UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
Facultad 1

Solución informática para la gestión de alumnos ayudantes en la gestión académica de pregrado

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

Autor:

Iván Leandro Rodríguez Rojas

Tutores:

Ing. Yanio García Vidal
Ing. Yasmany Tellez Collazo
Ing. Diana Rosa Pérez Santiesteban

La Habana. Junio, 2012

El auténtico desafío a vencer para llegar a ser un ser humano es aquel que afrontamos cuando, teniendo como alternativa hacer el mal se opta por un bien superior; ante el espejismo de la riqueza y la adulación mal habida se decide por el camino de lo legítimo. El auténtico desafío del ser humano es cuando ha logrado retar a su peor enemigo: él mismo, y dando una batalla sin tregua lucha día a día por conquistarse en las virtudes más altas y nobles; el desafío del ser humano es cuando no decide usurpar jamás un lugar que él mismo no ha conquistado, cuando se propone jamás ostentar méritos por él no logrados; cuando sus hazañas y proezas las guarda discretamente y ante el reconocimiento y elogio se mantiene humilde y con un espíritu de mansedumbre.

Miguel Ángel Cornejo

Declaración de autoría

Declaro ser el único autor del presente documento y autorizo al Centro de Informatización Universitaria de la Universidad de las Ciencias Informáticas y a la Universidad de las Ciencias Informáticas a dar el uso que estimen pertinente a este trabajo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Iván Leandro Rodríguez Rojas

_____

Firma del autor

Ing. Yanio García Vidal

_____

Firma del tutor

Ing. Yasmany Téllez Collazo

_____

Firma del tutor

Ing. Diana Rosa Pérez Santiesteban

_____

Firma del tutor

Datos de contacto

Autor:

Iván Leandro Rodríguez Rojas

Tutores:

Ing. Yanio García Vidal: Ingeniero en Ciencias Informáticas

Correo electrónico: yvidal@uci.cu

Ing. Yasmany Téllez Collazo: Ingeniero en Ciencias Informáticas

Correo electrónico: ytcollazo@uci.cu

Ing. Diana Rosa Pérez Santiesteban: Ingeniera en Ciencias Informáticas

Correo electrónico: dperezs@uci.cu

Consultantes:

Ing. Anay Díaz Estrada: Ingeniero en Ciencias Informáticas

Correo electrónico: adestrada@uci.cu

Dr. Melquiades Mendoza Pérez: Doctor en Ciencias Pedagógicas y Profesor Titular de la Universidad de Granma.

Correo electrónico: mmendozap@udg.co.cu

Agradecimientos

Agradezco a toda mi familia por el amor, dedicación, empeño y confianza puestos en mí.

A mis tutores Yasmany, Yanio y Diana por toda la ayuda brindada y por cada minuto dedicado a la realización de este trabajo.

A todos los grandes compañeros y amistades que me han acompañado en estos años de estudios, en especial a Jose, Yaima, Dianela, Lisbethy, Yaimy, Rodain, Irnel, Yusi, Claudia, Sandy, Enrique, Silvia, Tony, Yosvany, Marta, Yudith, Yaniel, Adonis, Carlos, Michael, Anailis y a todas las personas que conocí a lo largo de todo este tiempo en la universidad.

Y a todas las personas de que de una forma u otra han ayudado a mi formación en estos 5 años de estudio.

Dedicatorias

Dedico este trabajo de diploma a mis padres Ana y Nélsido, los cuáles han sido un ejemplo digno a seguir y a los cuales debo todo lo que tengo y lo que soy hoy en día.

A mis hermanas Yamila y Liudmila, mi hermano Gorki, por apoyarme y aconsejarme siempre.

A todos mis sobrinos, Tita, Chinchi, Ernestico, Tali, Marquito y Plin.

A mis cuñados Yaité e Iván por ser parte también de esta familia.

A todos en general por el amor y el apoyo brindados constantemente.

Resumen

En la sociedad actual deben ser evaluadas todas las tendencias que marcarán decisivamente el futuro. Una de estas tendencias está dirigida a informatizar, en su mayoría, las actividades que forman parte de la vida diaria del hombre con el fin de lograr una mejor gestión de las mismas. En el presente trabajo quedan descritos la fundamentación y los pasos seguidos en el desarrollo de las funcionalidades de un *software* para realizar acciones y mantener el control sobre la información generada por una organización que ha sido movimiento indisoluble del estudiantado universitario cubano: el Movimiento de Alumnos Ayudantes, específicamente dentro de la Universidad de las Ciencias Informáticas, con el objetivo de desarrollar una solución funcional para la consulta y gestión de la información de dicho movimiento. Para ello se realizó un estudio de todos los aspectos, conceptos y antecedentes teóricos a tener en cuenta y que servirían de base para guiar el desarrollo, luego aplicarlos y obtener resultados que dieron cumplimiento al objetivo trazado. Quedó definido el uso de UML como lenguaje de modelado y lenguajes de programación como PHP, JavaScript y HTML además de SQL como lenguaje de acceso a datos. Quedó definido el uso de herramientas para el modelado como Visual Paradigm, PgAdmin III para la administración de la base de datos y para el desarrollo del código fuente NetBeans. Con el fin de proveer respuestas rápidas y adaptables al cambio se utiliza un proceso de desarrollo de *software* con enfoque ágil basado en el nivel 2 de CMMI quedando validada la solución y comprobados los beneficios del resultado al finalizar dicho proceso.

Palabras clave: gestión de la información, movimiento de alumnos ayudantes, organización, universidad.

Abstract

In today's society we should be able to evaluate the trends that will check decisively the future. One of these trends is intended to computerize, most of the activities that are part of daily life of man in order to archive better management of them. In this document are described the rationale and the steps followed in developing the capabilities of software to perform actions and maintain control over the information generated by an organization that has been indissoluble movement of the Cuban university students: the Student Assistants Movement specifically in the University of Informatics Science, whit the objective of developing a functional solution for consulting and information management of the movement. For that was make a study of all aspects, concepts and theoretical antecedents to consider and provide a base for guiding the development, then apply and get results that gave effects to the target path. Was defined using UML as modeling language and programing languages such as PHP, JavaScript and HTML also SQL as data access language. Was a defined using modeling tool as Visual Paradigm, PgAdmin III for the database management and NetBeans for developing. In order to provide rapid responses and adaptable to change are use a process for software development with agile approach based on the CMMI level 2 being validated and the benefits of result.

Keywords: information management, organization, student assistants, university.

Contenido

INTRODUCCIÓN **1**

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA **4**

1.1 INTRODUCCIÓN 4

1.2 DESCRIPCIÓN GENERAL DEL OBJETO DE ESTUDIO 4

1.2.1Conceptos generales *4*

1.2.2 Gestión de la información del movimiento de alumnos ayudantes en la UCI *5*

1.3 ESTUDIO DEL ESTADO DEL ARTE 5

1.3.1 Estudio de los sistemas para la gestión de la información de organizaciones en el mundo *5*

1.3.2 Estudio de los sistemas para la gestión de la información de organizaciones en Cuba *7*

1.3.3 Valoración del estudio del estado del arte *8*

1.4 HERRAMIENTAS A UTILIZAR PARA LA CREACIÓN DE LA SOLUCIÓN 9

1.4.1 Herramienta para el desarrollo del código fuente *9*

1.4.2 Herramientas de modelado y diseño *9*

1.4.3 Herramienta para la administración de la base de datos y servidor web *10*

1.5 TECNOLOGÍAS A UTILIZAR PARA LA CREACIÓN DE LA SOLUCIÓN 11

1.5.1 Lenguaje de modelado *11*

1.5.2 Lenguajes de programación *12*

1.5.3 Lenguaje de acceso a datos *13*

1.5.4 Marco de trabajo *13*

1.5.5 Sistema gestor de base de datos *15*

1.5.6 Proceso de desarrollo con enfoque ágil centrado en el nivel 2 de CMMI *16*

1.5.7 Plataforma de desarrollo *17*

1.6 CONCLUSIONES PARCIALES 18

CAPÍTULO 2. CONCEPCIÓN DE LA PROPUESTA DE SOLUCIÓN **19**

2.1 INTRODUCCIÓN 19

2.2 MODELADO DEL NEGOCIO 19

2.2.1 Modelo de procesos del negocio *19*

2.2.2 Reglas del negocio *20*

2.3 INGENIERÍA DE REQUISITOS 21

2.3.1 Obtención y análisis de requisitos *22*

2.3.1.1 Descubrimiento de requisitos 22

2.3.1.2 Clasificación y organización de requisitos 23
2.3.2 Especificación de requisitos *25*
2.3.3 Validación de requisitos *27*
2.4 DISEÑO Y ARQUITECTURA DE LA SOLUCIÓN PROPUESTA 28
2.4.1 Estilos y patrones *29*
2.4.1.1 Estilo o modalidad arquitectónica 29
2.4.1.2 Patrón arquitectónico 30
2.4.1.3 Patrones de diseño 32
2.4.1.4 Patrones de base de datos 34
2.4.2 Modelo de datos *34*
2.4.3 Modelo de despliegue *36*
2.5 CONCLUSIONES PARCIALES 36
CAPÍTULO 3. IMPLEMENTACIÓN Y EVALUACIÓN DE LA PROPUESTA DE SOLUCIÓN 37
3.1 INTRODUCCIÓN 37
3.2 ESTÁNDARES DE CODIFICACIÓN 37
3.3 FUNDAMENTOS DE LA CONSTRUCCIÓN DE LA PROPUESTA DE SOLUCIÓN 41
3.4 DESCRIPCIÓN DE LA SOLUCIÓN 41
3.5 INTEGRACIÓN DE LA SOLUCIÓN AL SISTEMA DE GESTIÓN UNIVERSITARIA 45
3.5.1 Factores positivos que trae consigo la integración con los módulos del Sistema de Gestión Universitaria *45*
3.6 FUNDAMENTACIÓN DE LA EVALUACIÓN DEL RESULTADO 46
3.7 ASPECTOS IMPORTANTES DE LA EVALUACIÓN 47
3.7.1 Técnicas de evaluación estática *48*
3.7.1.1 Beneficios de las revisiones 48
3.7.1.2 Objetivos de la evaluación estática. 49
3.7.1.3 Técnicas de evaluación estáticas utilizadas 49
3.7.2 Técnicas de evaluación dinámica *49*
3.7.2.1 Estrategias y niveles de prueba 50
3.8 RESULTADOS DE LA EVALUACIÓN DEL SOFTWARE 51
3.9 ANÁLISIS DE LOS BENEFICIOS 53
3.10 CONCLUSIONES PARCIALES 54
CONCLUSIONES GENERALES 55
RECOMENDACIONES 56
BIBLIOGRAFÍA REFERENCIADA 57

BIBLIOGRAFÍA CONSULTADA **60**

GLOSARIO DE TÉRMINOS **62**

ANEXOS **64**

Índice de ilustraciones

Ilustración 1. Flujo de información de GUUD 15
Ilustración 2. Modelo de procesos de negocio 20
Ilustración 3. Interacción Cliente-Servidor 30
Ilustración 4. Funcionamiento del patrón Modelo-Vista-Controlador 31
Ilustración 5. MVC en GUUD 32
Ilustración 6. MVC en GUUD 32
Ilustración 7. Modelo de datos 35
Ilustración 8. Modelo de despliegue 36
Ilustración 9. Modelo en V de la evaluación del software 48
Ilustración 10. Resultado de las iteraciones 52
Ilustración 11. Resultados de las pruebas de integración 53

Índice de tablas

Tabla 1. Requisitos funcionales 23
Tabla 2. Requisitos no funcionales 23
Tabla 3. Especificación del requisito Crear solicitud de ayudantía 26
Tabla 4. Plan de iteraciones 27
Tabla 5. Estrategia de prueba definida para la solución 51

Introducción

Hoy en día el hombre se encuentra inmerso en un proceso cuyo objetivo principal es llegar a alcanzar la futura sociedad del conocimiento, al mismo tiempo, debe ser capaz de obtener y saber utilizar el capital intelectual del que dispone, y a la vez transferirlo en beneficio de la propia sociedad. La gestión del conocimiento y de la información ha de llevarse a cabo con todos los medios disponibles en la actualidad. Por su alta incidencia en la modernización y eficiencia en distintos sectores de la vida del hombre, los medios que más se adecuan son las denominadas Tecnologías de la Información y las Comunicaciones, también conocidas por las siglas TIC, pilares de la sociedad de la información. Las TIC son las herramientas, en nuestros días, más poderosas concebidas por el hombre para manipular información; estas cobran su verdadera dimensión como tecnología privilegiada de los sistemas de información, de manera primordial en las organizaciones.

Un sistema de gestión eficiente puede ser de gran ayuda para la erradicación de riesgos sociales, aumento de la satisfacción de los clientes, en la mejora de la efectividad de la producción, la reducción de costos y el logro de mejoras continuas a la hora de gestionar cualquier tipo de información. Este persigue un único objetivo: realizar operaciones tales como análisis, procesamiento, conservación y accesibilidad de datos organizados que tienen un significado sobre un proceso o fenómeno.

La manipulación de la información es un elemento fundamental para el control y la toma de decisiones sobre los distintos procesos que se desarrollan dentro de una entidad, que a su vez permiten establecer la mejor manera de lograr objetivos definidos por grupos sociales a través de métodos con el menor esfuerzo posible y reducen la duplicidad de funciones al delimitar responsabilidades.

En Cuba existen disímiles y variadas organizaciones e instituciones que aumentan la calidad de las actividades en cada uno de los sectores sociales. Específicamente en el sector de la educación, el Movimiento de Alumnos Ayudantes[1] (MAA) ha sido movimiento indisoluble de la vida académica de los estudiantes universitarios cubanos desde los años 60. Desde sus inicios la Universidad de las Ciencias Informáticas (UCI) ha hecho del MAA el pilar fundamental para enriquecer y apoyar el desarrollo académico de sus estudiantes, es por esto que cada curso académico se mantiene el control de la información, desde las solicitudes realizadas para vincularse al movimiento, la gestión de las mismas hasta el seguimiento que se les da a los estudiantes una vez que forman parte del MAA.

Actualmente las solicitudes para ingresar al movimiento se realizan a través de correos electrónicos. Teniendo en cuenta que el MAA de la universidad cuenta con un elevado número de integrantes, los

[1] El **Movimiento de Alumnos Ayudantes**, es una organización que agrupa a estudiantes de alto aprovechamiento docente para apoyar el desarrollo académico de los estudiantes de un centro de estudios determinado.

departamentos docentes a los cuales estos pertenecen manejan gran cantidad de información (desde datos personales, datos de la ayudantía, evaluaciones, planes de trabajo, entre otros aspectos). Toda la información tramitada a partir de este proceso queda archivada en documentos en formato digital creados y actualizados por los directivos del MAA solo con la ayuda de herramientas ofimáticas, por lo que se dificulta el rápido acceso, modificación y actualización de la información, se eleva el nivel de duplicidad de la información y esta a la vez no se encuentra centralizada.

De la **situación problemática** expuesta anteriormente se plantea el siguiente **problema de investigación**: ¿cómo contribuir a agilizar el manejo de la información del proceso de gestión de alumnos ayudantes (AA) de la UCI? Constituyendo como **objeto de estudio de la investigación** los procesos de gestión de la información en las organizaciones, por lo que el **objetivo general** que se persigue es desarrollar una agrupación funcional para la gestión de la información del MAA integrada al Subsistema de Gestión Académica de Pregrado (SGAP), del cual se derivan los siguientes **objetivos específicos**:

- Confeccionar el marco teórico-conceptual de la investigación.
- Proponer un modelo para la solución.
- Desarrollar la propuesta y aplicar validaciones antes, durante y después de la creación del código.
- Realizar un análisis de los beneficios del resultado obtenido.

Cuyo **campo de acción** abarca el proceso de gestión de la información del MAA en la UCI. Para guiar la investigación se propone la siguiente **idea a defender**: “con el desarrollo de una agrupación funcional para la gestión de la información del MAA se agilizará en gran medida dicho proceso en la UCI”.

Los **métodos científicos de investigación** utilizados son los siguientes: dentro de la clasificación de métodos teóricos se hizo uso del **analítico-sintético**, que posibilitó la construcción de un modelo de investigación y el análisis más a fondo de las características y elementos más importantes del proceso de gestión de AA en la universidad. El método **histórico-lógico** con el objetivo de analizar los antecedentes y cómo ha evolucionado el proceso desde sus inicios en el centro de estudios. Dentro de la clasificación de métodos empíricos fue utilizado el método de la **entrevista** para lograr un mejor entendimiento sobre las características y distintas etapas en el funcionamiento del proceso de gestión en la universidad, el diseño de la misma está accesible en el **anexo A**.

La presente investigación fue realizada para darle solución a un problema que afectaba de manera general a todos los individuos involucrados en el proceso de gestión de AA en la UCI, pues era necesario acceder y gestionar la información de manera rápida y eficiente. Como **posibles resultados**

de la investigación se espera obtener una agrupación funcional que cumpla con las expectativas y necesidades del manejo de la información del MAA, con la cual se homogenizará el proceso para todas las facultades de la universidad y se agilizará la captación, aceptación y seguimiento de los estudiantes vinculados a dicho movimiento.

A continuación se presenta una breve descripción de los apartados que componen el presente documento.

Capítulo 1. Fundamentación teórica: presenta un análisis de los elementos teóricos, investigaciones y antecedentes en general que sirvieron de base, orientaron y dirigieron conceptualmente el desarrollo de la investigación, así como un estudio de las herramientas y tecnologías que se utilizaron en el desarrollo de la solución.

Capítulo 2. Concepción de la propuesta de solución: se analiza la propuesta de solución planteada producto de la investigación, enmarcada en el proceso actual de gestión de alumnos ayudantes existente en la universidad. Se realiza un análisis del proceso que conforma el negocio y se presenta además una descripción detallada de los requisitos con los que debe cumplir la propuesta de solución, además de los elementos del diseño y la arquitectura.

Capítulo 3. Implementación y evaluación de la propuesta de solución: se realiza una breve descripción del componente implementado especificando las funcionalidades del mismo y como se realiza la integración de este con los demás módulos del Sistema de Gestión Universitaria (SGU). Por otra parte, se detalla todo el proceso de evaluación del producto desarrollado especificando las evaluaciones y validaciones realizadas antes, durante y después del proceso de creación del código. Además de un análisis general del producto desarrollado y validado, así como los beneficios que trae consigo el mismo.

Capítulo 1. Fundamentación teórica

1.1 Introducción

En el presente capítulo se realiza un análisis de los elementos teóricos y antecedentes investigativos sobre la gestión de información en las organizaciones, así como una valoración acerca de las tendencias actuales de los sistemas para gestionar información. Además se realiza un estudio de herramientas y tecnologías a utilizar durante el desarrollo de la propuesta de solución, con el objetivo de que toda la información recopilada sirva de base, oriente y guie conceptualmente el desarrollo de la investigación.

1.2 Descripción general del objeto de estudio

Si se piensa en los recursos humanos y todos los logros que se pueden alcanzar con ellos, se conocerá que su valor radica en el cúmulo de información adquirida por estos, mediante el procesamiento de datos, y que al relacionarla con una determinada actividad humana esta adquiere un valor específico transformándose en conocimiento.

1.2.1Conceptos generales

A continuación, serán expuestos varios conceptos de gran importancia, de los cuales es necesario tener conocimiento, antes de comprender el negocio. Los conceptos descritos mantienen una estrecha relación dentro del negocio analizado, esta relación se define a continuación: “en la UCI se realiza cada año la **gestión de la información** de los **alumnos ayudantes**, los cuales forman parte de una **organización** cuyo objetivo es apoyar el desarrollo académico de los estudiantes”.

- **Gestión de la información**: Carlota Bustelo Ruesta[2] en un intento de definición de la gestión de información se refiere a la misma como "el conjunto de actividades realizadas con el fin de controlar, almacenar y, posteriormente, recuperar adecuadamente la información producida, recibida o retenida por cualquier organización en el desarrollo de sus actividades” (Bustelo, 2001).
- **Alumno ayudante**: los alumnos ayudantes son aquellos estudiantes de alto aprovechamiento docente previamente seleccionados en las carreras, tanto en las sedes centrales como en las sedes universitarias, que se distinguen por mostrar ritmos de asimilación más rápidos, aptitudes favorables para el aprendizaje de alguna o algunas de las disciplinas del plan de estudio y para

[2] **Carlota Bustelo Ruesta** consultora independiente en gestión de la información y documentos.

la investigación científica o el trabajo de desarrollo técnico **¡Error! No se encuentra el origen de la referencia.**.

- **Organización**: la Real Academia Española define a una organización como "la asociación de personas regulada por un conjunto de normas en función de determinados fines" (RAE, 2012).

1.2.2 Gestión de la información del movimiento de alumnos ayudantes en la UCI

El Ministerio de Educación Superior define, en la resolución 210 del año 2007, el proceso como se describe a continuación. El proceso de selección de los alumnos ayudantes, en cada curso académico, es responsabilidad del decano de la facultad, contando para esta tarea con la participación activa de representantes de las organizaciones estudiantiles. El decano establecerá el procedimiento a seguir, así como designará al personal encargado de la misma. Los estudiantes que cursan las carreras que se desarrollan en la sede central y en las sedes universitarias, constituyen la cantera para la selección de los AA. Finalizado el proceso, el decano de la facultad aprueba oficialmente los estudiantes seleccionados e informa a los departamentos docentes y la dirección de la sede universitaria sobre estos resultados. Luego se realiza un seguimiento a estos estudiantes a lo largo de todo el curso académico (MES, 2007b).

1.3 Estudio del estado del arte

En la sociedad del mundo actual deben ser evaluadas, en todas sus dimensiones, las tendencias que marcarán decisivamente el futuro. Una de estas tendencias está dirigida a informatizar, en su mayoría, las actividades que forman parte de la vida diaria del hombre con el fin de lograr una mejor gestión de las mismas. Este concepto se ha hecho realidad desde hace algunos años donde empresas, instituciones y centros educacionales buscan agilizar sus procesos o servicios y obtener así mejores resultados.

En el mundo existen disímiles y variados sistemas para gestionar procesos, tareas, actividades y servicios. La implantación de sistemas web con estos fines se hace cada vez mayor dentro de la WWW dándole al hombre grandes ventajas sobre los distintos sectores en los que se enmarca la sociedad actual.

1.3.1 Estudio de los sistemas para la gestión de la información de organizaciones en el mundo

Intranet de la Universidad Politécnica de Valencia (UPV):

Intranet desde la cual los alumnos de la universidad podrán consultar información relacionada con sus estudios, y realizar diversas gestiones mediante la secretaría virtual.

Entre otras cosas, el estudiante, en el apartado personal tiene acceso a información académica (expediente, listas, datos personales, etc.) e información de asignaturas; en cuanto a la secretaría virtual podrá solicitar acreditaciones, certificados y expedición de títulos.

Dicha intranet cuenta con un Área de Sistemas de Información y Comunicaciones (ASIC). La unidad de aplicaciones del ASIC da soporte a la automatización del sistema de información corporativo. Igualmente, colabora con la planificación y modernización de los procesos administrativos de la universidad. Sus objetivos son los de diseñar, desarrollar, documentar y mantener cuantas aplicaciones de gestión se realizan en el ASIC. Por otra parte, trata de facilitar los procesos administrativos a los usuarios, ya sean alumnos, personal o proveedores. Dentro de las aplicaciones de gestión con las que cuenta se pueden mencionar las siguientes:

- Gestión académica.
- Gestión de RRHH.
- Gestión de investigación.
- Gestión de biblioteca.

Sistema web para alumnos ayudantes de la Universidad de Concepción de Chile:

Sistema web que es responsabilidad de la Dirección de Tecnologías de Información (DTI) de dicha universidad, el mismo permite crear postulaciones de alumnos ayudantes para laboratorios y soporte técnico de la DTI apoyando el proceso de provisión de las TIC. Cada sistema o producto de la DTI, si bien apoya un proceso de negocio particular, forma parte de un conjunto integrado de aplicaciones que se complementan entre sí tanto a nivel funcional como de datos. De esta forma, implementan el quehacer administrativo operativo la Universidad en los distintos ámbitos: finanzas, extensión, docencia y salud.

Sistemas Intranet:

Una intranet permite aglutinar la información que generan los distintos departamentos o secciones de la organización. Esto implica la posibilidad de consultar por un dato que puede estar localizado en distintas fuentes de información que la empresa u organización posee. Cada dato podría estar en bases de datos distintas, e incluso localizados en distintas ciudades o países.

Entonces, se puede hacer referencia a la ventaja de aglutinar la información en lugar de centralizarla. Si la máquina o la ruta de acceso fallan en el momento en que se desea consultar la información se

perderá el acceso a la misma. Además, el centralizar la información requiere el crecimiento progresivo del computador central y una inversión monetaria sostenida para mantenerlo eficiente.

La distribución de los datos en distintas máquinas permite disponer de distintos tipos de servidores y sistemas operativos con distintos tipos de bases de datos a distintos precios, y lograr obtener de todos ellos el dato necesario en la pantalla del usuario.

Una intranet trae consigo grades ventajas como identificar flujos de información (identificar quien genera y quien usa la información que maneja la empresa u organización), economizar recursos (cuando es capaz de ofrecer una interfaz estandarizada que incorpore y homologue la mayor parte de las aplicaciones computacionales requeridas para la gestión de la empresa), actualizar información, etc. (Pastor, 1998).

Facultad de Medicina de la Universidad de Chile:

Posee un sistema donde se gestionan las actividades de este centro. El mismo abarca distintas áreas como son la de pregrado, postgrado, egresados, investigación y académicos. Cuenta con un programa de alumnos ayudantes el cual contiene distintos documentos (resoluciones y normas) de tipo informativos para dicho movimiento y brinda la posibilidad de descargar documentos para que el estudiante solicite, cambie o se reincorpore a la ayudantía, así como para que los tutores y alumnos ayudantes presenten sus evaluaciones al finalizar el año académico. Dicho sistema brinda servicios para usuarios administrativos, estudiantes y personal docente.

1.3.2 Estudio de los sistemas para la gestión de la información de organizaciones en Cuba

Sistema automatizado para la gestión de la información del capital humano de un departamento docente:

Sistema que facilita la gestión de la información del capital humano de un departamento docente, en la Universidad de Matanzas “Camilo Cienfuegos” (UMCC). El mismo permite ordenar datos e interactuar dinámicamente con la información existente, así como tener disponible la información en la web. Dicha propuesta abarca también el trabajo conjunto entre profesores y alumnos ayudantes brindando la posibilidad de actualizar los datos de los alumnos ayudantes como son: datos personales, si imparten clases o no, el grupo al que le imparte estas, la asignatura sobre la que se desempeña como AA, cantidad de horas, tareas a desempeñar, así como la elaboración de reportes e informes que sirven para la toma de decisiones.

AKADEMOS:

Sistema de gestión académica utilizado en la UCI desde el curso 2004-2005, desarrollado con Microsoft Visual Studio .NET 2003 haciendo uso de las tecnologías ASP.NET y SQL Server 2000. Permite la gestión de estudiantes y estructuras, brinda información a los estudiantes sobre su desempeño académico, mantiene el control del desarrollo del proceso docente, notas y asistencia e incluye la generación de reportes personalizados. El mismo no maneja ningún tipo de información referente a los alumnos ayudantes.

Sistema de Gestión de la Nueva Universidad (SIGENU):

SIGENU es un sistema desarrollado en el Instituto Superior Politécnico "José Antonio Echeverría" con el fin de gestionar la información académica vinculada a la educación superior en Cuba. El sistema contiene entre sus aplicaciones la Aplicación Cliente Web de Secretaría, la misma cuenta con los módulos codificadores, matrícula, control de estudiantes, plan de estudio, evaluaciones y reportes. Mediante la opción Estudiantes del menú principal se accede a un listado de todos los estudiantes, este formulario permite también la búsqueda de estudiantes mostrando vistas diferentes en dependencia del rol. El usuario con el rol controlador de alumno ayudante tiene acceso a actualizar el estado de alumno ayudante de un estudiante o los grupos asociados a este.

1.3.3 Valoración del estudio del estado del arte

Durante el estudio de antecedentes o estado del arte fueron analizados diversos sistemas existentes en el mundo y en el país, los mismos poseen variedad de funcionalidades, en dependencia del área donde son aplicados, lo que los hace sistemas competentes a la hora de gestionar las distintas actividades realizadas en una empresa, institución u organización. Vale destacar que debido a que algunos de estos sistemas están enmarcados dentro de la calificación de *software* privativo (lo que entra en contradicción con la migración hacia el *software* libre[3] que está llevando a cabo nuestro país) y además la carencia de privilegios sobre algunos de ellos, no se pudo realizar un análisis a fondo de las funcionalidades que los mismos incluyen.

En cambio, de ellos fueron tomadas en consideración y como base algunas características que servirán para guiar el desarrollo de la propuesta de solución entre las que se encuentran: la actualización de la información a través del uso de interfaces que permitan a los involucrados generar y consultar información en tiempo real lo que otorga credibilidad al proceso. Mantener una mejor gestión de los recursos humanos, al existir una rápida actualización de los datos, los directivos del proceso pueden mejorar la gestión de sus recursos humanos porque pueden monitorear las acciones

[3] El **software libre** es una cuestión de la libertad de los usuarios de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software (FSF, 2012).

que realizan los involucrados para el logro de los objetivos o acuerdos definidos. Fortalecer la comunicación porque la información se sistematiza y fluye. La posibilidad de gestionar evaluaciones y emitir criterios cualitativos para la completitud de estas.

1.4 Herramientas a utilizar para la creación de la solución

Las herramientas estudiadas y analizadas a continuación, para el desarrollo de la solución, fueron definidas por el grupo de implantación, tecnología y soporte del Centro de Informatización Universitaria.

1.4.1 Herramienta para el desarrollo del código fuente

Entorno de desarrollo integrado (IDE) NetBeans (versión 6.7.1 o superior):

Es una herramienta destinada para escribir, compilar, depurar y ejecutar programas. NetBeans es un proyecto de código abierto que ha alcanzado mucho éxito. Presenta una gran base de usuarios, una comunidad en constante crecimiento. *Sun MicroSystems* fundó el proyecto de código abierto NetBeans en junio de 2000 y continúa siendo el patrocinador principal de los proyectos. Es un producto de código abierto y gratuito para uso tanto comercial como no comercial. El código fuente está disponible para su reutilización de acuerdo con la Licencia de Desarrollo Común y Distribución (CDDL) v1.0 y la Licencia Pública General de GNU (GPL[4]) v2 (OCA, 2012).

Quedó definido el uso de la herramienta NetBeans por ser un potente IDE de desarrollo de aplicaciones que brinda variedad de facilidades a los programadores a la hora de escribir el código fuente y por poseer una amplia comunidad que provee mejoras constantes.

1.4.2 Herramientas de modelado y diseño

Visual Paradigm (versión 8.0):

Es una poderosa herramienta CASE[5] que utiliza UML para el modelado, es la herramienta por excelencia para ser utilizada en un ambiente de *software* libre. Permite crear diferentes tipos de diagramas en un ambiente totalmente visual y genera código para varios lenguajes.

Visual Paradigm ofrece:

- Un entorno de creación de diagramas para UML 2.0.
- Diseño centrado en casos de uso y enfocado al negocio que genera un *software* de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la

[4] **GPL**, Licencia Pública General de GNU (en inglés, *General Public License).*
[5] **CASE**, Ingeniería de Software Asistida por Computadoras (en inglés, *Computer Aided Software Engineering*)

comunicación.

- Capacidad de ingeniería directa e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Disponibilidad de múltiples versiones.
- Disponibilidad de integrarse en los principales IDE.
- Disponibilidad en múltiples plataformas (VPILtd, 2011).

Evolus Pencil (versión 1.2.1):

Herramienta gratuita y de código abierto que posibilita la creación de diagramas y prototipos de interfaz gráfica de usuario. Evolus Pencil es la evolución de Pencil y con esta se pueden crear fácilmente las ventanas de prototipos arrastrando los diferentes elementos ya sea como extensión de Firefox o como aplicación estándar para Windows o Linux (PP, 2012).

Quedó definido el uso de las herramientas anteriormente descritas por las razones siguientes. Primeramente, Visual Paradigm por la inmensa variedad de posibilidades que brinda a la hora de realizar diferentes tipos de modelados, posibilitando la representación gráfica de diagramas, permitiendo ver el sistema desde diferentes perspectivas y mejorar así la comunicación entre los miembros del equipo usando un lenguaje gráfico y por ser una herramienta por excelencia para ser utilizada en un ambiente de *software* libre. Finalmente, Evolus Pencil por ser una herramienta gratuita que facilita el diseño de prototipos de interfaz gráfica sin necesidad de tener conocimientos avanzados en el tema.

1.4.3 Herramienta para la administración de la base de datos y servidor web

Administrador de base de datos PgAdmin III (versión 1.10.0)*:*

Herramienta gráfica para diseñar y gestionar sistemas de bases de datos PostgreSQL para sistemas Unix y Windows. Es de libre acceso bajo los términos de la licencia PostgreSQL y puede ser redistribuido siempre que las condiciones de la licencia se cumplan. Este proyecto es gestionado por el equipo de desarrollo de PgAdmin. Este *software* fue desarrollado como sucesor de los productos originales, PgAdmin y PgAdmin II, que, aunque populares había limitaciones en el diseño que les evitó ser trasladados al “siguiente nivel”. PgAdmin II está escrito en C++ y utiliza excelentes *wxWidgets* (antes conocidos como *wxWindows*) multiplataforma. La conexión a PostgreSQL se realiza utilizando la nativa librería *libpq*. Está diseñada para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todas las características de PostgreSQL y facilita enormemente la administración. La aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor y

un agente para lanzar *scripts* programados. La conexión al servidor puede hacerse mediante conexión TCP/IP y puede encriptar mediante SSL para mayor seguridad (GU, 2011).

Servidor web Apache (versión 2.2):

Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP, se desarrolla dentro del proyecto *HTTP Server* (httpd) de la *Apache Software Foundation*.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, tiene amplia aceptación en la red y desde 1996, Apache, es el servidor HTTP más usado. Entre las características más comunes de este servidor web se encuentran las siguientes:

- Modular.
- Código abierto.
- Multiplataforma.
- Extensible.
- Popular (fácil conseguir ayuda/soporte) (ASF, 2012).

Quedaron definidas como herramientas para la administración de bases de datos y como servidor web las anteriormente descritas, PgAdmin III y Apache respectivamente, por ser herramientas que permiten ser ejecutadas en ambientes de *software* libre y la variedad de posibilidades brindadas por estas.

1.5 Tecnologías a utilizar para la creación de la solución

Las tecnologías estudiadas y analizadas a continuación, para el desarrollo de la solución, fueron definidas por el grupo de implantación, tecnología y soporte del Centro de Informatización Universitaria.

1.5.1 Lenguaje de modelado

Lenguaje de modelado UML (versión 2.0):

El Lenguaje Unificado de Modelado es el lenguaje de modelado de sistemas de *software* más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de *software*. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de *software* reutilizables.

El Lenguaje Unificado de Modelado es la sucesión de una serie de métodos de análisis y diseño orientados a objetos que aparecen a fines de los 80 y principios de los 90. UML es llamado un lenguaje de modelado, no un método. Los métodos están conformados por, un lenguaje de modelado y un proceso. El UML, fusiona los conceptos de la orientación a objetos aportados por Booch, OMT y OOSE. UML incrementa la capacidad de lo que se puede hacer con otros métodos de análisis y diseño orientados a objetos (González, 2012).

Quedó definido como lenguaje de modelado UML por ser una tecnología que permite describir y documentar desde los más simples sistemas de *software* hasta los más complejos permitiendo un buen desempeño y comunicación entre el equipo de desarrollo.

1.5.2 Lenguajes de programación

Un lenguaje de programación es una técnica estándar de comunicación que permite expresar las instrucciones que han de ser ejecutadas en una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen un programa informático. Un lenguaje de programación permite a un programador especificar de manera precisa sobre qué datos una computadora debe operar, cómo deben ser estos almacenados y transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias.

HTML (versión 4):

HTML o lenguaje de marcado de hipertexto es el formato de los documentos de la WWW. Los documentos HTML son archivos de texto plano (también conocidos como ASCII) que pueden ser creados mediante cualquier editor de texto, aunque también existen programas específicos para editar HTML concebidos específicamente para editar páginas web en HTML. HTML no permite definir de forma estricta la apariencia de una página, aunque en la práctica, se utiliza también como un lenguaje de presentación. Los archivos de HTML se leen en un navegador web tal como *Netscape Navigator*, *Microsoft Explorer, Mozilla, Opera,* etc. La presentación de la página es muy dependiente del navegador o *browser* utilizado ya que el mismo documento no produce el mismo resultado en la pantalla si se visualiza con uno u otro, o sea, HTML se limita a describir la estructura y el contenido de un documento, y no el formato de la página y su apariencia (Lamarca, 2011).

JavaScript (versión 1.6):

Lenguaje de programación interpretado (por lo que no es necesario compilar los programas para ejecutarlos) orientado a páginas web, que se utiliza principalmente para crear páginas web dinámicas. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del DOM. Con la aparición de la Web 2.0 se ha convertido en un verdadero lenguaje de programación que

aporta la potencia de cálculo del navegador para aumentar la usabilidad de aplicaciones web con técnicas avanzadas como Ajax o JCC (Eguíluz, 2008).

PHP (versión 5.3):

Originado inicialmente con el nombre de *PHP Tools* o *Personal Home Page Tools*, es un lenguaje de programación interpretado, con licencia *Open Source*. Es un lenguaje multiplataforma que posee una capacidad de conexión con la mayoría de los manejadores de bases de datos que se utilizan en la actualidad, destaca la capacidad de expandir su potencial utilizando una gran cantidad de módulos (llamados *ext´s* o extensiones). Puede leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML. Es libre por lo que se presenta como una alternativa de fácil acceso para todos y permite las técnicas de programación orientada a objetos. El mismo es gratuito y de gran popularidad, enorme eficiencia, versatilidad y posee un gran número de funciones predefinidas (TPG, 2011).

1.5.3 Lenguaje de acceso a datos

SQL:

SQL o lenguaje estructurado de consultas es un lenguaje declarativo de alto nivel o de no procedimiento de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en estas. No es más que un lenguaje estándar de comunicación con bases de datos. Se habla por tanto de un lenguaje normalizado que nos permite trabajar con cualquier tipo de lenguaje (ASP, PHP…) en combinación con cualquier tipo de base de datos (MS Access, SQL Server, MySQL...) (Álvarez, 2001).

Quedaron definidos los lenguajes de programación PHP, HTML, JavaScript y SQL como lenguaje de acceso a datos por ser los más indicados para el desarrollo de aplicaciones que brinden seguridad, calidad, aceptación y el buen rendimiento de una aplicación web, a través del uso de las posibilidades brindadas por estos.

1.5.4 Marco de trabajo

GUUD (versión 1.0):

Es un marco de trabajo para personas que construyen aplicaciones web haciendo uso de PHP, propuesto por el equipo de arquitectura del Centro de Informatización Universitaria de la UCI cuya base la conforman los marcos de trabajo de desarrollo jQuery (versión 1.3.2) y CodeIgniter (versión 1.7.3) con el objetivo de permitir a los desarrolladores crear proyectos más fácilmente que si los escribiesen desde cero. El mismo posee todas las características de los marcos de trabajo antes

mencionados y a la vez quedaron incluidas un conjunto de novedades y adaptaciones, entre las cuales se destacan las siguientes.

Del lado del cliente fueron modificados una serie de *widgets* presentes en la base del jquery-ui para ser utilizados como interfaz de usuario como por ejemplo *date*, *tab* (ambos interfaces de los *widgets* del mismo nombre en jquery-ui) y el *popup* (interfaz del *dialog* de jquery-ui). Además, se implementaron otros nuevos como son los ejemplos del *attach*, *menu*, *message*, *tooltip*, *form* (construido de la unión del *form* de jQuery el cual se utiliza para el envío de formularios Ajax y el *validate* utilizado para validar formularios), *grid* (utiliza como *plugin* el *jqgrid*), *multiselect* (para selecciones múltiples), *navbar* (para el uso de barras de navegación), *tree* (para la creación de árboles jerárquicos) y el *graph* (utiliza la librería *highchart*). Fue implementado un *plugin* a jQuery para el manejo de espacios de nombres e internacionalización así como funciones comunes para todo el sistema, contenidas en los archivos *core.js* y *common.js*, entre las que se destacan: *loadIn*, *getDataJson*, *createSelect*, *isArray*, *isFunction*, *site_url*.

Del lado del servidor fue modificado el manejo de excepciones y mensajes, también se implementó el IOC (*inversion of control*) para la interacción entre los módulos del sistema y fue añadida la características de modularidad, o sea, que un sistema pueda dividirse en módulos. Además se añadieron los *plugins export_pi* (permite exportar a los formatos .pdf, .csv o .xls), *import_pi* (permite importar desde los formatos .csv o .xls) y se añadieron y modificaron los *helpers* o asistentes mencionados a continuación.

- Añadidos: *template* (posibilidad del uso de plantillas, para esto fue añadida la librería *template*), *assets* (utilizado para la integración de las vistas de JavaScript, css, imágenes y el *template*), *grid* y *json*.
- Modificados: *form, array* y *security* (García, 2011).

Los datos fluyen a través de GUUD de la siguiente manera (Ver Ilustración 1. Flujo de información de GUUD). El controlador frontal (*index.php*) inicializa todos los recursos que necesita el marco de trabajo para ejecutar, el ruteador examina la solicitud HTTP para determinar que debería hacer con esta. Si existe este archivo de caché, se lo envía directamente al navegador, sin pasar por la ejecución normal del sistema. Antes de cargar el controlador, por razones de seguridad, se filtran la solicitud HTTP y cualquier otro dato enviado por el usuario. El controlador carga las librerías (que funcionan como intermediarias entre las capas de negocio y acceso a datos), las bibliotecas del núcleo, *helpers* y cualquier otro recurso requerido para procesar una solicitud, además de procesar las vistas (estas a su vez obtienen información de los *scripts*).Si el caché está habilitado la vista se cachea para que futuras peticiones que la necesiten puedan ser servidas.

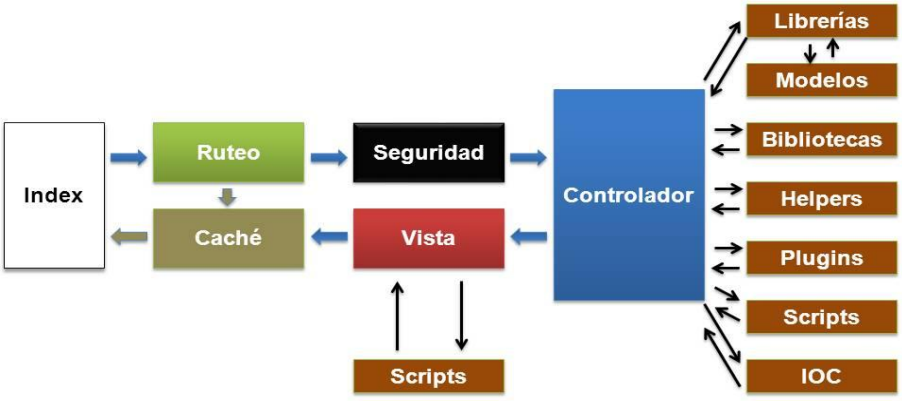


Ilustración 1. Flujo de información de GUUD

¿Qué es jQuery?

Es un marco de trabajo JavaScript que brinda una infraestructura al programador que facilita la creación de aplicaciones complejas del lado del cliente, por ejemplo, interfaces de usuario, efectos dinámicos, aplicaciones que hacen uso de Ajax, etc. El mismo ofrece una serie de funcionalidades basadas en JavaScript y permite cambiar el contenido de documentos HTML sin necesidad de recargarlos haciendo uso de DOM y Ajax. Dicho marco de trabajo tiene licencia para uso en cualquier tipo de plataforma, personal o comercial. Es un producto serio, estable, bien documentado y con gran equipo de desarrolladores a cargo de la mejora y actualización del marco de trabajo (Álvarez, 2011).

¿Qué es CodeIgniter?

Es un marco de trabajo para el desarrollo de aplicaciones web usando PHP. Provee una colección de librerías que facilitan la implementación de tareas, de desarrollo web, comúnmente necesarias como acceder a una base de datos, mandar un *email*, validar datos de un formulario, mantener secciones, manipular imágenes y mucho más, usa el patrón arquitectónico Modelo-Vista-Controlador el cual separa la lógica del negocio de la interfaz de usuario. Se encuentra bajo una licencia *open source* lo que permite que pueda ser usado libremente (CDI, 2012).

1.5.5 Sistema gestor de base de datos

PostgreSQL (versión 8.4.1):

Es un Sistema Gestor de Bases de Datos Objeto-Relacional (de sus siglas en inglés ORDBMS) basado en el proyecto Postgres, de la Universidad de Berkeley. Es una derivación libre de este proyecto. Debido a la licencia libre, PostgreSQL puede ser utilizado, modificado y distribuido por todo el mundo de forma gratuita para cualquier propósito, sea comercial, privado, o académico.

Fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, incluido más tarde en otros sistemas de gestión comerciales. PostgreSQL es un sistema objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional.

Es considerado como el más avanzado de las bases de datos de código abierto en el mundo. Proporciona una gran cantidad de características que normalmente solo se encuentran en bases de datos comerciales tal como Oracle (PGDG, 2012).

1.5.6 Proceso de desarrollo con enfoque ágil centrado en el nivel 2 de CMMI

Un proceso de desarrollo de *software* define quien está haciendo qué, cuándo y cómo con el objetivo de garantizar la solidez y la eficiencia en el proceso de creación de un *software*, para mayor comprensión: “es un conjunto de herramientas, técnicas, materiales y personas para producir o desarrollar *software*” (EVA, 2011); un proceso de desarrollo efectivo proporciona normas para el desarrollo eficiente de un *software*. Ahora, un proceso de desarrollo ágil es aquel capaz de proveer respuestas rápidas y adaptables al cambio sin exigir una excesiva generación de documentos, es decir, garantizan un proceso de desarrollo suficiente pero no excesivo.

La UCI es una universidad cuya misión se centra en la producción de *software* y servicios informáticos a raíz de la vinculación estudio-trabajo como modelo de formación y la cual se encuentra inmersa en un proceso de migración hacia el *software* libre. Con el fin de proveer respuestas rápidas, adaptables al cambio y desarrollar *software* de manera ágil, el Centro de Informatización Universitaria está acometiendo un proyecto de mejora de sus procesos basado en el nivel 2 del modelo CMMI (*Capability Madurity Model Integration*). Dicho proceso de mejora es el rector del ciclo de vida de la creación de *software* en este centro de desarrollo y por el cual se guiará el desarrollo de la agrupación funcional.

¿Qué es CMMI?

CMMI es un modelo de referencia para el crecimiento de capacidades y madurez, que se enfoca tanto en procesos de administración como de ingeniería de sistemas y s*oftware*, el mismo fue desarrollado por el SEI (*Software Engineering Institute*), centro de investigación y desarrollo, patrocinado por el Departamento de Defensa de los Estados Unidos y gestionado por la universidad de Carnegie Mellon. Este cuenta con 5 niveles de madurez: nivel 1 o inicial, nivel 2 o gestionado, nivel 3 o definido, nivel 4 o

gestionado cuantitativamente y el nivel 5 u optimizado. En el nivel de madurez número 2 la gestión del proyecto es más disciplinada, se establecen y siguen políticas organizativas, los planes de proyecto y las descripciones de procesos se documentan y se siguen, se asigna responsabilidad y autoridad sobre el ciclo de vida y el estado de las actividades y productos de trabajo es visible a la dirección en puntos definidos (EVA, 2011).

Según la definición dada por el SEI, CMMI es un enfoque de mejora de procesos que proporciona a las organizaciones los elementos esenciales de procesos efectivos, lo que mejorará su rendimiento. Basada en CMMI la mejora de procesos incluye la identificación de las fortalezas de su organización de procesos y debilidades y hacer cambios en el proceso de convertir debilidades en fortalezas (SEI, 2012).

1.5.7 Plataforma de desarrollo

GNU/Linux:

En los 90 la FSF[6] y su proyecto GNU, motivó a muchos programadores para promover el *software* de calidad y de distribución libre. El proyecto GNU aparte del *software* de utilidades, trabajaba en un núcleo (*kernel*) denominado HURD. Mientras, en octubre de 1991, un estudiante finlandés llamado Linus Trovalds presentó la primera versión de su *kernel* de sistema operativo, que denominó Linux. Algunas características que diferenciaron a Linux de los demás sistemas operativos de su tiempo y que siguen siendo aplicables, y otras heredadas de UNIX podrían ser:

- **Sistema operativo de código abierto.** Cualquiera puede disponer de sus fuentes, modificarlas y crear nuevas versiones que poder compartir bajo la licencia GPL.
- **Portabilidad.** Tal como el UNIX original, Linux está pensado para depender muy poco de una arquitectura concreta de máquina; consecuentemente, Linux es en su mayor parte, independiente de la máquina de destino y puede portarse a prácticamente cualquier arquitectura que disponga de un compilador C como el GNU GCC.
- **Módulos dinámicamente cargables.** Permiten poner partes del sistema operativo, como *filesystem*, o controladores de dispositivos, como pedazos externos que se cargan (o enlazan) con el *kernel* en tiempo de ejecución bajo demanda.

Los proyectos se sucedieron y –en el inicio de Linus con el *kernel*- a la gente de la FSF, con el *software* de utilidad GNU y, sobre todo, con su compilador de C (GCC), se les unieron otros proyectos importantes y acabaron por dar al *kernel* inicial Linux el recubrimiento de aplicaciones suficiente para

[6] **FSF**, Fundación para el Software Libre creada por Richard Matthew Stallman (en inglés, *Free Software Foundation).*

construir los sistemas GNU/Linux y competir en igualdad de condiciones con los sistemas propietarios (Jorba, "*et al.*", 2004).

1.6 Conclusiones parciales

Producto del análisis de antecedentes, investigaciones, herramientas y tecnologías para el desarrollo de sistemas informáticos así como tendencias actuales de los sistemas informáticos para la gestión de información de organizaciones, instituciones o empresas nacionalmente y mundialmente se concluyó que no todos los sistemas estudiados gestionan la información de la forma en la que se realiza actualmente en la UCI pero aportaron ideas novedosas para el desarrollo de la solución además se concluyó que las herramientas y tecnologías estudiadas se adaptan para lograr el completo desarrollo.

Capítulo 2. Concepción de la propuesta de solución

2.1 Introducción

En el presente capítulo se pretende obtener una descripción detallada del proceso de gestión y control de AA de la UCI así como las operaciones funcionales requeridas por este, al mismo tiempo, proponer un modelo para la solución, la organización fundamental de los componentes que forman parte de dicho modelo y las relaciones entre ellos; todo esto con el objetivo de definir y construir las bases de la solución propuesta para su implementación.

2.2 Modelado del negocio

El proceso comienza con la o las solicitudes realizadas por un estudiante (preferiblemente al concluir el primer año de la carrera), los alumnos ayudantes ratificados o las propuestas de los profesores. Después de evaluar las solicitudes y las propuestas, verificar que cumplan con los requisitos mínimos que debe tener un estudiante para ser seleccionado y según las necesidades de los departamentos docentes se aprueban oficialmente a los estudiantes seleccionados. Durante el proceso de aprobación cada estudiante deberá ser asignado a un departamento docente y se designará al profesor que atenderá su formación. Finalizada la aprobación se informa a los departamentos docentes y a la dirección de las facultades sobre estos resultados. Luego, a lo largo de todo el curso, se les da un seguimiento a estos estudiantes mientras ejercen como AA, controlando sus evaluaciones, preparación metodológica y planes de trabajos de estos.

2.2.1 Modelo de procesos del negocio

El siguiente modelo (Ver Ilustración 2. Modelo de procesos de negocio) contiene los procesos que conforman el negocio de gestión y control de alumnos ayudantes, este es la base para obtener una mayor comprensión de las operaciones realizadas dentro del mismo e identificar áreas de mejoras.

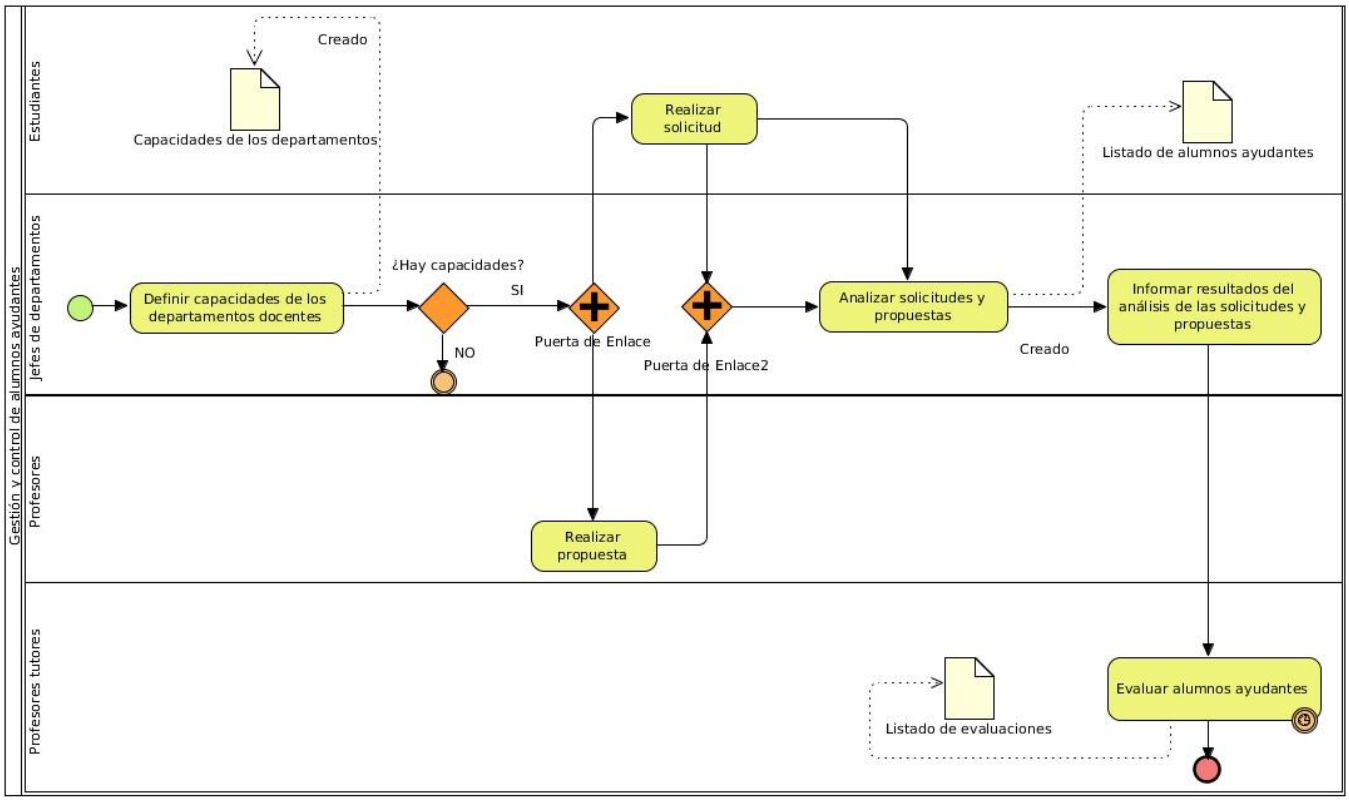


Ilustración 2. Modelo de procesos de negocio

2.2.2 Reglas del negocio

Reglas textuales:

- El estudiante debe estar matriculado para poder solicitar la ayudantía.
- El estudiante debe poseer un índice general académico igual o superior a 4 puntos.
- El estudiante debe poseer un valor de calificación final igual o superior a 4 puntos en la asignatura solicitada para la ayudantía.
- El estudiante debe haber vencido todas las asignaturas solicitadas.
- Solo será aceptada una solicitud por estudiante hasta satisfacer las necesidades existentes en los departamentos docentes de cada una de las facultades.
- Las solicitudes deberán ser realizadas en el período establecido.

Reglas de relación:

- Toda solicitud para ayudantía deberá ser realizada por un estudiante y el mismo podrá realizar tantas solicitudes como desee.
- Toda propuesta para ayudantía deberá ser realizada por un profesor y el mismo podrá realizar tantas propuestas como desee.

Reglas de derivación:

- La evaluación anual del AA se obtiene a partir los resultados alcanzados en las evaluaciones semestrales así como del criterio de su tutor y las organizaciones juveniles.
- La evaluación semestral del AA se obtiene a partir del cumplimiento del plan de trabajo y los resultados alcanzados en las esferas docente y científico-productiva así como del criterio de su tutor y las organizaciones juveniles.

2.3 Ingeniería de requisitos

Los requisitos reflejan las necesidades de los clientes de un sistema que ayudan a resolver un problema determinado. De igual forma Ian Sommerville[7] describe de la siguiente manera lo que es un requisito: "los requisitos para un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas" (Sommerville, 2005).

Sin embargo, en casi todos los sistemas los requisitos cambian por lo que es necesario mantener una constante gestión de estos. El proceso de gestionar cambios en los requisitos de un *software* se denomina gestión de requisitos. Sommerville presenta una alternativa sobre el proceso de ingeniería de requisitos, esta muestra el proceso como una actividad en cuatro etapas donde las actividades se organizan en un proceso iterativo alrededor de una espiral. En esta alternativa los requisitos se desarrollan en diferentes niveles de detalles: el análisis de si el sistema es factible para el negocio (estudio de viabilidad), el descubrimiento de requisitos (obtención y análisis); la transformación de estos en formularios estándar (especificación) y la verificación de que los requisitos realmente definen el sistema que quiere el cliente (validación). Esta alternativa permite que los requisitos y la implementación del sistema se desarrollen al mismo tiempo.

Estudio de viabilidad

Con la solución a desarrollar se pretende contribuir de manera general al proceso, descrito en acápites anteriores, facilitando el manejo de la información en cada una de las actividades pudiendo este ser implementado haciendo uso de las tecnologías y herramientas estudiadas con valores de coste y tiempo mínimos. El sistema podría integrarse a otros sistemas existentes para facilitar la obtención de información necesaria y disminuir de esta forma el esfuerzo de trabajo a la hora de desarrollar la misma.

[7] **Ian Sommerville** escritor, consultor y profesor de ingeniería del *software* en la Universidad St Andrews de Escocia.

2.3.1 Obtención y análisis de requisitos

2.3.1.1 Descubrimiento de requisitos

El descubrimiento de requisitos es el proceso de recoger información sobre el sistema propuesto y los existentes. Las fuentes de información durante esta fase incluyen la documentación, los *stakeholders*[8] del sistema y la especificación de sistemas similares. El analista se relaciona con los *stakeholders* a través de técnicas que se han desarrollado para darle soporte al descubrimiento de requisitos, entre las utilizadas están las siguientes (Sommerville, 2005).

Entrevista:

En las entrevistas se realizan preguntas a los *stakeholders* (con el fin de obtener una comprensión general de lo que hacen estos, como podrían interactuar con el sistema y las dificultades a las que se enfrentan con los sistemas actuales) sobre el sistema que utilizan y sobre el sistema a desarrollar, los requisitos provienen de las respuestas a estas preguntas. Las entrevistas pueden ser de dos tipos:

- Entrevistas cerradas donde los *stakeholders* responden a un conjunto predefinido de preguntas.
- Entrevistas abiertas donde no hay un programa predefinido.

En el **anexo B** se muestra el diseño de la entrevista, en este caso una entrevista abierta, aplicada. La misma no cuenta con un programa de preguntas predefinido, dichas preguntas surgen y tienen lugar a medida que avanza la entrevista dejando mayor libertad a la iniciativa de la persona entrevistada y del entrevistador. Desde un inicio fueron identificadas las personas a ser entrevistadas (personas que serán afectadas a todos los niveles por la solución), fueron definidos la fecha de la entrevista y el tiempo límite para la misma y por último se le dio ejecución.

Etnografía:

Es una técnica de observación que se puede utilizar para entender los requisitos sociales y organizacionales. A través de esta técnica el analista se sumerge por sí mismo en el entorno laboral donde será aplicado el sistema, observa el trabajo y anota las tareas reales en las que los participantes están involucrados. La etnografía ayuda a descubrir los requisitos implícitos que reflejan los procesos reales más que los formales en los que la gente está involucrada (Sommerville, 2005).

[8] **Stakeholders** es un término del inglés utilizado por primera vez por R. E. Freeman en su obra: "*Strategic Management: A Stakeholder Approach*" (Pitman, 1984), para referirse a *«quienes pueden afectar o son afectados por las actividades de una empresa»*.

2.3.1.2 Clasificación y organización de requisitos

Con el fin de prevenir la existencia de problemas durante el proceso de ingeniería de requisitos se realiza una clara separación de estos en diferentes niveles de descripción detallados a continuación.

Requisitos funcionales*:*

Durante al análisis fueron identificados un total de 27 requisitos funcionales.

Tabla 1. Requisitos funcionales

No.	Requisito	Prioridad
RF1	Asociar estudiante al MAA	Alta
RF2	Crear cargo de alumno ayudante	Alta
RF3	Crear evaluación anual de alumno ayudante	Alta
RF4	Crear evaluación semestral de alumno ayudante	Alta
RF5	Crear situación de alumno ayudante	Alta
RF6	Crear solicitud de ayudantía	Alta
RF7	Desvincular alumno ayudante	Alta
RF8	Modificar cargo de alumno ayudante	Alta
RF9	Modificar datos de alumno ayudante	Alta
RF10	Modificar estado de solicitud	Alta
RF11	Modificar evaluación anual	Alta
RF12	Modificar evaluación semestral	Alta
RF13	Modificar situación de alumno ayudante	Alta
RF14	Mostrar estructura del movimiento de alumnos ayudantes	Baja
RF15	Mostrar listado de alumnos ayudantes	Alta
RF16	Mostrar listado de cargos de alumno ayudante	Media
RF17	Mostrar listado de estudiantes a proponer	Alta
RF18	Mostrar listado de situación de alumno ayudante	Media
RF19	Mostrar listado de solicitudes aceptadas	Alta
RF20	Mostrar listado de solicitudes de ayudantía del estudiante	Media
RF21	Mostrar listado general de solicitudes	Alta
RF22	Mostrar registro de evaluaciones	Alta
RF23	Proponer estudiante para ayudantía	Alta
RF24	Ver detalles de alumno ayudante	Baja
RF25	Ver detalles de cargo de alumno ayudante	Baja
RF26	Ver detalles de situación de alumno ayudante	Baja
RF27	Ver detalles de solicitud de ayudantía	Baja

Requisitos no funcionales:

Tabla 2. Requisitos no funcionales

Usabilidad

- Facilidad de uso por parte de los usuarios: el sistema debe presentar una interfaz amigable que permita la fácil interacción con el mismo y llegar de manera rápida y efectiva a la información buscada. Debe, además, ser una interfaz de manejo cómodo que posibilite a los usuarios sin experiencia una rápida adaptación.
- Especificación de la terminología utilizada: el sistema debe adaptarse al lenguaje y términos utilizados por los usuarios en la rama abordada con vistas a una mayor comprensión por parte del cliente de la herramienta de trabajo.
- Potencialidades de capacitación orientadas a interfaces intuitivas: lo que enaltece la posibilidad de que el usuario aprenda mediante el uso y explotación de la herramienta.
- Emplear perfiles de usuario: diferenciar las interfaces y opciones para los usuarios que accedan al sistema según los diferentes roles que estos tengan dentro del mismo.

Seguridad

- La seguridad de la base de datos está a nivel de roles, con el fin de mantener la integridad de los datos en función del acceso de cada uno de ellos, brindando además la protección de la información.
- Políticas de seguridad por usuarios y roles: el sistema debe contar con un grupo de políticas de accesibilidad a las diferentes funcionalidades del mismo en dependencia del nivel de autorización que presente un usuario determinado.
- Registro sistemático de incidencias: el sistema debe ser capaz de registrar el accionar del usuario, así como permitir auditorías y exámenes de las trazas tanto en tiempo real como en históricos. Se precisa un monitor de incidencia para la visualización y tratamiento de las mismas.

Eficiencia

- El sistema debe soportar un tiempo de respuesta menor o igual a 5 segundos.
- El sistema debe soportar una conexión simultánea de 500 usuarios.

Soporte

- El sistema brinda como apoyo una ayuda contextual en la cual se refleja detalladamente la explicación de cada una de las pantallas con sus respectivas funcionalidades.
- Grupo de soporte y asesoría: el sistema contará con un grupo de soporte y asesoría al cliente del producto destinado a brindar asesoría y soporte técnico al mismo.

Restricciones del diseño

- Servidor de base de datos con Postgres 8.4 o superior bajo el sistema operativo Ubuntu Server

10.4.

- Servidor de aplicaciones Web: Apache 2.2 o superior.
- Navegador Web: Internet Explorer 7 o superior, Mozilla Firefox 2.3 o superior.
- Aplicación web: estilo arquitectónico cliente-servidor.

Documentación de usuarios en línea

- Manual de usuario: el sistema deberá presentar un manual de usuario, permitiendo con ello un correcto uso de sus funcionalidades y brindarle al usuario una mayor experiencia del trabajo con el mismo.
- Documentación actualizada del grupo de desarrollo: se precisa que la documentación del sistema esté actualizada en todos los aspectos, fases de trabajo y ciclos de desarrollo del mismo, permitiendo con ello un respaldo tanto ingenieril como legal del desarrollo de dicho sistema.

Interfaz

- Interfaz web: la interfaz debe ser sencilla con colores suaves a la vista y sin cúmulo de imágenes u objetos que distraigan al cliente del objetivo.

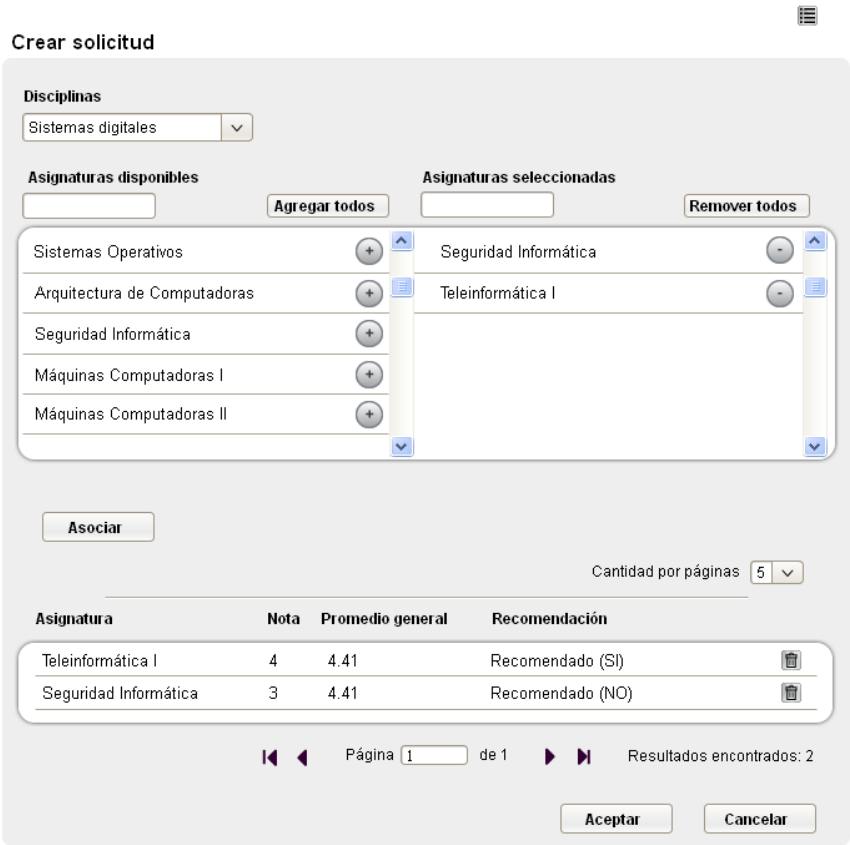
Estándares aplicables

- Se encuentran definidos en el documento de la arquitectura.

2.3.2 Especificación de requisitos

En el área de los sistemas informáticos el término especificación tiene significados diferentes para personas distintas, desde un documento escrito, modelos gráficos, modelos matemáticos, prototipos o la combinación de estos. La especificación es el producto del trabajo final que genera la ingeniería de requisitos, describe la función y el desempeño de un *software* y las restricciones que regirán su desarrollo. Y además sirve de base para las actividades de ingeniería subsecuentes, en este caso los requisitos quedan especificados a través de una plantilla donde se combinan descripciones en el lenguaje natural y modelos gráficos, quedando representados de una manera más consistente y entendible. En la siguiente tabla se muestra un ejemplo de la especificación del requisito *Crear solicitud de ayudantía,* el resto de las descripciones se pueden encontrar en el **anexo C**.

Tabla 3. Especificación del requisito Crear solicitud de ayudantía

Nº	Nombre	Descripción	Complejidad	Prioridad para cliente
RF1	Crear solicitud de ayudantía	Permite crear a un estudiante una solicitud para ejercer la ayudantía, dando la posibilidad de cancelar la acción.	Alta	Alta
	Prototipo			
				
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• selectDisciplinas • multiselectAsignaturas • gridEvaluaciones	• Varchar • Varchar • No procede	• Campo obligatorio • Campo obligatorio • Solo lectura	

	Observaciones	

Una vez especificados los requisitos funcionales es necesario crear un plan de iteraciones en el cual se establecen las iteraciones necesarias para desarrollar el producto y se define cuáles son los requisitos a desarrollar en cada una de estas.

Tabla 4. Plan de iteraciones

Iteración	Requisitos	Duración
Iteración 1	RF1 al RF11	15/01/2012 – 15/02/2012
Iteración 2	RF12 al RF19	20/02/2012 – 10/03/2012
Iteración 3	RF18 al RF27	15/03/2012 – 30/04/2012

2.3.3 Validación de requisitos

El proceso de validación de requisitos trata de mostrar que estos realmente definen el sistema que el cliente desea, implica encontrar problemas con los requisitos (Sommerville, 2005).

Durante este proceso se llevaron a cabo diferentes verificaciones sobre los requisitos.

- Verificaciones de validez: un usuario puede pensar que se necesita un sistema para llevar a cabo ciertas funciones. Sin embargo, el razonamiento y el análisis pueden identificar que se requieren funciones adicionales o diferentes. Los sistemas tienen diversos *stakeholders* con diferentes necesidades, y cualquier conjunto de requisitos es inevitablemente un compromiso en el entorno del *stakeholder*.
- Verificaciones de consistencia: los requisitos en el documento no deben contradecirse. Esto es, no debe haber restricciones o descripciones contradictorias de la misma función del sistema.
- Verificaciones de completitud: el documento de requisitos debe incluir requisitos que definan todas las funciones y restricciones propuestas por el usuario del sistema.
- Verificaciones de realismo: utilizando el conocimiento de la tecnología existente, los requisitos deben verificarse para asegurar que se pueden implementar.
- Verificabilidad: para reducir la posibilidad de discusiones entre el cliente y el contratista, los requisitos del sistema siempre deben redactarse de tal forma que sean verificables

De las diferentes técnicas de validación de requisitos existentes se hizo uso de las siguientes.

- Revisión de requisitos: comprueba consistencia, integridad, verificabilidad, comprensibilidad y rastreabilidad. Proceso manual que involucra a varios lectores, tanto del cliente como del

contratista, puede ser formal o informal, en la revisión formal el equipo de desarrollo conduce al cliente a través de los requisitos, explicándole las implicaciones de cada uno.

- Construcción de prototipos: en este enfoque de validación, se muestra un modelo ejecutable del sistema a los usuarios finales y a los clientes. Estos pueden experimentar con los modelos para ver si cumplen sus necesidades reales, los cuales puede encontrar en el **anexo C**.
- Generación de casos de prueba: los requisitos deben poder probarse. Si las pruebas para estos se conciben como parte del proceso de validación, a menudo revela los problemas en los requisitos. Si una prueba es difícil o imposible de diseñar, normalmente, significa que los requisitos serán difíciles de implementar y deberían ser considerados nuevamente. Los mismos los puede encontrar en el **anexo D**.

Las técnicas anteriormente descritas fueron aplicadas con el fin de mostrar que los requisitos planteados definen el sistema que el cliente desea, además de aumentar la calidad de las especificaciones de estos. Antes del proceso de creación del código fuente y a fin de obtener modelos ejecutables para mostrarlos a los clientes fueron creados prototipos de interfaz; además para verificar la completitud de los requisitos fueron realizadas detalladas revisiones a los artefactos 0132_Evaluación_de_requisitos y 0113_Especifricación_de_requisitos_de_software. Una vez concluida la codificación fueron diseñados casos de pruebas para cada uno de los requisitos especificados permitiendo verificar que todos se pudieran probar; mediante el uso del criterio de particiones de equivalencia a través del cual son divididos los campos de entrada en clases de datos de los cuales se derivan los casos de prueba. En otras palabras, con el diseño de casos de pruebas fue dividido el dominio de entrada de la solución en un número finito de clases de equivalencia, representando un conjunto de estados válidos y no válidos.

2.4 Diseño y arquitectura de la solución propuesta

Durante el ciclo de vida del *software*, y a fin de garantizar diversos factores de calidad, se produce un modelo del mismo procedente de los requisitos, donde, para la creación y evolución de dicho modelo se define una organización fundamental de sus componentes y las relaciones entre ellos. A grandes rasgos esta es la relación existente entre el diseño y la arquitectura de un *software*, sin ánimo de que lo anteriormente mencionado represente una definición más pues la bibliografía sobre el tema es tan extensa como la cantidad de conceptos relacionados que se pueden encontrar.

Teniendo en cuenta que la propuesta de solución a desarrollar está integrada al SGAP del SGU, la misma deberá cumplir con todas las pautas especificadas para el desarrollo de dicho sistema.

2.4.1 Estilos y patrones

Los diseñadores de *software* van creando un amplio repertorio de principios y expresiones que los guían a desarrollar el *software*, a unas y a otras se les puede asignar el nombre de patrones siempre y cuando se codifiquen en un formato estructurado que describa el problema y su solución. Por lo que se puede afirmar que un patrón es una descripción de un problema y su solución que recibe un nombre y que puede emplearse en otros contextos; en teoría, indica la manera de utilizarlo en circunstancias adversas (Larman, 1999). Los patrones no se proponen descubrir ni expresar nuevos principios de la ingeniería de *software*, solo intenta codificar el conocimiento, las expresiones y los principios ya existentes. Los patrones son soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos, son soluciones basadas en la experiencia y que se han demostrado que funcionan. Pero existe una diferencia notable entre estilo arquitectónico, patrón arquitectónico y patrón de diseño aunque se relacionan entre sí formando parte de un todo, estos son los que en gran medida aportan calidad al sistema resultante. El estilo arquitectónico, patrón arquitectónico y patrón de diseño, representan, de lo general a lo particular, los niveles de abstracción que componen la arquitectura de *software*.

El **estilo arquitectónico** es el encargado de:

- Describir la estructura general de un sistema, independientemente de otros estilos.
- Definir los componentes del sistema, su relación e interactividad.

El **patrón arquitectónico** es el nivel en el cual la arquitectura de *software*:

- Define la estructura básica de un sistema, pudiendo estar relacionada con otros patrones.
- Representa una plantilla de construcción que provee un conjunto de subsistemas aportando las normas para su organización.

El **patrón de diseño** es el tercer nivel de abstracción de la arquitectura de *software*, cuya finalidad es la de precisar en detalles los subsistemas y componentes de la aplicación (Bahit, 2011).

Habiendo marcado la diferencia entre estos tres conceptos que conforman los niveles de abstracción de la arquitectura de *software*, están creadas las condiciones de abordar los detalles de los niveles de la arquitectura definidos a utilizar en el desarrollo del SGU, los cuales serán aplicados a la propuesta de solución.

2.4.1.1 Estilo o modalidad arquitectónica

Cliente-Servidor: en el estilo Cliente-Servidor el cliente envía un mensaje a un servidor solicitando un determinado servicio (petición) y este envía uno o varios mensajes de respuesta (provee el servicio) (Ver Ilustración 3. Interacción Cliente-Servidor). Usualmente la mayoría del trabajo pesado lo hace el

servidor mientras que los procesos del cliente solo se ocupan de la interacción con el usuario a través de interfaces gráficas; esto permite distribuir físicamente los procesos y los datos en forma más eficiente reduciendo grandemente el tráfico de la red.

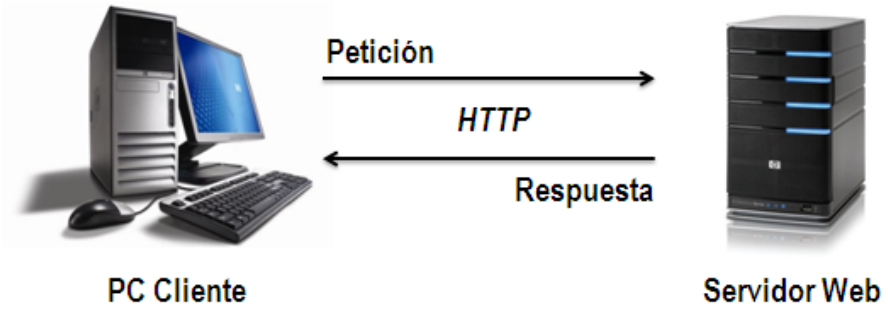


Ilustración 3. Interacción Cliente-Servidor

2.4.1.2 Patrón arquitectónico

Modelo-Vista-Controlador: el patrón MVC (Ver Ilustración 4. Funcionamiento del patrón Modelo-Vista-Controlador) es un patrón de arquitectura de *software* encargado de separar la lógica del negocio de la interfaz del usuario y es el más utilizado en aplicaciones web, ya que facilita la funcionalidad, mantenibilidad y escalabilidad del sistema, de forma simple y sencilla, a la vez que permite "no mezclar lenguajes de programación en el mismo código". El patrón MVC divide la aplicación en tres niveles de abstracción (Bahit, 2011).

Modelo: representa la lógica del negocio, encargado de acceder a los datos actuando como intermediario con la base de datos.

Vista: es la encargada de mostrar la información al usuario de forma legible.

Controlador: sirve como intermediario entre el modelo, la vista y cualquier otro recurso necesario para procesar la solicitud HTTP y generar una página web.

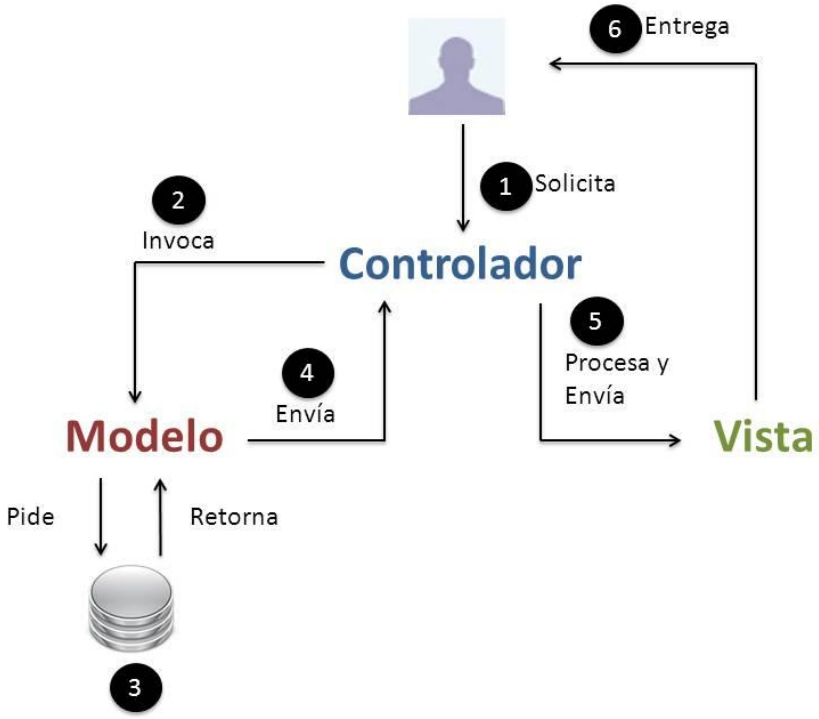


Ilustración 4. Funcionamiento del patrón Modelo-Vista-Controlador

Una vez descrito el funcionamiento del patrón Modelo-Vista-Controlador se pasará a explicar como queda evidenciada su utilización dentro del marco de trabajo sobre el cual se desarrolla la solución. El marco de trabajo cuenta, en su interior, con carpetas definidas para cada uno de los subsistemas (Ver Ilustración 5. MVC en GUUD) que forman parte del *software* a desarrollar. Al mismo tiempo, cada uno de estos subsistemas, teniendo en cuenta que el marco de trabajo cuenta con la característica de modularidad, define carpetas para cada uno de sus módulos (Ver Ilustración 6. MVC en GUUD) dentro de las cuales queda evidenciado el uso del patrón de la siguiente manera: la carpeta *views* donde serán almacenadas las páginas o fragmentos de páginas web que mostrarán la información al usuario, la carpeta *controllers* donde se encuentran los archivos que son los encargados de procesar la información y cargar cualquier recurso necesario para procesar la solicitud HTTP, la carpeta *models* donde se encuentran los archivos que contienen una serie de funciones que permiten devolver, insertar y actualizar la información de la base de datos y por último la carpeta *libraries* en la cual están los archivos que sirven de intermediarios entre el controlador y el modelo, además de ser usadas para obtener información de los restantes módulos. Esta última como una adaptación realizada para disminuir la cantidad de responsabilidades en el controlador.

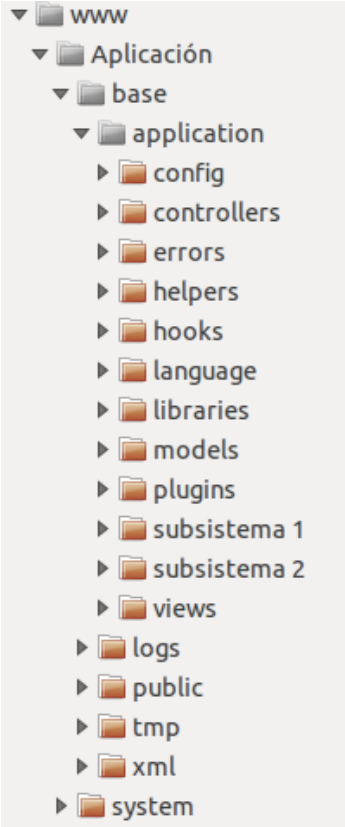


Ilustración 5. MVC en GUUD

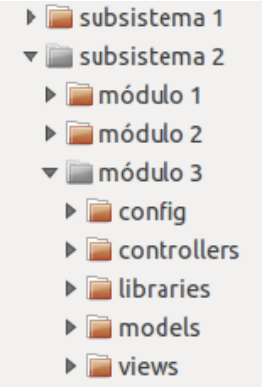


Ilustración 6. MVC en GUUD

2.4.1.3 Patrones de diseño

Patrones que expresan esquemas para definir estructuras de diseño (o sus relaciones) con las que construir sistemas de *software*.

Los patrones GoF (*Gang Of Four*), describen las formas comunes en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros. Tratan la relación entre clases, la combinación de clases y la formación de estructuras de mayor complejidad. Permiten crear grupos de objetos para ayudar a realizar tareas complejas. Existen tres clasificaciones de patrones GoF: de creación, estructurales y de comportamiento.

Los patrones de creación abstraen la forma en la que se crean los objetos, permitiendo tratar las clases a crear de forma genérica dejando para más tarde la decisión de qué clases crear o cómo crearlas.

- Instancia única *(Singleton):* garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia.

Los patrones de comportamiento estudian las relaciones de llamadas entre los diferentes objetos, normalmente, ligados con la dimensión temporal.

- Mediador *(Mediator)*: define un objeto que coordine la comunicación entre objetos de distintas clases, pero que funcionan como un conjunto.
- Observador *(Observer)*: define una dependencia de uno-a-muchos entre objetos, de forma que cuando un objeto cambie de estado se notifique y actualicen automáticamente todos los objetos que dependen de él.

Los patrones de asignación de responsabilidades «GRASP», los cuales dan la medida de un refinamiento del diseño.

- Experto *(Expert)*: asignar una responsabilidad al experto en información, la clase que cuenta con la información necesaria para cumplir la responsabilidad.
- Creador *(Creator)*: asignarle a la clase B la responsabilidad de crear una instancia de clase A.
- Controlador *(Controller)*: asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase.
- Bajo acoplamiento *(Low Coupling)*: asignar una responsabilidad para mantener bajo acoplamiento. El grado de acoplamiento no puede considerarse aisladamente de otros principios como Experto y Alta Cohesión. Sin embargo, es un factor a considerar cuando se intente mejorar el diseño.
- Alta cohesión *(High Cohesion)*: asignar una responsabilidad, de modo que la cohesión siga siendo alta (García, 2011).

¿Cómo se evidencia la aplicación de estos patrones de diseño en la aplicación?

Aplicación de patrones GOF:

- Instancia única: todas las clases con la categoría de controlador son instancias únicas como lo es también la *IOC* para la interacción entre módulos.
- Mediador: todas las clases con la categoría de librería funcionan como mediadoras entre las clases controladoras y modelo o acceso a datos.
- Observador: en la clase *loader*, objeto *load* dentro de las clases controladoras, la misma es la responsable de cargar los elementos del marco de trabajo como son librerías, modelos, vistas y de actualizar a la controladora instanciada.

Aplicación de patrones GRASP:

- Experto: se evidencia en las clases con categoría de librería, que son las que cuentan con la información necesaria para cumplir las responsabilidades sobre los elementos de negocio.

- Creador: en la clase *loader*, objeto *load* dentro de las clases controladoras, la misma se encarga de cargar los elementos del marco de trabajo como son librerías, modelos y vistas. Además, en el módulo seguridad, en la librería fabrica_ma_lib, que se encarga de crear los objetos de los modos de autenticación (ma).
- Controlador: las clases con la categoría de controlador son las encargadas de obtener datos y enviar estos hacia las librerías y vistas.
- Bajo acoplamiento y alta cohesión: la propia implementación de CodeIgniter contiene ambos patrones nivelados, pues permiten el uso de los componentes de forma individual, evidenciando el bajo acoplamiento y la dependencia entre estos o alta cohesión.

2.4.1.4 Patrones de base de datos

Entre los patrones de diseño de bases de datos existentes fueron utilizados los siguientes:

- Llaves subrogadas: plantea que se genere una llave primaria única para cada entidad, en vez de usar un atributo identificador en el contexto dado.

Aplicación:

Cada una de las entidades del modelo de datos que conforma la solución posee una llave primaria.

2.4.2 Modelo de datos

El modelo de datos (Ver Ilustración 7. Modelo de datos) es el lenguaje mediante el cual queda descrita la base de datos especificando la estructura de los datos a almacenar (tipos, restricciones, operaciones de manipulación y relaciones entre estos) (Pressman, 2002), el mismo cuenta con un total de 8 tablas persistentes que almacenan toda la información del negocio.

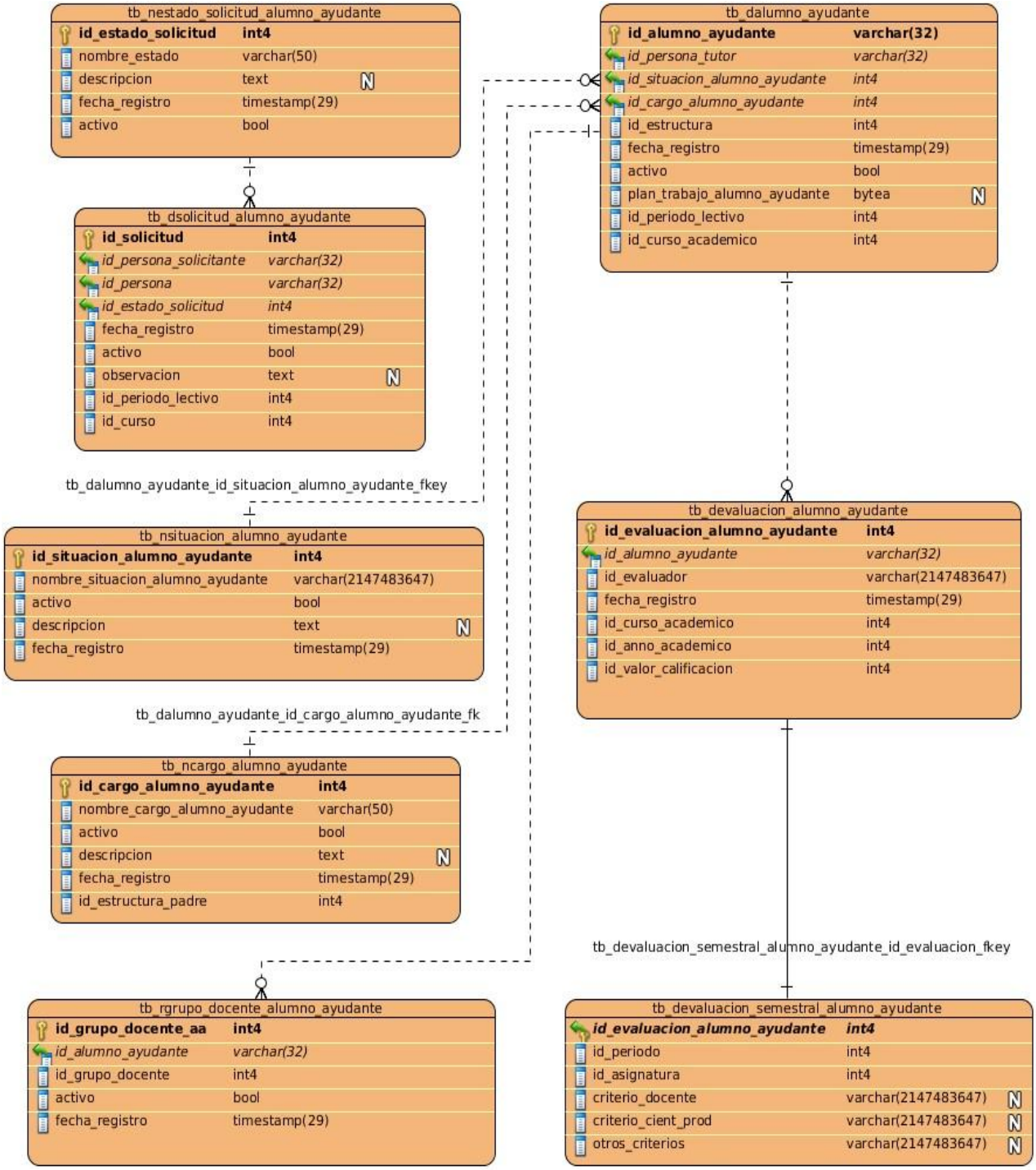


Ilustración 7. Modelo de datos

2.4.3 Modelo de despliegue

El modelo de despliegue (Ver Ilustración 8. Modelo de despliegue) es un modelo de objetos que describe la distribución física del sistema en función de cómo se distribuye la funcionalidad entre los nodos de cómputo (Booch, *"et al."*, 2000).

- El nodo cliente es el encargado de convertir las peticiones del usuario en peticiones del sistema.
- El servidor de aplicaciones web se encarga de procesar todas las peticiones HTTP.
- El servidor de base de datos es el encargado de obtener y modificar la información.
- La impresora es un dispositivo mediante el cual el usuario puede extraer la información gestionada por el sistema.

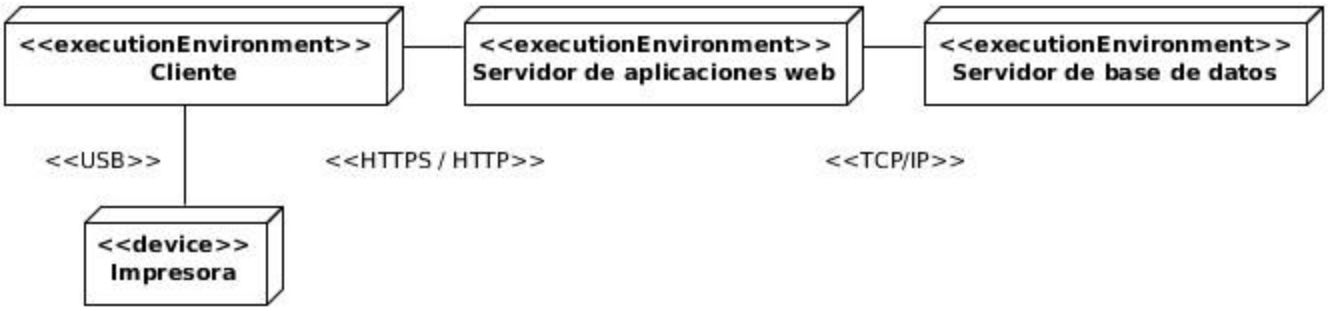


Ilustración 8. Modelo de despliegue

2.5 Conclusiones parciales

Con la identificación del proceso para la gestión de la información del MAA en la UCI, así como el estudio de conceptos asociados al tema y el posterior establecimiento de las pautas del diseño y la arquitectura, se comprobó que quedaron establecidas las bases para comenzar el proceso de codificación de la solución.

Capítulo 3. Implementación y evaluación de la propuesta de solución

3.1 Introducción

Una vez concluidos el análisis y el diseño correspondientes al ciclo de desarrollo de la solución, se dispone de suficientes condiciones para generar el código fuente, obtener una solución funcional y evaluar la misma. En muchos aspectos, la evaluación de un *software* es un proceso tanto individualista como solidario, y el número de tipos de pruebas varía tanto como el número de enfoques que se le pueda dar al desarrollo. Por un largo tiempo la única defensa contra los errores cometidos en la elaboración de *software* fue la capacidad de ser cuidadosos a la hora de realizar diseños y el nivel de inteligencia del cual disponían los desarrolladores. En estos tiempos el hombre se encuentra en una era, dentro del desarrollo de *software*, en la que se imponen modernas técnicas que ayudan a reducir y mitigar defectos; de manera similar comienzan a surgir variadas filosofías y enfoques diferentes de pruebas.

3.2 Estándares de codificación

¿Por qué es necesario utilizar estándares de codificación?

La utilización de estándares de codificación en un proyecto permite que todos los desarrolladores e involucrados lo puedan entender en menor tiempo, estos estándares les dicen a los desarrolladores como deben escribir el código, asegurando que un gran proyecto sea codificado en un estilo coherente, las partes no se escriben de manera diferente por diferentes programadores.

Los estándares de codificación definidos (García, 2011), para la creación del código fuente de la solución, se describen a continuación.

Identación, llaves de apertura y cierre y tamaño de líneas

Usar una identación sin tabulaciones, con un equivalente a 4 espacios, para mantener integridad en las revisiones svn. El uso de las llaves “{}” será en una nueva línea. La longitud de las líneas de código es aproximadamente de 75-80 caracteres.

Ejemplo:

```
1....$a = $b;
2
3....function ejemplo ()
4.... {
5....//BI
6.... }
```

Convención de nomenclatura

Variables: se rigen por la nomenclatura camelCase. Siempre comienzan por minúscula y en caso de nombres compuestos la primera letra de cada palabra comienza con mayúscula.

Ejemplo:

```
1....$variable
2....$variableNombreCompuesto
```

Constantes: constantes siempre debe ser todo en mayúsculas, con caracteres de subrayado "_" para separar palabras en caso de nombres compuestos.

Ejemplo:

```
1....define (CONSTANTE, valor);
2....define (CONSTANTE_COMPUESTO, valor);
```

Clases: siempre comienzan con mayúscula, en caso de nombres compuestos las palabras se separan con el carácter subrayado "_" y el resto en minúscula.

Ejemplo:

```
1....class Clase
2.... {
3....//BI
4....}
5
6....class Clase_nombre_compuesto
7.... {
8....//BI
9....}
```

Funciones: se rigen por la nomenclatura camelCase. Siempre comienzan con minúscula y en caso de nombres compuestos la primera letra de cada palabra comienza con mayúscula. Los parámetros son separados por espacio luego de la coma que los separa.

Ejemplo:

```
1....function funcion ($parametro1,$parametro2)
2.... {
3....//BI
4....}
5
6....function funcionNombreCompuesto ($parametro1,$parametro2)
7.... {
8....//BI
9....}
```

Ficheros: todo siempre en minúscula y en caso de nombres compuestos se usa el carácter subrayado "_".

- Vistas: intuitivo y relacionado con el formulario y/o vista que representa.
- Modelos: con el mismo nombre de la clase que representa que contiene en el nombre el sufijo _mdl o _base en caso de ser modelos base.

- Librerías: con el mismo nombre de la clase que representa que contiene en el nombre el sufijo _lib.
- Controladoras: con el mismo nombre de la clase que representa.

Estructuras de control

Se incluye *if*, *for*, *foreach*, *while* y *switch* entre las estructuras de control. Se recomienda utilizar siempre llaves de apertura y cierre, incluso en situaciones en las que técnicamente son opcionales. Esto aumenta la legibilidad y disminuye la probabilidad de errores lógicos.

Ejemplo:

```
1....if.(condicion)
2.... {
3....//BI
4....}
5....elseif (condicion)
6.... {
7....//BI
8....}
9....else
10.... {
11....//BI
12....}
13
14....switch.(valor)
15.... {
16....case valor1:
17....//BI para valor1
18....break;
19....case valor2:
20....//BI para valor2
21....break;
22....default:
23....//BI por defecto
24....}
```

Si las condiciones son muy largas que sobrepasan el tamaño de la línea, estas se dividen en varias líneas.

Ejemplo:

```
1....if.(condicion1
2....|| condicion2)
3....||(condicion3
4....&& condicion4))
5.... {
6....//BI
7....}
```

En el mejor de los casos cuando la condición es muy extensa, se puede dividir esta en variables y compararlas dentro de la estructura de control.

Ejemplo:

```
1....$variableCondicion1 = condicion1 || condicion2;
2....$variableCondicion2 = condicion3 &&condicion4;
3
4....if ($variableCondicion1 || $variableCondicion2)
5.... {
6....//BI
7....}
```

Documentación

Todos los archivos deben tener la documentación asociada al mismo. Para esto debe de cumplir con el siguiente bloque al principio de cada clase.

Clase:

```
1 /**
2 *Breve descripción de la clase
3 *
4 *PHP versión #
5 *
6 *@category Categoría de la clase implementada "Librería,
7 * Controladora, Modelo"
8 *@package Nombre del paquete o módulo al que pertenece
9 *@author Nombre y Apellidos del autor y correo electrónico
10*/
```

Funciones:

```
1 /**
2 *Breve descripción de la función
3 *
4 *@param tipo y nombre del parámetro(por cada parámetro que
5 * recibe la función)
6 *@return tipo que retorna
7 *@author Nombre y Apellidos del autor y correo electrónico
8 */
```

Buenas prácticas

Los valores booleanos y nulos siempre se escriben con mayúscula, para facilitar la legibilidad del código, usar un "enter" antes de las estructuras de control y definición de las funciones.

```
1....$variableBooleana = FALSE;
2....$variableNula = NULL;
3....
4....if (condicion)
5.... {
6....//BI
7....}
```

Para adaptar todo lo anteriormente descrito a la construcción de la propuesta y con la cual se dará solución al problema de investigación, se hará uso de la Programación Orientada a Objetos. Esta como paradigma de programación combina e incorpora técnicas diferentes y está compuesta por elementos que favorecerán y agilizarán el desarrollo como son: clases, objetos, métodos, mensajes, propiedades y atributos. Posee además características que la identifican como son el aislamiento de un elemento de su contexto (abstracción); reúne al mismo nivel de abstracción a todos los elementos que puedan considerarse pertenecientes a una misma entidad (encapsulación); permite dividir una aplicación en varias partes más pequeñas independientes unas de otras (modularidad); los objetos están aislados del exterior protegiendo a sus propiedades para no ser modificadas por aquellos que no tengan derecho a acceder a las mismas (ocultación); brinda la capacidad a diferentes objetos de contar con métodos, propiedades y atributos de igual nombre, sin que los de un objeto interfieran con el de otro (polimorfismo); da la posibilidad de relacionar clases donde una es la principal y otras son secundarias y dependen de ellas; donde a la vez, los objetos obtienen las características de los otros objetos de los cuales heredan (herencia) y permite destruirlos cuando ya no son necesarios (recolección de basura) (Bahit, 2011).

3.3 Fundamentos de la construcción de la propuesta de solución

La fase de implementación no es un paso fácil en el ciclo de desarrollo de la propuesta de solución, durante esta todavía pueden tomarse muchas decisiones y efectuarse cambios en el diseño; pero, en teoría, la arquitectura global y las decisiones más importantes fueron establecidas totalmente antes del comienzo de esta fase. Por todos los riesgos que implicarían una deficiente elaboración del código se ha trabajado de forma excepcional poniendo en práctica todo el conocimiento del que se dispone a fin de minimizar la complejidad. Facilitando al usuario una mejor manipulación y uso de estructuras complejas y brindando gran cantidad de información que el mismo no puede almacenar en su memoria operativa. Anticipándose a los cambios, pues el *software* puede cambiar con el paso del tiempo y construyendo para verificar mediante el uso de estándares de codificación que permitan las futuras revisiones del código.

3.4 Descripción de la solución

Como resultado de todo el proceso de implementación se obtuvo una agrupación funcional que cuenta con las principales funcionalidades (a las cuales se accede de forma restringida debido a los niveles de privilegios definidos para cada usuario) que permiten gestionar la información del proceso de captación y seguimiento de los alumnos ayudantes en la UCI. La misma permite la gestión de las solicitudes, los

datos de alumnos ayudantes y otras informaciones asociadas a estos; para ello se hace uso de buscadores que facilitan y agilizan la búsqueda a través de filtros, formularios de envío de datos y un componente para visualizar estructuras.

A continuación se describen cada una de las funcionalidades que conforman la solución:

(En las descripciones se hace referencia al término usuario, refiriéndose al usuario que interactúa con el sistema, el cual varía según los niveles de privilegios que les han asignado).

Crear solicitud para ayudantía:

El usuario tiene la posibilidad de realizar solicitudes para ejercer la ayudantía, haciendo uso, para ello, de un formulario de envío en el que podrá seleccionar la o las asignaturas de su preferencia realizando un filtrado por disciplinas, para una mejor organización; de las asignaturas seleccionadas se mostrará la evaluación de esta y si se considera que pueda ejercer la ayudantía para esta (dependiendo de la evaluación y el promedio general, ambas deben ser igual o superior a 4 puntos). A través de dicha interfaz el sistema le comunicará si puede ser realizada la solicitud para la o las asignaturas elegidas. El usuario finalmente podrá realizar el envío de las solicitudes o cancelarlo.

Mostrar listado de solicitudes al estudiante:

El usuario tendrá acceso a ver las solicitudes realizadas por él y comprobar el estado en el que se encuentran las mismas.

Proponer estudiantes:

A través del uso de un buscador el usuario podrá realizar una búsqueda sobre todos los estudiantes matriculados en el centro y filtrar la misma por varios parámetros hasta encontrar al estudiante o los estudiantes deseados, seleccionarlos y enviar la propuesta o cancelarla si así lo desea.

Mostrar registro general de solicitudes realizadas:

A través del uso de un buscador el usuario podrá realizar una búsqueda sobre todas las solicitudes realizadas y filtrar la misma por varios parámetros. Las solicitudes mostradas contarán con dos opciones, una para acceder a los detalles de la misma y otra para modificar el estado en el que se encuentra.

Detalles de la solicitud:

Muestra todos los detalles de la solicitud.

Modificar estado de la solicitud:

Brinda la posibilidad al usuario de modificar el estado en el que se encuentra la solicitud. Ya sea para cancelarla, aceptarla, cerrarla, rechazarla o mantenerla pendiente. Una vez especificados los datos, el usuario puede realizar la modificación o cancelar la misma.

Mostrar listado de solicitudes aceptadas:

Muestra un listado con todas las solicitudes que fueron aceptadas, brindando la posibilidad de filtrar la búsqueda a través de varios parámetros. Las solicitudes mostradas contarán con una opción que le permitirá al usuario asociar al estudiante que la realizó al Movimiento de Alumnos Ayudantes.

Asociar estudiante al movimiento de alumnos ayudantes:

Esta funcionalidad da la posibilidad de asociar a un estudiarte con todos los datos necesarios para que el mismo pueda pasar a formar parte del movimiento; en el formulario de envío se relacionarán datos como son el tipo de curso en el que el estudiante ejercerá la ayudantía, el profesor que estará a cargo de dicho estudiante, el cargo que ejercerá dentro del movimiento en caso de ser necesario, los grupos que le serán asignados, se especifica si estará activo o no y por último el plan de trabajo que deberá cumplir. Una vez especificados los datos el usuario puede asociar al estudiante o cancelar la acción.

Mostrar registro general de alumnos ayudantes:

A través del uso de un buscador el usuario podrá realizar una búsqueda sobre todos los alumnos ayudantes y filtrar la misma por varios parámetros. Los alumnos ayudantes mostrados contarán con una serie de opciones las cuales se mencionan a continuación. Las opciones a mostrar son: activar o desactivar, crear evaluación semestral, crear evaluación anual, mostrar detalles, listar evaluaciones, modificar datos y desvincular.

Activar o desactivar*:*

Permite al usuario modificar el estado del alumno ayudante entre activo o inactivo.

Crear evaluación semestral:

Permite al usuario registrar la evaluación semestral al alumno ayudante, permitiendo registrar además de la evaluación cualitativa, criterios sobre el desempeño del alumno en las esferas docente, científico-productiva y otras consideraciones que crea necesarias. Una vez especificados los datos, el usuario puede crear la evaluación o cancelar la acción.

Crear evaluación anual:

Permite al usuario registrar la evaluación anual teniendo en cuenta las evaluaciones semestrales de ese curso académico, en su desempeño como alumno ayudante. Una vez especificados los datos el usuario puede crear la evaluación o cancelar la acción.

Mostrar registro de evaluaciones:

Se muestra el registro de todas las evaluaciones del estudiante durante su desempeño como alumno ayudante, evaluaciones anuales y semestrales, permitiendo al usuario ver los criterios emitidos en cada una de estas y permitiendo realizar modificaciones.

Modificar evaluación semestral:

Permite al usuario modificar la evaluación semestral del alumno ayudante así como los criterios emitidos. Una vez especificados los datos el usuario puede realizar la modificación o cancelar la misma.

Modificar evaluación anual:

Permite al usuario modificar la evaluación anual del alumno ayudante. Una vez especificados los datos el usuario puede realizar la modificación o cancelar la misma.

Detalles del alumno ayudante:

Se muestran los datos relacionados con el alumno ayudante: datos del tutor, departamento al que pertenece y los grupos que tiene asociados.

Modificar datos del alumno ayudante:

Se muestra un formulario en el cual aparecen los datos actuales del alumno ayudante, los cuales fueron especificados cuando se vinculó al MAA permitiendo su modificación, además de dar la posibilidad de realizar un cambio de asignatura o de departamento para el alumno. Una vez especificados los datos el usuario puede realizar la modificación o cancelar la misma.

Desvincular al alumno ayudante:

Da la posibilidad de desvincular al alumno ayudante, una vez desvinculado, el estudiante no podrá volver a formar parte del movimiento.

Mostrar listado de cargos de alumno ayudante:

Muestra un listado con todos los cargos de los alumnos ayudantes configurados en el sistema brindando las opciones de ver detalles y modificar.

Crear cargo:

Muestra un formulario de envío en el cual se especifican el nombre, la estructura de la cual depende, una breve descripción y si el cargo estará activo en el sistema. Una vez especificados los datos el usuario puede crear el cargo o cancelar la acción.

Detalles de cargo:

Muestra los detalles del cargo de alumno ayudante.

Modificar cargo:

Permite al usuario modificar los datos del cargo. Una vez especificados los datos el usuario puede realizar la modificación o cancelar la misma.

Mostrar listado de situación de alumno ayudante:

Muestra un listado con todas las situaciones de alumno ayudante configuradas en el sistema brindando las opciones de ver detalles y modificar.

Crear situación:

Muestra un formulario de envío en el cual se especifican el nombre, una breve descripción y si la situación estará activa en el sistema. Una vez especificados los datos el usuario puede crear la situación o cancelar la acción.

Detalles de situación:

Muestra los detalles de la situación de alumno ayudante.

Modificar situación:

Permite al usuario modificar los datos de la situación. Una vez especificados los datos el usuario puede realizar la modificación o cancelar la misma.

Mostrar estructura del movimiento de alumnos ayudantes:

Muestra en un árbol jerárquico la estructura del MAA de manera general en la universidad y de forma específica en cada una de las facultades, teniendo en cuenta los datos especificados para cada uno de los estudiantes vinculados al movimiento.

A dicha agrupación se pueden incorporar funcionalidades en la medida que lo exijan las necesidades del MAA y la universidad.

3.5 Integración de la solución al Sistema de Gestión Universitaria

La propuesta de solución, como parte del módulo de Personal y Secretaría del SGAP, se encuentra integrada al SGU facilitando así el manejo de gran parte de la información necesaria para realizar una buena gestión dentro del negocio y la utilización de varios componentes y funcionalidades implementadas en los diferentes módulos que conforman el SGAP.

La integración de la propuesta de solución con las funcionalidades horizontales del núcleo del sistema y los diferentes módulos del subsistema se gestiona a través de la variable IOC del núcleo, con la cual se puede acceder a todas las funcionalidades de los diferentes módulos, compartidas en el fichero *ioc.xml*. De esta forma, se evitan las redundancias en el código.

3.5.1 Factores positivos que trae consigo la integración con los módulos del Sistema de Gestión Universitaria

El núcleo es el encargado de gestionar todas las funcionalidades horizontales del SGU permitiendo estandarizar las mismas para ser usadas por cada uno de los subsistemas, esto beneficia en gran medida al negocio de la siguiente manera:

- La integración con el módulo de seguridad garantiza que el acceso a la información sea restringido debido a los niveles de privilegios definidos para cada usuario, a través de grupos de usuarios y roles.

- La integración con el módulo de trazas permite el acceso a todas las incidencias de un usuario sobre el sistema brindando acceso a la información referente a: usuario, acción realizada, fecha y hora de la ejecución.
- La integración con el módulo de estructura y composición garantiza el acceso a toda la estructura administrativa definida por la universidad.
- La integración con el módulo de personal permite el acceso a la información de todas las personas registradas en el sistema.

En cuanto a la integración con los módulos del propio SGAP, la propuesta de solución se beneficia de la siguiente forma:

- La integración con el módulo de carrera brinda la posibilidad de consultar toda la información referente a tipos de cursos, marcos lectivos, disciplinas, asignaturas, entre otros.
- La integración con el módulo de control docente permite la consulta de información referente a grupos docentes, profesores y estudiantes vinculados a estos y sus evaluaciones.

Al mismo tiempo se brinda información, de los estudiantes vinculados al MAA, al subsistema de caracterización integral de los estudiantes (SCIE) para realizar las caracterizaciones de los mismos.

3.6 Fundamentación de la evaluación del resultado

La construcción de la propuesta tiene como objetivo satisfacer las necesidades y cumplir con las expectativas del cliente, pero, ¿cómo estar seguros si el producto desarrollado se corresponde con lo solicitado por el cliente? o ¿cómo estar seguros de que el producto funcionará correctamente?

Para ello se hace necesario llevar a cabo, en paralelo al proceso de desarrollo, un proceso de evaluación o comprobación del producto que se va generando; con el fin de entregar a los clientes un resultado satisfactorio garantizando así la funcionalidad, fiabilidad, eficiencia, usabilidad, mantenibilidad y portabilidad del mismo. Controlar y corregir las faltas y fallos existentes en el producto afecta positivamente la funcionalidad y fiabilidad como atributos de calidad.

Antes de continuar es necesario saber distinguir entre cuatro conceptos para describir funcionamientos incorrectos, relacionados entre sí pero distintos (esta terminología se define detalladamente en el estándar IEEE 610.12-1990):

- Error: es la acción humana que produce una falta.
- Falta: es algo que está mal en un producto (código, documento, etc.).
- Fallo: manifestación de una falta.
- Defecto: error, falta o fallo (IEEE, 1990).

3.7 Aspectos importantes de la evaluación

Durante el desarrollo de un *software* las distintas técnicas de evaluación son la principal estrategia para detectar faltas y fallos. En términos generales se pueden distinguir dos tipos de evaluaciones durante el proceso de desarrollo: verificación y validaciones y según el IEEE[9] Std 729-1983 definidas como:

- Verificación: proceso de determinar si los productos de una determinada fase del desarrollo de *software* cumplen o no los requisitos establecidos durante la fase anterior.
- Validación: proceso de evaluación del *software* al final del proceso de desarrollo para asegurar el cumplimiento de las necesidades del cliente.

De esta forma se puede apreciar que la verificación se encarga de comprobar si se ha construido el producto correctamente, mientras que la validación ayuda a comprobar si se ha construido el producto correcto. Tanto para realizar verificaciones como validaciones se pueden utilizar distintos tipos de técnicas estas últimas agrupadas en dos categorías:

- Técnicas de evaluación estáticas: buscan faltas sobre el sistema en reposo. Es decir, estudian los distintos modelos que componen el sistema buscando posibles faltas en los mismos. Así pues, estas técnicas se pueden aplicar, tanto a requisitos como a modelos de análisis, diseño y código.
- Técnicas de evaluación dinámica: generan entradas al sistema con el objetivo de detectar fallos, al ejecutar el sistema sobre esas entradas. Es decir, se pone el sistema a funcionar buscando posibles incongruencias entre la salida esperada y la salida real (Juristo, *"et al."*, 2005).

La aplicación de técnicas dinámicas es también conocida como pruebas del *software* o *testing* y se aplican generalmente sobre el código, las interfaces y el sistema en general.

¿Cómo se realiza la evaluación?

Las técnicas de evaluación estática se aplican a medida que se van generando los distintos productos del desarrollo siguiendo una filosofía *top-down*. Es decir, la evaluación estática acompaña a las actividades de desarrollo, a diferencia de la evaluación dinámica que únicamente puede comenzar una vez finalizada la codificación, siguiendo esta última una estrategia *botom-up*. La siguiente imagen muestra de forma detallada la aplicación de las técnicas estáticas y dinámicas para evaluar un *software* (Juristo, *"et al."*, 2005).

[9] **IEEE**. Instituto para Ingenieros Eléctricos y Electrónicos (en inglés, *Institute for Electrical and Electronics Engineers)*

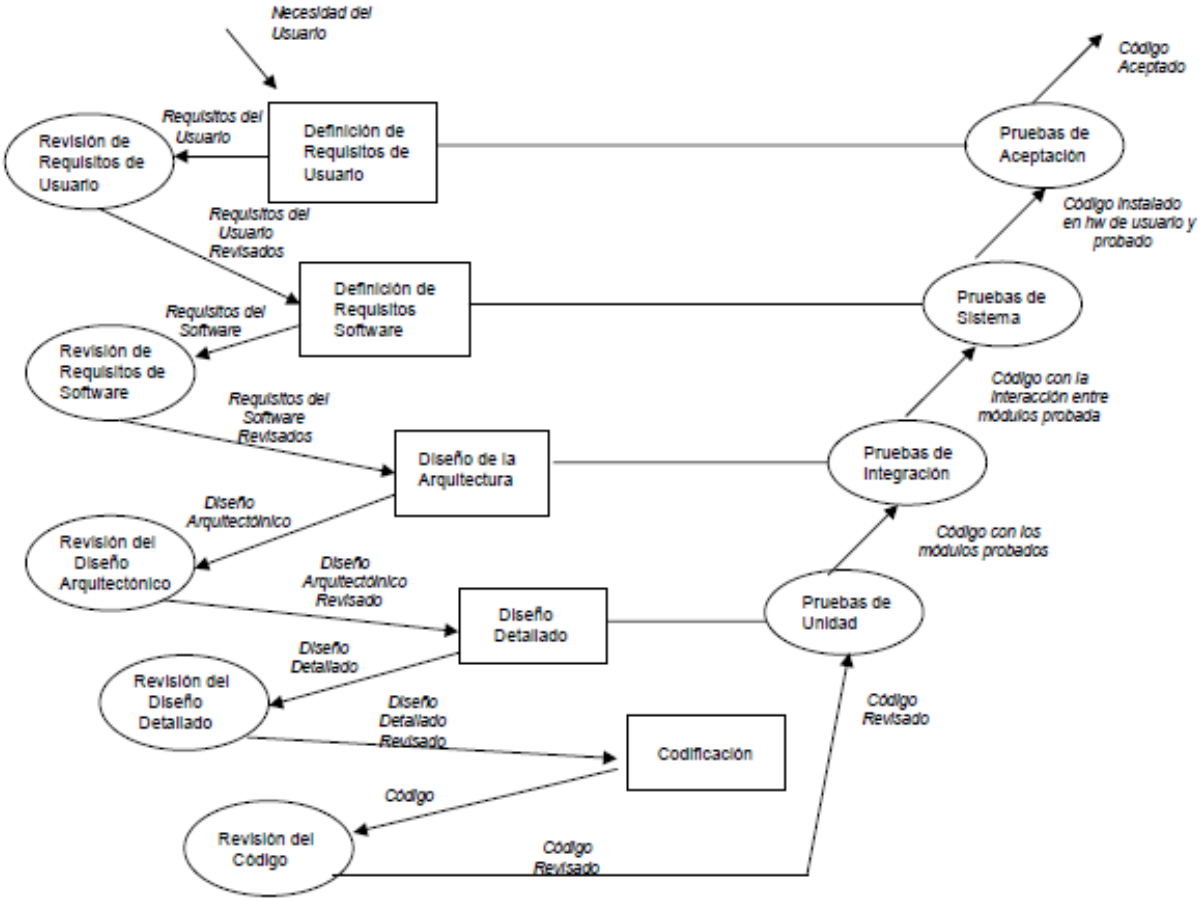


Ilustración 9. Modelo en V de la evaluación del software

3.7.1 Técnicas de evaluación estática

3.7.1.1 Beneficios de las revisiones

La razón por la cual es imprescindible buscar defectos de manera temprana en los productos es porque estos pueden traducirse en defectos en el producto final. Si el análisis y diseño realizados contienen defectos, seguramente estos serán transmitidos al código cuando los desarrolladores usen el diseño como guía para realizar su trabajo. La detección temprana de errores implica grandes beneficios mejorando la calidad, reduciendo costos al proyecto, potenciando las capacidades de los participantes, proporcionando aprendizaje, retroalimentación y prevención entre otras, esta detección a su vez es más fácil realizarla durante la evaluación estática que durante la dinámica.

3.7.1.2 Objetivos de la evaluación estática.

La evaluación estática de los distintos artefactos o productos que fueron generados en el diseño del *software* (especificación de requisitos, modelos conceptuales, diseño, etc.) tiene como principal objetivo comprobar su calidad. Por tanto, es imprescindible, cuando se está evaluando estáticamente un producto de *software* que el evaluador sepa identificar cuáles defectos está buscando.

Los defectos que fueron objetivo de búsqueda al evaluar estáticamente la solución quedan definidos a continuación:

- Para los requisitos: corrección, compleción, ambigüedad y claridad.
- Para el diseño: corrección, compleción, consistencia, factibilidad y trazabilidad.
- Para el código fuente: corrección, compleción, consistencia y trazabilidad.

3.7.1.3 Técnicas de evaluación estáticas utilizadas

Las técnicas de evaluación estáticas se les conocen de modo genérico como revisiones, estas pretenden detectar manualmente defectos en cualquier producto del desarrollo; al decir manualmente hace referencia a que el producto (sea requisito, diseño, código, etc.) está en formato digital o impreso en papel y su revisión se realiza mediante la lectura (Juristo, *"et al."*, 2005).

Entre los distintos tipos de revisiones existentes fueron aplicadas las revisiones informales donde se realiza un intercambio de opiniones entre los participantes**,** revisiones formales o inspecciones donde a través de la utilización de una *técnica de lectura*[10], específicamente la técnica *Ad-hoc*[11], fue analizado el producto en busca de defectos.

3.7.2 Técnicas de evaluación dinámica

A la aplicación de técnicas de evaluación dinámica se le denomina también prueba del *software*. Concretamente la prueba del *software* se puede definir como una actividad en la cual un sistema o algunos de sus componentes se ejecuta en circunstancias previamente especificadas en la configuración de la prueba y se registran los resultados obtenidos, seguido de un proceso en el que los resultados obtenidos se comparan con los esperados, para de esta forma detectar los posibles fallos

[10] Las **técnicas de lectura** son guías que ayudan a detectar defectos en los productos de *software,* básicamente son una serie de pasos o procedimientos cuyo propósito es que el inspector adquiera un conocimiento profundo del producto de *software* que inspecciona.

[11] La técnica **Ad-hoc** o **sin listas de chequeo**, se entrega al inspector el producto de *software* sin ninguna indicación o guía sobre como proceder con el producto ni qué buscar. La detección de los defectos depende completamente de las habilidades, conocimientos y experiencia del inspector.

en el *software*. Estos fallos son la entrada a un proceso de depuración en el que se identifica la falta asociada al fallo y corregirla dando lugar de esta manera a una nueva prueba (Juristo, *"et al."*, 2005).

3.7.2.1 Estrategias y niveles de prueba

Una estrategia de prueba del *software* proporciona una descripción o planificación bien detallada de los pasos a llevar a cabo como parte de la prueba que da como resultado una correcta construcción del *software*. Por tanto cualquier estrategia de prueba debe incorporar el diseño de la prueba, la generación de los casos de prueba, la ejecución de estos y la evaluación de los resultados y la misma debe encaminarse a través de varios niveles, desde pruebas de bajo nivel que verifiquen que todos los pequeños segmentos de código se han implementado correctamente hasta pruebas de alto nivel que validen la integración con otros componentes del *software*, compatibilidad con el *hardware*, entre otros aspectos.

Conceptualmente se pueden distinguir cuatro grandes niveles de pruebas:

- Unidad o Unitarias: es la primera fase de las pruebas dinámicas y estas verifican el funcionamiento aislado de partes del *software* que se pueden probar de manera independiente. Dependiendo del contexto, estas partes podrían ser módulos, una unidad funcional de estos, entre otras y el objetivo es comprobar que estos están correctamente codificados. Normalmente, estas pruebas se realizan con acceso al código fuente y con el soporte de herramientas de depuración pudiendo implicar a la persona que lo creó.
- De Integración: es el proceso de verificar la integración entre componentes de un *software*.
- Del Sistema: se ocupan del completo comportamiento de un sistema para comprobar que se han integrado adecuadamente todos los elementos del mismo (*hardware* y *software*) comparándolo con los requisitos no funcionales como seguridad, confiabilidad, interconexiones externas con otras aplicaciones o dispositivos de *hardware*.
- De Aceptación: al momento de aplicarlas el *software* está listo para implantarse, entonces debe ser el cliente quien realice estas pruebas apoyado por personas del equipo de pruebas directamente en el entorno donde se va a explotar el mismo (Juristo, *"et al."*, 2005).

Para probar dinámicamente la solución se definió una estrategia a seguir. Las pruebas serán realizadas a diferentes niveles dependiendo del objetivo que se persiga con la misma, para cada uno de los cuales se definen los tipos de pruebas a aplicarse y las técnicas por cada uno de estos tipos y finalmente se evalúan las consideraciones y criterios de éxito.

Tabla 5. Estrategia de prueba definida para la solución

NIVEL	OBJETIVO	TIPO DE PRUEBA	MÉTODO DE PRUEBA	TÉCNICA DE PRUEBA
Unidad	Comprobar que las especificaciones funcionales se han implementado correctamente. Verificar que se cubren los flujos de control y de datos.	Función	Caja blanca[12]	Manual (Camino básico)
			Caja negra[13]	Manual (Particiones o clases de equivalencia)
Integración	Verificar el correcto funcionamiento de los componentes del *software* integrados como un todo.	Estructura, Seguridad	Caja negra	Manual (No incremental)
Sistema	Verificar que el *software* cumple los requisitos de rendimiento especificados, específicamente los de capacidad y tiempo de respuesta.	Carga, Stress	Caja negra	Automática

3.8 Resultados de la evaluación del software

Con la evaluación estática o realización de revisiones, aplicada tanto a los requisitos como a los distintos modelos del análisis y el diseño, y con el fin de comprobar su calidad fueron identificados defectos de forma temprana en el ciclo de vida del desarrollo de la solución. Se puede concluir que con la realización de dicha evaluación fueron corregidos, se verificó la completitud y se eliminaron ambigüedades en los requisitos y en el diseño realizado, todo esto gracias a la aplicación de evaluaciones formales e informales que permitió evaluar estáticamente los distintos artefactos generados; permitiendo el pase a la siguiente fase de la evaluación del *software*, con un porcentaje superior en cuanto al ahorro del coste de detectar y corregir defectos.

Con la evaluación dinámica, o realización de pruebas al *software* fueron identificados y corregidos los defectos presentes no descubiertos en la fase de evaluación estática, con la realización de cuatro

[12] **Técnica de caja blanca o estructural**, se basan en un minucioso examen de los detalles procedimentales del código a evaluar.

[13] **Técnica de caja negra o funcional**, se realizan pruebas sobre la interfaz del programa a probar, entendiendo por interfaz las entradas y salidas de dicho programa.

iteraciones de pruebas realizadas (Ver Ilustración 10. Resultado de las iteraciones), arrojando resultados que permitieron aumentar la calidad de la solución. La aplicación de las pruebas de unidad permitió la verificación del funcionamiento aislado de cada una de las partes que componen la solución, permitiendo corregir los errores presentes en el comportamiento interno y estructura del código fuente inspeccionado (Ver **anexo E**) así como también fueron identificadas y mitigadas, a la vez, inconsistencias en el flujo de datos (entrada y salida) sobre las interfaces de usuario (Ver **anexo D**). Con la realización de pruebas de integración (Ver Ilustración 11. Resultados de las pruebas de integración) quedó validada la combinación de la solución con los demás módulos del SGU, quedando comprobados: la correcta asignación de responsabilidades brindadas por el módulo de seguridad, la obtención de información ajena al negocio analizado, procedente de los módulos de carrera, control docente y estructura y composición (Ver **anexo F**). Por otra parte con la aplicación de pruebas al sistema (Ver **anexo G**) fueron mitigados defectos, y a la vez, se verificó el total cumplimiento de los requisitos no funcionales definidos para la solución como la velocidad de respuesta, carga y confiabilidad con la utilización de la herramienta JMeter[14].

Todo esto permitió mantener el número de fallos a niveles mínimos, lo que trajo como resultado que la solución pudiese mejorar los atributos de funcionalidad y eficiencia.

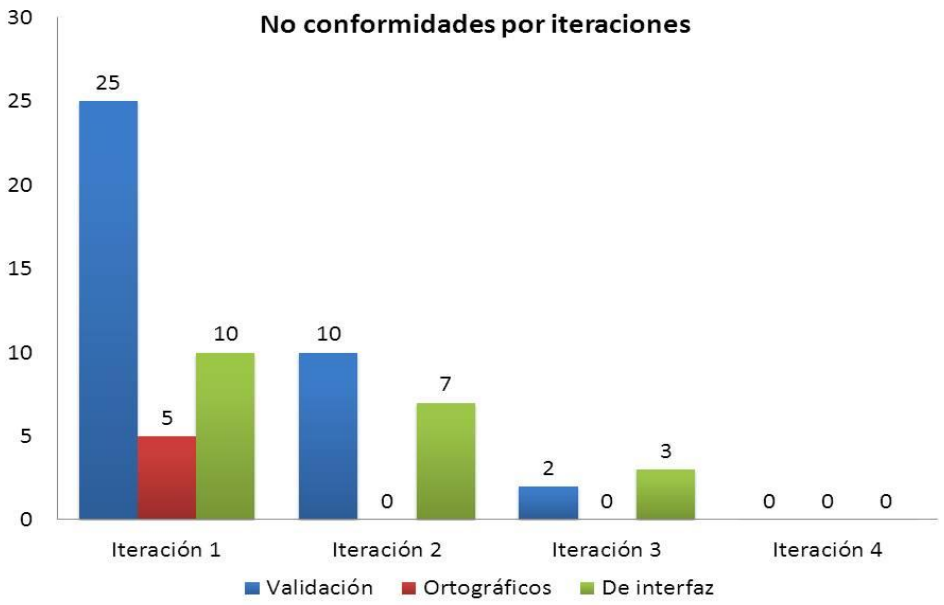


Ilustración 10. Resultado de las iteraciones

[14] **JMeter** es una herramienta Java dentro del proyecto Jakarta, que permite realizar pruebas de rendimiento y pruebas funcionales sobre aplicaciones web.

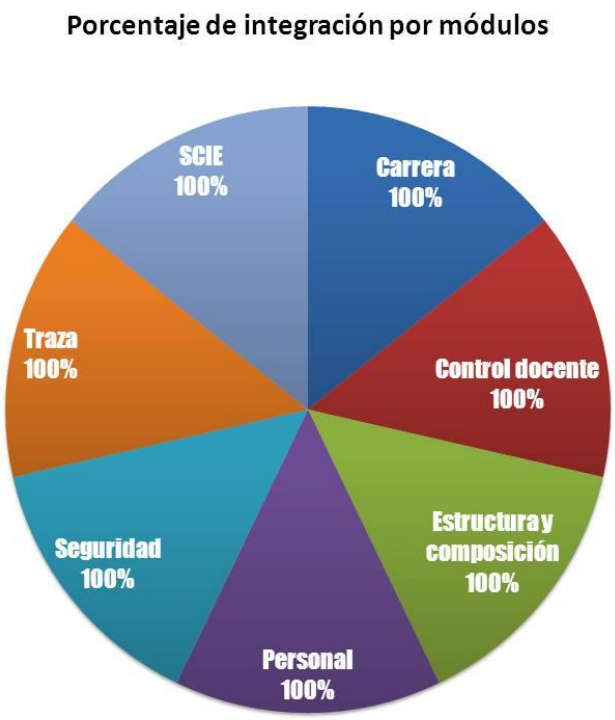


Ilustración 11. Resultados de las pruebas de integración

3.9 Análisis de los beneficios

¿Satisface el sistema a los objetivos generales del proceso y de la universidad con respecto al MAA? El producto terminado forma parte del módulo de Personal y Secretaría del SGAP perteneciente al SGU de la UCI, con el cual se gestionarán los procesos universitarios. El mismo será utilizado por los jefes de los departamentos docentes de cada una de las facultades de la universidad, profesores que ejercen como tutores de AA, los propios AA y demás involucrados en las actividades que conforman el proceso de gestión de dichos estudiantes, para mantener toda la información centralizada y facilitar la divulgación de la misma. El mismo contribuye a las exigencias, necesidades y objetivos de la universidad y del proceso en general; a través de la gestión de solicitudes, evaluaciones, e información general referente al proceso; quedando satisfechas todas necesidades del cliente; por lo que está disponible para su explotación en cada una de las facultades de la universidad y con la cual se agilizará el proceso de captación, aceptación y seguimiento de los estudiantes vinculados al MAA.

Desde un punto de vista técnico la solución para la gestión de la información de AA permite mostrar datos e información a cualquiera de los usuarios conectados a este desde una computadora personal.

Desde el punto de vista funcional la solución para la gestión de la información de AA proporciona grandes beneficios a quienes hacen uso de esta, permitiendo concentrar y consultar información procedente de las distintas facultades de la universidad, identificar quien genera y usa la información que maneja el MAA, actualizar, generar y consultar datos en tiempo real otorgando credibilidad al proceso, mantener una mejor gestión de los recursos humanos pues posibilita monitorear la situación actual en la que se encuentran cada uno de los involucrados en el proceso, además provoca un elevado nivel de comunicación dentro del centro de estudios ya que la información fluye y se sistematiza. De manera general la gestión de los recursos humanos e información asociada a estos se sistematiza, se ahorra tiempo, recursos y beneficia al cliente con información oportuna, producto de la actualización constante y centralización de los datos.

3.10 Conclusiones parciales

Tras el completo desarrollo de la solución, y posterior a este, la aplicación de técnicas de evaluación y una valoración del resultado obtenido tras la puesta en práctica de estas, se obtuvo una serie de funcionalidades integradas al SGAP del SGU que dan cumplimiento al objetivo trazado con la investigación y que satisfacen las necesidades y expectativas del proceso de gestión de la información del MAA dentro de la UCI. Quedó comprobado además el correcto funcionamiento de la solución tanto internamente como en el cumplimiento de los requisitos funcionales y no funcionales.

Conclusiones generales

Con la investigación realizada, el desarrollo y puesta en práctica de una agrupación funcional para la gestión de la información del MAA dentro de la UCI y en correspondencia con los resultados esperados se puede concluir que:

- Fueron adquiridos y puestos en práctica los elementos teóricos más importantes que sirvieron de base y guiaron conceptualmente la investigación.
- Quedó definido un modelo para la solución mediante el cual se obtuvo un grupo de funcionalidades que satisfacen las necesidades de la gestión de información del MAA en la UCI.
- Quedó garantizado el correcto funcionamiento de la solución mediante la evaluación realizada, verificando el cumplimiento de los requisitos funcionales, no funcionales y brindando beneficios para la organización y la universidad.

Por lo que se concluye que fue desarrollada una agrupación funcional para consultar y mantener el control sobre la información de los estudiantes vinculados al MAA y del movimiento en general que cumple con las expectativas y necesidades de dicha organización.

Recomendaciones

Con el desarrollo de futuras investigaciones se recomienda perfeccionar e incorporar nuevas funcionalidades para lograr mejoras superiores en la gestión de la información dentro del objeto de estudio de la presente investigación, entre las cuales se pueden considerar las siguientes:

- Aprovechar la integración con el módulo de control docente para una mejor actualización de la situación del alumno ayudante.
- La posibilidad de exportar la información a los formatos pdf y xls.
- Realizar notificaciones, vía correo electrónico, una vez que el estudiante sea vinculado al MAA o cambie su situación como alumno ayudante.

Bibliografía referenciada

(Álvarez, 2011) **ÁLVAREZ, Miguel Ángel.** *Manual de jQuery*. [en línea] <http://www.desarrolloweb.com> [consultada: 12 noviembre 2011].

(Álvarez, 2001) **ÁLVAREZ, Rubén**. *Qué es y para que sirve SQL*. 2001. [en línea] < http://www.desarrolloweb.com/articulos/262.php > [consultada: 12 noviembre 2011].

(ASF, 2012) **Apache Software Foundation.** *Apache HTTP Server*. 2012. [en línea] <http://httpd.apache.org/> [consultada: 12 noviembre 2011].

(Bahit, 2011) **BAHIT, Eugenia**. *POO y MVC en PHP: el paradigma de la Programación Orientada a Objetos en PHP y el patrón de arquitectura de software MVC*. 2011 [en línea]<http://sunshine.prod.uci.cu/gridfs/sunshine/books/MVC_con_PHP.pdf > [consultada: 06 abril 2012].

(Booch, *"et al."*, 2000) **BOOCH, Grady; JACOBSON, Ivar; RUMBAUGH, James**. *El proceso unificado de desarrollo de software*. España, Ed, Pearson Educación, 2000. 464p.

(Bustelo, 2001) **BUSTELO, C; Amarilla, R**. *Gestión del Conocimiento y Gestión de la Información*. Boletín del Instituto Andaluz de Patrimonio histórico; 2001 VIII(34): 226-230.

(CDI, 2012) **CodeIgniter.** *Bienvenidos a CodeIgniter*. [en línea] <http://www.codeigniter.com> [consultada: 12 noviembre 2011].

(Eguíluz, 2008) **EGUÍLUZ Pérez, Javier**. *Introducción a JavaScript*. 2008. [en línea] <http://www.librosweb.es> [consultada: 12 noviembre 2011].

(EVA, 2011) **EVA (Entorno Virtual de Aprendizaje).** *Mejora del proceso de software. El modelo CMMI*. 2011 [en línea] <http://eva.uci.cu/file.php/257/Documentos/Recursos_didacticos/Presentaciones_digitales_UD3/PresentacionCMMI.ppt> [consultada: 29 mayo 2012].

(FSF, 2012) **Free Software Foundation.** *¿Qué es el software libre?* 2012 [en línea] < http://www.gnu.org/philosophy/free-sw.es.html > [consultada: 12 mayo 2012].

(García, 2011) **GARCÍA Vidal, Yanio**. *Documento de arquitectura de software*. Sistema de Gestión Académica de Pregrado. (1.2), 2011.

(González, 2012) **GONZÁLEZ Cornejo, José Enrique.** *¿Qué es UML?.* 2012 [en línea] <http://www.docirs.cl/uml.htm> [consultada: 01 abril 2012].

(GU, 2011) **Guía Ubuntu.** *PgAdmin III*. 2008. [en línea] <http://guia-ubuntu.org/index.php?title=PgAdmin_III> [consultada: 12 noviembre 2011].

(IEEE, 1990) **IEEE Std 610.12-1990(R2002).** *IEEE Standard Glossary of Software Engineering Terminology,* IEEE, 1990.

(Jorba, "*et al.*", 2004) **JORBA Esteve, Josep; SUPPI Boldrito, Remo**. *Administración avanzada de GNU/Linux*. 2004 [en línea] <http://sunshine.prod.uci.cu/gridfs/sunshine/books/Administracin_avanzada_de_GNU_Linux.pdf> [consultada: 18 abril 2012].

(Juristo, *"et al."*, 2005) **JURISTO, Natalia; MORENO, Ana M; VEGAS Sira**. *Técnicas de evaluación de software*. 2005. [en línea] < http://eva.uci.cu/file.php/158/Documentos/Recursos_bibliograficos/Libros_y_articulos_UD_2/Comun/Tecnicas_de_evaluacion_de_software_Jurisco-Moreno.pdf > [consultada: 25 abril 2012].

(Lamarca, 2011) **LAMARCA Lapuente, María Jesús**. *Hipertexto: El nuevo concepto de documento en la cultura de la imagen*. 2011. [en línea] < http://www.hipertexto.info/documentos/html.htm> [consultada: 19 noviembre 2011].

(Larman, 1999) **LARMAN, Craig**. *UML y Patrones*. Ed, Prentice Hall, 1ra Edición, México. 1999 [en línea] < http://eva.uci.cu/mod/resource/view.php?id=8500&subdir=/UML_y_Patrones > [consultada: 22 mayo 2012].

(OCA, 2012) **Oracle Corporation and/or its affiliates.** *¿Qué es NetBeans?* 2012. [en línea] <http://netbeans.org/index_es.html> [consultada: 12 noviembre 2011].

(Pastor, 1998) **PASTOR H, Patricio**. *Intranet: un sistema para la gestión de información*. En: Taller en Tecnologías de Redes Internet para América Latina y el Caribe: 13 y 18 de julio de 1998. Rio de Janeiro, Brasil. Internet Society, ENRED y Mercosur.

(PP, 2012) **Pencil Project**. *Pencil*. [en línea] < http://sudokill.blogspot.com/2010/10/evolus-pencil.html > [consultada: 11 abril 2012].

(PGDG, 2012) **PostgreSQL Global Development Group.** *A brief history of PostgreSQL***.**

2012 [en línea] <http://www.postgresql.org/docs/8.2/interactive/intro-whatis.html.> [consultada: 12 noviembre 2012].

(Pressman, 2002) **PRESSMAN, Roger**. *Ingeniería del Software. Un enfoque práctico.* 5ta edición. McGraw-Hill, 2002. Pp. 201-209.

(RAE, 2012) **Real Academia Española**. *Organización*. [en línea] <http://www.rae.es/RAE/Noticias.nsf/Home?ReadForm> [consultada: 08 abril 2012].

(MES, 2007a) **Resolución 210 del Ministerio de Educación Superior**. *Capítulo 5, artículo 214*. 2007.

(MES, 2007b) **Resolución 210 del Ministerio de Educación Superior**. *Capítulo 5, artículo 215*. 2007.

(SEI, 2012) **SEI (Software Engineering Institute).** *Información general*. 2012. [en línea] <http://www.sei.cmu.edu/cmmi/> [consultada: 11 abril 2012].

(Sommerville, 2005) **SOMMERVILLE, Ian**. *Ingeniería del software*. 7ma edición. Pearson Educación. S.A., 2005. [en línea] < http://eva.uci.cu/mod/resource/view.php?id=8501&subdir=/Ediciones_del_Sommerville/Sommerville_7ma_edicion> [consultada: 1 febrero 2012].

(TPG, 2011) **The PHP Group**. *¿Qué es PHP?* 2011. [en línea] <http:///www.php.net> [consultada: 9 noviembre 2011].

(VPILtd, 2011) **Visual Paradigm International Ltd.** *Visual Paradigm for UML*. 2011 [en línea] < http://www.visual-paradigm.com/> [consultada: 12 noviembre 2011].

Bibliografía consultada

Apache Jakarta Project. *JMeter*. [en línea] <http://jakarta.apache.org/jmeter/> [consultada: 22 mayo 2012]

ARTOLA Pimentel, María de Lourdes; BAÑOS Pinedo, Irelys; *et al*. Sistema Automatizado para la *Gestión de la Información del Capital Humano de un Departamento Docente.* [en línea] <http://monografias.umcc.cu/monos/2006/Informatica/SistemaAutomatizadoparalaGestióndelaInformaciónde lCapital.pdf> [consultada: 01 abril 2012].

EGUILUZ Pérez, Javier. *Introducción a CSS*. [en línea] <http://www.librosweb.es/css> [consultada: 01 abril 2012].

Facultad de Medicina de la Universidad de Chile. [en línea] <http://www.med.uchile.cl/> [consultada: 20 noviembre 2011].

GALLEGO Vázquez, José Antonio. *Desarrollo web con PHP y MySQL*. Anaya, 2003.

GONZÁLEZ Fernández, Yisel; BENÍTEZ Campo, Yanitza. *Sistema de apoyo al ingreso, seguimiento y control del movimiento de alumnos ayudantes en la facultad 1*. Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas, Universidad de las Ciencias Informáticas, 2010.

GONZÁLEZ, C. D. *Base de Datos PostgreSQL, SQL avanzado y PHP*. 2008.

HINOJOSA Domínguez, Yusdani; HURTADOS Díaz, Rangel. *Sistema de gestión del Movimiento de Alumnos Ayudantes*. Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas, Universidad de las Ciencias Informáticas, 2010.

LARMAN, Craig. *UML y Patrones*. Ed, Prentice Hall, 1ra Edición, México. [en línea] < http://eva.uci.cu/mod/resource/view.php?id=8500&subdir=/UML_y_Patrones > [consultada: 22 mayo 2012]

MEYER, Bertrand. *Construcción de software Orientado a Objetos*. Ed, Prentice Hall.

PONJUÁN Dante, Gloria. *Gestión de información. Dimensiones e implementación para el éxito organizacional. Rosario: Nuevo Paradigma*, 2004.

PONJUÁN Dante, Gloria. *Gestión de la Información en las Organizaciones. Principios, conceptos y aplicaciones*. Santiago de Chile: CECAPI, 1998. Pp.84.

PRESSMAN, Roger. *Estrategias de prueba del software. Ingeniería del Software. Un enfoque práctico*. 6ta edición. McGraw-Hill, 2005. Pp.383-417.

PRESSMAN, Roger. *Ingeniería de requisitos. Ingeniería del Software. Un enfoque práctico*. 6ta edición. McGraw-Hill, 2005. Pp.155-190.

PRESSMAN, Roger. *Ingeniería del Software. Un enfoque práctico.* 5ta edición. McGraw-Hill, 2002

PRESSMAN, Roger. *Software Engineering: A Practitioner´s Approach. The software process. Chapter 2 "Process models"*. 7ma edición. McGraw-Hill, 2010. Pp.31-64.

PRESSMAN, Roger. *Software Engineering: A Practitioner´s Approach. The software process. Chapter 3 "Agile Development"*. 7ma edición. McGraw-Hill, 2010. Pp.65-92.

PRESSMAN, Roger. *Técnicas de prueba del software. Ingeniería del Software Un enfoque práctico*.6ta edición. McGraw-Hill, 2005. Pp.418-461.

SEVIUS (Secretaría Virtual de la Universidad de Sevilla). [en línea] <https://sevius.us.es> [consultada: 18 noviembre 2011].

SIU Guaraní. [en línea] <http://alumnos.unsa.edu.ar> [consultada: 20 noviembre 2011].

SOMMERVILLE, Ian. *Ingeniería del software*. 7ma edición. Pearson Educación. S.A., 2005. [en línea] < http://eva.uci.cu/mod/resource/view.php?id=8501&subdir=/Ediciones_del_Sommerville/Sommerville_7ma_edicion> [consultada: 1 febrero 2012]

Universidad de Concepción de Chile. *Dirección de Tecnologías de Información*. [en línea] <http://www.udec.cl/dti/> [consultada: 20 noviembre 2011].

Universidad Politécnica de Valencia. [en línea] <http://www.upv.es/index-es.html> [consultada: 20 noviembre 2011].

WHITE, Stephen A; MIERS, Derek. *Guía de Referencia y Modelado BPMN. Comprendiendo y utilizando BPMN*, Future Strategies Inc.

Glosario de términos

Ajax: acrónimo de JavaScript y XML asíncronos (en inglés, *Asynchronous JavaScript And XML*) es una técnica de desarrollo web para crear aplicaciones interactivas.

Centro de Informatización Universitaria (CENIA): es un centro productivo de la Universidad de las Ciencias Informáticas que se encarga de la informatización de la gestión universitaria y el desarrollo del concepto de universidad digital. En el mismo se trabaja con un enfoque de solución integral para la gestión de los procesos universitarios, con proyectos para las áreas de procesos de la formación académica de pregrado y postgrado, de las investigaciones, la cooperación y la gestión de residencia.

DOM: es el Modelo de Objetos de Documento (en inglés, *Document Object Model*) es una forma de representar documentos estructurados que es independiente de cualquier lenguaje orientado a objetos.

Entorno de Desarrollo Integrado (IDE)*:* programa compuesto por un conjunto de herramientas para un programador. Cuenta con un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI).

Helpers: los *helpers* en informática no son más que una colección de funciones (simples y de procedimiento) de una categoría en particular que facilitan a los programadores la realización de varias tareas.

HTTPS: es el protocolo seguro de transferencia de hipertexto (en inglés, *Secure HyperText Transfer Protocol*), el mismo es usado en cada transacción realizada en la web.

Intranet: red interna de una empresa o institución.

Marco de trabajo: Es una estructura de soporte definida para la creación de *software* de una forma más rápida, brindando a los desarrolladores posibilidades (funciones, librerías, *script*, etc.) que le facilitan la creación de este si se desea comenzar desde cero.

Software privativo: La expresión de *software* privativo comenzó a ser utilizada por Richard Matthew Stallman en sus conferencias sobre libre. Es el *software* sobre el cual existen restricciones de derechos o libertades.

Software: son todos los componentes intangibles de un ordenador o computadora, es decir, al conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica.

SQL Server: Microsoft SQL Server es uno de los mejores sistemas gestores de bases de datos relacionales, ideal para clientes corporativos y fabricantes independientes de software.

SSL: acrónimo de Protocolo de Capa de Conexión Segura (en inglés, *Secure Sockets Layer*) proporciona conjuntamente con TLS (en inglés, *Transport Layer Security*) comunicaciones seguras en internet.

UML*:* Lenguaje Unificado de Modelado (en inglés, *Unified Modeling Language*) es un lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de *software*.

UNIX: sistema operativo portable, multitarea y multiusuario desarrollado por un grupo de empleados de los laboratorios Bell de AT&T.

Visual Studio .NET: es un entorno de desarrollo integrado desarrollado por Microsoft para el sistema operativo Microsoft Windows.

Widget: es una pequeña aplicación, presentada en pequeños archivos, que son ejecutados por un motor de *widgets* o *widgets engine*. Estos brindan fácil acceso a funciones frecuentemente usadas y en algunos casos proveen información visual.

WWW*:* es la telaraña mundial (en ingles, *World Wide Web*) es un sistema de hipertexto que funciona sobre internet.

Anexos

Anexo A. Entrevista (Método científico de investigación)

Entrevistado: Eugenio Rosales Rosa.

Cargo: Presidente del MAA de la Facultad 1.

Preguntas:

¿Cuál es el proceso a seguir para la selección de los estudiantes que formarán parte del Movimiento de Alumnos Ayudantes?

¿Cuál es el personal encargado de la selección de estos estudiantes?

¿Qué requisitos debe cumplir el estudiante para poder ejercer la ayudantía?

¿Qué inconformidades existen hoy en día por parte de los estudiantes, vinculados a dicho movimiento, con respecto a este?

¿Qué inconformidades existen hoy en día por parte de los estudiantes, que no están vinculados a dicho movimiento, con respecto a este?

¿Existe la necesidad de una aplicación para la gestión de la información?

Anexo B. Entrevista (Obtención de requisitos)

Entrevistados: Jose Hilario Quintana Álvarez (Segundo Jefe de Departamento de Ciencias Básicas, Facultad 1), Yalice Gómez Batista (Vicedecano de formación, Facultad 3), Dunia Suarez Ferreiro (Jefe de Departamento de Programación, Facultad 1).

Preguntas:

¿Cuáles son las principales funciones o acciones que se ejecutan dentro del proceso?

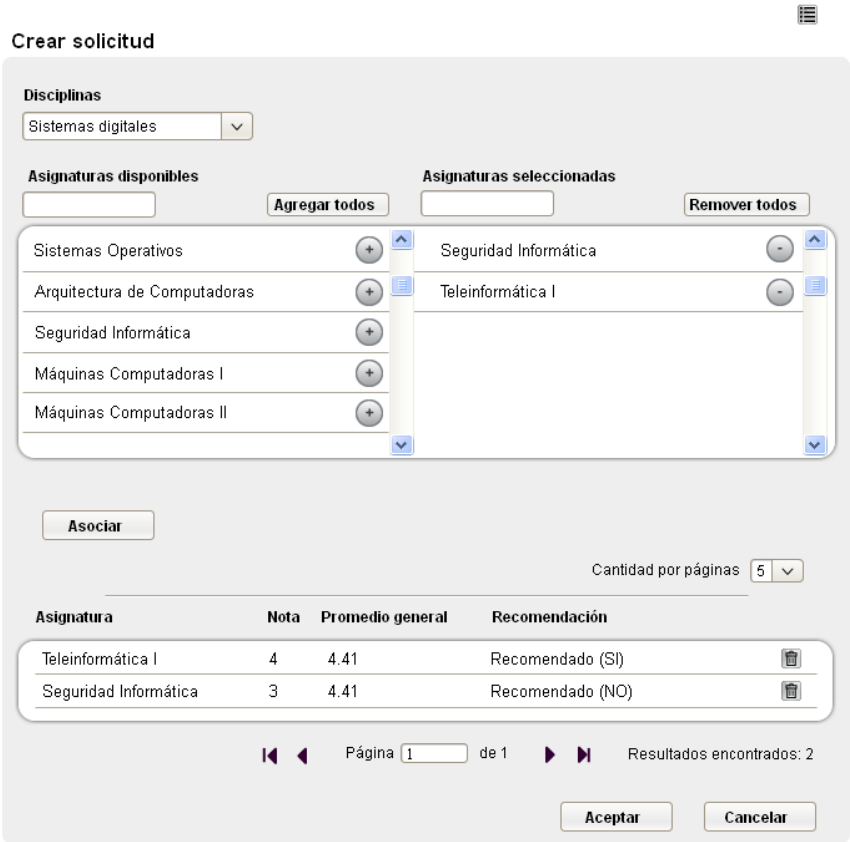
Podría describir el proceso de ratificación de los alumnos ayudantes.

¿De que dependen las evaluaciones semestrales de los alumnos ayudantes?

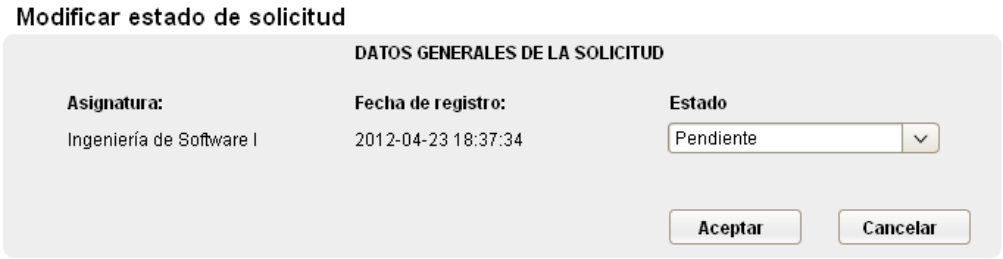
¿De que dependen las evaluaciones anuales de los alumnos ayudantes?

¿El estudiante puede ejercer la ayudantía en más de una asignatura al mismo tiempo?

Anexo C. Especificación de requisitos de software

Nº	Nombre	Descripción	Complejidad	Prioridad para cliente
RF1	Crear solicitud de ayudantía	Permite crear a un estudiante una solicitud para ejercer la ayudantía, dando la posibilidad de cancelar la acción.	Alta	Alta
	Prototipo			
				
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• selectDisciplinas • multiselect Asignaturas	• Varchar • Varchar • No procede	• Campo obligatorio • Campo obligatorio • Campo obligatorio	

	• gridEvaluaciones			
	Observaciones			

RF2	Modificar estado de solicitud	Permite al jefe de un departamento docente modificar el estado de la solicitud realizada.	Alta	Alta
	Prototipo			
				
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• select_Estados_Solicitud • textAsigantura • textFechaRegistro	• Varchar • Varchar • Date	• Campo obligatorio • Solo lectura • Solo lectura	
	Observaciones			

RF3	Mostrar listado de solicitudes de ayudantía del estudiante	Muestra al estudiante un listado con las solicitudes realizadas por él brindándole la opción de "Ver detalles" en el área de íconos internos así como la opción de	Media	Media

		"Crear solicitud" en el área de íconos flotantes.		
	Prototipo			
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• textNombreApellidos • textUsuario • textSolapín • gridSolicitudesEstudiante	• Varchar • Varchar • Varchar • Varchar	• Solo lectura • Solo lectura • Solo lectura • Solo lectura	
	Observaciones			

RF4	Mostrar listado general de solicitudes	Muestra a los jefes de departamentos docentes un listado con las solicitudes realizadas por los estudiantes, brindándoles las opciones de "Ver detalles de solicitud" y "Modificar estado de	Media	Alta

		solicitud" en el área de íconos internos, así como también permite especificar parámetros de búsqueda (nombre, usuario y solapín del solicitante) y filtrar la búsqueda por "Facultad del solicitante", "Grupo del solicitante", "Sexo del solicitante", "Asignatura solicitada", "Estado de la solicitud" y "Fecha de registro".		
	Prototipo			
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• textBuscar • selectFiltros • selectFacultadSolicitante • selectGrupoSolicitante • selectSexoSolicitante	• Varchar • Varchar • Varchar • Int • Char • Varchar • Varchar • Date • Varchar	• Se debe escribir en el campo algún parámetro de búsqueda como: nombre, usuario o solapín del solicitante. • Campo no obligatorio • Campo no obligatorio	

	• selectAsignaturaSolicitada • selectEstadoSolicitud • dateFechaRegistro • gridSolicitudesGeneral		• Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Solo lectura
	Observaciones		

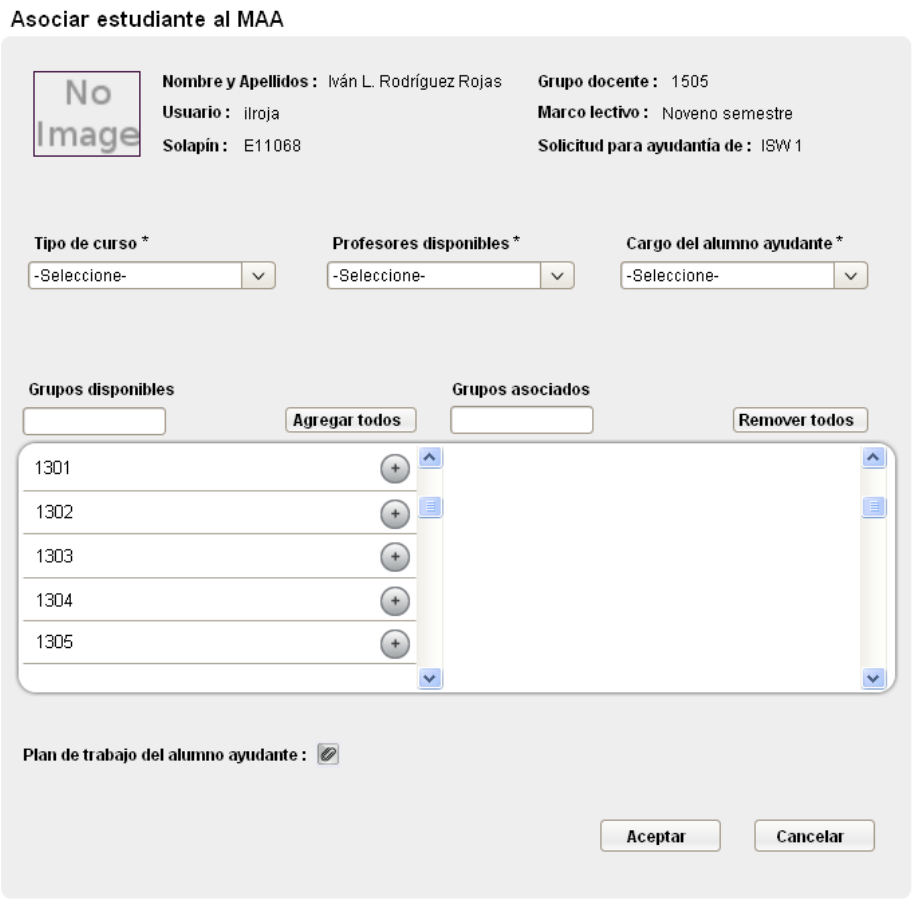
RF5	Ver detalles de solicitud de ayudantía	Muestra todos los detalles de la solicitud realizada como son: quien realizó la solicitud, en caso de ser una propuesta quien propuso y quien fue propuesto, asignatura solicitada, nota alcanzada por el estudiante e índice general del mismo.	Baja	Baja
	Prototipo			
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• textAsignatura • textFechaRegistro	• Varchar • Date • Varchar	• Solo lectura • Solo lectura • Solo lectura	

	• textEstado			
	Observaciones			

RF6	Mostrar listado de solicitudes aceptadas	Muestra a los jefes de departamentos docentes un listado con todas las solicitudes aceptadas brindándole la opción de "Asociar estudiante al MAA" en el área de íconos internos y filtrar la búsqueda por "Facultad del solicitante", "Grupo del solicitante", "Sexo del solicitante", "Asignatura solicitada".	Media	Media
	Prototipo			
	Campos	**Tipos de Datos**		**Reglas o Restricciones**
	• textBuscar • selectFiltros • selectFacultadSolicitante	• Varchar • Varchar • Varchar • Int • Char • Varchar		• Se debe escribir en el campo algún parámetro de búsqueda como: nombre, usuario o solapín del solicitante.

	• selectGrupoSolicitante • selectSexoSolicitante • selectAsignaturaSolicitada • dateFechaRegistro • gridSolicitudesAceptadas	• Date • Varchar	• Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Solo lectura
	Observaciones		

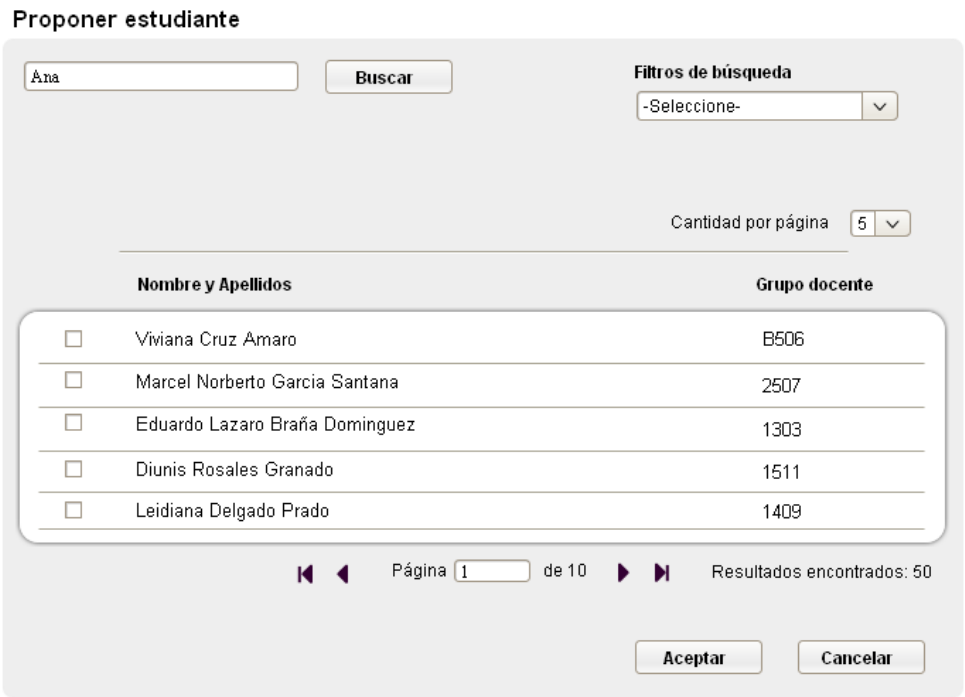
RF7	Asociar estudiante al MAA	Permite al jefe del departamento docente asociar a el estudiante seleccionado al MAA, asignándole a este un tutor, un tipo de curso, un plan de trabajo del alumno ayudante y un cargo de alumno ayudante en caso que sea necesario.	Alta	Alta
	Prototipo			



Campos	Tipos de Datos	Reglas o Restricciones
• textNombreApellidos • textUsuario • textSolapín • textGrupo • textMarcoLectivo • textAsignatura • selectTutor • selectTipoCurso • selectCargoAA	• Varchar • Varchar • Varchar • Int • Varchar • Varchar • Varchar • Varchar • Varchar • Varchar • File	• Solo lectura • Solo lectura • Solo lectura • Solo lectura • Solo lectura • Solo lectura • Campo obligatorio • Campo obligatorio • Campo obligatorio • Campo no obligatorio • El archivo cargado debe ser en formato .pdf

	• multiselect Grupos • archivoPlan TrabajoAA			
	Observaciones			

RF8	Mostrar listado de estudiantes a proponer	Muestra a los profesores un listado con los estudiantes a proponer brindándole la opción de "Proponer estudiante" en el área de íconos internos, así como también permite especificar parámetros de búsqueda (nombre y apellidos, ci, usuario y solapín del estudiante) y filtrar la búsqueda por "Facultad", "Grupo", "Sexo", "Año académico" y "Provincia".	Media	Media
	Prototipo			



Campos	Tipos de Datos	Reglas o Restricciones
• txtBuscar • selectFiltros • selectFacultad • selectGrupo • selectAño • selectSexo • gridEstudiantes	• Varchar • Varchar • Varchar • Int • Int • Char • Varchar	• Se debe escribir en el campo algún parámetro de búsqueda como son nombre, apellidos, ci, usuario o solapín del estudiante que se desea buscar • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Solo lectura

	Observaciones	

RF9	Proponer estudiante para ayudantía	Al profesor dar click sobre el botón aceptar del listado de estudiantes a proponer se realiza la propuesta registrando junto con los datos del estudiante los datos del profesor que realizó la propuesta y la asignatura que imparte.	Alta	Alta
	Prototipo			

	Campos	**Tipos de Datos**	**Reglas o Restricciones**
	• txtBuscar • selectFiltros • selectFacultad • selectGrup	• Varchar • Varchar • Varchar • Int • Int • Char	• Se debe escribir en el campo algún parámetro de búsqueda como son nombre, apellidos, ci, usuario o solapín

	o • selectAño • selectSexo • gridEstudiantes	• Varchar	del estudiante que se desea buscar • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Solo lectura
	Observaciones		

RF10	Mostrar listado de alumnos ayudantes	Muestra un listado de los estudiantes que ejercen como alumnos ayudantes brindando las opciones de “Activar o desactivar”, “Ver detalles del alumno ayudante”, “Modificar datos del alumno ayudante”, “Crear evaluación semestral”, “Crear evaluación anual”, “Ver listado de evaluaciones” y “Desvincular al alumno ayudante” en el área de íconos internos, así como también permite especificar parámetros de búsqueda (nombre y apellidos, ci, usuario y solapín del estudiante) y filtrar la búsqueda por “Facultad”, “Grupo”, “Sexo”, ”Asignatura”, “Departamento”, “Situación” y “Cargo del alumno ayudante”.	Media	Media
	Prototipo			

Campos	Tipos de Datos	Reglas o Restricciones
• txtBuscar • selectFiltros • selectFacultad • selectGrupo • selectSexo • selectAsignatura • selectDepartamentoDocente • selectSituación • selectTutor • gridDatos	• Varchar • Varchar • Varchar • Int • Char • Varchar • Varchar • Varchar • Varchar • Varchar	• Se debe escribir en el campo algún parámetro de búsqueda como son nombre, apellidos, ci, usuario o solapín del alumno ayudante. • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio • Campo no obligatorio

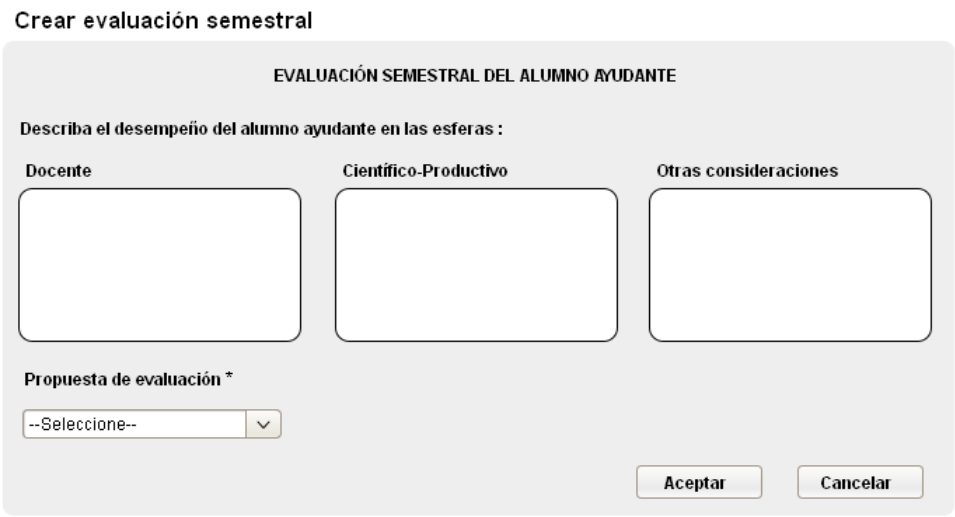
			• Campo no obligatorio • Campo no obligatorio • Solo lectura
	Observaciones		

RF11	Ver detalles de alumno ayudante	Muestra los detalles del alumno ayudante y su tutor, entre los datos a mostrar se encuentran el nombre del tutor, el departamento al que pertenece, los grupos que tiene asignados y el estado en el que se encuentra el alumno ayudante.	Baja	Baja
	Prototipo			
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• textNombreTutor • textDepartamento • textGrupos • textEstado	• Varchar • Varchar • Varchar • Varchar	• Solo lectura • Solo lectura • Solo lectura • Solo lectura	
	Observaciones			

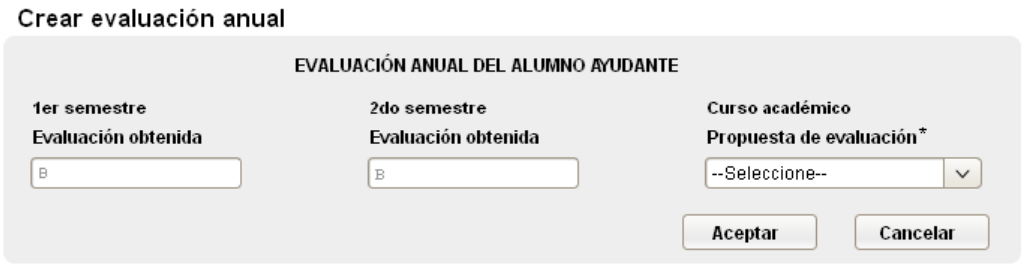
RF12	Modificar datos del alumno ayudante	Permite al jefe del departamento docente especificar nuevos datos al alumno ayudante (tutor, tipo curso, plan de trabajo, cargo y grupos asociados) o mantener los datos anteriores; así como la posibilidad de cambiar la asignatura del mismo o ubicarlo en otro departamento docente asignándoles una nueva asignatura y tutor en este último paso.	Alta	Alta
	Prototipo			

	Campos	Tipos de Datos	Reglas o Restricciones
	• textNombreApellidos • textUsuario • textSolapín • textGrupo • textMarcoLectivo • textAsignatura • selectTutor • selectTipoCurso • selectCargoAA • multiselectGrupos • archivoPlanTrabajoAA • radioCambioAsignatura • radioCambioDepartamento • selectAsignaturaCambioAsignatura • selectDepartamentoCambioDepartamenrto • selectAsignaturaCambioDepartamento • selectTutorCambioDep	• Varchar • Varchar • Varchar • Int • Varchar • Varchar • Varchar • Varchar • Varchar • Varchar • File • Bool • Bool • Varchar • Varchar • Varchar • Varchar	• Solo lectura • Solo lectura • Solo lectura • Solo lectura • Solo lectura • Solo lectura • Campo obligatorio • Campo obligatorio • Campo obligatorio • Campo no obligatorio • El archivo cargado debe ser en formato .pdf • Campo no obligatorio • Campo no obligatorio • Campo obligatorio • Campo obligatorio • Campo obligatorio • Campo obligatorio

	artamento			
	Observaciones			

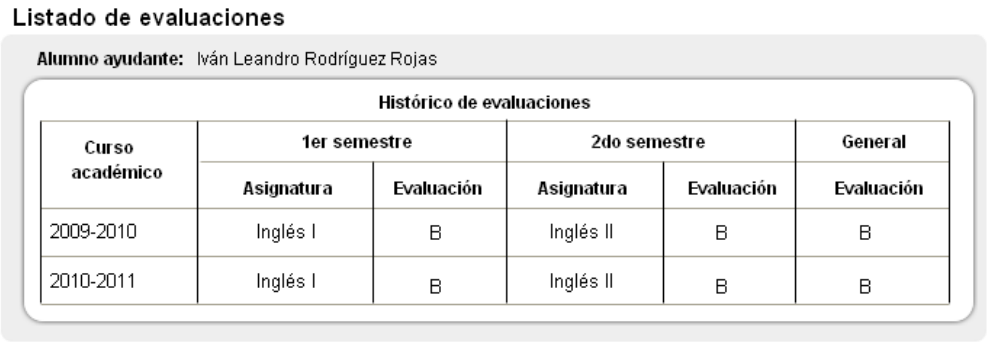
RF13	Crear evaluación semestral de alumno ayudante	Permite al tutor del alumno ayudante crear una evaluación semestral.	Alta	Alta
	Prototipo			
				
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• textAreaDocente • textareaCientificoProductivo • textAreaOtras • selectEvaluaciónFinal	• Varchar • Varchar • Varchar • Char	• El máximo de palabras admitidas es de un total de 300 de no más de 16 caracteres cada una • El máximo de palabras admitidas es de un total de 300 de no más de 16 caracteres cada una • El máximo de palabras admitidas	

			es de un total de 300 de no más de 16 caracteres cada una • Campo obligatorio y solo admite los siguientes caracteres: B, R y M
	Observaciones		

RF14	Crear evaluación anual de alumno ayudante	Permite al tutor del alumno ayudante crear una evaluación anual.	Alta	Alta
	Prototipo			
				
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• textEvaluaciónPrimerSemestre • textEvaluaciónSegundoSemestre • selectEvaluaciónAnual	• Char • Char • Char	• Solo lectura • Solo lectura • Campo obligatorio y solo admite los siguientes caracteres: B, R y M	
	Observaciones			

RF15	Mostrar registro de	Permite mostrar un listado con todas las evaluaciones del alumno	Alta	Alta

	evaluaciones	ayudante durante todo su desempeño, permitiendo la posibilidad de modificar estas.		
	Prototipo			



Listado de evaluaciones

Alumno ayudante: Iván Leandro Rodríguez Rojas

Histórico de evaluaciones

Curso académico	1er semestre		2do semestre		General
	Asignatura	Evaluación	Asignatura	Evaluación	Evaluación
2009-2010	Inglés I	B	Inglés II	B	B
2010-2011	Inglés I	B	Inglés II	B	B

	Campos	**Tipos de Datos**	**Reglas o Restricciones**
	• tablaEvaluaciones	• Varchar	• Solo lectura
	Observaciones		

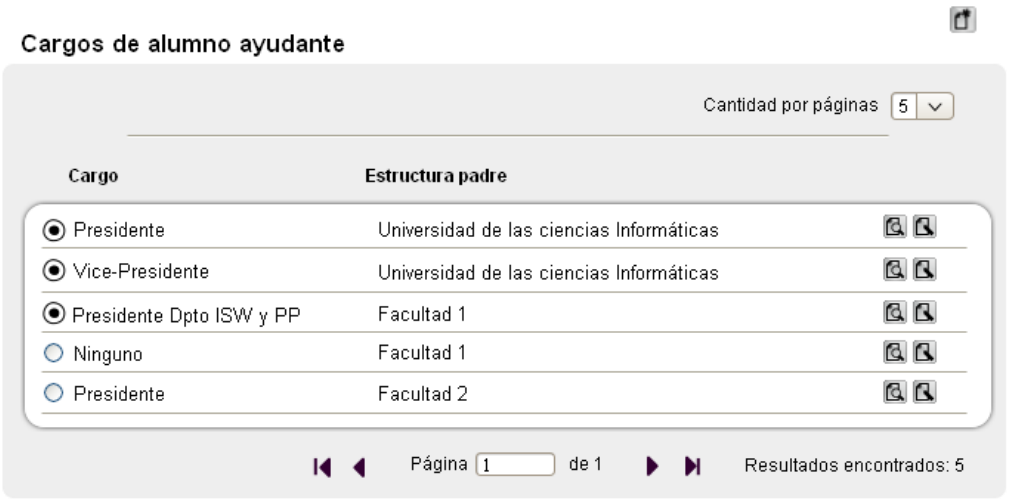
RF16	Desvincular alumno ayudante	Permite al jefe del dpto. docente desvincular al alumno ayudante del MAA	Alta	Alta
	Prototipo			

	Campos	Tipos de Datos	Reglas o Restricciones
	No procede	No procede	No procede
	Observaciones		

RF17	Mostrar estructura del movimiento de alumnos ayudantes	Se muestra un árbol jerárquico con la estructura del movimiento en la universidad	Alta	Alta
	Prototipo			

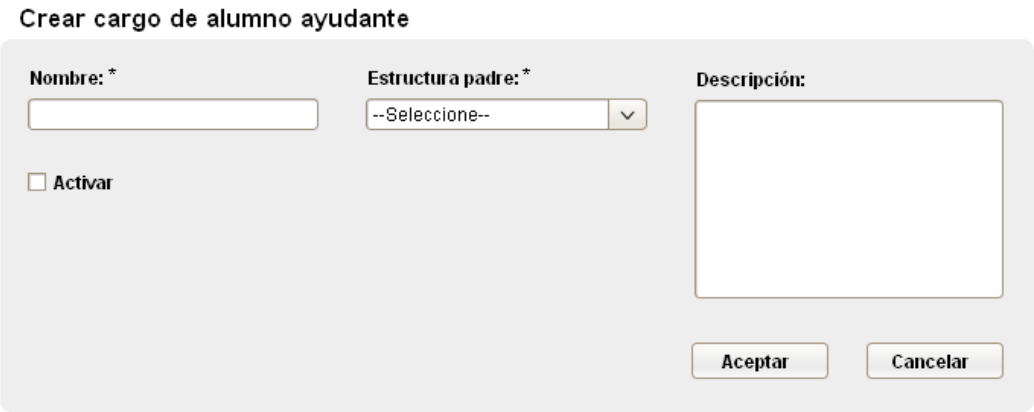
	Campos	Tipos de Datos	Reglas o Restricciones
	• arbolestructura	• Varchar	• Solo lectura
	Observaciones		

RF18	Mostrar listado de cargos de alumno ayudante	Muestra un listado con todos los cargos de alumnos ayudantes que han sido creados, brindando las opciones de “Ver detalles” y “Modificar cargo” en el área de íconos internos.	Media	Media
	Prototipo			

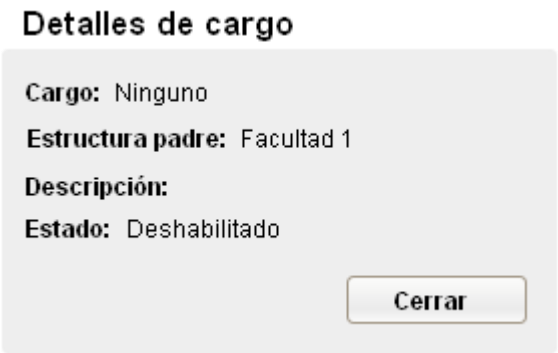


	Campos	Tipos de Datos	Reglas o Restricciones
	• gridCargos	• Varchar	• Solo lectura
	Observaciones		

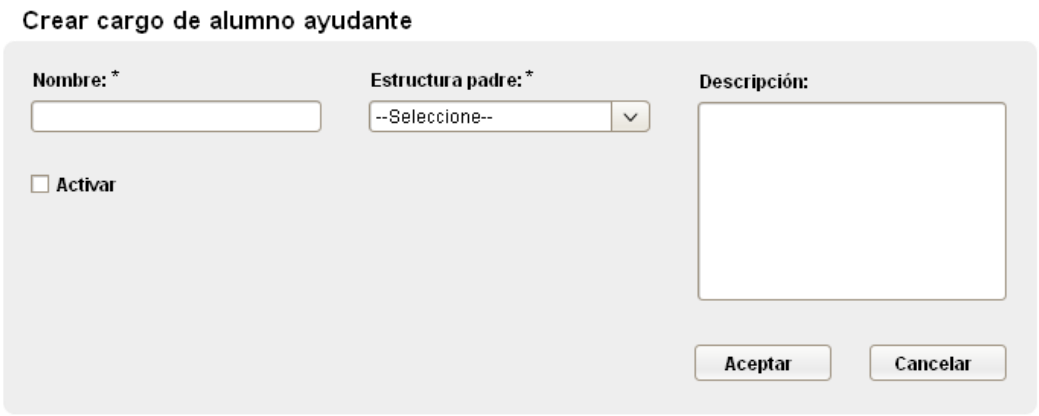
RF19	Crear cargo de alumno ayudante	Permite crear un cargo de alumno ayudante asociándole los campos: nombre, estructura padre, descripción y activo	Alta	Alta
	Prototipo			



	Campos	Tipos de Datos	Reglas o Restricciones
	• txtBuscar • selectEstructuraPadre • textareaDescripción • checkboxActivo	• Varchar • Varchar • Varchar • Bool	• Campo obligatorio • Campo obligatorio • Campo no obligatorio • Campo no obligatorio
	Observaciones		

RF20	Ver detalles de cargo de alumno ayudante	Permite mostrar los detalles del cargo seleccionado	Baja	Baja
	Prototipo			
				
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• textCargo • textEstructuraPadre • textDescripción • textEstado	• Varchar • Varchar • Varchar • Varchar	• Solo lectura • Solo lectura • Solo lectura • Solo lectura	

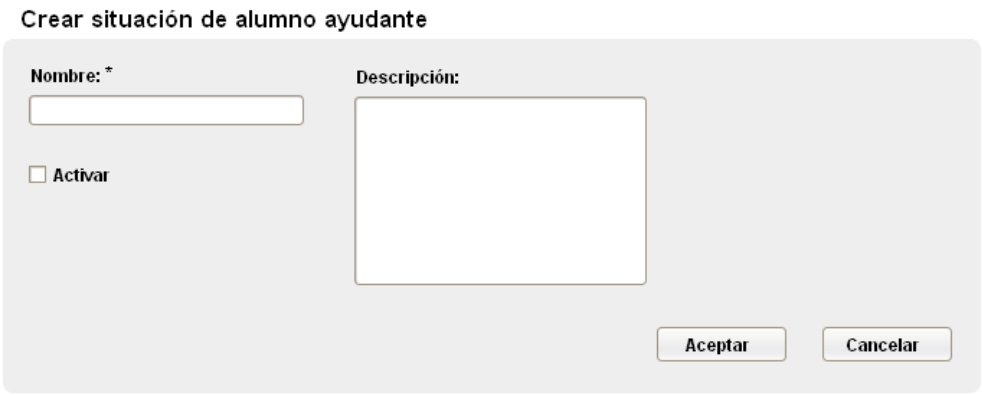
	Observaciones			

RF21	Modificar cargo de alumno ayudante	Permite modificar los datos del cargo seleccionado	Alta	Alta
	Prototipo			
				
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• txtBuscar • selectEstructuraPadre • textareaDescripción • checkboxActivo	• Varchar • Varchar • Varchar • Bool	• Campo obligatorio • Campo obligatorio • Campo no obligatorio • Campo no obligatorio	
	Observaciones			

RF22	Mostrar listado de situación de alumno ayudante	Muestra un listado con todas las situaciones de alumnos ayudantes creadas brindando las opciones de “Ver detalles” y “Modificar” en el	Media	Media

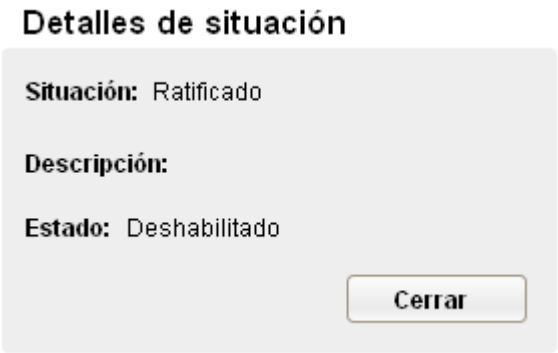
		área de íconos internos.		
	Prototipo			
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• gridSituaciones	• Varchar	• Solo lectura	
	Observaciones			

RF23	Crear situación de alumno ayudante	Permite crear una situación de alumno ayudante asociándole los campos: nombre, descripción y activo	Alta	Alta
	Prototipo			



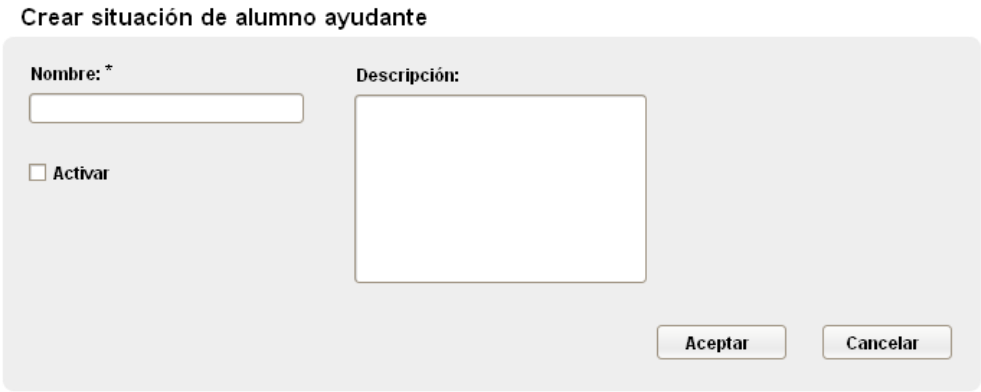
	Campos	Tipos de Datos	Reglas o Restricciones
	• textSituación • textareaDescripción • checkboxActivo	• Varchar • Varchar • Bool	• Campo obligatorio • Campo no obligatorio • Campo no obligatorio
	Observaciones		

RF24	Ver detalles de situación de alumno ayudante	Muestra los detalles de la situación de alumno ayudante seleccionada.	Baja	Baja
	Prototipo			



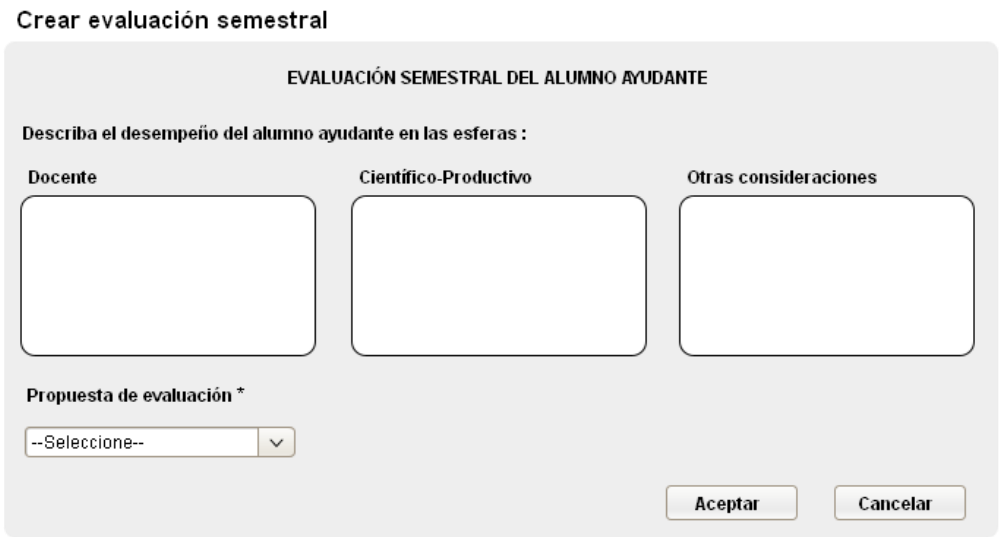
	Campos	Tipos de Datos	Reglas o Restricciones
	• textSituación • textDescripción • textEstado	• Varchar • Varchar • Varchar	• Solo lectura • Solo lectura • Solo lectura
	Observaciones		

RF25	Modificar situación de alumno ayudante	Permite modificar los datos de la situación de alumno ayudante seleccionada.	Alta	Alta
	Prototipo			



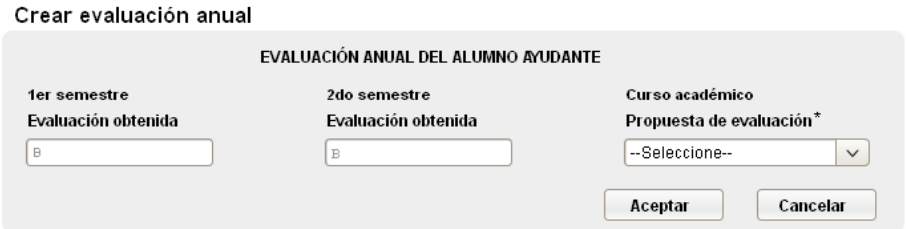
	Campos	Tipos de Datos	Reglas o Restricciones
	• textSituación • textAareaDescripción • checkboxActivo	• Varchar • Varchar • Bool	• Campo obligatorio • Campo no obligatorio • Campo no obligatorio
	Observaciones		

RF26	Modificar evaluación semestral de alumno ayudante	Permite modificar la evaluación semestral.	Alta	Alta
	Prototipo			



	Campos	Tipos de Datos	Reglas o Restricciones
	• textAreaDocente • textareaCientificoProd	• Varchar • Varchar • Varchar • Char	• El máximo de palabras admitidas es de un total de 300 de no más de 16

	uctivo • textAreaOtras • selectEvaluaciónFinal		caracteres cada una • El máximo de palabras admitidas es de un total de 300 de no más de 16 caracteres cada una • El máximo de palabras admitidas es de un total de 300 de no más de 16 caracteres cada una • Campo obligatorio y solo admite los siguientes caracteres: B, R y M
	Observaciones		

RF27	Modificar evaluación anual	Permite modificar la evaluación anual	Alta	Alta
	Prototipo			
				
	Campos	**Tipos de Datos**	**Reglas o Restricciones**	
	• textEvaluaciónPrimerSemestre • textEvaluaciónSegundoSemestre • selectEvalu	• Char • Char • Char	• Solo lectura • Solo lectura • Campo obligatorio y solo admite los siguientes caracteres: B, R y M	

	aciónAnual		
	Observaciones		

Anexo D. Diseño de casos de prueba (Técnica de caja negra)

DCP Asignación de plazas

- SC 1 Mostrar solicitudes aceptadas

Escenario	Descripción	Página	Cantidad por página	Respuesta del sistema	Flujo central
EC 1.1: Mostrar datos correctamente.	Mediante este escenario se muestra el listado de las solicitudes aceptadas.	NA	NA	El sistema muestra el listado de las solicitudes aceptadas	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se muestra un listado de las solicitudes aceptadas.

EC 1.2: Seleccionar cantidad por páginas .	Mediante este escenario se puede seleccionar la cantidad de elementos que se desean mostrar por página los cuales pueden ser: 5, 10, 15, 20.	V	V	El sistema muestra la cantidad de elementos por página seleccionado por el usuario.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se muestra un listado de las solicitudes aceptadas. 4- Se selecciona la cantidad de elementos a mostrar por página. 5- El sistema muestra la cantidad de elementos por página seleccionado por el usuario.
		NA	(5)		
EC 1.3: Seleccionar página Siguiente.	Mediante este escenario se puede seleccionar la próxima página.	NA	NA	El sistema muestra la página a continuación.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se muestra un listado de las solicitudes aceptadas. 4- Se selecciona la opción página siguiente. 5- El sistema muestra la página a continuación.
EC 1.4: Seleccionar página	Mediante este escenario se puede seleccionar	NA	NA	El sistema muestra la página que antecede.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y

Anterior.	la página. Anterior.				Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se muestra un listado de las solicitudes aceptadas. 4- Se selecciona la opción página anterior. 5- El sistema muestra la página anterior.

- SC 2 Asociar

Escenario	Descripción	Tipo de curso	Profesor disponibles (tutoría)	Cargo del alumno ayudante	Grupos disponibles	Activo	Respuesta del sistema	Flujo central
SC2.1: Asociar estudiante al Movimiento de Alumnos Ayudantes correctamente.	Se asocia un estudiante al movimiento de Alumnos Ayudantes correctamente.	V (Continuidad de estudios)	V (Alexey Peña Paz)	V (Presidente)	V (1301)	V (true)	El sistema muestra el mensaje “El elemento se ha asociado satisfactoriamente”.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se selecciona ícono Asignar el cual se encuentra a la derecha de cada solicitud aceptada.

								4- Se muestra un formulario donde se deben seleccionar todos los datos: - Tipo de curso - Profesor disponibles(tutoría) - Cargo del alumno ayudante - Grupos disponibles 5- Se selecciona la opción Aceptar. 6- El sistema muestra el mensaje “El elemento se ha asociado satisfactoriamente.” 7- Se muestra el listado de las solicitudes aceptadas.
SC2.2: Asociar estudiante al Movimiento de Alumnos Ayudantes dejando campos sin	Asociar un estudiante al Movimiento de alumnos Ayudantes dejando campos sin seleccionar.	I (vacío)	V (Alexey Peña Paz)	V (Presidente)	V (1301)	V (false)	El sistema muestra en cada campo obligatorio el mensaje “Campo requerido”.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a
		V (Curso Regular Diurno)	I (vacío)	V (Vice presidente)	V (vacío)	V (true)		

seleccionar.		V (Curso por encuentros)	V (Ariel Reyes Antuan)	I (vacío)	V (vacío)	V (false)		la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se selecciona ícono Asignar el cual se encuentra a la derecha de cada solicitud aceptada. 4- Se dejan campos sin seleccionar en el formulario que se muestra. 5- Se selecciona la opción Aceptar. 6- El sistema muestra en cada campo obligatorio el mensaje “Campo requerido”.
SC2.3: Asociar estudiante al Movimiento de Alumnos Ayudantes adjuntando plan de trabajo correctamente.	Asociar un estudiante al Movimiento de alumnos Ayudantes adjuntando plan de trabajo correctamente.	V (Continuidad de estudios)	V (Alexey Peña Paz)	V (Presidente)	V (1306)	V (true)	El sistema adjunta plan de trabajo al alumno ayudante.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación

								de plazas. 3- Se selecciona ícono Asignar el cual se encuentra a la derecha de cada solicitud aceptada. 4- Se seleccionan todos los campos del formulario que se muestra. 5- Se selecciona la opción Plan de trabajo del alumno ayudante. 6- Se selecciona el fichero a adjuntar, el mismo debe estar en formato pdf 7- Se selecciona la opción Aceptar. 8- El sistema adjunta plan de trabajo al alumno ayudante.
SC2.4: Asociar estudiante al Movimiento de Alumnos Ayudantes adjuntando plan de trabajo con formato incorrecto.	Asociar un estudiante al Movimiento de alumnos Ayudantes adjuntando plan de trabajo con formato incorrecto..	V (Continuidad de estudios)	V (Alexey Peña Paz)	V (Presidente)	V (1306)	V (true)	El sistema muestra el mensaje “Extensión no aceptada”	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía.

								2- Se selecciona la funcionalidad Asignación de plazas. 3- Se selecciona ícono Asignar el cual se encuentra a la derecha de cada solicitud aceptada. 4- Se seleccionan todos los campos del formulario que se muestra. 5- Se selecciona la opción Plan de trabajo del alumno ayudante. 6- Se selecciona el fichero a adjuntar, con formato inválido. 7- Se selecciona la opción Aceptar. 8- El sistema muestra el mensaje “Extensión no aceptada”.

SC2.5: Asociar estudiante al Movimiento de Alumnos Ayudantes adjuntando plan de trabajo sin especificar la ruta.	Asociar un estudiante al Movimiento de alumnos Ayudantes adjuntando plan de trabajo sin especificar la ruta.	V (Continuidad de estudios)	V (Alexey Peña Paz)	V (Presidente)	V (1306)	V (true)	El sistema muestra el mensaje “Campo requerido”	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se selecciona ícono Asignar el cual se encuentra a la derecha de cada solicitud aceptada. 4- Se seleccionan todos los campos del formulario que se muestra. 5- Se selecciona la opción Plan de trabajo del alumno ayudante. 6- Se selecciona el fichero a adjuntar, con formato válido. 7- Se selecciona la opción Aceptar. 8- El sistema muestra el mensaje

								“Campo requerido”.
SC2.6: Asociar estudiante al Movimiento de Alumnos Ayudantes adjuntando plan de trabajo incorrectamente.	Asociar un estudiante al Movimiento de alumnos Ayudantes adjuntando plan de trabajo incorrectamente.	V (Continuidad de estudios)	V (Alexey Peña Paz)	V (Presidente)	V (1306)	V (true)	El sistema muestra el mensaje “Campo requerido”	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se selecciona ícono Asignar el cual se encuentra a la derecha de cada solicitud aceptada. 4- Se seleccionan todos los campos del formulario que se muestra. 5- Se selecciona la opción Plan de trabajo del alumno ayudante. 6- Se selecciona el fichero a

								adjuntar.. 7- Se selecciona la opción Aceptar. 8- El sistema muestra el mensaje “El fichero no se pudo subir correctamente”.
EC 2.7 Asignar un alumno ayudante seleccionando la opción Cancelar	Asignar alumno ayudante seleccionando la opción Cancelar	NA	NA	NA	NA	NA	El sistema muestra el mensaje "¿Seguro desea realizar esta acción?".	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se selecciona ícono Asignar el cual se encuentra a la derecha de cada solicitud aceptada. 4- Se seleccionan todos los campos del formulario que se muestra. 5- Se adjunta plan de trabajo del alumno ayudante. 6- Se

								selecciona la opción Cancelar. 7- El sistema muestra el mensaje "¿Seguro desea realizar esta acción?". 8- Se selecciona la opci8ón aceptar del mensaje. 9- Se muestra el listado de solicitudes aceptadas.

- SC 3 Filtrar búsqueda por facultad del solicitante

Escenario	Descripción	Facultad	Respuesta del sistema	Flujo central
EC 3.1: Filtrar búsqueda por facultad del solicitante correctamente.	Se muestran las solicitudes aceptadas de la facultad por la que se filtró.	V (Facultad 1)	El sistema muestra un listado con las solicitudes aceptadas pertenecientes a la facultad seleccionada en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Facultas del solicitante. 4- Se selecciona la facultad por la cual se desea filtrar la búsqueda. 5- Se muestra un listado con las solicitudes aceptadas pertenecientes a la

				facultad seleccionada en el filtro.

- SC 4 Filtrar búsqueda por asignatura solicitada

Escenario	Descripción	Asignatura	Respuesta del sistema	Flujo central
EC 4.1: Filtrar búsqueda por asignatura solicitada correctamente.	Se muestran las solicitudes aceptadas de la asignatura por la que se filtró.	V (Algoritmia)	El sistema muestra un listado con las solicitudes aceptadas donde la asignatura coincida con la seleccionada en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Asignación de plazas. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Asignatura solicitada. 4- Se selecciona la asignatura por la cual se desea filtrar la búsqueda. 5- Se muestra un listado con las solicitudes aceptadas donde la asignatura coincida con la seleccionada en el filtro.

DCP Estructura

- SC 1 Mostrar estructura del movimiento de alumnos ayudantes

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1: Mostrar estructura del Movimiento de Alumnos Ayudantes	Se muestra la estructura del Movimiento de Alumnos Ayudantes	El sistema muestra la estructura del Movimiento de Alumnos Ayudantes	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Estructura. 3- Se muestra la estructura del Movimiento de Alumnos Ayudantes en la Universidad y por cada una de las facultades.

DCP Propuesta

- SC 1 Mostrar estudiantes

Escenario	Descripción	Página	Cantidad por página	Respuesta del sistema	Flujo central

EC 1.1: Mostrar datos correctamente.	Mediante este escenario se muestra el listado de los estudiantes	NA	NA	El sistema muestra el listado de estudiantes	4- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 5- Se selecciona la funcionalidad Propuesta. 6- Se muestra un listado de los estudiantes.
EC 1.2:	Mediante este	V	V	El sistema muestra la cantidad de	6- Se accede al subsistema de

Seleccionar cantidad por páginas .	escenario se puede seleccionar la cantidad de elementos que se desean mostrar por página los cuales pueden ser: 5, 10, 15, 20.	NA	(5)	elementos por página seleccionado por el usuario.	pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 7- Se selecciona la funcionalidad Propuesta. 8- Se muestra un listado de estudiantes. 9- Se selecciona la cantidad de elementos a mostrar por página. 10- El sistema muestra la cantidad de elementos por página seleccionado por el usuario.
EC 1.3: Seleccionar página Siguiente.	Mediante este escenario se puede seleccionar la próxima página.	NA	NA	El sistema muestra la página a continuación.	6- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 7- Se selecciona la funcionalidad Propuesta. 8- Se muestra un listado de estudiantes. 9- Se selecciona la opción página siguiente. 10- El sistema muestra la página a continuación.
EC 1.4: Seleccionar página Anterior.	Mediante este escenario se puede seleccionar la página. Anterior.	NA	NA	El sistema muestra la página que antecede.	6- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía.

					7- Se selecciona la funcionalidad Propuesta. 8- Se muestra un listado de estudiantes a proponer. 9- Se selecciona la opción página anterior. 10- El sistema muestra la página anterior.

- SC 2 Proponer estudiante

Escenario	Descripción	Buscar(nombre, apellidos, CI, usuario, solapín)	Respuesta del sistema	Flujo central
EC 2.1: Proponer estudiante correctamente.	Proponer estudiante correctamente	V (dgalardy)	El sistema muestra el mensaje “El elemento ha sido creado satisfactoriamente.”.	8- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 9- Se selecciona la funcionalidad Propuesta. 10- Se introduce el criterio de búsqueda, se selecciona la opción Buscar. 11- Se muestra un listado de estudiantes según la búsqueda realizada. 12- Se selecciona el estudiante o los estudiantes que se deseen proponer. 13- Se selecciona la opción Aceptar. 14- Se muestra el mensaje “El elemento ha sido creado satisfactoriamente.”. 15- Se selecciona la opción Aceptar del mensaje. 16- Se muestra la interfaz para realizar otra búsqueda.
		V (89101403817)		
		V (E11952)		
		V (Yaima)		
		V (Salina)		

EC 2.2: Proponer estudiante sin realizar búsqueda.	Proponer estudiante sin realizar búsqueda	I (vacío)	El sistema muestra el mensaje “Debe proponer al menos un estudiante”.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Propuesta. 3- No se introduce criterio de búsqueda. 4- Se selecciona la opción Aceptar. 5- Se muestra el mensaje “Debe proponer al menos un estudiante”.
EC 2.3: Realizar propuesta seleccionando la opción Cancelar	Realizar seleccionando la opción Cancelar	NA	El sistema muestra el mensaje "¿Seguro desea realizar esta acción?".	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Propuesta. 3- Se introduce el criterio de búsqueda. 4- Se selecciona la opción Cancelar. 5- Se muestra el mensaje "¿Seguro desea realizar esta acción?". 6- Se selecciona la opción Aceptar del mensaje. 7- Se muestra la interfaz para realizar una nueva búsqueda.

- SC 3 Filtrar búsqueda por facultad

Escenario	Descripción	Facultad	Respuesta del sistema	Flujo central

EC 3.1: Filtrar búsqueda por facultad del solicitante correctamente.	Se muestran las solicitudes aceptadas de la facultad por la que se filtró.	V (Facultad 1)	El sistema muestra un listado con los estudiantes pertenecientes a la facultad seleccionada en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Propuesta. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Facultad del estudiante. 4- Se selecciona la facultad por la cual se desea filtrar la búsqueda. 5- Se muestra un listado con los estudiantes pertenecientes a la facultad seleccionada en el filtro.

- SC 4 Filtrar búsqueda por grupo

Escenario	Descripción	Grupo docente	Respuesta del sistema	Flujo central

EC 4.1: Filtrar búsqueda por grupo correctamente.	Se muestran los estudiantes del grupo por el que se filtró.	V (1304)	El sistema muestra un listado con los estudiantes que pertenecen al grupo seleccionado en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Propuesta. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Grupo. 4- Se selecciona el grupo por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con los estudiantes que pertenecen al grupo seleccionado en el filtro.

- SC 5 Filtrar búsqueda por año académico

Escenario	Descripción	Año académico	Respuesta del sistema	Flujo central

EC 5.1: Filtrar búsqueda por año académico.	Se muestran los estudiantes del año académico seleccionado.	V (Cuarto año)	El sistema muestra un listado con los estudiantes que pertenecen al año académico seleccionado.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Propuesta. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro año académico. 4- Se selecciona el año académico por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con los estudiantes que pertenecen al año académico seleccionado en el filtro.

- SC 6 Filtrar búsqueda por sexo

Escenario	Descripción	Sexo	Respuesta del sistema	Flujo central
EC 5.1: Filtrar búsqueda *por sexo*.	Se muestran los estudiantes del sexo seleccionado.	V (Femenino)	El sistema muestra un listado con los estudiantes que pertenecen al sexo seleccionado.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Propuesta. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro sexo. 4- Se selecciona el sexo por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con los estudiantes que

				pertenecen al sexo seleccionado en el filtro.

DCP Registro general de alumnos ayudantes

- SC 1 Mostrar listado de alumnos ayudantes

Escenario	Descripción	Página	Cantidad por página	Respuesta del sistema	Flujo central

EC 1.1: Mostrar datos correctamente.	Mediante este escenario se muestra el listado de los AA.	NA	NA	El sistema muestra el listado de los AA.	7- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 8- Se selecciona la funcionalidad Registro general de AA. 9- Se 10- Se muestra un listado de los AA.
EC 1.2: Seleccionar cantidad por páginas .	Mediante este escenario se puede seleccionar la cantidad de elementos que se desean mostrar por página los cuales pueden ser: 5, 10, 15, 20.	V NA	V (5)	El sistema muestra la cantidad de elementos por página seleccionado por el usuario.	11- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 12- Se selecciona la funcionalidad Registro general de AA. 13- Se muestra un listado de AA. 14- Se selecciona la cantidad de elementos a mostrar por página. 15- El sistema muestra la

					cantidad de elementos por página seleccionado por el usuario.
EC 1.3: Seleccionar página Siguiente.	Mediante este escenario se puede seleccionar la próxima página.	NA	NA	El sistema muestra la página a continuación.	11- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 12- Se selecciona la funcionalidad Registro general de AA. 13- Se muestra un listado de AA. 14- Se selecciona la opción página siguiente. 15- El sistema muestra la página a continuación.
EC 1.4: Seleccionar página Anterior.	Mediante este escenario se puede seleccionar la página. Anterior.	NA	NA	El sistema muestra la página que antecede.	11- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 12- Se selecciona la funcionalidad Registro general de AA. 13- Se muestra un listado

					de AA. 14- Se selecciona la opción página anterior. 15- El sistema muestra la página anterior.

- SC 2 Buscar alumno ayudante

Escenario	Descripción	Buscar(nombre, apellidos, CI, usuario, solapín)	Respuesta del sistema	Flujo central
EC 2.1: Buscar AA	Buscar AA	V (dgalardy)	El sistema muestra un listado con los AA asociados a la búsqueda realizada.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se introduce el criterio de búsqueda, se selecciona la opción Buscar. 4- Se muestra un listado de AA según el criterio de búsqueda introducido.
		V (89101403817)		
		V (E11952)		
		V (Yaima)		
		V (Salina)		

- SC 3 Activar/Desactivar

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 3.1: Desactivar o activar alumno ayudante.	Desactivar o activar alumno ayudante	El sistema muestra un listado con los AA asociados a la búsqueda realizada.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se introduce el criterio de búsqueda, se selecciona la opción Buscar. 4- Se muestra un listado de AA según el criterio

			de búsqueda introducido. 5- Se selecciona la opción Activar/Desactivar que se encuentra a la derecha de cada AA. 6- Se activa o desactiva el AA en el sistema.

- SC 4 Crear evaluación semestral

Escenario	Descripción	Docente	Científico-Productivo	Otras consideraciones	Propuesta de evaluación	Respuesta del sistema	Flujo central
EC 4.1: Crear evaluación semestral correctamente.	Se crea una evaluación semestral correctamente.	V (Muy buen desempaño)	V(Muy buen desempeño)	V (vacio)	V(B)	El sistema muestra el mensaje "El elemento ha sido creado satisfactoriamente."	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Crear evaluación semestral el cual se encuentra a la derecha de cada AA. 4- Se muestra un formulario el cual debe ser llenado correctamente: - Docente - Científico-productivo - Otras considera
		V (vacío)	V (vacío)	V (vacío)	V(M)		

							ciones - Propuesta de evaluación 5- Se selecciona la opción Aceptar. 6- Se muestra el mensaje “El elemento ha sido creado satisfactoriamente.” 7- Se selecciona la opción Aceptar del mensaje mostrado. 8- Se muestra la interfaz para realizar la búsqueda nuevamente.
EC 4.2: Crear evaluación semestral sin seleccionar el campo Propuesta de evaluación.	Crear una evaluación semestral sin seleccionar el campo Propuesta de evaluación.	V (vacío)	V(Muy buen desempeño)	V (vacio)	I (vacío)	El sistema muestra el mensaje “Campo requerido”.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Crear evaluación semestral el cual se encuentra a la derecha de cada AA. 4- Se llena el

							formulario sin seleccionar el campo Propuesta de evaluación. 5- Se selecciona la opción Aceptar. 6- En el campo obligatorio se muestra el mensaje “Campo requerido”.
EC 4.3: Crear evaluación semestral seleccionando la opción Cancelar.	Crear una evaluación semestral seleccionando la opción Cancelar.	NA	NA	NA	NA	El sistema muestra el mensaje “¿Está seguro de realizar la acción?”.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Crear evaluación semestral el cual se encuentra a la derecha de cada AA. 4- Se llena el formulario y se selecciona la opción Cancelar. 5- Se muestra el mensaje “¿Está seguro de realizar la acción?”. 6- Se selecciona

							la opción Aceptar del mensaje. 7- Se muestra la interfaz para realizar la búsqueda nuevamente.

- SC 5 Crear evaluación anual

Escenario	Descripción	Primer semestre	Segundo semestre	Propuesta de evaluación	Respuesta del sistema	Flujo central
EC 5.1: Crear evaluación anual correctamente.	Se crea una evaluación anual correctamente.	NA	NA	V (M)	El sistema muestra el mensaje "El elemento ha sido creado satisfactoriamente."	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Crear evaluación anual el cual se encuentra a la derecha de cada AA. 4- Se muestra un formulario el cual debe ser llenado correctamente: - Propuesta de evaluación 5- Se selecciona la opción Aceptar. 6- Se muestra el mensaje "El elemento ha sido creado satisfactoriamente."

						7- Se selecciona la opción Aceptar del mensaje mostrado. 8- Se muestra la interfaz para realizar la búsqueda nuevamente.
EC 5.2: Crear evaluación anual sin seleccionar el campo Propuesta de evaluación.	Crear una evaluación anual sin seleccionar el campo Propuesta de evaluación.	NA	NA	I (vacío)	El sistema muestra el mensaje “Campo requerido”.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Crear evaluación anual el cual se encuentra a la derecha de cada AA. 4- Se llena el formulario sin seleccionar el campo Propuesta de

						evaluación. 5- Se selecciona la opción Aceptar. 6- En el campo obligatorio se muestra el mensaje “Campo requerido”.
EC 5.3: Crear evaluación anual seleccionando la opción Cancelar.	Crear una evaluación anual seleccionando la opción Cancelar.	NA	NA	NA	El sistema muestra el mensaje “¿Está seguro de realizar la acción?”.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Crear evaluación anual el cual se encuentra a la derecha de cada AA. 4- Se llena el formulario y se selecciona la opción Cancelar. 5- Se muestra el mensaje “¿Está seguro de realizar la acción?”. 6- Se selecciona la opción Aceptar del mensaje. 7- Se muestra la interfaz para realizar la búsqueda nuevamente.

- SC 6 Detalles de alumno ayudante

Escenario	Descripción	Respuesta del sistema	Flujo central

EC 6.1: Detalles del AA.	Se muestran los detalles del AA seleccionado.	El sistema muestra una ventana con los detalles del AA seleccionado.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Detalles el cual se encuentra a la derecha de cada AA. 4- Se muestra una ventana con los detalles del AA seleccionado: - Tutor - Departamento - Grupos asociados - Estado 5- Se selecciona la opción Aceptar de la ventana. 6- Se muestra el listado de los AA registrados en el sistema.

- SC 7 Listar evaluaciones de alumno ayudante

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 7.1: Listar evaluaciones del AA.	Se muestran las evaluaciones del AA seleccionado.	El sistema muestra un listado con las evaluaciones del AA seleccionado.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Listar evaluaciones del AA seleccionado el cual se encuentra a la derecha de cada uno. 4- Se muestra un listado con las evaluaciones asociadas al AA seleccionado.

- SC 8 Modificar datos de alumno ayudante

Escenario	Descripción	Tipo de curso	Profesores disponi	Cargo del AA	Asignatura	Departamento	Profesores disponi	Respuesta del sistema	Flujo central

			bles)				**bles**		
EC 8.1: Modificar datos de AA correctamente seleccionando la opción Cambio de asignatura.	Se modifican los datos de un AA correctamente Cambio de asignatura	V (Curso Regular Diurno)	V (Jordanis Viltres Chávez)	V (Ninguno)	V (Idioma Extranjero 2)	NA	NA	El sistema muestra el mensaje "El elemento ha sido modificado satisfactoriamente."	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona el ícono Modificar el cual se encuentra a la derecha de cada AA. 4- Se muestra un formulario donde se podrán modificar los datos del AA. 5- Se

									seleccion a la opción Cambio de asignatur a. 6- Se seleccion a la opción Aceptar. 7- El sistema muestra el mensaje “El elemento ha sido modificad o satisfacto riamente. ” 8- Se seleccion a la opción Aceptar del mensaje. 9- Se muestra la interfaz para la realizar la búsqued a nuevame nte.

EC 8.2: Modificar datos de AA correctamente seleccionando la opción Cambio de departamento.	Se modifica los datos de un AA correctamente Cambio de departamento.	V (Curso por encuentros)	V (Olivia Rodriguez Abril)	V (Presidente UCI)	V (Idioma Extranjero 2)	V (Departamento Ciencias Básicas)	V (Yeily Delgado Cruz)	El sistema muestra el mensaje “El elemento ha sido modificado satisfactoriamente.”	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Modificar el cual se encuentra a la derecha de cada AA. 4- Se muestra un formulario donde se podrán modificar los datos del AA. 5- Se selecciona la

									opción Cambio de departamento. 6- Se selecciona la opción Aceptar. 7- El sistema muestra el mensaje “El elemento ha sido modificado satisfactoriamente.” 8- Se selecciona la opción Aceptar del mensaje. 9- Se muestra la interfaz para la realizar la búsqueda nuevamente.
EC 8.3: Modificar datos de AA dejando campos requerido	Se modifica los datos de un AA dejando campos requerido	V (Educación a distancia)	V (Ariel Reyes Antuan)	V (Vice-Presidente UCI)	I (vacío)	I (vacío)	I (vacío)	El sistema muestra el mensaje “Debe completar todos los campos.”	1- Se accede al subsistema de pregrado, se seleccion

s sin seleccionar.	s sin seleccionar.	V (Educación a distancia)	V (Ariel Reyes Antuan)	V (Vice-Presidente UCI)	I (vacío)	NA	NA		a al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Modificar el cual se encuentra a la derecha de cada AA. 4- Se muestra un formulario donde se podrán modificar los datos del AA. 5- Se dejan campos requeridos sin seleccionar. 6- Se selecciona la opción

									Aceptar. 7- El sistema muestra el mensaje “Debe completar todos los campos.”
EC 8.4: Modificar datos de AA adjuntando plan de trabajo correctamente.	Se modifica los datos de un AA adjuntando plan de trabajo correctamente.	V (Educación a distancia)	V (Ariel Reyes Antuan)	V (Vice-Presidente UCI)	V (Matemática 3)	NA	NA	El sistema adjunta plan de trabajo al alumno ayudante.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Modificar el cual se encuentra a la derecha

									de cada AA. 4- Se muestra un formulario donde se podrán modificar los datos del AA. 5- Se selecciona la opción Plan de trabajo del alumno ayudante. 6- Se selecciona el fichero a adjuntar, el mismo debe estar en formato pdf 7- Se selecciona la opción Aceptar. 8- El sistema adjunta plan de trabajo al alumno ayudante.

EC 8.5: Modificar datos de AA adjuntando plan de trabajo con formato incorrecto	Se modifican los datos de un AA adjuntando plan de trabajo con formato incorrecto.	V (Educación a distancia)	V (Ariel Reyes Antuan)	V (Vice-Presidente UCI)	V (Matemática 3)	V (Departamento Ciencias Básicas)	V (Yeily Delgado Cruz)	El sistema muestra el mensaje “Extensión no aceptada”	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Modificar el cual se encuentra a la derecha de cada AA. 4- Se muestra un formulario donde se podrán modificar los datos del AA. 5- Se selecciona la

									opción Plan de trabajo del alumno ayudante. 6- Se selecciona el fichero a adjuntar, con formato inválido. 7- Se selecciona la opción Aceptar. 8- El sistema muestra el mensaje “Extensión no aceptada”.
EC 8.6: Modificar datos de AA adjuntando plan de trabajo sin especificar la ruta.	Se modifican los datos de un AA adjuntando plan de trabajo sin especificar la ruta.	V (Educación a distancia)	V (Ariel Reyes Antuan)	V (Vice-Presidente UCI)	V (Matemática 3)	V (Departamento Ciencias Básicas)	V (Yeily Delgado Cruz)	El sistema muestra el mensaje “Campo requerido”.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se seleccion

									a la funcionalidad Registro general de AA. 3- Se selecciona ícono Modificar el cual se encuentra a la derecha de cada AA. 4- Se muestra un formulario donde se podrán modificar los datos del AA. 5- Se selecciona la opción Plan de trabajo del alumno ayudante. 6- Se selecciona el fichero a adjuntar, con formato válido. 7- Se selecciona la opción Aceptar. 8- El

									sistema muestra el mensaje “Campo requerido”.
EC 8.7: Modificar datos de AA seleccionando la opción Cancelar	Modificar datos de AA seleccionando la opción Cancelar	NA	NA	NA	NA	NA	NA	El sistema muestra el mensaje “¿Está seguro de realizar la acción?”	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona ícono Modificar el cual se encuentra a la derecha de cada solicitud. 4- Se

									muestra un formulario donde se podrá modificar el estado de la solicitud. 5- Se selecciona la opción Cancelar de la ventana. 6- Se muestra el mensaje "¿Está seguro de realizar la acción?" 7- Se selecciona la opción Aceptar del mensaje. 8- Se muestra la interfaz para realizar la búsqueda.

- SC 9 Eliminar alumno ayudante

Escenario	Descripción	Respuesta del sistema	Flujo central

EC 9.1: Eliminar AA correctamente.	Se elimina un AA correctamente.	El sistema muestra el mensaje "¿Está seguro de realizar la acción?".	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona la opción Eliminar que se encuentra a la derecha de cada AA. 4- Se muestra el mensaje "¿Está seguro de realizar la acción?". 5- Se selecciona la opción Aceptar del mensaje. 6- Se muestra la interfaz para realizar la búsqueda.

- SC 10 Actualizar

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 10.1: Actualizar listado de AA.	Se actualiza el listado de AA.	El sistema actualiza el listado de AA	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se accede al ícono Actualizar que se encuentra en la parte superior derecha de la interfaz. 4- Se actualiza el listado de AA.

- SC 11 Ayuda

Escenario	Descripción	Respuesta del sistema	Flujo central

EC 11.1: Ayuda.	Se brinda ayuda para una mejor comprensión de la funcionalidad.	El sistema muestra una ayuda.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona la opción Ayuda que se encuentra en la esquina superior derecha de la interfaz. 4- Se muestra una ventana explicando las acciones que se pueden realizar en esta funcionalidad. 5- Se selecciona la opción Cerrar. 6- Se muestra el listado con los AA.

- SC 12 Filtrar búsqueda por facultad

Escenario	Descripción	Facultad	Respuesta del sistema	Flujo central
EC 12.1: Filtrar búsqueda por facultad del AA correctamente.	Se muestran los AA de la facultad por la que se filtró.	V (Facultad 1)	El sistema muestra un listado con los AA pertenecientes a la facultad seleccionada en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Facultad. 4- Se selecciona la facultad por la cual se desea filtrar la búsqueda. 5- Se muestra un listado con los AA pertenecientes a la facultad seleccionada en el filtro.

- SC 13 Filtrar búsqueda por grupo

Escenario	Descripción	Grupo	Respuesta del sistema	Flujo central
EC 13.1: Filtrar búsqueda por grupo correctamente.	Se muestran los AA del grupo por el que se filtró.	V (1304)	El sistema muestra un listado con los AA pertenecientes al grupo seleccionado en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Grupo. 4- Se selecciona el grupo por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con los AA que pertenecen al grupo seleccionado en el filtro.

- SC 14 Filtrar búsqueda por sexo

Escenario	Descripción	Sexo	Respuesta del sistema	Flujo central

EC 14.1: Filtrar búsqueda por sexo.	Se muestran los AA del sexo seleccionado.	V (Femenino)	El sistema muestra los AA que han realizado solicitudes y que pertenecen al sexo seleccionado en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Sexo. 4- Se selecciona el sexo por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con los AA que pertenecen al sexo seleccionado en el filtro.

- SC 15 Filtrar búsqueda por asignatura

Escenario	Descripción	Asignatura	Respuesta del sistema	Flujo central
EC 15.1: Filtrar búsqueda por asignatura.	Mostrar los AA de la asignatura seleccionada.	V (Física I)	El sistema muestra los AA de la asignatura seleccionada.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Asignatura. 4- Se selecciona la asignatura por la cual se desea filtrar la

				búsqueda. 5- Se muestra un listado con los AA correspondientes a la asignatura seleccionada.

- SC 16 Filtrar búsqueda por departamento

Escenario	Descripción	Departamento	Respuesta del sistema	Flujo central
EC 16.1: Filtrar búsqueda por departamento.	Mostrar los AA del departamento seleccionado.	V (Ingeniería de Software)	El sistema muestra los AA que pertenecen al departamento seleccionado.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Departamento. 4- Se selecciona el departamento. 5- Se muestra un listado con los AA correspondientes al departamento seleccionado.

- SC 17 Filtrar búsqueda por situación

Escenario	Descripción	Situación	Respuesta del sistema	Flujo central
EC 17.1: Filtrar búsqueda por situación.	Mostrar los AA pertenecientes a la situación seleccionada.	V (Desvinculado)	El sistema muestra los AA que se encuentra en la situación seleccionada.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Situación. 4- Se selecciona la situación por la cual se desea filtrar la búsqueda. 5- Se muestra un listado con los AA correspondientes a la situación seleccionada.

- SC 18 Filtrar búsqueda por tutor

Escenario	Descripción	Tutor	Respuesta del sistema	Flujo central

EC 18.1: Filtrar búsqueda por tutor.	Mostrar los AA que son tutorados por el tutor seleccionado.	V (Yeily Delgado Cruz)	El sistema muestra los AA tutorados por el tutor seleccionado.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de AA. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Tutor. 4- Se selecciona el tutor por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con los AA que son tutorados por el tutor seleccionado.

DCP Registro general de solicitudes

- SC 1 Mostrar solicitudes

Escenario	Descripción	Página	Cantidad por página	Respuesta del sistema	Flujo central

EC 1.1: Mostrar datos correctamente.	Mediante este escenario se muestra el listado de las solicitudes	NA	NA	El sistema muestra el listado de las solicitudes	11- Se accede al subsistema de pregrado, se selecciona el módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 12- Se selecciona la funcionalidad Registro general de solicitudes. 13- Se muestra un listado de las solicitudes.
EC 1.2: Seleccionar cantidad por páginas .	Mediante este escenario se puede seleccionar la cantidad de elementos que se desean mostrar por página los cuales pueden ser: 5, 10, 15, 20.	V NA	V (5)	El sistema muestra la cantidad de elementos por página seleccionado por el usuario.	16- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 17- Se selecciona la funcionalidad Registro general de solicitudes. 18- Se muestra un listado de solicitudes. 19- Se selecciona la cantidad de elementos a mostrar por página. 20- El sistema muestra la

					cantidad de elementos por página seleccionado por el usuario.
EC 1.3: Seleccionar página Siguiente.	Mediante este escenario se puede seleccionar la próxima página.	NA	NA	El sistema muestra la página a continuación.	16- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 17- Se selecciona la funcionalidad Registro general de solicitudes. 18- Se muestra un listado de solicitudes. 19- Se selecciona la opción página siguiente. 20- El sistema muestra la página a continuación.
EC 1.4: Seleccionar página Anterior.	Mediante este escenario se puede seleccionar la página. Anterior.	NA	NA	El sistema muestra la página que antecede.	16- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 17- Se selecciona la funcionalidad Registro general de solicitudes. 18- Se muestra un listado

					de solicitudes. 19- Se selecciona la opción página anterior. 20- El sistema muestra la página anterior.

- SC 2 Registro general de solicitudes

Escenario	Descripción	Buscar(nombre, apellidos, CI, usuario, solapín)	Respuesta del sistema	Flujo central
EC 2.1: Acceder al registro de solicitudes de un estudiante.	Acceder al registro de solicitudes de un estudiante.	V (dgalardy)	El sistema muestra un listado con las solicitudes asociadas a la búsqueda realizada.	17- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 18- Se selecciona la funcionalidad Registro general de solicitudes. 19- Se introduce el criterio de búsqueda, se selecciona la opción Buscar. 20- Se muestra un listado de solicitudes según la el criterio de búsqueda introducido.
		V (89101403817)		
		V (E11952)		
		V (Yaima)		
		V (Salina)		
EC 2.2: Acceder al registro de solicitudes sin realizar búsqueda del estudiante.	Acceder al registro de solicitudes sin realizar búsqueda del estudiante.	V (vacío)	El sistema muestra un listado con las solicitudes realizadas por los estudiantes	6- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 7- Se selecciona la funcionalidad Registro general de solicitudes. 8- Se introduce el criterio de búsqueda o se deja vacío. 9- Se selecciona la opción Buscar. 10- Se muestra el listado de las solicitudes realizadas de todos los estudiantes.
		V (ysalinaa)		

- SC 3 Detalles de solicitud

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 3.1: Detalles de la solicitud.	Se muestran los detalles de la solicitud seleccionada.	El sistema muestra una ventana con los detalles de la solicitud seleccionada.	9- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 10- Se selecciona la funcionalidad Registro general de solicitudes. 11- Se selecciona ícono Detalles el cual se encuentra a la derecha de cada solicitud. 12- Se muestra una ventana con los detalles de la solicitud seleccionada: - Asignatura - Fecha de registro - Estado 13- Se selecciona la opción Aceptar de la ventana 14- Se muestra el listado de las solicitudes registradas en el sistema.

- SC 4 Modificar estado de solicitud

Escenario	Descripción	Asignatura	Fecha de registro	Estado	Respuesta del sistema	Flujo central
EC 4.1: Modificar solicitud correctamente.	Se modifica una solicitud correctamente	NA	NA	V (Aceptada)	El sistema muestra el mensaje “El elemento ha sido modificado satisfactoriamente.”	10- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 11- Se selecciona la funcionalidad Registro general de solicitudes. 12- Se selecciona ícono Modificar el cual se

						encuentra a la derecha de cada solicitud. 13- Se muestra un formulario donde se podrá modificar el estado de la solicitud. 14- Se selecciona la opción Aceptar de la ventana 15- Se muestra el listado de las solicitudes registradas en el sistema.
EC 4.2: Modificar solicitud seleccionando la opción Cancelar	Modificar una solicitud seleccionando la opción Cancelar	NA	NA	NA	El sistema muestra el mensaje "¿Está seguro de realizar la acción?"	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de solicitudes. 3- Se selecciona ícono Modificar el cual se encuentra a la derecha de cada solicitud. 4- Se muestra un formulario donde se podrá modificar el estado de la solicitud. 5- Se selecciona la opción Cancelar de la ventana. 6- Se muestra el mensaje "¿Está seguro de realizar la acción?" 7- Se selecciona la opción Aceptar del mensaje.

						8- Se muestra la interfaz para realiza la búsqueda.

- SC 5 Filtrar búsqueda por facultad

Escenario	Descripción	Facultad	Respuesta del sistema	Flujo central
EC 5.1: Filtrar búsqueda por facultad del solicitante correctamente.	Se muestran las solicitudes de la facultad por la que se filtró.	V (Facultad 1)	El sistema muestra un listado con los estudiantes pertenecientes a la facultad seleccionada en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de solicitudes. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Facultad del solicitante. 4- Se selecciona la facultad por la cual se desea filtrar la búsqueda. 5- Se muestra un listado con los estudiantes pertenecientes a la facultad seleccionada en el filtro.

- SC 6 Filtrar búsqueda por grupo

Escenario	Descripción	Grupo docente	Respuesta del sistema	Flujo central
EC 6.1: Filtrar búsqueda por grupo docente correctamente.	Se muestran los estudiantes del grupo por el que se filtró.	V (1304)	El sistema muestra un listado con las solicitudes pertenecientes al grupo seleccionado en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de solicitudes. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Grupo del solicitante. 4- Se selecciona el grupo por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con las solicitudes que pertenecen al grupo seleccionado en el filtro.

- SC 7 Filtrar búsqueda por sexo del solicitante

Escenario	Descripción	Sexo	Respuesta del sistema	Flujo central

EC 7.1: Filtrar búsqueda por sexo del solicitante.	Se muestran los estudiantes del sexo seleccionado.	V (Femenino)	El sistema muestra los estudiantes que han realizado solicitudes y que pertenecen al sexo seleccionado en el filtro.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de solicitudes. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Sexo del solicitante. 4- Se selecciona el sexo por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con los estudiantes que ha realizado solicitudes y pertenecen al sexo seleccionado en el filtro.

- SC 8 Filtrar búsqueda por asignatura solicitada

Escenario	Descripción	Asignatura	Respuesta del sistema	Flujo central
EC 8.1: Filtrar búsqueda por asignatura solicitada.	Mostrar las solicitudes de la asignatura seleccionada.	V (Física I)	El sistema muestra las solicitudes de la asignatura seleccionada.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de solicitudes. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Asignatura solicitada. 4- Se selecciona la asignatura por la cual

				se desea filtrar la búsqueda. 5- Se muestra un listado con las solicitudes correspondientes a la asignatura seleccionada.

- SC 9 Filtrar búsqueda por estado de solicitud

Escenario	Descripción	Estado	Respuesta del sistema	Flujo central
EC 9.1: Filtrar búsqueda por estado de solicitud.	Mostrar las solicitudes de la asignatura seleccionada.	V (Aceptada)	El sistema muestra las solicitudes en estado igual al seleccionado.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de solicitudes. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Estado de solicitud. 4- Se selecciona el estado de solicitud por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con las solicitudes correspondientes al estado seleccionado.

- SC 10 Filtrar búsqueda por fecha de registro

Escenario	Descripción	Fecha de registro	Respuesta del sistema	Flujo central
EC 10.1: Filtrar búsqueda por fecha de registro.	Mostrar las solicitudes de las pertenecientes a la fecha de registro seleccionada.	V (18/01/2002)	El sistema muestra las solicitudes registradas en la fecha seleccionada.	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Registro general de solicitudes. 3- Se selecciona de la lista desplegable Filtro de búsqueda el filtro Fecha de registro. 4- Se selecciona la fecha de registro por el cual se desea filtrar la búsqueda. 5- Se muestra un listado con las solicitudes correspondientes a la fecha de registro seleccionada.

DCP Solicitud

- SC 1 Mostrar solicitudes

Escenario	Descripción	Página	Cantidad por página	Respuesta del sistema	Flujo central

EC 1.1: Mostrar datos correctamente.	Mediante este escenario se muestra el listado de las solicitudes realizadas asociadas al usuario registrado en el sistema	V NA	V NA	El sistema muestra el listado de las solicitudes realizadas	14- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 15- Se selecciona la funcionalidad Solicitud. 16- Se muestra un listado de las solicitudes asociadas al usuario autenticado en el sistema.
EC 1.2:	Mediante este	V	V	El sistema muestra la cantidad de	21- Se accede al subsistema de

Seleccionar cantidad por páginas .	escenario se puede seleccionar la cantidad de elementos que se desean mostrar por página los cuales pueden ser: 5, 10, 15, 20.	NA	(5)	elementos por página seleccionado por el usuario.	pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 22- Se selecciona la funcionalidad Solicitud. 23- Se muestra un listado de las solicitudes registradas. 24- Se selecciona la cantidad de elementos a mostrar por página. 25- El sistema muestra la cantidad de elementos por página seleccionado por el usuario.
EC 1.3: Seleccionar página Siguiente.	Mediante este escenario se puede seleccionar la próxima página.	NA	NA	El sistema muestra la página a continuación.	21- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 22- Se selecciona la funcionalidad Solicitud. 23- Se muestra un listado de las solicitudes registradas. 24- Se selecciona la opción página siguiente. 25- El sistema muestra la página a continuación.
EC 1.4: Seleccionar página Anterior.	Mediante este escenario se puede seleccionar la página.	NA	NA	El sistema muestra la página que antecede.	21- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación

	Anterior.				funcional Ayudantía. 22- Se selecciona la funcionalidad Solicitud. 23- Se muestra un listado de las solicitudes registradas. 24- Se selecciona la opción página anterior. 25- El sistema muestra la página anterior.

- SC 2 Registrar nueva solicitud

Escenario	Descripción	Disciplina	Asignaturas	Respuesta del sistema	Flujo central

EC 2.1: Registrar una solicitud correctamente	Se registra una solicitud correctamente	V (Preparación para la Defensa)	V (Seguridad Nacional)	El sistema muestra el mensaje “”.	17- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 18- Se selecciona la funcionalidad Solicitud. 19- Se selecciona ícono registrar nueva solicitud el cual se encuentra en la esquina superior izquierda de la interfaz. 20- Se selecciona la disciplina y según la disciplina seleccionada se mostraran las asignaturas disponibles pertenecientes a la misma. 21- Se selecciona(n) la(s) asignatura(s). 22- Se selecciona la opción Asociar. 23- Se muestra un listado con los resultados finales de la solicitud donde se muestra de cada asignatura nota, promedio general y recomendación. 24- Se selecciona la opción Aceptar. 25- En caso de que la nota y el promedio general sean mayor o igual que 4 puntos se muestra el mensaje “.”. 26- Se muestra un

					listado con las solicitudes asociadas al usuario registrado en el sistema.
EC 2.2: Registrar una	Registrar una solicitud sin	I	V	El sistema muestra	26- Se accede al subsistema de

solicitud sin seleccionar una disciplina.	seleccionar una disciplina.	(vacío)	NA	el mensaje "Debe seleccionar una disciplina."	pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 27- Se selecciona la funcionalidad Solicitud. 28- Se selecciona ícono registrar nueva solicitud el cual se encuentra en la esquina superior izquierda de la interfaz. 29- No se selecciona ninguna disciplina 30- Se selecciona la opción Asociar. 31- El sistema muestra el mensaje "Debe seleccionar una disciplina." 32- Se selecciona la opción Aceptar del mensaje. 33- Se muestra la interfaz inicial para seleccionar una disciplina.
EC 2.3: Registrar una solicitud sin seleccionar asignaturas.	Registrar una solicitud sin seleccionar asignatura.	V (Matemática)	I (vacío)	El sistema muestra el mensaje "Debe seleccionar al menos una asignatura."	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Solicitud. 3- Se selecciona ícono registrar nueva solicitud el cual se encuentra en la

					esquina superior izquierda de la interfaz. 4- Se selecciona una disciplina. 5- No se selecciona ninguna asignatura. 6- Se selecciona la opción Asociar. 7- El sistema muestra el mensaje “Debe seleccionar al menos una asignatura.” 8- Se selecciona la opción Aceptar del mensaje. 9- Se muestra la interfaz inicial para seleccionar una disciplina.
EC 2.4: Registrar una solicitud no recomendada	Registrar una solicitud no recomendada	V (Física)	V (Física 1)	El sistema muestra el mensaje “Por el momento no puede crear la solicitud.”	1- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 2- Se selecciona la funcionalidad Solicitud. 3- Se selecciona ícono registrar nueva solicitud el cual se encuentra en la esquina superior izquierda de la interfaz. 4- Se selecciona una disciplina. 5- Se selecciona una o varias asignaturas. 6- Se selecciona la opción Asociar. 7- Se selecciona la

					opción Aceptar. 8- En caso que la nota o el promedio general fuesen menor de 4 puntos se muestra el mensaje “Por el momento no puede crear la solicitud.”. 9- Se selecciona la opción Aceptar del mensaje. 10- Se muestra la interfaz inicial para seleccionar una disciplina.

- SC 3 Detalles de la solicitud

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 3.1: Detalles de la solicitud.	Se muestran los detalles de la solicitud seleccionada.	El sistema muestra una ventana con los detalles de la solicitud seleccionada.	21- Se accede al subsistema de pregrado, se selecciona al módulo Personal y Secretaría, se accede a la agrupación funcional Ayudantía. 22- Se selecciona la funcionalidad Solicitud. 23- Se selecciona ícono Detalles el cual se encuentra a la derecha de cada solicitud. 24- Se muestra una ventana con los detalles de la solicitud seleccionada: - Asignatura - Fecha de registro - Estado 25- Se selecciona la opción Aceptar de la ventana 26- Se muestra un listado de las solicitudes asociadas al usuario registrado en el sistema.

Anexo E. Resultados de las pruebas de unidad (Técnica de caja blanca)

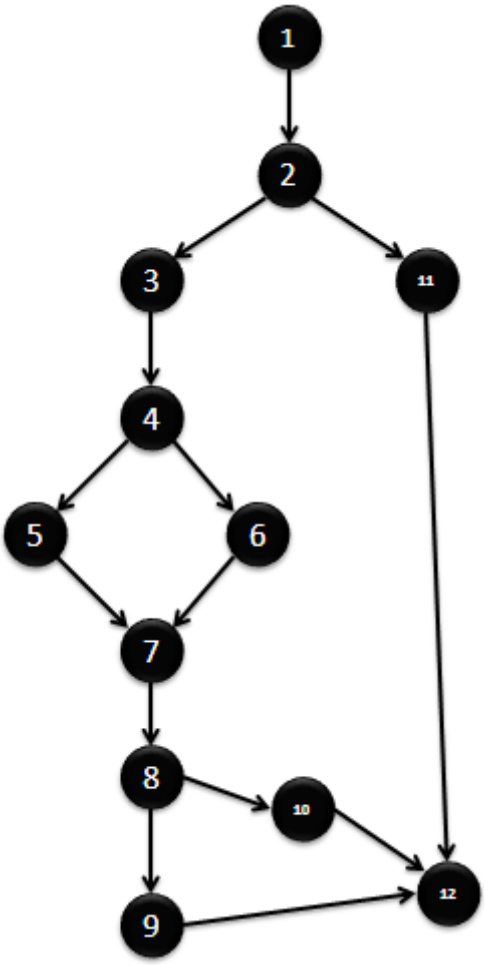
Diseño de caso de prueba #1. Unit_CB_Registrar_alumno_ayudante.

Criterio: Cobertura de caminos

Código fuente.

```
        public function registrarAlumnoAyudante() {
1                       $verificar = array('idPersona');
2                       if ($this->input->is_post_back($verificar)) {
3                               $post = $this->input->all_post(true);
3                               $pL=$this->ayudantia_lib->obtenerPL($post['id']);
3                               $cursoAcademico = $this->ioc->carrera->obtenerCursoAcadActivo();
3                               $datos = array();
4                               if(isset($post['grupos']))
5                                       $datos['idsGrupos'] = $post['grupos'];
6                               else
6                                       $datos['idsGrupos'] = array();
7                               $datos['id_cargo_aa'] = $post['cargosAA'];
7                               // declaración de variables
7                               $res=$this->ayudantia_lib->registrarAA($datos);
8                               if ($res == true)
9                                       $this->message('El elemento ha sido registrado');
10                              else
10                                      throw new Exception_Error('No se pudo realizar la acción');
                        }
11                      else
11                              throw new Exception_Error('No existe el id de la persona');
12              }
```

Grafo de flujo



Cálculo de la complejidad ciclomática. V (G) = Aristas – Nodos + 2 = 14 - 12 + 2 = 4	**Conjunto básico de caminos.** Camino 1: 1 - 2 - 3 – 4 – 5 – 7 – 8 – 9 - 12 Camino 2: 1 - 2 - 3 – 4 – 5 – 7 – 8 – 10 – 12 Camino 3: 1 - 2 - 3 – 4 – 6 – 7 – 8 – 10 – 12 Camino 4: 1 - 2 – 11 - 12

Casos de prueba		
Camino 1		
Condición de ejecución: Existe el id del estudiante a registrar y los grupos que le serán asignados y estos son registrados correctamente.		
Datos de prueba	**Resultado esperado**	**Evaluación**
$post['id']='df65rf0ergy8'; $post['grupos']= array('1504','1505');	'El elemento ha sido registrado'	**SATISFACTORIO**
Camino 2		
Condición de ejecución: Existe el id del estudiante a registrar y los grupos que le serán asignados y estos no pueden ser		

registrados correctamente.		
Datos de prueba	**Resultado esperado**	**Evaluación**
$post['id']='df65rf0ergy8'; $post['grupos']= array('1504','1505');	'No se pudo realizar la acción'	**SATISFACTORIO**
Camino 3		
Condición de ejecución: Existe el id del estudiante a registrar y no existen grupos a asignar y estos no pueden ser registrados correctamente.		
Datos de prueba	**Resultado esperado**	**Evaluación**
$post['id']='df65rf0ergy8'; $post['grupos']=array();	'No se pudo realizar la acción'	**SATISFACTORIO**
Camino 4		
Condición de ejecución: No existe el id del estudiante a registrar.		
Datos de prueba	**Resultado esperado**	**Evaluación**
$post['id']=' ';	'No existe el id de la persona'	**SATISFACTORIO**

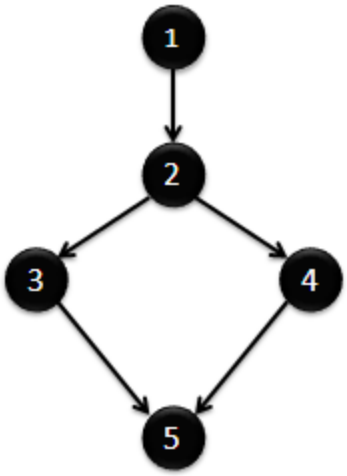
Diseño de caso de prueba #2. Unit_CB_Modificar_estado_de_solicitud

Criterio: Cobertura de caminos

Código fuente.

```
        public function registrarPropuestaAlumnoAyudante () {
1                       $post = $this->input->all_post(true);
1                       $datos = array ();
1           $datos ['id_solicitud'] = $post ['id_solicitud'];
1           $datos ['id_estado] = $post ['id_estado];
1           $datos ['id_estado_anterior'] = $post ['id_estado_anterior'];
1           $datos ['id_estudiante'] = $post ['id_estudiante'];
2                       if ($this->ayudantia_lib-> modificarEstadoSolicitud ($datos) == true)
3                               $this->message('El elemento ha sido registrado');
4                       else
4                               throw new Exception_Error('No se pudo realizar la acción');
5               }
```

Grafo de flujo



Cálculo de la complejidad ciclomática. V (G) = Aristas – Nodos + 2 = 5 -5 + 2 = 2	**Conjunto básico de caminos.** Camino 1: 1 - 2 - 3 – 5 Camino 1: 1 - 2 - 4 - 5

Casos de prueba

Camino 1

Condición de ejecución: Los datos introducidos son válidos

Datos de prueba	Resultado esperado	Evaluación
$datos = array('id_solicitud'=>125478, 'id_estado'=>5, 'id_estado_anterior'=>1, 'id_estudiante'=>'df65rf0ergy8');	'El elemento ha sido registrado'	**SATISFACTORIO**

Camino 2

Condición de ejecución: Los datos introducidos no son válidos.

Datos de prueba	Resultado esperado	Evaluación
$datos = array('id_solicitud'=>125478, 'id_estado'=>5, 'id_estado_anterior'=>1, 'id_estudiante'=>'df65rf0ergy8');	'No se pudo realizar la acción'	**SATISFACTORIO**

Anexo F. Resultados de las pruebas de integración

Diseño de caso de prueba #1. Int_Módulo_Carrera.

Módulo a integrar: Carrera
Descripción de la prueba: Comprobar que el módulo de carrera proporcione los datos referentes la carrera como son: tipos de cursos, disciplinas, asignaturas, marco lectivo, etc.
Condiciones de ejecución: Se realiza la petición de los datos
Pasos de la ejecución: El controlador de la agrupación funcional de ayudantía realiza una petición a la librería para obtener los datos de la carrera, gestionados por el módulo de carrera
Resultado esperado: El módulo de control docente brinda toda la información solicitada
Evaluación: SATISFACTORIO

Diseño de caso de prueba #2. Int_Módulo_Control_docente.

Módulo a integrar: Control docente
Descripción de la prueba: Comprobar que el módulo de control docente proporcione los datos referentes a las evaluaciones de cada uno de los estudiantes
Condiciones de ejecución: Se realiza la petición de los datos
Pasos de la ejecución: El controlador de la agrupación funcional de ayudantía realiza una petición a la librería para obtener los datos de todas la evaluaciones de cada una de las asignaturas vencidas por los estudiantes, gestionados por el módulo de control docente
Resultado esperado: El módulo de control docente brinda toda la información solicitada
Evaluación: SATISFACTORIO

Diseño de caso de prueba #3. Int_Módulo_Estructura_Composición.

Módulo a integrar: Estructura y composición
Descripción de la prueba: Comprobar que el módulo de estructura y composición proporcione los datos referentes a toda la estructura organizativa definida por la universidad
Condiciones de ejecución: Se realiza la petición de los datos
Pasos de la ejecución: El controlador de la agrupación funcional de ayudantía realiza una petición a la librería para obtener los datos de las facultades, grupos docentes, departamentos docentes, etc. gestionados por el módulo de estructura y composición
Resultado esperado: El módulo de estructura y composición brinda toda la información solicitada
Evaluación: SATISFACTORIO

Diseño de caso de prueba #4. Int_Módulo_Personal.

Módulo a integrar: Personal
Descripción de la prueba: Comprobar que el módulo de personal proporcione la información del personal de la universidad.
Condiciones de ejecución: Se realiza la petición de los datos

Pasos de la ejecución: El controlador de la agrupación funcional de ayudantía realiza una petición a la librería para obtener los datos de una persona, gestionada por el módulo de personal
Resultado esperado: El módulo de personal brinda todas la información solicitada
Evaluación: SATISFACTORIO

Diseño de caso de prueba #5. Int_Módulo_Seguridad.

Módulo a integrar: Seguridad
Descripción de la prueba: Comprobar que el módulo de seguridad proporcione el acceso restringido a cada usuario según el rol que el mismo le asigne.
Condiciones de ejecución: El usuario debe estar autenticado en el sistema.
Pasos de la ejecución: Una vez autenticado el usuario procede a tratar de acceder a una de las funcionalidades de la solución.
Resultado esperado: El sistema muestra un mensaje de error de acceso denegado y en caso contrario accede a la funcionalidad.
Evaluación: SATISFACTORIO

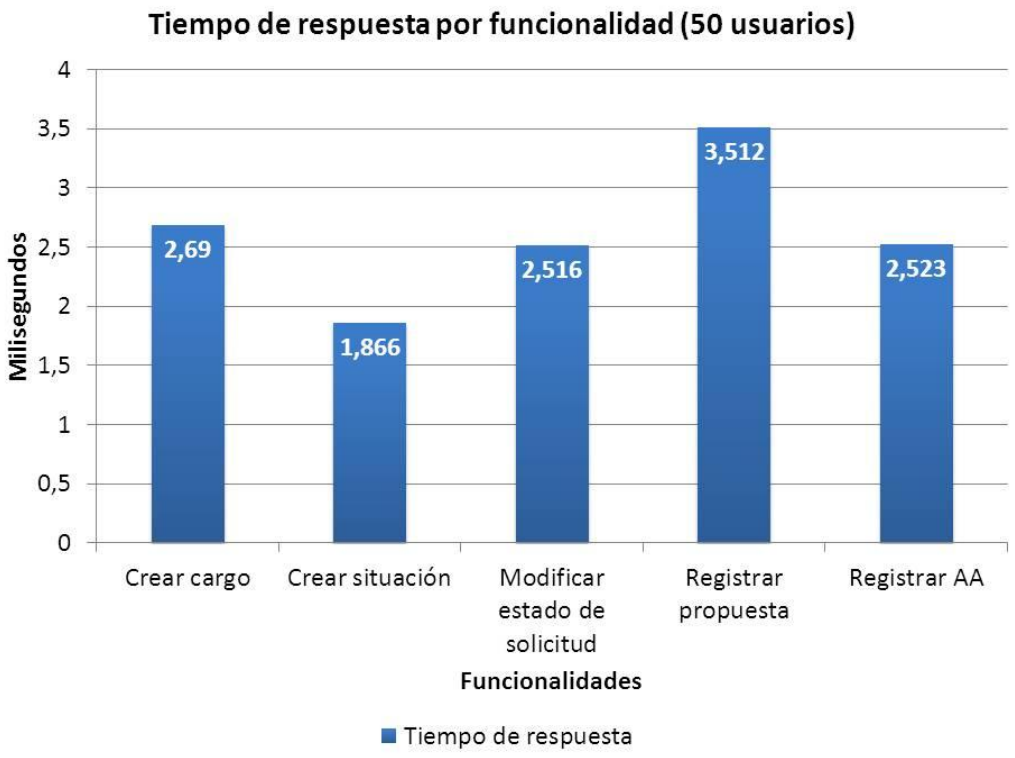
Diseño de caso de prueba #6. Int_Módulo_Traza.

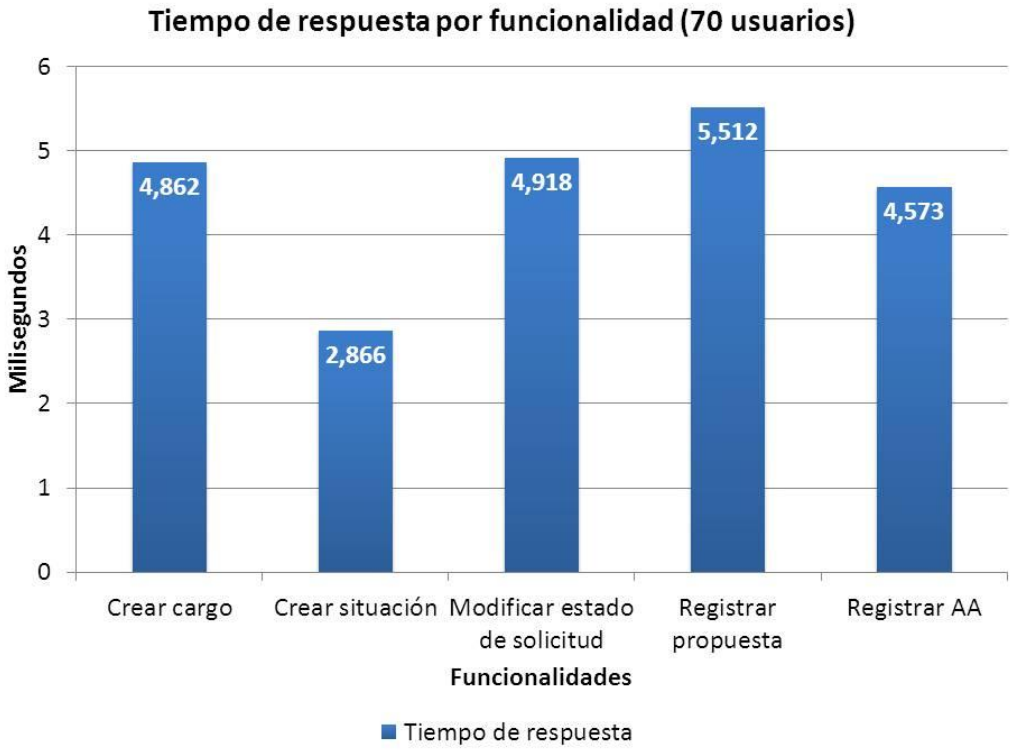
Módulo a integrar: Traza
Descripción de la prueba: Comprobar que son registradas todas la incidencias de un usuario sobre la agrupación funcional de ayudantía
Condiciones de ejecución: El usuario realiza cualquier acción sobre la agrupación funcional
Pasos de la ejecución: Una vez realizada una acción el sistema registra los datos de la incidencia, como son: nombre de usuario, dirección IP, acción ejecutada, fecha de la ejecución, etc. Luego se procede a buscar mediante el registro de trazas la incidencia deseada
Resultado esperado: Se muestran los datos de la traza registrada
Evaluación: SATISFACTORIO

Anexo G. Resultados de las pruebas del sistema

Resultados arrojados por la herramienta Apache JMeter para una conexión de 50 usuarios en línea con el sistema.

Etiqueta	# de muestras	Media	Mediana	Mín.	Máx.	% de error	Rendimiento	Kb / seg.
Crear cargo	50	2690	3694	500	7441	0,00 %	3,4/seg	3,7
Crear situación	50	1866	2213	538	6749	0,00 %	2,0/seg	2,3
Modificar estado de solicitud	50	2516	2980	1340	5032	0,00 %	1,6/seg	1,01
Registrar propuesta	50	3512	3411	1478	9778	0,00 %	1,0/seg	1,7
Registrar AA	50	2523	2055	2002	5520	0,00 %	1,04/seg	3,0
Total	250	2503	3003	538	6430	0,00 %	5,4/seg	7,6



Resultados arrojados por la herramienta Apache JMeter para una conexión de 70 usuarios en línea con el sistema.

Etiqueta	# de muestras	Media	Mediana	Mín.	Máx.	% de error	Rendimiento	Kb / seg.
Crear cargo	70	4862	5866	580	8491	0,00 %	5,1/seg	3,9
Crear situación	70	2866	3213	538	6749	0,00 %	4,0/seg	3,0
Modificar estado de solicitud	70	4918	5382	1354	7093	0,00 %	2,8/seg	1,5
Registrar propuesta	70	5512	5411	2852	7778	0,00 %	2,3/seg	1,9
Registrar AA	70	4573	4105	2225	7563	0,00 %	2,4/seg	3,4
Total	350	4546	4803	538	8491	0,00 %	9,4/seg	8,1