ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL.

Facultad de Ingeniería en Electricidad y Computación

"IMPLEMENTACIÓN DE UN SISTEMA PARA GESTIÓN Y SEGUIMIENTO DE LOS DOCUMENTOS PROCESADOS POR LA FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN."

TESIS DE GRADO

Previa a la obtención del Título de:

INGENIERO EN COMPUTACIÓN ESPECIALIZACIÓN SISTEMAS TECNOLÓGICOS

Presentado por:

Karla Abad Sacoto

Larry Beltrán Flores

Guayaquil - Ecuador

Año 2009

AGRADECIMIENTO

Agradecemos a Dios, a nuestros padres y hermanos que nos han brindado su apoyo a lo largo de nuestra carrera. A la Ing. Carmen Vaca por ser una excelente maestra y guía.

DEDICATORIA

A nuestra querida familia y amigos.

TRIBUNAL DE GRADUACIÓN

Ing. Jorge Aragundi	Msc. Carmen Vaca R.
Presidente del Tribunal	Director de Tesis
Mba. Ana Tapia R.	Phd. Katherine Chiluiza G.
Miembro Principal	Miembro Principal

DECLARACIÓN EXPRESA

La responsabilidad del contenido de esta Tesis de Grado, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral.

____

Karla Abad Sacoto

____

Larry Beltrán Flores

RESUMEN

Para la presente tesis se ha desarrollado un sistema que permitirá tramitar solicitudes realizadas en la Facultad de Ingeniería en Electricidad y Computación de la ESPOL. Este sistema automatizará procesos que actualmente se realizan en papel, disminuyendo el tiempo en que el trámite es realizado y permitiendo que los involucrados puedan dar seguimiento a sus documentos.

El primer capítulo, revisa los antecedentes y justificación del sistema. Presenta la solución propuesta (flujos de trabajo), enfatizando en las ventajas de utilizarlo en la automatización de los procesos (solicitudes) que se realizan en FIEC.

El segundo capítulo especifica los objetivos, requerimientos y alcances del sistema. Justifica la selección del motor de flujo de trabajo y describe la plataforma tecnológica sobre la cual se realizó el sistema.

El tercer capítulo explica cómo fueron modelados los procesos para poder ser automatizados a través del motor de flujo de trabajo. Además define cada componente de dichos procesos.

El cuarto capítulo explica el diseño e implementación del sistema detallando cada módulo existente.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

Código de campo cambiado

ÍNDICE DE TABLAS

INTRODUCCIÓN

En la actualidad, la FIEC no cuenta con un sistema para automatizar los trámites generados por las solicitudes recibidas de sus estudiantes o profesores. Estas solicitudes se procesan de forma manual ocasionando que exista, en algunas ocasiones, pérdida de tiempo o de documentos. Además, la persona que solicita el trámite no tiene manera de conocer el estado de su solicitud.

Por los motivos anteriores se plantea el diseño e implementación de un sistema para gestión y seguimiento de los documentos procesados por la Facultad de Ingeniería en Electricidad y Computación, con el objetivo de mejorar el servicio prestado por la facultad a los solicitantes y optimizar el manejo de la información.

Los motores de flujo de trabajo son herramientas idóneas para la implementación de sistemas de seguimiento de documentos. En esta tesis se analizan soluciones de software libre que implementan esta tecnología y luego se detalla el desarrollo de la implementación del flujo de trabajo de cinco procesos de la FIEC utilizando una de las herramientas analizadas.

CAPÍTULO I

1. ANÁLISIS DEL PROBLEMA

1.1. Antecedentes

En la actualidad uno de los principales problemas que tiene la Facultad es la administración, organización y seguimiento de documentos asociados a un trámite. El proceso de ejecución del trámite es lento, el estudiante no tiene manera de conocer el estado del mismo y las personas involucradas no son informadas a tiempo del ingreso, salida o cambio de estado de los documentos. Todas estas dificultades ocasionan que el estudiante perciba una atención inadecuada y además la pérdida de tiempo para muchas de las personas involucradas en el proceso.

El manejo de estos trámites se realiza de manera manual a través de solicitudes por escrito, en hojas especiales (especies valoradas), las cuales sólo se expenden en unos pocos lugares de la universidad. Además de esto el estudiante está obligado a seguir un formato especial para la realización de cada tipo de solicitud. Todo esto hace de cada solicitud un proceso molestoso para el estudiante.

Otro punto que se debe considerar es que al manejar procesos a través de papeles es más difícil llevar una organización de ellos, haciendo que exista la posibilidad de pérdida de los documentos. También se debe tener en cuenta que al ser un proceso manual la rapidez con que un documento pasa de un estado a otro depende únicamente de las personas involucradas en él.

La realización de actividades, que normalmente se desarrollan pasando formularios en papel de una persona a otra, son costosas y demandan recursos importantes. La automatización de lo expuesto anteriormente puede representar ahorros significativos para la facultad. Por ello, se explorará y analizará la tecnología de flujos de trabajo.

1.2. Flujos de trabajo [1]

La tecnología de flujos de trabajo, en inglés Workflow, se originó como una forma de reducir el tiempo y costo de llevar a cabo los procesos de negocios y asegurar que las tareas sean realizadas consistentemente para mejorar la calidad de los servicios.

Se puede decir que esta tecnología se basa sobre la asunción de que los humanos son buenos para tomar decisiones, innovar, identificar hechos inesperados. Pero usualmente no son eficientes en actividades tales como: buscar un documento entre cientos, tener presentes los vencimientos de las tareas que se tienen que realizar dentro de ciertos plazos, así como también el asegurarse de que el trabajo terminado pase de un lugar a otro respetando la secuencia definida.

A fines de esta década se busca mejorar el flujo de la información, el desafío que se plantea es obtener la información rápida y eficientemente. Surgen las necesidades de incrementar la eficiencia, optimizar la productividad, acortar los tiempos de procesos, tener un control sobre estos, así como también de reducir los costos y mejorar la gestión. Todo esto como consecuencia del incremento de la competitividad y de la exigencia de mejores productos, dentro de un mercado que avanza a gran velocidad. En este momento existe una proliferación de diversos mecanismos de intercambio de información, los mismos facilitan el manejo del flujo de la información en general.

1.2.1. Definiciones

En esta sección se presentan las definiciones de los términos que se utilizarán a lo largo de la tesis.

Workflow: de acuerdo a [2] un Workflow es la automatización de un proceso del negocio, total o parcial, durante los cuales documentos, información o tareas son pasadas de un participante a otro para su procesamiento, de acuerdo a un conjunto de reglas establecidas.

Desde otro punto de vista, se puede definir Workflow como un conjunto de actividades almacenadas como un modelo que describe un proceso del mundo real. El trabajo pasa por el modelo de principio a fin y las actividades pueden ser ejecutadas por personas o por las funciones del sistema.

Proceso: se define como un conjunto de actividades, asociados a un modelo de proceso. El orden de las actividades va de acuerdo a las reglas de navegación definidos en el modelo.

Modelo de proceso: un proceso de negocio es traducido del mundo real a una definición formal y procesable por computadora a través de una o más técnicas de análisis, definición y modelado. El resultado es llamado definición o modelo de proceso.

Un modelo o definición de proceso comprende un conjunto discreto de actividades, con operaciones y reglas asociadas, las cuales gobiernan el flujo entre esas actividades. Esta definición puede ser expresada en forma grafica a través de un editor o textual mediante un lenguaje de notación formal.

Lenguaje de modelado de procesos: esquema XML que define la lógica de un proceso.

Herramientas de modelado de Proceso: son usadas para crear una descripción del proceso en forma que una computadora pueda entenderlo y procesarlo, esto se realiza a través de un lenguaje formal de definición de procesos. Algunas herramientas son gráficas y permiten dibujar el proceso generando por sí mismas el archivo con el texto en el lenguaje correcto, también se puede escribir la definición de forma manual si se conoce el lenguaje de definición de proceso. La herramienta puede ser provista como parte del producto o puede ser obtenido de forma separada. Si es el último caso se debe revisar que el formato para crear las definiciones de procesos sea compatible con el software Workflow que se esté utilizando.

Instancia de Proceso: es una instancia en ejecución de modelo de proceso.

Actividad: una tarea que forma un paso lógico en una definición de proceso. Puede ser automática o manual.

Usuarios: son las personas que hacen uso del sistema.

Participantes: son todos los usuarios a los que se les permite jugar algún rol (responsabilidades de un participante en una actividad) en un proceso dado. Uno de los tipos siguientes: un recurso del sistema, un recurso específico, una unidad de la organización, un rol (una función de un ser humano dentro de una organización), un ser humano, o un sistema (un agente automático).

Rutas: secuencia de pasos a seguir por los documentos (o información) dentro de un sistema de Workflow. La capacidad de direccionar las tareas a los usuarios es vital en una aplicación de este tipo.

Para asegurar el éxito del flujo de información y decisiones, todos los miembros del equipo deben ser capaces de tomar parte en este proceso.

Reglas de Transición: Son reglas lógicas que determinan la navegación del documento dentro del sistema. Expresan que acción se va a tomar dependiendo del valor de expresiones lógicas. La definición de las reglas puede ser muy complicada, con múltiples opciones, variaciones, y excepciones. Un ejemplo sencillo podría ser el siguiente: Un profesor solicita al Decano la contratación de ayudantes, entonces la siguiente regla expresa el camino a seguir sobre la base de la solicitud: “SI el Decano aprueba la solicitud entonces enviar un correo a todos los estudiantes comunicando que se requieren ayudantes, SINO, terminar la solicitud”. La regla anterior muestra, de manera sencilla, el tipo de reglas que comúnmente se expresan.

Eventos: un evento es una interrupción que contiene información, el mismo tiene un origen y uno o más destinatarios. La información contenida en el mensaje que se produjo por el evento puede ser implícita o dada por el usuario. Los eventos pueden ser disparados voluntariamente por el usuario; o en forma implícita durante un proceso según el estado de los datos o de decisiones tomadas por el usuario; o en forma automática. Por ejemplo, cuando se decide rechazar una solicitud, se dispara un evento para que envíe un correo notificando a los involucrados.

Plazos: puede verse a los plazos como los tiempos que se le asignan a ciertos elementos. Ejemplos de plazos pueden ser: el tiempo máximo que se le asigna a una tarea para que sea terminada; el tiempo máximo para recorrer una ruta; terminar una tarea antes de cierta fecha; terminar el recorrido de una ruta antes de cierta fecha; y así se podría seguir.

A los plazos puede asignárseles eventos, de forma tal que cuando venza determinado plazo se disparen ciertos eventos asignados por el usuario, o programados para que se disparen automáticamente.

1.2.2. Uso de Workflow en empresas y negocios

De acuerdo a [1], la gerencia de flujos de trabajo es una tecnología de rápido crecimiento, la cual está incrementando el número de negocios, de varios tipos de industria, en las que es utilizado.

Su característica principal es la automatización de procesos, combinando actividades realizadas por humanos y actividades realizadas por computadoras. Sin embargo, su uso más relevante es dentro de ambientes de oficina en tareas del grupo de trabajo como seguros bancarios, administración legal y general, etc., y también es aplicable a algunos tipos de aplicaciones industriales y de manufactura.

Dentro del desarrollo actual del Workflow, como tecnología, es posible identificar la evolución de su uso en diferentes áreas, las cuales se nombran a continuación:

• **Procesamiento de imágenes:** En este caso se captura en forma de imagen electrónica (por ejemplo mediante un escáner) la información o documento que se desea, para luego ser pasada entre los diferentes participantes con distintos propósitos, durante la realización de un proceso.

• **Administración de documentos**: Esta tecnología se relaciona con la administración del ciclo de vida de los documentos. Esta incluye facilidades para guardar en un depósito común aquellos documentos que se comparten, así como también las facilidades para el acceso o modificación de los mismos mediante un conjunto predefinido de reglas.

• **Aplicaciones basadas en transacciones:** Las transacciones de esta tecnología guardan la información, reglas, roles, y otros elementos sobre un servidor de Bases de Datos Relacionados, ejecutando la aplicación sobre una interfaz gráfica para los usuarios. Estas aplicaciones típicamente incluyen componentes gráficos para el ingreso de los datos.

• **Procesamiento de Formularios:** El ambiente de los formularios es amigable y familiar para muchos usuarios. Éste es un excelente vehículo para el manejo de la información dentro de una aplicación de flujo de trabajo, basado en el valor de los campos de un formulario.

La tecnología de flujos de trabajo ha alcanzado un grado de madurez razonable, en cuanto a productos desarrollados y a experiencias industriales en las que se ha aplicado, demostrándose su efectividad para soportar las distintas actividades involucradas en procesos de la industria.

Actualmente, muchas empresas están redefiniendo la forma en que ellos hacen negocio. Esta transformación involucra dejar afuera las viejas estructuras para construir equipos que respondan a las necesidades de los clientes y encarar la amenaza competitiva. Para que estos equipos operen eficientemente necesitan un software que permita a los miembros del equipo compartir ideas, información y tareas para ayudarlos a completar los procesos de negocios tan eficientemente como sea posible. Estos tipos de software ofrecen herramientas para la definición, ejecución y administración de los flujos de trabajo.

1.2.3. Ventajas

En esta sección se presenta un análisis de las principales ventajas que encontramos como consecuencia de utilizar una herramienta Workflow.

Incremento de la productividad: la implementación de una solución Workflow mejorará la productividad de varias maneras:

- Reduciendo el tiempo requerido para resolver el procesamiento de papel (solicitud).
- Enlazando formularios electrónicos con sus bases de datos y creando tablas de consulta.
- Evitando que se pierda información valiosa recaudada durante los procesos de negocio.
- Direccionando los procesos a los individuos apropiados y garantizando que las entradas corresponda a las reglas preestablecidas.
- Evitando que se pierdan documentos, procesos.

Reducción del tiempo de respuesta: la automatización de procesos es proactiva, asegura que las tareas avancen sin detenerse. Las buenas soluciones de Workflow permiten definir los tiempos permisibles para la realización de una tarea, esto mejora la respuesta de los procesos administrativos en beneficio de la eficiencia y la calidad.

Asignación de actividades a los participantes: la asignación de actividades se realiza mediante la definición de roles, eliminando la tediosa tarea de asignar los trabajos caso por caso.

Consistencia y confiabilidad: la automatización de Workflow asegura consistencia y confiabilidad cada vez que se ejecuta un proceso. Los errores son reducidos dramáticamente y la aplicación garantiza el cumplimiento de las reglas de negocio preestablecidas.

Control de los Procesos: utilizando la tecnología de Workflow es posible monitorear el estado actual de los procesos así como también observar cómo evolucionan los planes de trabajo realizados. Permite ver cuáles son los embotellamientos dentro del sistema, es decir aquellas tareas o decisiones que están requiriendo de tiempo no planificado.

Documentación de procesos: uno de los beneficios colaterales de la automatización de procesos de negocio o administrativos, es que obliga a las organizaciones a analizar y documentar sus procesos. No se puede automatizar lo que no se conoce. Asimismo el ejercicio de analizar y documentar por sí solo suele poner en evidencia redundancias e ineficiencias. Documentar los procesos y asegurar el seguimiento de las reglas es central para los estándares de sistemas de calidad como ISO.

1.2.4. Limitaciones

Una de las limitaciones que se ha encontrado al implementar un sistema de este tipo es el costo de aprendizaje de la herramienta y sus funciones, debido a que es una tecnología relativamente nueva y la información disponible aún no es abundante. La mayoría de las implementaciones estudiadas presentan diferentes lenguajes propios para la definición de procesos, lo cual añade complejidad al aprendizaje de cada una de ellas.

Además se debe considerar que si una empresa maneja actualmente sus procesos de forma manual, existirá resistencia al cambio por parte de sus empleados pues deben aprender a utilizar el sistema implementado.

1.3. Planteamiento de la solución

Luego de analizar las ventajas y limitaciones de una solución Workflow se expone el análisis de por qué usar esta solución.

El problema planteado se centra en procesos que se inician con una solicitud. Cada uno de estos procesos encierran un conjunto de actividades que deben realizarse en secuencia, considerando la información obtenida para decidir el siguiente paso a dar. Las diferentes actividades son realizadas por los usuarios del sistema, dependiendo de cuál es su rol y cuáles son sus funciones.

Los sistemas Workflow ayudan a manejar los procesos de negocios asegurando que las tareas sean priorizadas para ser llevadas a cabo:

• Tan pronto como sea posible

• Por la persona correcta

• En el orden adecuado

El Workflow entonces, da a una organización la posibilidad de automatizar sus procesos, reducir costos; esto mejora la eficiencia y los servicios brindados. En la figura 1.1 se muestra los beneficios de una solución Workflow.

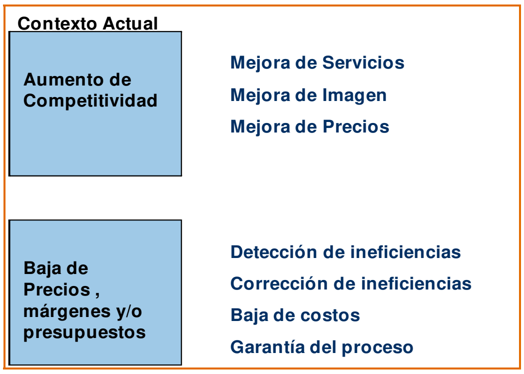


Figura 1.1 Beneficios de una solución Workflow.

La tecnología de flujos de trabajo es ideal para resolver los problemas de procesamiento de trámites existentes en la facultad, descritos en la sección Antecedentes.

La solución consistirá en implementar una aplicación que use un motor de flujo de trabajo que permita definir los procesos asociados a cada trámite. Como parte de la aplicación se desarrollara una interfaz web para la interacción con los usuarios.

1.4. Justificación

Con el propósito de mejorar la calidad de la Facultad en cuanto a organización y atención al estudiante, se plantea el desarrollo de un Sistema de flujo de trabajo para gestionar los documentos procesados por la FIEC.

El sistema que se plantea agilizará los procesos tramitados en la facultad debido a que la información podrá ser accedida de manera rápida y desde cualquier lugar donde se encuentre el usuario.

Los usuarios involucrados serán notificados inmediatamente de cada actividad que deban realizar durante los procesos, reduciendo así el tiempo que conlleva realizarlo.

El sistema permitirá ahorrar tiempo y trabajo. Este ahorro se traduce en un incremento en la calidad de los servicios prestados por la facultad.

También mejora la imagen de la administración frente a los estudiantes, debido a que sus trámites son realizados con mayor eficiencia y en un menor tiempo.

Beneficios del sistema

El sistema tendrá múltiples beneficios, algunos de los cuales se listan a continuación:

- Automatizar los procesos que siguen los documentos en la Facultad.
- Actualizar de forma continua los cambios de estado y de localización de los documentos.
- Mostrar a los estudiantes de la Facultad el estado de sus solicitudes.
- Mostrar las rutas tomadas de los documentos durante su procesamiento.
- G
  enerar una fuente de supervisión de los documentos para los directivos de la Facultad.
- Agilitar trámites manteniendo informados a los usuarios sobre las revisiones/aprobaciones pendientes.
- Notificar el cambio de estado de los documentos a los usuarios correspondientes del sistema utilizando como medio de comunicación el correo electrónico.

En este capítulo se ha revisado los antecedentes y justificación del sistema, así como también se ha presentado la solución propuesta (flujos de trabajo).

Para implementar un sistema de flujos de trabajo es necesario un análisis de la plataforma a utilizar, este análisis se lo realiza en el siguiente capítulo.

CAPÍTULO II

2. ESPECIFICACIONES Y PLATAFORMA TECNOLÓGICA

2.1. Especificaciones del Sistema

2.1.1. Descripción

El sistema permite dar seguimiento a los documentos procesados por la facultad, de tal manera que el solicitante pueda revisar de forma oportuna el resultado de su petición. El sistema también asegura que la solicitud llegue de forma automática a manos de la persona encargada de evaluarla.

El usuario interactúa con el sistema a través de una aplicación web en la cual podrá iniciar la solicitud que requiera y revisar continuamente el estado de la misma. Adicionalmente, los directivos podrán hacer uso de la aplicación para consultar si existe alguna solicitud en espera de su respuesta.

2.1.2. Objetivos

Después de definir el uso de un Workflow para la implementación del sistema, especificamos los objetivos a cumplir:

Objetivo General:

Implementación de un sistema que permita automatizar el procesamiento de documentos para que constituya una herramienta de comunicación entre la facultad y las personas involucradas en el proceso, de tal manera que estos puedan conocer el estado de todos sus trámites en cualquier momento.

Objetivos Específicos:

- Evaluar y determinar la herramienta de código abierto más apropiada para la implementación del flujo de trabajo (Workflow) de la facultad.
- Levantar información sobre el proceso a seguir de cada uno de los tipos de documentos cuyo trámite podrá ser gestionado a través del sistema.
- Definir los roles de los participantes en la aprobación de cada uno de los tipos de documentos a procesar.
- Implementar un flujo de trabajo para cada uno de los documentos.

- Configurar notificaciones vía correo electrónico a los usuarios que tengan pendientes aprobaciones de distintos documentos en el sistema.
- Implementar páginas Web a través de las cuales todos los usuarios del sistema (incluidos estudiantes) puedan consultar el estado actual del documento asociado a un trámite dado.

2.1.3. Requerimientos del Proyecto

Para el correcto funcionamiento del sistema es necesario que se cumplan con ciertos requerimientos, los cuales se detallan a continuación:

Requerimientos Funcionales

Definen las funciones que el sistema será capaz de realizar. Seguidamente se mencionan los requerimientos funcionales del sistema:

- Permitir a los usuarios iniciar una solicitud a través del llenado de un formulario web. Las solicitudes que cada usuario puede iniciar, dependen de su rol dentro del sistema. Por ejemplo, un estudiante podrá iniciar una solicitud para ser ayudante pero no puede iniciar una solicitud de contratación de ayudantes.

- Mostrar los procesos en curso (solicitudes activas) que el usuario tiene. Aquí se muestran las actividades por las que el proceso ha pasado y en cual se encuentra actualmente.

- Mostrar al usuario una lista (por cada uno de los tipos de solicitudes) de las actividades pendientes por realizar. En esta parte el usuario puede acceder desde la lista al formulario correspondiente de la actividad para ejecutarla.

- Mostrar un historial de las solicitudes ya completadas.

- Permitir al administrador del sistema manejar usuarios y roles, así como asignar usuarios a los roles.

- Permitir al administrador del sistema registrar o modificar carreras y especializaciones.

- Permite realizar búsqueda de solicitudes.

- Permitir a las secretarias la impresión de los documentos finales que luego serán llevados a las diferentes autoridades fuera de FIEC (ejemplo: Secretaria Técnica Académica).

- Enviar notificaciones mediante correo electrónico.

Requerimientos No Funcionales

Los requerimientos no funcionales se desarrollarán sobre áreas clave de calidad, cada una de las cuales presentan características que permitirán ejecutar con eficiencia y eficacia las funcionalidades planteadas. Los requerimientos no funcionales que se han encontrado en el sistema son:

- El sistema deberá estar en capacidad de prestar servicio con niveles aceptables de desempeño y tiempos de respuesta, teniendo en cuenta la concurrencia de usuarios. Deberá estar en capacidad de atender, sin que implique deterioro del servicio, a un número finito de usuarios realizando procesos en línea.

- El sistema deberá garantizar el soporte en el crecimiento del volumen de la información almacenada en la base de datos.

- Proveer canales cifrados de transmisión de datos confidenciales, como los empleados durante la autenticación del usuario.

- Garantizar capacidad de operar sobre los sistemas operativos más usados. Además ofrecer compatibilidad con navegadores de uso común.

- Permitir formatos de salida compatibles con herramientas de ofimática.

- El sistema deberá contar con una especificación de funcionalidad para que el usuario pueda interactuar con el mismo, esto será entendido como el manual de usuario para la aplicación.

- El sistema deberá ser de uso intuitivo, de tal forma que se reduzca los tiempos de entrenamiento, soporte y prueba por parte del usuario. Los mensajes de error deberán ser lo suficientemente informativos para dar a conocer la severidad del error.

2.1.4. Usuarios del Sistema

Los usuarios que harán uso del sistema son:

Estudiantes, profesores, directivos (Decano, Subdecano) y personal administrativo de la Facultad de Ingeniería en Electricidad y Computación de la ESPOL.

2.1.5. Alcances

Solo pueden hacer uso del sistema los usuarios que pertenecen a la Facultad de Ingeniería en Electricidad y

Computación de la ESPOL.

El sistema permitirá el ingreso y seguimiento de las siguientes solicitudes:

- Solicitudes de Ayudantías.
- Solicitudes de convalidación de materias.
- Solicitudes de cambio de carrera-especialización.
- Solicitudes de aprobación diplomado/maestría.

El sistema permitirá imprimir los resultados finales de las solicitudes que así lo ameriten. Sin embargo, el sistema no debe considerar como los resultados impresos o en medios físicos llegan a las entidades que están fuera de línea con el mismo.

El sistema sólo aceptará el ingreso de usuarios que cuenten con un usuario y clave en ESPOL. Estas credenciales serán verificadas utilizando un servicio Web de autenticación provisto por el Centro de Servicios Informáticos de la ESPOL.

Las solicitudes que maneja el sistema sólo pueden ser iniciadas por los estudiantes y profesores que pertenezcan a la FIEC.

2.1.6. Un escenario de ejemplo

Se describe a continuación cómo funcionaría el sistema para un escenario de ejemplo.

Proceso que automatiza el trámite donde un estudiante solicita cambiarse de carrera y/o especialización.

El primer paso es definir el flujo de actividades por las que el proceso debe pasar, basándonos en la información obtenida en documentos escritos, entrevistas a las autoridades, etc.

Siendo el flujo el siguiente: el estudiante ingresa su solicitud a través un formulario web, indicando la carrera a la que desea pertenecer, el Subdecano recibe esta solicitud y solicita el informe del coordinador de la nueva carrera. El coordinador analiza si el cambio es factible y emite su informe favorable o no para que el Subdecano evalúe y decida si la solicitud debe ser aprobada. La decisión de la evaluación es notificada al solicitante a través del sistema y también por correo electrónico. Si la solicitud es aprobada la secretaria la recibe e imprime un informe para que sea llevado a la Secretaría Técnica Académica y continúe su proceso fuera de FIEC.

Como se puede notar, por los párrafos anteriores, en cada actividad se debe identificar el rol que la ejecutará, por ejemplo: el coordinador de carrera es quien emite el informe sobre la solicitud.

Una vez que se tiene completamente claro el desarrollo del proceso se utiliza un editor gráfico para elaborar el diagrama. El editor ayuda a colocar las actividades y definir el rol participante en cada una de ellas, así como también nos permite colocar actividades de decisión cuyo resultado define la ruta que sigue el proceso.

El usuario del sistema puede revisar el estado de su proceso y también identificar en que parte del flujo se encuentra su solicitud a través de formularios web. A continuación la figura 2.1 muestra el proceso elaborado:

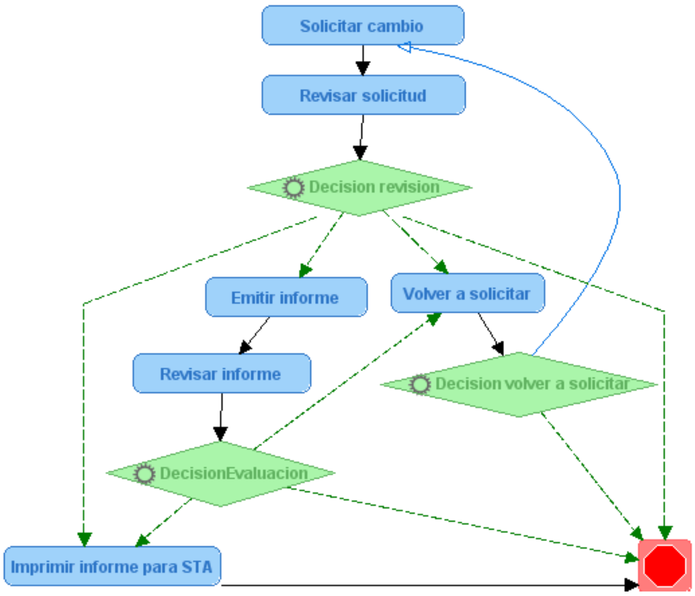


Figura 2.1 Diagrama de flujo del proceso solicitar cambio de carrera/especialización.

El ejemplo anterior describe de manera sencilla el flujo que debe seguir una solicitud en el sistema a implementar. Más adelante, en el siguiente capítulo, se explicará en detalle el proceso que sigue cada solicitud a automatizar.

Después de considerar las especificaciones generales del sistema, en la siguiente sección se examina la plataforma tecnológica a utilizar.

2.2. Plataforma Tecnológica

2.2.1. Análisis y selección del motor de flujo de trabajo a utilizar en el sistema.

Descripción de los motores estudiados

En el mercado actualmente es posible encontrar muchos motores de Workflow, tanto comerciales como no comerciales.

Los Workflows comerciales son bastantes costosos y además impiden hacer cambios en su estructura debido a que no hay acceso al código con el cual fue implementado; por ello los usuarios de estos tipos de implementaciones obligan a sus sistemas a acoplarse al Workflow con poca oportunidad de hacerlo más ajustable a la situación real del negocio. Estas razones son las que llevaron a analizar herramientas de software libre, gratuitas y de ser posible de código abierto.

A continuación se listan las soluciones más factibles (es decir son más robustas y brindan mayor soporte). Además todas han sido escritas sobre la plataforma Java que es la tecnología con la que se trabajará a lo largo de la implementación del sistema.

- JBPM
- Netbeans BPEL
- Object Web Bonita
- Enhydra Shark
- YAWL

Con el objetivo de encontrar la mejor herramienta de Workflow, se revisarán las principales características de estos motores para luego analizar y concluir cuál es la mejor.

JBPM (Java Business Process Management) [3]: permite crear y automatizar procesos de negocio y coordina a las personas, aplicaciones y servicios involucrados. Es una biblioteca escrita en Java, como consecuencia se puede utilizar en cualquier ambiente, por ejemplo una aplicación web, una aplicación swing, un EJB[1], un servicio web. Utiliza el lenguaje

[1] EJB: (Enterprise Java Beans). Proporcionan componentes del lado del servidor. Dotan al programador de un modelo que le permite abstraerse de los problemas generales de una aplicación para centrarse en el desarrollo de la lógica del negocio.

de definición de procesos JPDL (Java Process Definition Language) y permite la integración con el marco de trabajo SEAM.

Netbeans Proyecto BPEL (Business Process Execution Language) [4]: es un tipo de proyecto que utiliza el lenguaje de definición de procesos BPEL. Además provee y consume Servicios y utiliza proyectos compuestos de uno o más módulos de integración JBI (especificación desarrollada para mejorar la implementación de las arquitecturas orientadas a servicios).

YAWL (Yet Another Workflow Language) [5]: es un original y sofisticado lenguaje de Workflow. YAWL está construido sobre la investigación de patrones de Workflow y combina esto con el poder del lenguaje de las redes Petri. Provee un poderoso, lenguaje para los modeladores de proceso para describir relaciones complejas del control de flujo entre los procesos de negocio. Además provee soporte para datos basada en las tecnologías de XML.

Object Web Bonita [6]: Bonita es una solución de flujo de trabajo de código abierto que permite manejar procesos orientados al usuario y de larga duración, proveyendo funcionalidades de BPM (Business Process Management) para la administración de los procesos de negocio. Bonita implementa XPDL (XML Process Definition Language), lenguaje utilizado por la WFMC[2].

El sistema de Workflow Bonita integra un gran número de servicios para el control y simplificación de muchos aspectos relativos a la cooperación.

Enhydra Shark [7]: Enhydra Shark es un motor de Workflow completamente basado en las especificaciones WfMC (Workflow Management Coalition. Shark utiliza el Lenguaje de Definición de Procesos (XPDL), basado en XML. Cada componente (capa de persistencia, administrador de sesión, etc) puede ser usado con su implementación estándar o extendida. De esta manera Enhydra Shark puede ser utilizado como una simple "librería Java" en servlets, en aplicaciones swing, o corriendo en un contenedor J2EE. El almacenamiento de procesos y actividades se hace a través de un API[3] de persistencia configurable.

---

[2] WFMC: (Workflow Management Coalition). Organización internacional sin fines de lucro, conformada por usuarios, analistas y grupos de investigación de Workflow.

[3] Un API es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Características de un Workflow

Para analizar las herramientas de Workflow, se ha definido parámetros importantes que deben cumplir, a fin de que sea provechosa su utilización.

- Ser libre, y de ser posible de código abierto para poder realizar pruebas y modificaciones en caso de ser necesario.
- Cumplir con los estándares propuestos por la WFMC.
- De fácil implantación y de ser posible integrable con diferentes tecnologías.
- Poseer una herramienta gráfica para el diseño de los modelos de proceso, la cual permita definir gráficamente la ruta de cada documento y los estados por los que va pasando.
- Proveer un conjunto de APIs que permiten un control completo de la definición y ejecución del proceso.
- Que posea una herramienta para definir las reglas de negocio y asociación de roles a actividades.
- Permitir monitorear las instancias y actividades en ejecución.
- Permitir manejar tiempos plazos y eventos.
- Permitir anexar datos en cada actividad.
- Facilitar la interconexión con aplicaciones externas.
- Listar el trabajo asignado a cada usuario.

- Poseer gestión de históricos, es decir que permita revisar procesos que han sido realizados anteriormente.
- Ofrecer gran cantidad de documentación e información disponible.

Estas características han sido los referentes para concluir cuál es la herramienta más óptima en la implementación de un sistema para manejo de procesos. Para realizar el análisis se ha revisado cada una de las características a través de lectura de documentos y pruebas de implementación.

A continuación la tabla 2.1 muestra la comparación entre las diferentes herramientas estudiadas y los parámetros de análisis.

	JBPM	BPEL	Bonita	YAWL	Enhydra
Herramienta Libre y de código abierto	✓		✓	✓	
Cumpla Estándares de la WFMC			✓	✓	✓
De fácil implantación e integrable con otras tecnologías	✓	✓	✓		
Herramientas gráficas para diseños de procesos	✓	✓	✓	✓	✓
Provee APIs	✓	✓	✓	✓	✓
Permite definir reglas de negocio	✓	✓	✓	✓	✓
Monitoreo de las instancias y actividades en ejecución	✓	✓	✓	✓	✓
Manejo de tiempos plazos y eventos	✓		✓		✓
Anexar datos en cada actividad	✓	✓	✓	✓	✓
Definir perfiles de usuario y roles	✓	✓	✓	✓	✓
Integración con bases de datos	✓	✓	✓	✓	✓
Facilidad para la interconexión con aplicaciones externas	✓	✓	✓		
Listado de trabajo asignado	✓		✓	✓	✓
Gestión de históricos			✓		
Gran cantidad de documentación e información disponible			✓		

Tabla 2.1 Comparación entre diferentes motores de flujo de trabajo.

Análisis de las herramientas comparadas

Como se puede apreciar en la tabla anterior, las implementaciones estudiadas comparten muchas características comunes.

Todas proveen herramientas gráficas para diseño de procesos, aunque cabe mencionar que las herramientas ofrecidas por YAWL, Bonita y Enhydra son más completas y además permiten realizar diseños de procesos complejos (esto es debido a que permiten definir de mejor manera las regles del negocio).

Otra característica común es que todas se pueden configurar con diferentes motores de base de datos y servidor de aplicaciones. Sin embargo, cada una utiliza un servidor de aplicaciones, base de datos e IDE (entorno de desarrollo integrado) por defecto. Un punto para tomar la decisión de que herramienta elegir ha sido precisamente la facilidad de la herramienta para configurarse sobre la plataforma que usaremos en la implementación.

Todas son libres pues se pueden conseguir gratuitamente sin embargo BPEL y Enhydra no permiten obtener el código fuente de su implementación.

Para este análisis se realizaron pruebas con cada uno de los motores, por ello se conoce que Enhydra y YAWL son bastante complejos de instalar y probar. Además no existe suficiente información sobre cómo instalar la herramienta o cómo resolver los problemas que se pueden suscitar en el momento de la instalación. Por otra parte tanto Netbeans BPEL y JBPM vienen integrados en un IDE de desarrollo por lo que basta con conseguir el instalador adecuado del IDE y éste se encargara de la instalación y configuración por defecto del motor. En cuanto a Bonita podemos decir que instalarlo y probarlo bajo sus requerimientos por defecto (es decir servidor de aplicaciones y motor de base de datos) es sencillo.

Aunque todas las herramientas estudiadas proveen aplicaciones para monitoreo de los procesos en curso y de las actividades pendientes, se debe tener en consideración que unas son más sofisticadas que otras. Un ejemplo de ello es Bonita, el cual provee una aplicación llamada jiapAdmin (usada a través de un explorador web) que bien podría ser usada a nivel empresarial sin necesidad de implementar algo más, debido a que esta herramienta permite realizar actividades, ver procesos en curso y terminados, ver actividades pendientes, administración de usuario y roles, etc. En el caso del sistema a implementarse en esta tesis una herramienta de estas características sería muy valiosa para poder realizar pruebas de manera rápida y sencilla.

Uno de los puntos más importantes a tomar en cuenta para la elección, ha sido el conjunto de APIs que provee cada herramienta. Los APIs son el medio de comunicación entre los componentes de software, que permiten acceder a las diferentes funciones y procedimientos de cada uno. En este caso se utilizarán para poder acceder a todas las funcionalidades que el motor seleccionado otorgue durante la implementación del sistema.

Después de haber probado y analizado cada una de las herramientas presentadas concluimos que la mejor opción en Workflow es Object Web Bonita. Una de las principales ventajas de esta herramienta es la gran cantidad de documentos e información existentes, que se puede conseguir de manera libre y gratuita.

También cuenta con un gran número de componentes, entre ellos una herramienta gráfica para diseño de procesos que facilita de gran manera la tarea de modelar. Además es integrable con diferentes plataformas y ofrece un API completo que permite hacer uso de todas las funcionalidades del motor.

En la siguiente sección se presenta de manera general las principales características del motor seleccionado.

2.2.2. Workflow Bonita

Bonita es una solución Workflow para definir flujos de procesos orientados al usuario, permitiendo brindar a sus procesos de negocio la funcionalidad de “out the box” es decir el usuario simplemente se encarga de definir el flujo de proceso, siendo el motor de Bonita el que mantendrá la lógica del Workflow.

Bonita ofrece una gran cantidad de características innovadoras, como la infraestructura que permite a los usuarios estar informados de cualquier acontecimiento ocurrido durante la ejecución de un determinado proceso. También posee las características tradicionales de flujos de trabajo como la resolución dinámica de usuarios/roles, la ejecución secuencial de la actividades, que le permite dar soporte a procesos de flujo de trabajo cooperativos y administrativos.

En las últimas versiones de Bonita se ha estandarizado la definición del proceso del flujo de trabajo, a fin de permitir a los usuarios migrar fácilmente su definición de proceso con un mínimo costo. Esa es la razón por la que se decidió prestar apoyo a la norma XPDL. XPDL (XML Process Definition Language), un lenguaje basado en XML definido por la WFMC con el objetivo de tener interoperabilidad entre los distintos proveedores de flujos de trabajo. De esa manera podríamos fácilmente importar una definición de proceso desde un proveedor compatible con XPDL.

La mayoría de estas características representan un importante valor añadido con respecto a los otros motores de flujo de trabajo de código abierto, en términos de flexibilidad, soporte para procesos corporativos y el beneficio que brinda el servidor de aplicaciones a través de la calidad de servicios. Estos servicios están disponibles para el usuario final a través de un conjunto de ricas APIs Java. De esta forma, los usuarios pueden definir y ejecutar los procesos de flujo de trabajo a través del uso de los APIs [8].

Calidad de Servicios

Uno de los principales logros en Bonita Workflow es que todos los datos, incluidos los datos relevantes de flujo de trabajo y los estados internos, se almacenan durante la ejecución de todo el proceso. Esto significa que cada vez que una actividad modifica su estado este cambio se guarda en la base de datos de Bonita.

El mecanismo de persistencia y de otros servicios, como el de seguridad, correo electrónico, conectores JCA (conecta servidores de aplicaciones y sistemas de información de empresa) o notificaciones de eventos son gestionados automáticamente por el servidor de aplicaciones en la que el motor esté en marcha. Este enfoque tiene muchas ventajas en términos de calidad del servicio.

Arquitectura [9]

Bonita es un sistema de flujo de trabajo J2EE compatible con Enterprise JavaBeans 2.0 (EJB) que proporciona una especificación más flexible y portátil para el entorno de aplicaciones distribuidas. Usando EJB 2.0 y el motor de generación de código Xdoclet (motor de código abierto para el lenguaje de programación Java, cuya función es la generación de código), el ambiente Bonita genera descriptores XML y todas las interfaces de Enterprise Java Beans. Los Beans son componentes de software reutilizables. Todos los Beans en la última versión de Bonita se manejan dentro del contenedor de persistencia CMP 2.0, el cual maneja la persistencia de estas entidades automáticamente en tiempo de ejecución. En la figura 2.2 se muestra la arquitectura de Bonita.

Código de campo cambiado

Código de campo cambiado

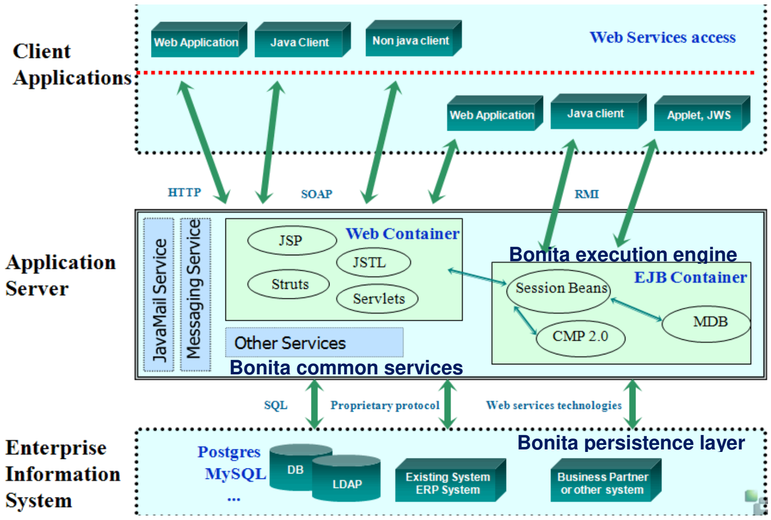


Figura 2.2 Arquitectura de Bonita.

Persistencia en Bonita: las principales unidades de la capa de persistencia son los proyectos. Los proyectos de Bonita representan flujos de trabajo que serán definidos por las organizaciones empresariales. Cada proceso de flujo de trabajo contiene básicamente información (nombre de proceso, el usuario creador, el estado de ejecución del proceso) y un gran número de datos asociados, en términos de actividades, las conexiones entre estas actividades, proceso de datos, proceso de las propiedades, los participantes, etc. Para separar los datos del proyecto de los elementos que lo componen, se ha definido beans portables para las relaciones con las demás entidades que componen representación del

Workflow Bonita: actividades, usuarios, roles, propiedades plazos y eventos. El gestor de persistencia mapea el bean de entidad a la base de datos de Bonita y permite implementar y ejecutar métodos basados en el lenguaje de consulta de bases de datos EJB QL.

Motor de ejecución de Bonita: el motor de ejecución está diseñado para mejorar la flexibilidad de la ejecución del proceso. Esta flexibilidad permite a las actividades compartir resultados intermedios durante la ejecución, por lo tanto, la ejecución no se realiza de una manera aislada. La ejecución del flujo de trabajo se basa en el principio de la anticipación, lo que permite una actividad alcanzar el tradicional modelo de sincronización inicio-fin. De la misma manera, el usuario puede cancelar una actividad en ejecución o cambiar el modo de ejecución.

El componente de ejecución en Bonita está compuesto por el EngineSession Bean, los modos de ejecución y la aplicación Worklist. El motor de estado de Sesión Bean, define a todas las operaciones de ejecución del proceso: inicio de actividad, fin de actividad, cancelar la actividad y terminar el proceso de ejecución. Se basa en una implementación recursiva que gestiona las operaciones de ejecución y propaga los cambios en la actividad a las demás actividades que están conectadas a ésta.

El motor de ejecución se encarga de gestionar los distintos modos de ejecución de Bonita. Los participantes en un proceso de flujo de trabajo pueden modificar el modo de ejecución de cada una de las actividades que componen su proyecto mediante el establecimiento anticipado del valor condicional: verdadero o falso.

El motor de ejecución de Bonita también introduce el concepto de hook o evento. Los hooks son las unidades de código fuente que están asociados a las actividades del proceso de flujo de trabajo. Estos eventos deben ser escritos en Java o en uno de lenguajes de scripting disponibles en Bonita (TCL, BeanShell, etc). El usuario puede asociar hooks en diferentes momentos del ciclo de vida de la ejecución de la actividad: antes de iniciar, después de iniciar, crear, antes de terminar, después de terminar, etc. Con la finalidad de ejecutar alguna acción necesaria.

Servicios Comunes: el sistema de Workflow Bonita integra un gran número de servicios para el control y simplificación de muchos aspectos relativos a la cooperación.

- Servicio de mensajes JMS: implementación para notificar la definición y cambios en la ejecución dentro de un proceso de flujo de trabajo. Cada interacción con el usuario es notificado al núcleo de Bonita y lanza un evento JMS (API de Servicios de Mensajería de Java).

Código de campo cambiado

- Servicio de notificación Jabber: que permite a los usuarios recibir notificaciones en tiempo real e intercambio de diferentes tipos de mensajes.

Después de describir el motor de flujo de trabajo a utilizar, debemos dar un vistazo al resto de la plataforma tecnológica. En la siguiente sección describiremos la plataforma general que se utilizará para implementar el sistema.

2.2.3. Descripción completa de la Plataforma tecnológica

La plataforma tecnológica se refiere a todas las tecnologías que se utilizarán para la implementación y puesta en marcha de un sistema, así como la función que cumple cada una y la interacción entre estas.

La plataforma ha sido elegida teniendo en cuenta factores como confiabilidad, integridad y accesibilidad de cada una de las tecnologías. En el gráfico 2.3 muestra la plataforma tecnológica utilizada en el sistema.

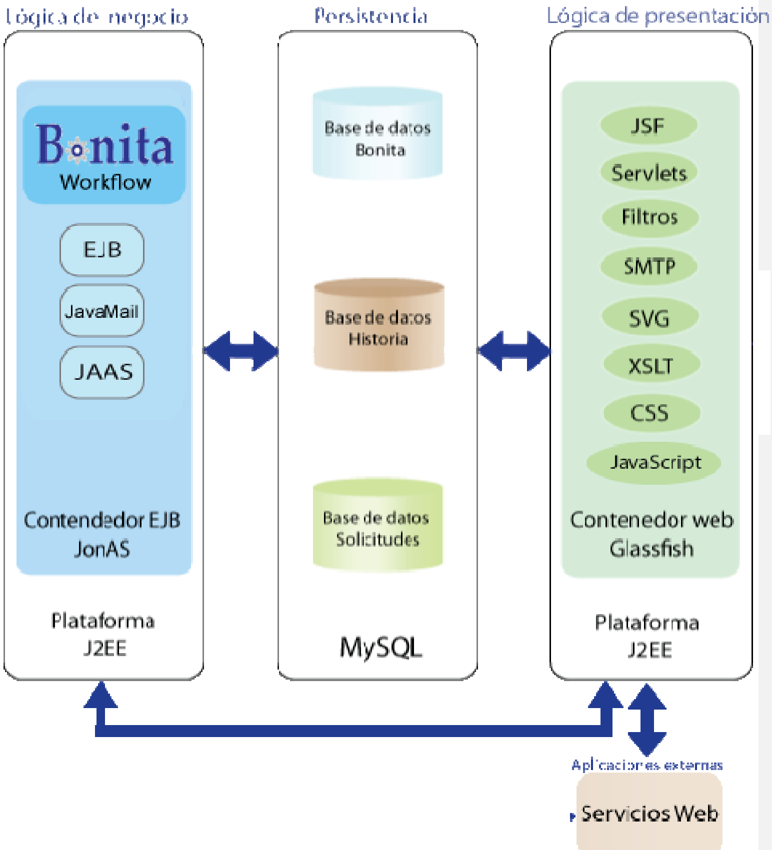


Figura 2.3 Plataforma tecnológica del sistema.

Tanto la lógica de negocio como la lógica de presentación fueron elaboradas sobre la tecnología J2EE (Java 2 Enterprise Edition). Además, todo el sistema fue implementado utilizando el lenguaje de programación Java y el entorno de desarrollo Netbeans 6.

Java, fue la primera solución de rápido desarrollo y despliegue para el negocio de escritorio y las aplicaciones empresariales.

La tecnología Java está disponible para el entorno operativo Solaris, Linux y Microsoft Windows, así como otras plataformas a través de licenciatarios. Y es considerado uno de los lenguajes más robustos del mercado.

Lógica del negocio:

La lógica del negocio en el sistema se refiere a los procesos para lo cual se utiliza el motor de flujo de trabajo Bonita.

Bonita Workflow: motor de flujo de trabajo el cual fue descrito en detalle en la sección 2.2.2.

Bonita J2EE es un sistema cooperativo de flujo de trabajo compatible con Enterprise JavaBeans 2.0 (EJB) especificación que proporciona una mayor flexibilidad y portabilidad para aplicaciones distribuidas. Utiliza el contenedor de EJBs de JonAs para correr los EJBs que administra.

El servidor de aplicaciones Jonas [10]: es un servidor de aplicaciones de código abierto implementado en la plataforma de programación J2EE. Su alto nivel de modularidad permite que sea utilizado como:

- Un servidor J2EE, para desplegar y ejecutar aplicaciones EAR (aplicaciones compuestas con componentes web y EJB).
- Un contenedor Web, para desplegar y ejecutar aplicaciones

web.

- Un contenedor EJB, para desplegar y ejecutar componentes EJB.

El motor de flujo de trabajo Bonita aprovecha esta modularidad para desplegar sus diferentes componentes, incluyendo su consola de administración “jiapAdmin”.

Este servidor de aplicaciones también cuenta con un servicio de correo, el cual es utilizado en el sistema para el envío de correos electrónicos de notificación. Este servicio está basado en la tecnología **JavaMail Service** el cual es un API de Java usado para recibir y enviar correos.

Otro servicio interesante que provee esta herramienta es el servicio de autenticación **JAAS (Java Authentication and Authorization Service)**, un conjunto de paquetes usado para autenticar y controlar el acceso. El sistema utiliza este servicio para autenticar a los usuarios que ingresan utilizando su cuenta de ESPOL.

Lógica de presentación:

Para la presentación se construyó una aplicación web que contiene todas las páginas con las que los usuarios interactuarán. Durante la construcción de ésta aplicación se han utilizado los siguientes elementos:

El Servidor **GlassFish J2EE** es un servidor de aplicaciones que implementa las tecnologías definidas en la plataforma Java y permite ejecutar aplicaciones que siguen esta especificación. La aplicación web Solicitudes FIEC corre sobre el contenedor web del servidor Glassfish.

JSF (Java Server Faces) [11]: es un marco de trabajo para aplicaciones basadas en web, que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. JSF usa Java Server Pages (JSP[4]) como la tecnología que permite hacer el despliegue de las páginas. Todas las páginas web del sistema han sido construidas utilizando esta tecnología.

JSF provee:

- Un conjunto de APIs para representar componentes de una interfaz de usuario y administrar su estado, manejar eventos, validar entradas, definir un esquema de navegación de las páginas y dar soporte para internacionalización y accesibilidad.
- Un conjunto por defecto de componentes para la interfaz de usuario.

---

[4] JSP: (Java Server Pages). Es una tecnología Java que permite generar contenido dinámico para web.

- Dos librerías de etiquetas personalizadas para JavaServer Pages que permiten expresar una interfaz JavaServer Faces dentro de una página JSP.
- Un modelo de eventos en el lado del servidor.
- Beans administrados.

Servlets: un servlet es un objeto que se ejecuta en un servidor o contenedor J2EE, fue especialmente diseñado para ofrecer contenido dinámico desde un servidor web.

Filtros: un filtro recepta pedidos y respuestas dinámicamente para usar la información contenida en ellos. Se utilizan para bloquear la autenticación dependiendo de la identidad del usuario.

SVG (Scalable Vector Graphics) es un lenguaje para describir gráficos vectoriales bidimensionales, tanto estáticos como animados. SVG es un estándar abierto desarrollado por la W3C (World Wide Web Consortium). Los archivos .svg son compactos y proveen gráficos de alta calidad y pueden ser manipulados con herramientas libres con por ejemplo Inkscape.

XSLT o **Transformaciones XSL [12]** es un estándar de la organización W3C que presenta una forma de transformar documentos XML en otros e incluso a formatos que no son XML. Las hojas de estilo XSLT realizan la transformación del documento utilizando una o varias reglas de plantilla. Estas reglas de plantilla, unidas al documento fuente a transformar, alimentan un procesador de XSLT que realiza las transformaciones deseadas poniendo el resultado en un archivo de salida, o, como en el caso de una página web, las hace directamente en un dispositivo de presentación tal como el monitor del usuario.

Las tecnologías SVG y XSLT se utilizan en el sistema para elaborar los contratos de ayudantes en el proceso *Solicitar Ayudantes*.

También se ha utilizado algunos componentes de la librería Ext JS para mejorar la interfaz gráfica de la aplicación. **Ext JS** es una librería JavaScript para construir aplicaciones ricas en internet. JavaScript es un lenguaje de programación interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web.

Persistencia de información:

Para almacenar la información del sistema se ha escogido el servidor de base de datos My SQL.

My SQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Es muy utilizado en aplicaciones web porque hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

Aplicaciones externas:

Para proveer una interfaz de comunicación con sistemas externos se utilizaron Servicios Web o Webservices. Un **servicio web** [13] es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos. Por ejemplo, en el sistema se ha utilizado esta tecnología para obtener información de los usuarios, proporcionada por el Centro de Servicios Informáticos de ESPOL (CSI).

Después de revisar en detalle la plataforma a utilizar, en el siguiente capítulo se analizarán los procesos a implementar.

CAPÍTULO III

3. ANÁLISIS DE LOS PROCESOS

En este capítulo se analizarán los procesos a automatizar en el sistema, detallando cada uno de los elementos necesarios para su implementación.

Estos elementos son:

- **Roles:** responsabilidades de los participantes en una actividad.
- **Actividades:** conjunto de acciones que generalmente son desempeñadas por una única persona dentro de los roles que pueden realizar dicha tarea.
- **Rutas del Proceso:** caminos que va tomando un proceso en base a reglas de transición.
- **Actividades de ruteo:** permiten tomar decisiones que determinan la navegación (ruta) dentro del proceso. Expresan qué acción se va a tomar dependiendo de la evaluación de reglas de transición.

- **Iteraciones:** una iteración permite regresar a un punto determinado del proceso. Es decir, recorrer el proceso nuevamente con la posibilidad de cambiar ciertos parámetros que nos lleven a la ruta deseada.
- **Plazos:** los plazos colocados en los procesos analizados determinarán la fecha límite para realizar una actividad.

3.1. Proceso Solicitar Ayudantes

Proceso que inicia el profesor para la aprobación de la contratación de ayudantes académicos o de actividades varias. La solicitud va dirigida al Decano de la facultad para su evaluación, si ésta es aprobada se inicia el concurso de ayudantía en el cual podrán participar todos los estudiantes de la facultad.

Roles: los distintos roles que participan en el proceso son:

- Profesor (solicitante).
- Sistema (parte del motor, que permite la ejecución automática de actividades).
- Decano (autoridad encargada de la evaluación de la solicitud).
- Secretaria (administrativo encargado de la impresión de los contratos de ayudantes).

Actividades: las actividades identificadas en el proceso de solicitar ayudantes son:

- **Solicitar ayudantes:** una persona con rol profesor inicia una solicitud dirigida al Decano de la facultad requiriendo la contratación de ayudantes. El profesor deberá ingresar datos necesarios para el concurso de ayudantías, como por ejemplo los requisitos que deben cumplir los aspirantes, fecha máxima de recepción de solicitudes, fecha inicio y fin de la ayudantía, etc.
- **Revisar solicitud:** la solicitud enviada por el profesor llega al Decano para su evaluación.
- **Volver a solicitar:** notifica al profesor que su solicitud fue devuelta por el Decano y da la posibilidad de reiniciarla o finalizarla.
- **Configurar datos para convocatoria:** si el resultado de la evaluación fue favorable, el profesor podrá realizar esta actividad para configurar la información a enviar en la convocatoria. Además en esta actividad se puede modificar el plazo límite para recepción de solicitudes de ayudantía.
- **Enviar convocatoria a estudiantes:** una vez que la solicitud ha sido aprobada, se debe lanzar la convocatoria a concurso de ayudantías. Esta actividad es realizada automáticamente por el sistema, el cual se encarga de enviar un correo los estudiantes de la FIEC notificando la convocatoria para la ayudantía. Los estudiantes de la facultad podrán enviar sus solicitudes a través del sistema, durante el período de recepción de solicitudes.

- **Registrar estudiantes:** en esta actividad se espera a que los estudiantes envíen sus solicitudes antes de que se cumpla la fecha máxima de recepción de solicitudes, la cual fue configurada por el profesor en el momento de enviar su solicitud o modificada durante la actividad *Configurar datos para convocatoria.*
- **Seleccionar ayudantes:** una vez finalizado el período de recepción de solicitudes, el profesor puede elegir a los ayudantes entre las solicitudes recibidas.
- **Enviar notificación a estudiantes seleccionados:** una vez realizada la selección se envía una notificación a los estudiantes elegidos. Esta actividad es realizada automáticamente por el sistema, el cual se encarga de enviar un correo a los estudiantes seleccionados.
- **Esperar confirmación de Estudiantes Seleccionados:** en esta actividad se espera a que los estudiantes seleccionados confirmen si aceptan o no la ayudantía. Permanece activa mientras se cumple la fecha máxima de recepción de confirmación, la cual fue configurada por el profesor cuando realizó la selección de los ayudantes.
- **Revisar resultados:** una vez terminada la recepción de las confirmaciones, el profesor las analiza; y decida la próxima acción a tomar.

- **Revisar informe de selección:** el informe del proceso de selección de los nuevos ayudantes es revisada por el Decano.
- **Imprimir Contratos:** una vez aprobados, los contratos son impresos por la secretaria para ser enviados a las entidades correspondientes fuera de FIEC.

En la tabla 3.1 se resume la relación entre actividades y roles dentro del proceso Solicitar ayudantes.

Actividad	**Rol ejecutor**
Solicitar ayudantes	Profesor
Revisar solicitud	Decano
Configurar datos para convocatoria	Profesor
Enviar convocatoria a estudiantes	Sistema
Registrar estudiantes	Sistema
Seleccionar ayudantes	Profesor
Enviar notificación a estudiantes seleccionados	Sistema
Esperar confirmación de estudiantes seleccionados	Sistema
Revisar solicitudes	Profesor
Revisar informe de selección	Decano
Imprimir Contratos	Secretaria

Tabla 3.1 Relación actividad-rol proceso Solicitar ayudantes.

Rutas del Proceso: la figura 3.1 muestra las rutas existentes en el proceso Solicitar Ayudantes.

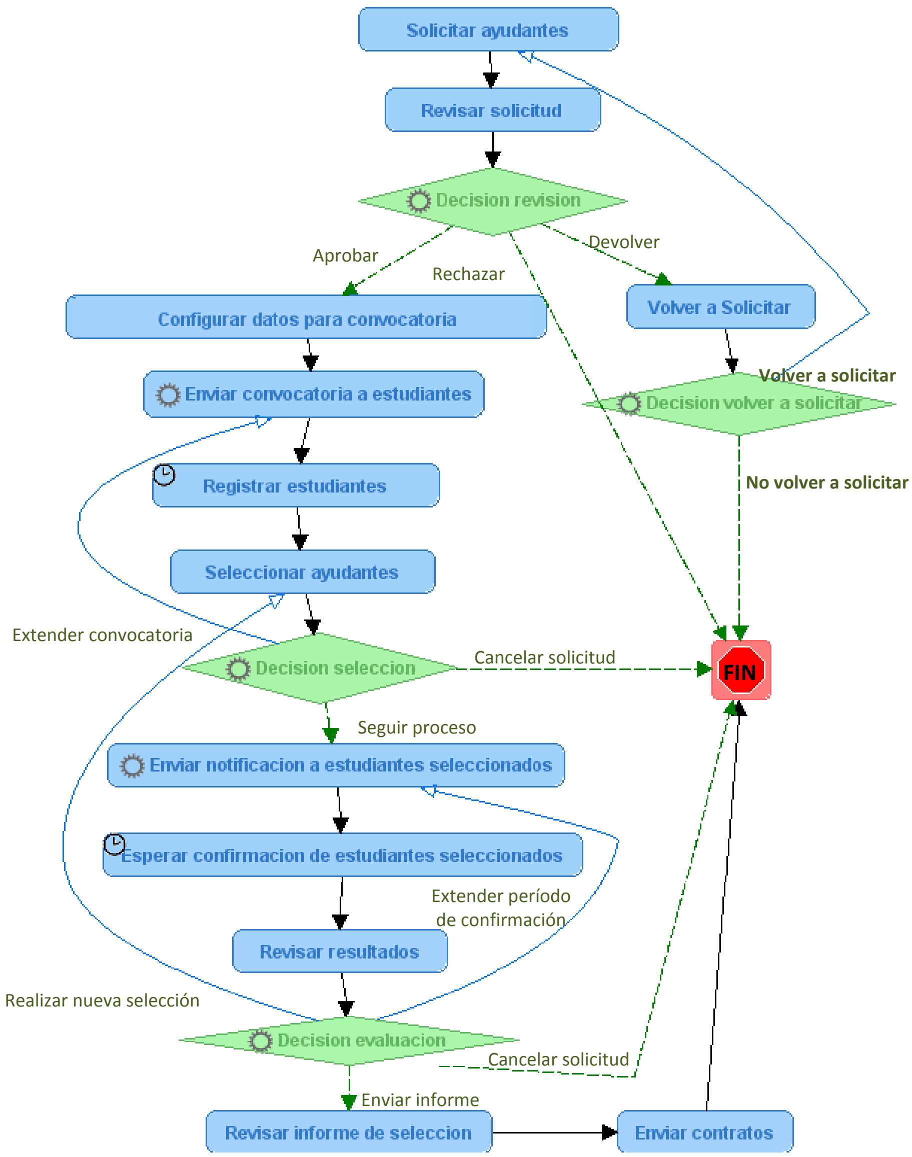


Figura 3.1 Rutas del proceso Solicitar Ayudantes.

Actividades de ruteo: en el proceso Solicitar ayudantes se encontraron las siguientes actividades de ruteo:

- **Decisión revisión:** en esta actividad el Decano decide si aprueba, rechaza o devuelve la solicitud enviada por el profesor.
- **Decisión volver a solicitar:** en caso de que la solicitud del profesor haya sido devuelta, él puede decidir reiniciar su solicitud o finalizarla.
- **Decisión selección:** en esta actividad el profesor puede decidir seguir con el proceso, cancelar la solicitud o extender el período de recepción de solicitudes.
- **Decisión evaluación:** después de recibir las confirmaciones de los estudiantes seleccionados, el profesor evalúa si: debe enviar el informe de selección, extender el período de confirmación, realizar una nueva selección de ayudantes o cancelar la solicitud.

Iteraciones: las iteraciones existentes en este proceso son:

- Si la solicitud es devuelta por el Decano, el profesor puede elegir volver a solicitar ayudantes, esta elección hace que el proceso regrese a la actividad *Solicitar ayudantes* generando la iteración **Volver a solicitar.** El proceso regresa al punto en que el profesor puede hacer las correcciones necesarias y volver a enviar su solicitud.

- Durante la selección de ayudantes el profesor puede elegir volver a *Enviar convocatoria a estudiantes* para dar más tiempo a los estudiantes que deseen participar en la selección. Esto genera la iteración **Extender convocatoria.**
- Durante la actividad *Revisar solicitudes* se encontraron dos posibles iteraciones. La primera se da si el profesor decide dar más tiempo a los estudiantes seleccionados para confirmar su aceptación o rechazo de la ayudantía (en caso de que no todos hayan respondido). Entonces el proceso regresa a la actividad *Enviar notificación a estudiantes seleccionados* y se genera la iteración **Extender período de confirmación**. La segunda iteración se produce cuando el profesor necesita seleccionar nuevos estudiantes para completar el número de horas solicitadas. En este caso el proceso regresa a la actividad *Seleccionar ayudantes* y genera la iteración **Realizar nueva selección**.

Plazos: en el análisis del proceso se determinó la necesidad de colocar plazos máximos para determinadas tareas, los cuales se describen a continuación:

- **Plazo para registro de estudiantes en concurso de ayudantía:** es la fecha límite en que los estudiantes podrán subir sus solicitudes. Se ejecuta en la actividad *Registrar estudiantes* haciendo que ésta permanezca en ejecución durante el tiempo determinado por el profesor para recibir las solicitudes, una vez terminado este plazo la actividad finaliza, así como la recepción de solicitudes. El profesor solicitante deberá esperar a que finalice esta actividad para poder seguir con el trámite.

- **Plazo para confirmación de estudiantes seleccionados:** es el tiempo máximo en que los estudiantes seleccionados podrán responder si aceptan o no la ayudantía. Se ejecuta en la actividad *Esperar confirmación de estudiantes seleccionados*, la cual permanece en ejecución durante el tiempo que el profesor solicitante haya determinado. Luego de este tiempo la actividad finaliza y los estudiantes que no hayan contestado ya no podrán hacerlo. Igual que en el caso anterior, el profesor deberá esperar a que esta actividad finalice para poder seguir su trámite.

3.2. Proceso Solicitar Ser Ayudante

Proceso que inicia el estudiante para solicitar una ayudantía actualmente disponible. Está relacionada con una solicitud de ayudantes iniciada por un profesor.

Roles: los distintos roles que participan en el proceso son:

- Sistema (parte del motor, que permite la ejecución automática de actividades).
- Estudiante (solicitante).

- Profesor (persona que evalúa la solicitud).

Actividades: Dentro del conjunto de actividades desempeñadas en este proceso se incluyen:

- **Solicitar ayudantía:** un estudiante envía su solicitud para concursar en la selección de ayudantes. El estudiante debe ingresar la información necesaria para el concurso, como por ejemplo la cantidad de horas disponibles para la ayudantía o adjuntar su hoja de vida.
- **Revisar solicitud:** la solicitud enviada por el Estudiante llega al Profesor para su evaluación.
- **Configurar fecha máxima de confirmación:** esta actividad es realizada automáticamente por el sistema y sirve para determinar el plazo límite de recepción de confirmaciones.
- **Confirmar aceptación de ayudantía:** en caso de haber sido seleccionado, el estudiante debe confirmar su aceptación de la ayudantía para proceder a la contratación.
- **Prorrogar confirmación:** esta actividad se realiza si el profesor decide expandir el tiempo de recepción de confirmaciones. Esto puede darse por ejemplo, si uno o más de los estudiantes seleccionados no han contestado la confirmación.

La tabla 3.2 resume la relación entre actividades y roles dentro del proceso Solicitar ser ayudante.

Actividad	Rol ejecutor
Solicitar ayudantía	Estudiante
Revisar solicitud	Profesor
Configurar fecha máxima de confirmación	Sistema
Confirmar aceptación de ayudantía	Estudiante
Prorrogar confirmación	Profesor

Tabla 3.2 Relación actividad-rol proceso Solicitar se ayudante.

Rutas del Proceso:

A continuación la figura 3.2 muestra las rutas que puede tomar el proceso de Solicitar Ser Ayudante.

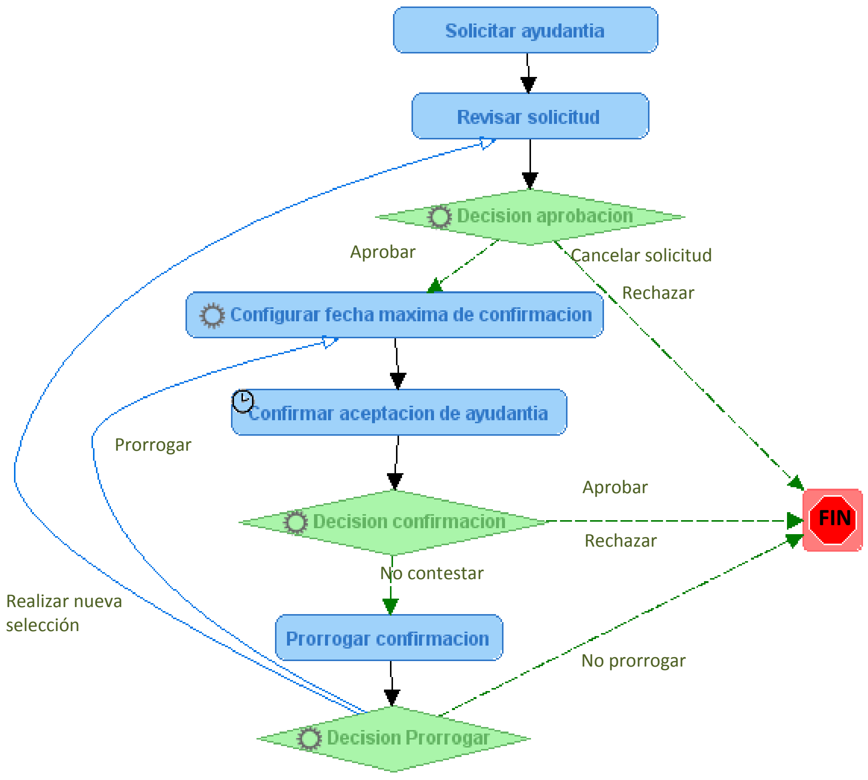


Figura 3.2 Rutas del proceso Solicitar Ser Ayudante.

Actividades de ruteo: en este proceso existen las siguientes actividades de enrutamiento:

- **Decisión aprobación:** la condición colocada en esta actividad direcciona el proceso de los estudiantes dependiendo si su solicitud de ayudantía fue aprobada o no.
- **Decisión confirmación:** si el estudiante seleccionado no acepta la ayudantía el proceso termina, caso contrario sigue el flujo del proceso.
- **Decisión prorrogar:** el profesor puede decidir extender el período de confirmación para que los estudiantes seleccionados confirmen su aceptación de la ayudantía. Además puede optar por realizar una nueva selección de ayudantes o cancelar la solicitud.

Iteraciones: en este proceso existen dos iteraciones, la primera se da cuando el profesor decide dar una prórroga de tiempo a los estudiantes que no han confirmado si aceptan o no la ayudantía. En este caso el proceso regresa a la actividad *Configurar fecha máxima de confirmación*, y genera la iteración **Prorrogar**.

La segunda iteración se da cuando el profesor solicitante decide volver a seleccionar ayudantes, en este caso los procesos de los estudiantes seleccionados, que no confirmaron si aceptan a no la ayudantía, regresan a la actividad *Revisar solicitud*, y están disponibles nuevamente para la selección. La iteración generada es **Realizar nueva selección**.

Plazos: el plazo existente en este proceso se ejecuta en la actividad *Confirmar aceptación de ayudantía*, y caracteriza el tiempo máximo que tiene el estudiante seleccionado para confirmar si acepta o no la ayudantía. La actividad mencionada puede finalizar por medio de la contestación del estudiante o porque expiró el plazo.

3.3. Proceso Solicitar Cambio de Carrera o Especialización

Proceso que inicia el estudiante para solicitar ser cambiado de carrera o especialización. La solicitud va dirigida al Subdecano de la facultad y puede requerir el informe del coordinador de la carrera.

Roles: Los distintos roles que participan en el proceso son:

- Estudiante (solicitante).
- Coordinador de Carrera (profesor encargado de emitir un informe acerca de la solicitud).
- Subdecano (autoridad encargada de evaluar la solicitud).
- Secretaria (administrativo encargado de la impresión del informe final de la solicitud).

Actividades: las actividades identificadas en este proceso son:

- **Solicitar cambio:** una persona con rol estudiante inicia una solicitud dirigida a Subdecano de la facultad para pedir se apruebe su cambio de carrera o especialización.
- **Revisar solicitud:** la solicitud del estudiante llega al Subdecano para su evaluación.
- **Volver a solicitar:** notifica al estudiante que su solicitud fue devuelta y da la posibilidad de reiniciarla o finalizarla. Si una solicitud fue devuelta el solicitante puede reiniciarla poniendo en práctica las recomendaciones del evaluador.
- **Emitir informe:** la solicitud llega al coordinador de la carrera, el cual emite un informe basado en la factibilidad del cambio solicitado.
- **Revisar informe:** el Subdecano evalúa la solicitud en base al informe del coordinador.
- **Imprimir solicitud para STA:** una vez que la solicitud es aprobada por el Subdecano, la secretaria imprime el informe resultante para enviarlo a la Secretaría Técnica Académica.

La tabla 3.3 resume la relación entre actividades y roles dentro del proceso Solicitar Cambio de Carrera o Especialización.

Actividad	Rol ejecutor
Solicitar cambio	Estudiante
Revisar solicitud	Subdecano
Volver a solicitar	Estudiante
Emitir informe	Coordinador de carrera
Revisar informe	Subdecano
Imprimir solicitud para STA	Secretaria

Tabla 3.3 Relación actividad-rol proceso Solicitar Cambio de Carrera o Especialización.

Rutas del Proceso: Las rutas posibles para el proceso de cambio de carrera o especialización se muestran en la figura 3.3:

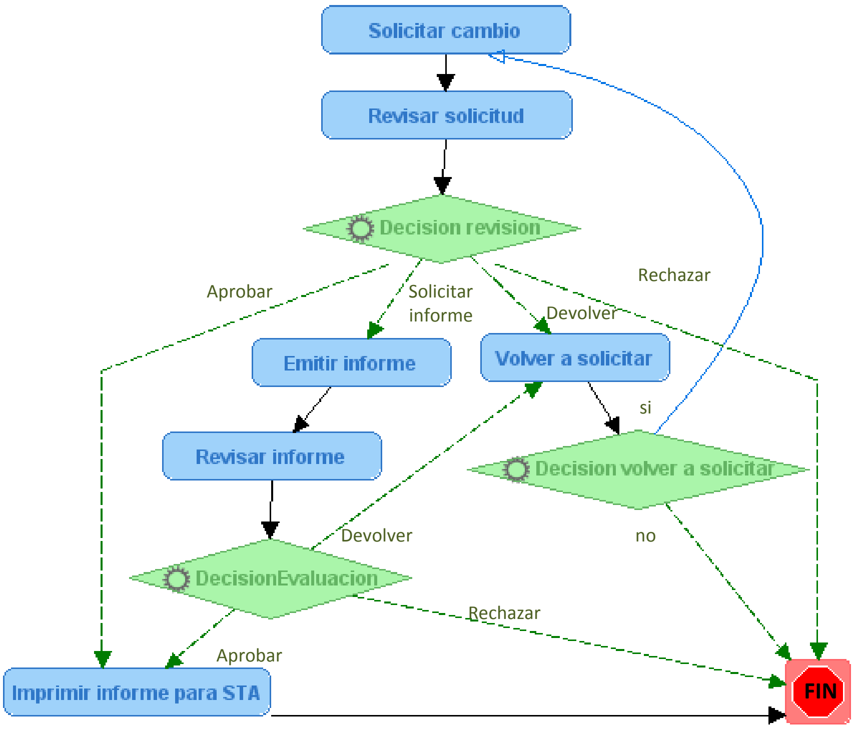


Figura 3.3 Rutas del proceso Solicitar Cambio de Carrera o Especialización.

Actividades de ruteo: en el proceso de Solicitar cambio de carrera o especialización encontramos las siguientes actividades de ruteo:

- **Decisión revisión:** en esta actividad el Subdecano puede solicitar el informe del coordinador de carrera o también aprobar, devolver, o rechazar la solicitud del estudiante.
- **Decisión volver a solicitar:** esta actividad permite al estudiante decidir entre terminar la solicitud o volver a solicitar después de que ha sido devuelta.
- **Decisión Evaluación:** en esta actividad el Subdecano decide si aprueba, devuelve o rechaza la solicitud del estudiante, en base al informe emitido por el coordinador.

Iteraciones: en este proceso se encontró necesaria una iteración, la cual se da cuando el estudiante es notificado de que su solicitud fue devuelta y decide volver a solicitar el cambio de carrera o especialización. El proceso regresa a la actividad *Solicitar cambio*, esto genera la iteración **Volver a solicitar.**

En el caso de este proceso no fue necesario colocar plazos.

3.4. Proceso Solicitar Convalidación de materia

Proceso que inicia el estudiante para solicitar la convalidación de un curso o seminario por una materia de FIEC. La solicitud va dirigida al

Subdecano de la facultad y puede requerir del informe de un evaluador.

Roles: Los distintos roles que participan en el proceso son:

- Estudiante (solicitante).
- Profesor (profesor encargado de emitir un informe acerca de la solicitud).
- Subdecano (autoridad encargada de evaluar la solicitud).
- Secretaria (administrativo encargado de la impresión del informe final de la solicitud).

Actividades: las actividades identificadas en este proceso son:

- **Solicitar convalidación:** persona con rol estudiante inicia una solicitud dirigida al Subdecano para convalidar una materia de FIEC. El estudiante deberá adjuntar los archivos necesarios para validar la convalidación, por ejemplo certificados de aprobación de cursos.
- **Revisar solicitud:** la solicitud del estudiante llega al Subdecano para su evaluación.
- **Volver a solicitar:** notifica al estudiante que su solicitud fue devuelta por el Subdecano y brinda la posibilidad de reiniciarla o finalizarla.

- **Emitir informe:** la solicitud llega al coordinador de la carrera, el cual emite un informe de favorable o no acerca de la convalidación.
- **Revisar informe:** el Subdecano aprueba o desaprueba la solicitud en base al informe del evaluador.
- **Imprimir solicitud para STA:** una vez que la solicitud es aprobada por el Subdecano, la secretaria imprime el informe resultante para enviarlo a la Secretaría Técnica Académica.

En la tabla 3.4 se resume la relación entre actividades y roles dentro del proceso Solicitar convalidación de materia.

Actividad	Rol ejecutor
Solicitar convalidacion	Estudiante
Revisar solicitud	Subdecano
Volver a solicitar	Estudiante
Emitir informe	Evaluador
Revisar informe	Subdecano
Enviar solicitud a STA	Secretaria

Tabla 3.4 Relación actividad-rol proceso Solicitar Convalidación de Materia.

Rutas del Proceso: las rutas que va tomando el proceso de convalidación de materia se muestran en la figura 3.4:

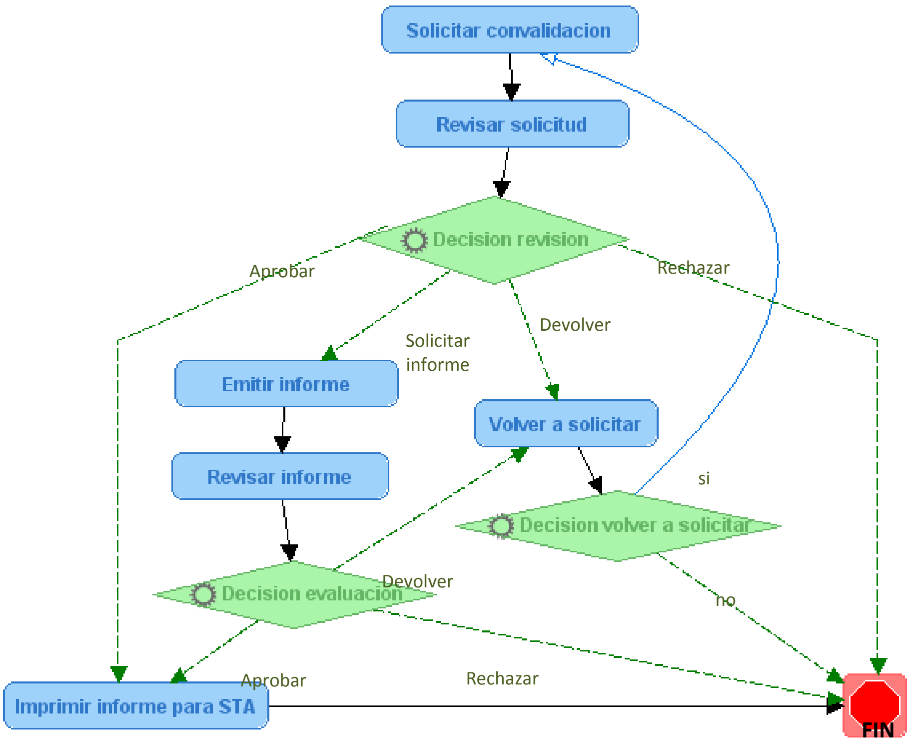


Figura 3.4 Rutas del proceso Solicitar Convalidación de Materia.

Actividades de ruteo:

- **Decisión revisión:** en esta actividad el Subdecano puede solicitar el informe de un profesor evaluador o también aprobar, devolver, o rechazar la solicitud del estudiante.
- **Decisión volver a solicitar:** esta actividad permite al estudiante decidir entre terminar la solicitud o reiniciarla después de que ha sido devuelta.

- **Decisión evaluación:** en esta actividad el Subdecano decide si aprueba, devuelve o rechaza la solicitud del estudiante.

Iteraciones: en este proceso se encontró necesaria una iteración, la cual se da cuando el estudiante es notificado de que su solicitud fue devuelta y decide reiniciarla. Al tomar esta decisión el proceso regresa a la actividad *Solicitar convalidación,* generándose la iteración **Volver a solicitar.**

En el análisis de este proceso se determinó que no es necesaria la utilización de plazos.

3.5. Proceso Solicitar Apertura Postgrado.

Proceso que inicia un profesor de FIEC para solicitar la apertura de un nuevo postgrado (diplomado o maestría). La solicitud va dirigida al Decano de la facultad y puede requerir del informe de evaluadores escogidos por el Decano. Luego pasa a ser evaluada por el consejo directivo de la FIEC y la secretaria ingresa la resolución final.

Roles: Los distintos roles que participan en el proceso son:

- Profesor (solicitante).
- Decano (autoridad encargada de evaluar la solicitud).
- Profesores (profesor encargado de emitir un informe acerca de la solicitud).

- Secretaria (administrativo encargado del ingreso e impresión de la resolución final de la solicitud).

Actividades: las actividades identificadas en este proceso son:

- **Solicitar apertura postgrado:** un profesor de FIEC solicita la apertura de un nuevo postgrado.
- **Revisar solicitud:** la solicitud es revisada por el Decano de la facultad.
- **Volver a solicitar:** el profesor es notificado de la devolución de su solicitud y puede optar por reiniciarla o finalizarla.
- **Emitir informe evaluador:** existirá una actividad de este tipo por cada uno de los evaluadores, los cuales analizarán la propuesta y emitirán el informe pertinente.
- **Revisar informes:** luego de que los evaluadores emiten su informe, el Decano evalúa la solicitud. Si el Decano opina que es factible la propuesta de apertura de postgrado, la enviará a consejo directivo para su evaluación final.
- **Ingresar resolución consejo directivo:** Una vez que la solicitud es analizada por el consejo directivo, la secretaria de la facultad ingresa la resolución tomada.

En la tabla 3.5 se resume la relación entre actividades y roles dentro del proceso Solicitar apertura postgrado.

Actividad	Rol ejecutor
Solicitar apertura postgrado	Profesor
Revisar solicitud	Decano
Volver a solicitar	Profesor
Emitir informe evaluador	Evaluador
Revisar Informes	Decano
Ingresar resolución consejo directivo	Secretaria

Tabla 3.6 Relación actividad-rol proceso Solicitar Apertura Postgrado

Rutas del Proceso:

La figura 3.5 muestra las rutas que puede tomar el proceso de Solicitar Apertura postgrado.

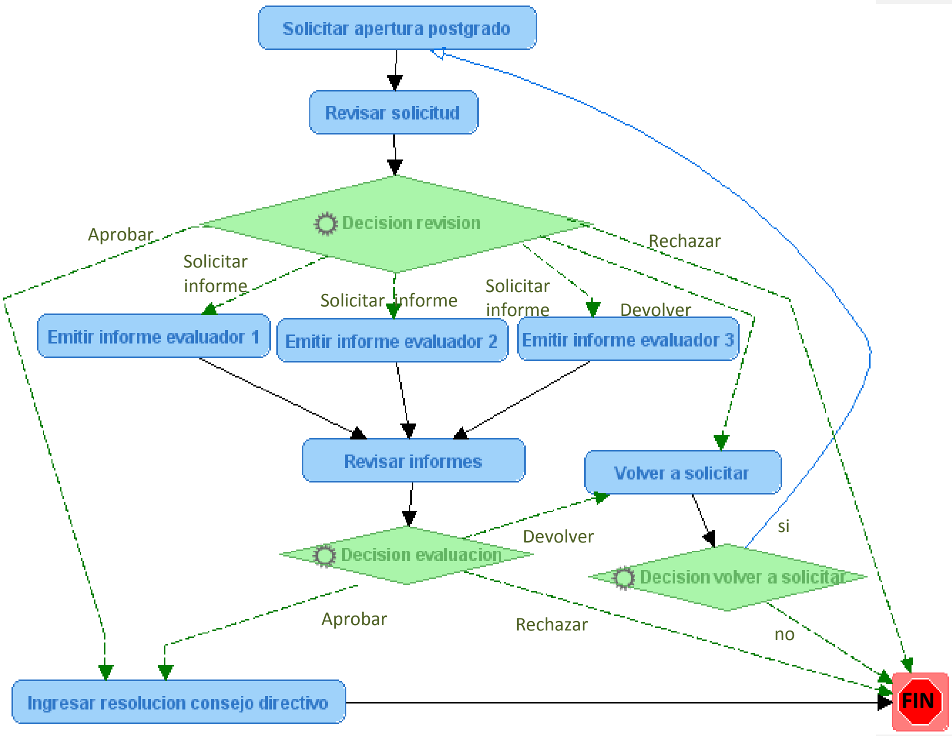


Figura 3.5 Rutas del proceso Solicitar Apertura postgrado.

Actividades de ruteo:

- **Decisión revisión:** en esta actividad el Decano puede solicitar el informe de evaluadores (de uno a tres) o también aprobar, devolver, o rechazar la solicitud de apertura de postgrado. Si la solicitud es aprobada por el Decano esta pasará a ser tratada en el Consejo Directivo de la FIEC.
- **Decisión volver a solicitar:** esta actividad permite al profesor reiniciar una solicitud luego de que haya sido devuelta por el Decano.
- **Decisión Evaluación:** En esta actividad se decide enviar a Consejo Directivo, devolver o rechazar lo solicitud.

Iteraciones: la iteración se produce cuando el profesor decide reiniciar su solicitud luego de que esta fue devuelta. El proceso regresa a la actividad *Solicitar apertura postgrado* y genera la iteración **Volver a solicitar.**

En este proceso no fue necesaria la colocación de plazos.

CAPÍTULO IV

4. DISEÑO E IMPLEMENTACIÓN

4.1. Esquema general del sistema.

El Sistema Solicitudes FIEC es una aplicación web, a la cual se podrá acceder a través de internet, con una cuenta de usuario de ESPOL. Analizando el sistema encontramos que es posible dividirlo en tres grandes módulos que son: los flujos de proceso y el motor Bonita, la aplicación web encargada de interactuar con el usuario y las bases de datos que almacenarán toda la información que recopile el sistema. En la figura 4.1 se esquematiza el diseño general del sistema.

Las siguientes secciones de este capítulo desglosan el desarrollo de cada uno de estos módulos.

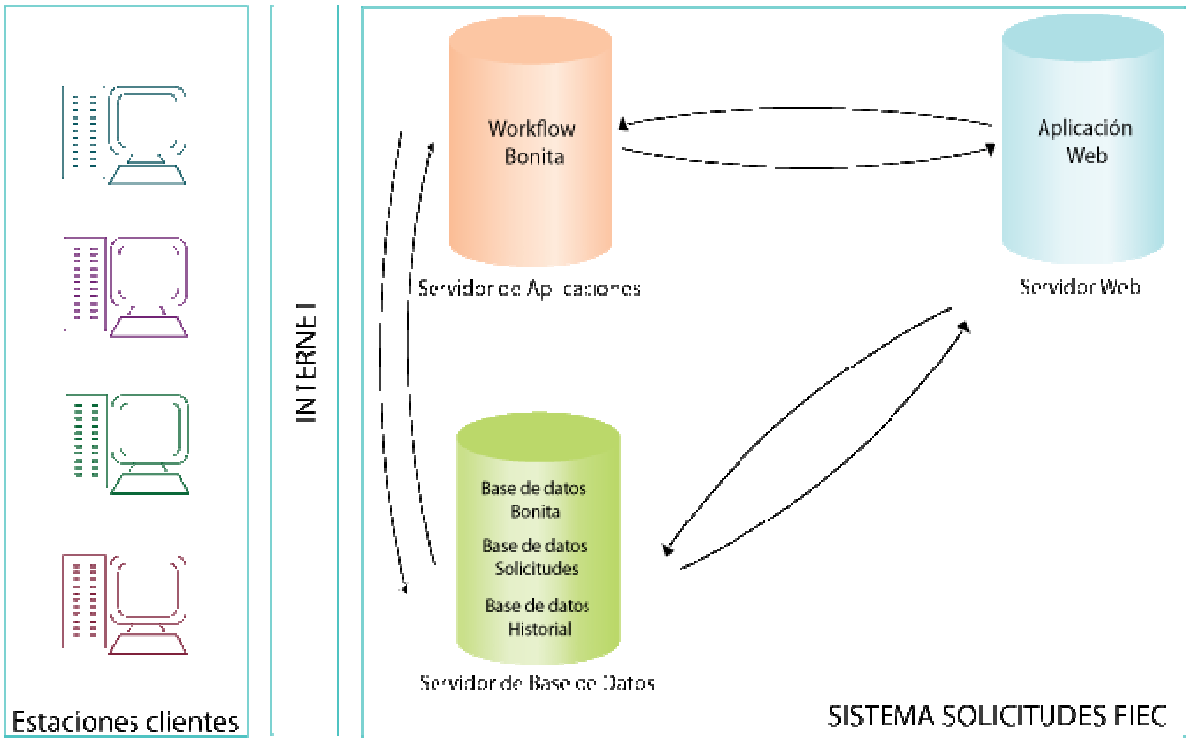


Figura 4.1 Esquema General del Sistema.

4.2. Diseño e Implementación de los procesos.

En esta sección se explicará cómo fueron implementados los procesos (analizados en el capítulo anterior), utilizando el Workflow Bonita.

El primer paso para implementar un proceso es definir el modelo. Se debe recordar que Bonita usa el lenguaje de definición de procesos XPDL, pero ventajosamente no es necesario definir el modelo utilizando el lenguaje, dado que el motor provee una herramienta de modelado que permite definir el proceso de manera gráfica, llamada **proed[5].** El diagrama generado (modelo de proceso) es almacenado como un archivo XPDL y luego debe ser importado al motor de Workflow Bonita para poder iniciar nuevas instancias del proceso.

Para administrar los procesos se utilizaron las herramientas provistas por Bonita: la consola de administración **jiapAdmin** para gestionar usuarios, importar definiciones de procesos en formato XPDL, eliminar procesos, administrar instancias, etc., así como el editor gráfico **proed** para construir modelos de procesos. Las figuras 4.1 y 4.2 muestran las interfaces de éstas herramientas.

---

[5] Para revisar el funcionamiento de esta herramienta se utilizó la guía de desarrollo de Bonita que viene dentro del paquete de instalación.

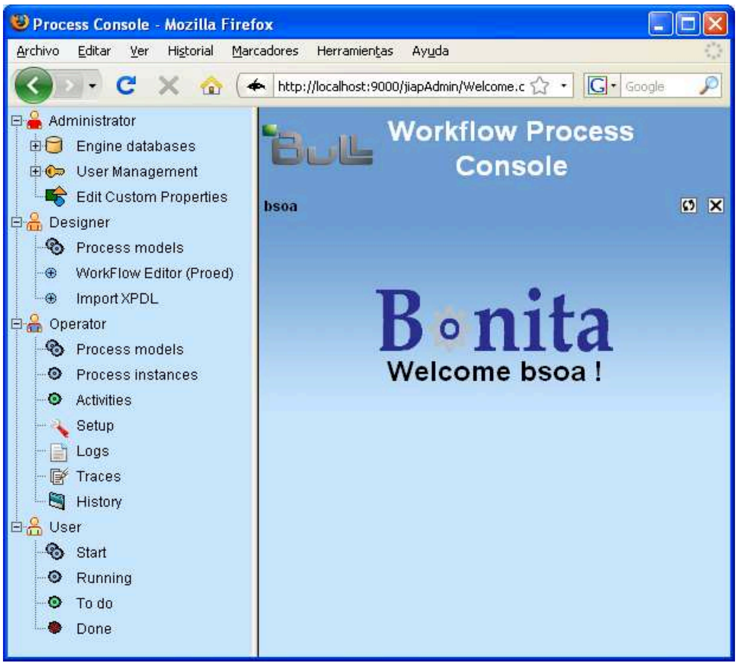


Figura 4.2 Interfaz de la consola JiapAdmin

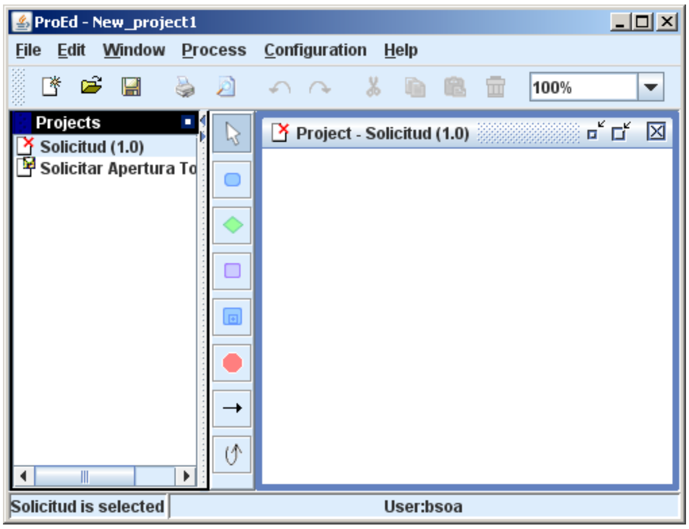


Figura 4.3 Interfaz del editor proed.

En diferentes puntos del modelo o definición de proceso podría ser necesario llamar a clases externas de Java, que desempeñen tareas específicas. Estas clases están divididas en categorías, acordes a la tarea que desempeñan durante el proceso. Estas categorías son:

Initiator Mapper: especifica qué personas pueden iniciar instancias de los procesos.

Mappers: especifica qué personas pertenecen a un rol del proceso en particular.

Hooks: ejecuta acciones en momentos específicos durante un proceso o actividad.

A continuación se detalla cómo se crea un modelo de proceso utilizando la herramienta de modelado provista por el motor Bonita.

4.2.1. Creación de un nuevo proceso.

Para crear un nuevo proceso o modelo de proceso se debe especificar un nombre, descripción y la clase que actuará como initiatorMapper. La figura 4.4 muestra la ventana de creación de un nuevo proceso.

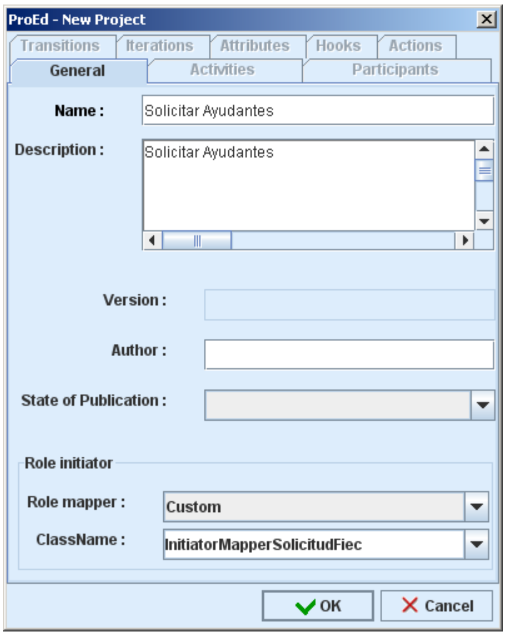


Figura 4.4 Nuevo proceso.

Un nuevo proceso tendrá por default una versión inicial de 1.0. Bonita incluye el versionamiento de los proyectos, esto permite que más de una versión del mismo proyecto convivan dentro del motor, lo cual será ventajoso en el caso de tener una transición entre procesos que deban ser modificados.

El creador determinará cuando es necesario cambiar la versión del proyecto en base a los cambios que se hayan realizado desde su creación.

Iniciadores: al crear el proyecto se debe definir la clase que funcionará como **initiatorMapper**. Esta clase se encargará de

especificar qué usuarios estarán autorizados para iniciar instancias del nuevo proceso. El servidor JonAS ejecuta el código de la clase cuando un usuario inicia sesión en el Workflow, lo cual permite obtener la lista de los procesos que puede iniciar el usuario, a través del API de Bonita.
Como se puede apreciar en la figura 4.2 cada proceso creado permite colocar el nombre de la clase que será utilizado como initiatiorMapper, para el Sistema Solicitudes FIEC se creó una sola clase llamada InitiatorMapperSolicitudFiec.

Atributos de proceso: un proyecto tiene propiedades o atributos, los cuales son pares propiedad/valor. Los atributos de proceso se propagan a todas las actividades, es decir podemos obtener su valor en cualquier momento del proceso después de la instanciación. El principal atributo que fue necesario crear es el idSolicitudProceso, el cual es el identificador de la solicitud dentro de la base de datos solicitudes FIEC. Este atributo permite relacionarnos con el registro que contiene la información actual de la solicitud.
Cuando un proyecto es creado, Bonita lo registra en su base de datos incluyendo los atributos que se hayan añadido. La figura 4.5 ilustra los atributos de un proceso.

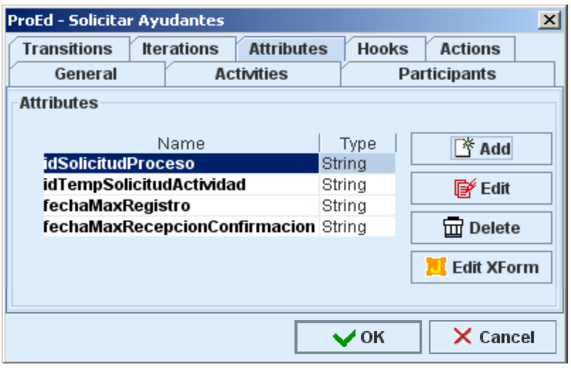


Figura 4.5 Ventana atributos de proceso.

4.2.2. Definición del flujo de actividades

Después de creado el proceso, se debe crear el flujo de las actividades que lo formarán.

El editor proed cuenta con una barra de herramientas donde encontramos los componentes necesarios para construir un modelo de proceso. La tabla 4.1 muestra la descripción de estos componentes:

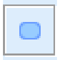
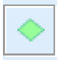
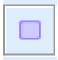
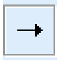
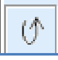
Botón	Descripción
	Cursor: configura el puntero a su uso estándar.
	Añadir actividad básica: crea una actividad básica (la unidad más pequeña de trabajo).
	Añadir actividad de ruteo: crea una actividad con condiciones de transición.
	Añadir un bloque de actividades: crea un conjunto de actividades.
	Añadir Sub-proceso: crea un modelo completo de un proceso de Workflow como una actividad.
	Añadir actividad fin: crea una actividad fin, la cual es en el proceso el punto final donde terminará una instancia del flujo de trabajo.
	Añadir transición: añade una transición entre dos actividades.
	Añadir iteración: añade una iteración entre dos actividades.

Tabla 4.1 Descripción de las herramientas para el diseño de un modelo de proceso.

Para crear una actividad usamos el botón añadir actividad básica de la barra de herramientas. Lo cual nos presenta la figura 4.6:

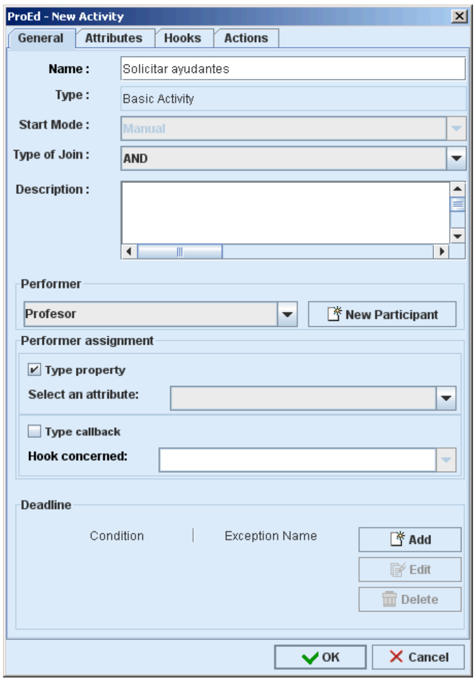


Figura 4.6 Nueva actividad.

Aquí es necesario colocar el nombre de la actividad, y el performer (ejecutor) el cual puede ser seleccionado de una lista donde están los participantes que se hayan creado durante la creación del proceso o también se puede optar por crear nuevos participantes. Este tipo de actividades inician de manera manual, es decir es necesario especificar quién la realizará.

Atributos de actividad: al igual que los atributos de proceso, los atributos de actividad son pares propiedad/valor, con la diferencia de que pueden o no ser propagados dependiendo de lo que se necesite. El principal atributo de actividad que tienen los procesos creados en esta tesis es el idSolicitudActividad el cual identifica la solicitud dentro de la base de datos solicitudes FIEC. Este atributo permite conocer el registro que contiene la información de la solicitud en el momento que pasó por la actividad dueña del atributo. De esta manera el sistema almacena un historial para que luego los participantes puedan conocer qué sucedió (como estuvo su solicitud) en cada paso del proceso

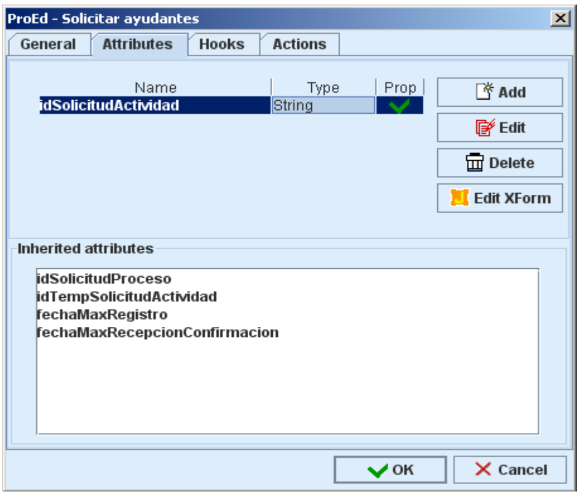


Figura 4.7 Atributos de actividad.

Actividades de ruteo: para añadir una actividad de ruteo se utiliza la herramienta añadir actividad de ruteo de la barra descrita en la tabla 4.1. Este tipo de actividades son puntos de control de flujo usados para condiciones transicionales complejas. Las actividades de ruteo serán realizadas por el sistema sin intervención de los participantes del proceso. Otra característica de este tipo de actividades es que no pueden invocar a clases hooks.

Participantes y mappers:

Para añadir nuevos participantes al proceso podemos hacerlo desde la ventana nuevo proyecto (pestaña participantes) o también desde la ventana nueva actividad (botón nuevo participante), en cualquiera de los casos obtendremos la ventana que muestra la figura 4.5. Aquí se puede observar que hay diferentes tipos de participantes, pero para la elaboración de los procesos sólo ha sido necesario utilizar el tipo Rol y sistema. Si el participante es de tipo Rol significa que todas las personas que pertenezcan a dicho rol podrán realizar la actividad y si el participante es de tipo sistema significa que el sistema será el encargado de realizarla. La figura 4.8 muestra cómo crear un nuevo participante.

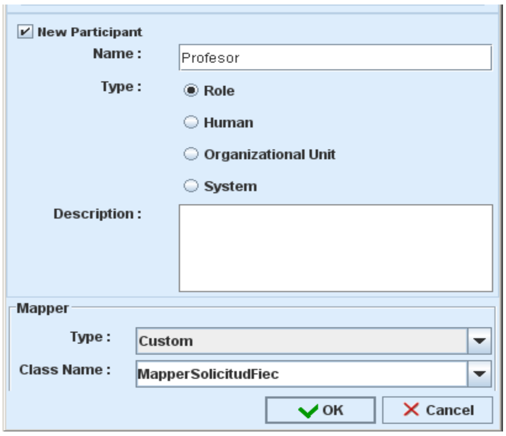


Figura 4.8 Nuevo participante.

Para poder enlazar los roles participantes con el conjunto de usuarios se utiliza una clase **mapper**. Esta clase se encargará de especificar qué personas corresponden a un rol específico definido en el modelo de proceso y su código es ejecutado cada vez que se crea una nueva instancia de proceso, es decir en este caso, cuando un usuario envía una solicitud.

Performer assignment:

Cuando en una actividad se coloca como performer (ejecutor) a un participante de tipo rol, se está especificando que todos los usuarios que pertenezcan a dicho rol pueden realizar la actividad. Pero ¿qué sucedería si se requiere que no todo el grupo, sino uno solo de ellos tenga acceso a realizar la actividad? En caso de que se conozca de antemano el usuario, se podría definir un nuevo participante de tipo humano, escribir el usuario y asunto arreglado. Pero cuando no se conoce qué usuario debe desempeñar la actividad sino hasta justo antes de que la actividad deba ser realizada, es necesario utilizar un performer assignment. Este caso se presentó en los procesos Solicitar convalidación de materia, Solicitar apertura de tópico y Solicitar apertura postgrado, en los cuales el evaluador de la solicitud requiere el informe de uno o varios profesores elegidos por él durante la actividad Revisar solicitud.

Los performer assignment son una característica de Bonita que permite colocar reglas de asignación adicionales tales como:

- Asignar la actividad a un usuario de un grupo llamando a una clase de Java (un hook) para hacer la selección del usuario.
- Dinámicamente asignar la actividad a un usuario usando una propiedad de la actividad.

En el sistema se utilizó la segunda regla de asignación, es decir, para saber quién debe realizar la actividad el motor toma el valor del atributo especificado. Este valor debe ser el usuario de la persona que debe desempeñar la actividad.
Una condición es que este usuario pertenezca al rol definido en performer. La figura 4.9 ilustra la colocación de un performer

assignment por propiedad.

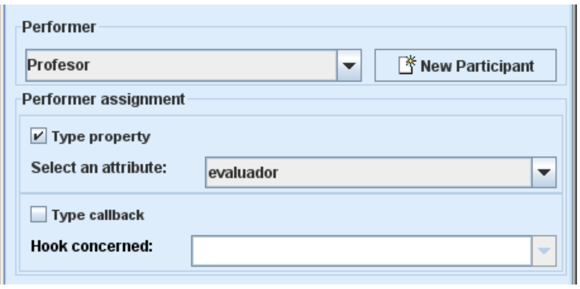


Figura 4.9 Performer assignment para una actividad.

Hooks y deadlines:

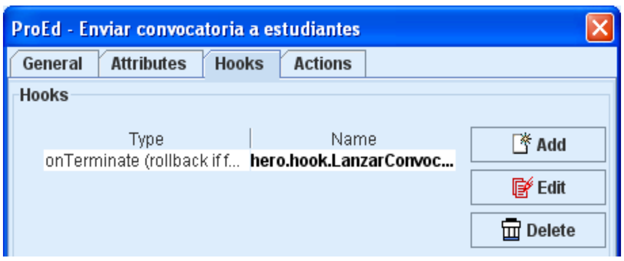
En ciertos casos es necesario ejecutar algún evento durante la realización de una actividad o de un proceso, para ello se utiliza las clases hooks. Los hooks, en el contexto del Workflow Bonita, son entidades lógicas (clases Java) que desempañan operaciones definidas por el usuario. Básicamente existen dos categorías:

- **Los hooks de proceso:** pueden ser llamados al inicio o al final del tiempo de vida del proceso.
- **Los hooks de actividad:** pueden ser llamados en diferentes momentos del tiempo de vida de la actividad, como se describe a continuación:
  - OnStart: el hook es llamado justo antes de iniciar la actividad.

- OnStart (rollback if failed): el hook es llamado justo después de iniciar la actividad.
- OnCancel: el hook es llamado antes de cancelar una actividad.
- OnTerminate: el hook es llamado justo antes de terminar la actividad.
- OnTerminate (rollback if failed): el hook es llamado justo después de que la actividad es terminada.
- On Ready: el hook es llamado cuando la actividad llega a estar lista para realizarse.
- OnDeadline: El hook es llamado cuando el plazo de la actividad expira.

En el Sistema Solicitudes FIEC se utilizaron los hooks para configurar las fechas límites de las actividades que contenían plazos. Por ejemplo para el proceso solicitar ayudantes, actividad *Enviar convocatoria a estudiantes,* se utilizó un hook para configurar el valor del atributo de proceso que representa la fecha máxima en que los estudiantes podrán enviar sus solicitudes para concursar en la selección de ayudantes.

En las figuras 4.10 y 4.11 se puede observar como añadir un hook a una actividad.



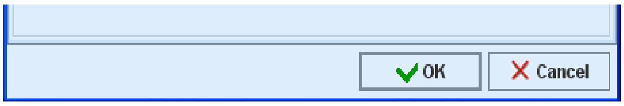


Figura 4.10 Hooks de actividad.

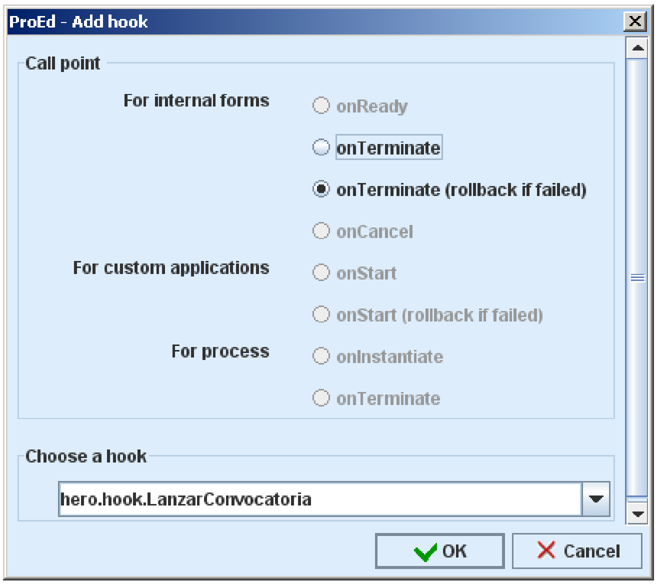


Figura 4.11 Añadir hook a una actividad.

Deadlines:

Otro uso de los hooks son los deadlines o plazos. Estas clases se ejecutan cuando se ha cumplido el plazo colocado para realizar la actividad y se encarga de finalizarla.

En el proceso solicitar ayudantes, por ejemplo, se utilizaron deadlines en la actividad *Registrar estudiantes* y *Esperar confirmación de estudiantes seleccionados.* Como se puede apreciar en las figuras 4.12 y 4.13 es posible configurar una duración o fecha para plantear el deadline. En este caso la fecha tuvo que ser colocada a través de un atributo en la actividad anterior dado que no era conocida de antemano.

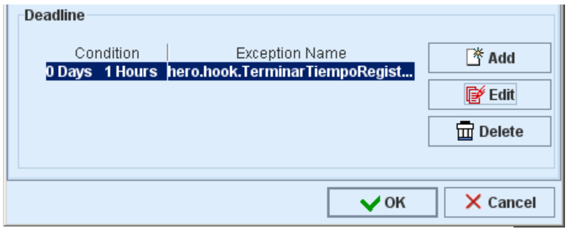


Figura 4.12 Deadlines de actividad.

Figura 4.13 Añadir deadline a una actividad.

4.2.3. Especificación de transiciones e iteraciones.

Para añadir una **transición** se utiliza el componente correspondiente de la barra de herramientas descrita en la tabla 4.1. Una transición es el nexo entre dos actividades y puede o no tener condiciones de cuyo valor dependerá la ruta que tome el proceso. La ventana que muestra la figura 4.14 se obtuvo de las propiedades de una transición y como se observa es posible colocar condiciones en base a atributos de actividad o proceso.

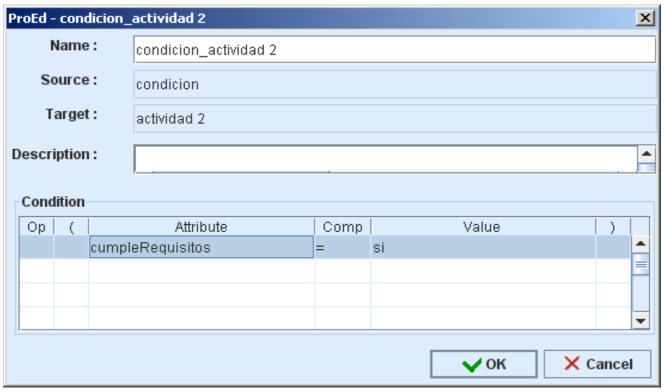


Figura 4.14 Propiedades de una transición.

Las **iteraciones** son necesarias cuando se requiere regresar a una actividad anterior en el flujo. Para añadir una iteración es necesario especificar cuál es la actividad origen, actividad destino y qué condición se debe cumplir para que el flujo regrese a la actividad origen. La condición se especifica mediante el valor de uno o más atributos (pueden ser de actividad o de proceso).

En las figuras 4.15 y 4.16 podemos ver ejemplos de cómo añadir transiciones e iteraciones a un modelo de proceso.

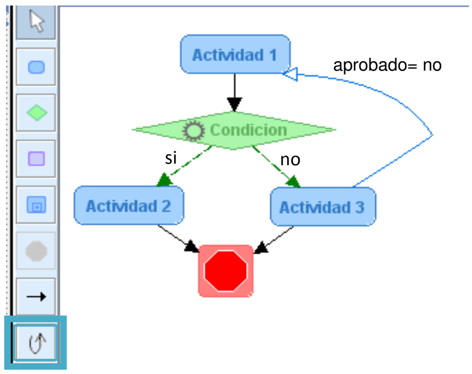


Figura 4.15 Ejemplo modelo de proceso con transiciones e iteraciones.

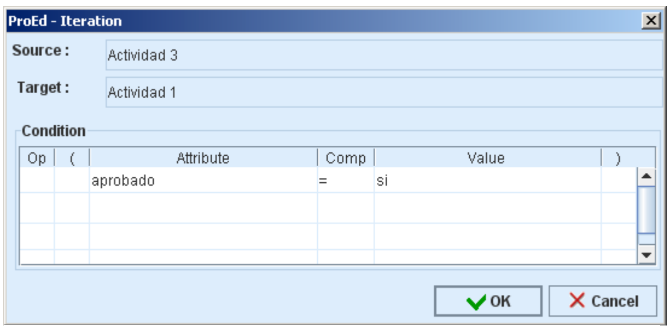


Figura 4.16 Añadir iteración entre dos actividades.

4.2.4. Puesta en marcha de un modelo de proceso en el motor Bonita.

Luego de realizar el modelo de proceso y guardarlo como un archivo XPDL, el siguiente paso es importarlo al motor Bonita. Para esto utilizamos la consola de procesos jiapAdmin. Una vez importado el modelo se puede empezar a crear instancias del proceso.

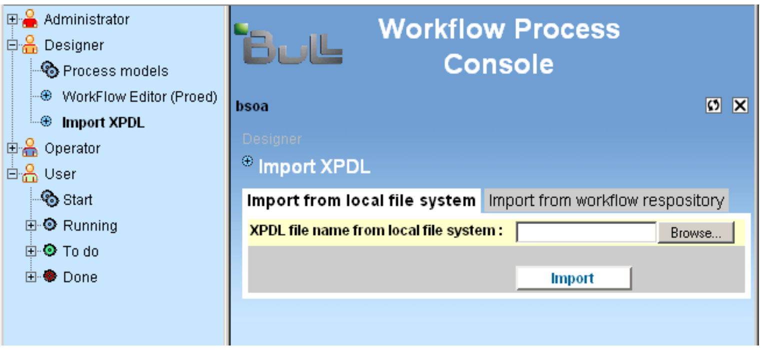


Figura 4.17 Consola de administración – importar procesos

4.3. Implementación aplicación Web.

El uso de las aplicaciones Web tiene múltiples ventajas, no necesitan ser descargadas ni configuradas permitiéndoles a los usuarios acceder en cualquier momento y desde cualquier lugar utilizando su respectivo usuario y contraseña.

Otra de las ventajas de las aplicaciones Web radica en la practicidad del navegador Web como cliente ligero y su habilidad de actualización y mantenimiento sin distribuir e instalar software en las computadoras de los potenciales usuarios.

Por los beneficios nombrados anteriormente se decidió que la aplicación a implementar debía ser Web. Dentro del diseño de la aplicación encontramos varios módulos importantes, los cuales detallamos a continuación:

4.3.1. Módulo ingreso.

Para obtener la información personal de los usuarios del sistema (estudiantes, profesores y personal administrativo de la FIEC) así como para realizar el proceso de autenticación se utilizó los servicios web que provee el Centro de Servicios Informáticos de la ESPOL (CSI).

Cuando el usuario ingresa su cuenta de ESPOL, el sistema llama al servicio web del CSI y verifica que los parámetros sean correctos. Si pasa esta revisión entonces verificará si el usuario ésta o no registrado en el sistema, si no lo está se procede a registrar sus datos y crear el usuario en la base de datos de Bonita para que pueda iniciar sesión en el motor y acceder a sus procesos (solicitudes).Con los datos obtenidos se podrá determinar a qué rol pertenece el usuario (estudiante, profesor, personal administrativo).

La figura 4.18 muestra la venta de ingreso del Sistema Solicitudes FIEC.

Figura 4.18 Ventana de ingreso sistema Solicitudes FIEC.

4.3.2. Módulo Procesamiento de solicitudes.

El punto central de la aplicación web son las solicitudes. Dentro de esto tenemos:

- Solicitudes a iniciar
- Actividades pendientes
- Solicitudes Activas
- Historial de solicitudes
- Búsqueda de solicitudes

Para implementar cada una de estas secciones se utilizó el API de Bonita.

Solicitudes a iniciar: en esta parte del sistema el usuario podrá ver la lista de las solicitudes que puede iniciar, de acuerdo a su rol y a la clase iniciadora que el motor ejecuta cuando inicia su sesión. Cuando un usuario inicia sesión en el sistema también lo hace en el Workflow.

Para iniciar una solicitud el usuario logoneado debe llenar el formulario correspondiente. Una vez que este formulario es enviado se crea una nueva instancia del proceso y se realiza la primera actividad del modelo, esta instancia representa a la solicitud dentro del motor de flujo de trabajo.

Los datos de esta nueva instancia de proceso son almacenados, por el motor, en la base de datos de Bonita y además los datos granulares de la solicitud, tales como nombres de archivos adjuntos, identificadores de los solicitantes, comentarios, etc. son almacenados en la base de datos de solicitudes. Como revisamos en la sección 4.2.7 cada actividad almacena el identificador de su registro en la base de datos solicitudes como un atributo de actividad.

Actividades pendientes: aquí obtenemos la lista de las actividades pendientes del usuario en los diferentes procesos activos. Para obtener esta lista recurrimos a métodos del API de Bonita que nos proveen de todas las actividades de una instancia que se encuentran en estado *listo para ejecución* en el motor.

Solicitudes activas: en esta sección el usuario podrá revisar el estado actual de las solicitudes activas en las que participa, ya sea como solicitante o como realizador de alguna de las actividades. Por estado actual se comprende la actividad actual en la que el proceso se encuentra, el nombre del solicitante, la última fecha de modificación y el nombre del ejecutor de la actividad actual (es decir quién tiene la solicitud actualmente). Además del estado actual, también se podrá revisar en detalle cómo estuvieron los datos de la solicitud en cada actividad por la que ha pasado.

Historial de solicitudes: aquí se localizan las solicitudes terminadas en las que el usuario ha participado, ya sea como solicitante o como realizador de alguna actividad. El sistema mostrará la fecha en que finalizó, el nombre del solicitante y además se podrá acceder al detalle de las solicitudes para revisar que sucedió en cada actividad. Para obtener los procesos finalizados hacemos uso de la base de datos historial, ya que la base de datos Bonita solamente almacena los procesos que se encuentran activos.

Búsqueda de solicitudes: en esta sección el sistema permite realizar búsquedas de solicitudes por diferentes criterios como: nombre de solicitante, tipo de solicitud, fecha de creación y finalización, etc.

4.3.3. Módulo de administración

El modulo de administración podrá ser accedido sólo por los usuarios administradores. En esta sección el sistema permitirá gestionar el ingreso, modificación y eliminación de roles, usuarios, carreras, materias, ayudantías.

4.3.4. Interfaz gráfica

Las vistas de cada uno de los módulos del sistema Solicitudes FIEC se crearon utilizando componentes JSF. Estos componentes fueron provistos por el marco de trabajo Visual Web Java Server Faces que viene junto con el entorno de desarrollo Netbeans 6. En el anexo 1 (Manual de usuario) se incluyen detalles de la interfaz gráfica.

4.4. Diseño de las bases de datos

Para el sistema Solicitudes FIEC se han utilizado tres bases de datos las cuales almacenan la información del sistema de acuerdo a:

Base de datos Bonita: esta base de datos es administrada totalmente por el motor de Workflow Bonita y aquí se almacenan los registros correspondientes a los procesos, instancias de procesos, actividades, roles, usuarios y demás necesarios para describir un proceso. La figura 4.19 muestra el diagrama entidad-relación de la base de datos.

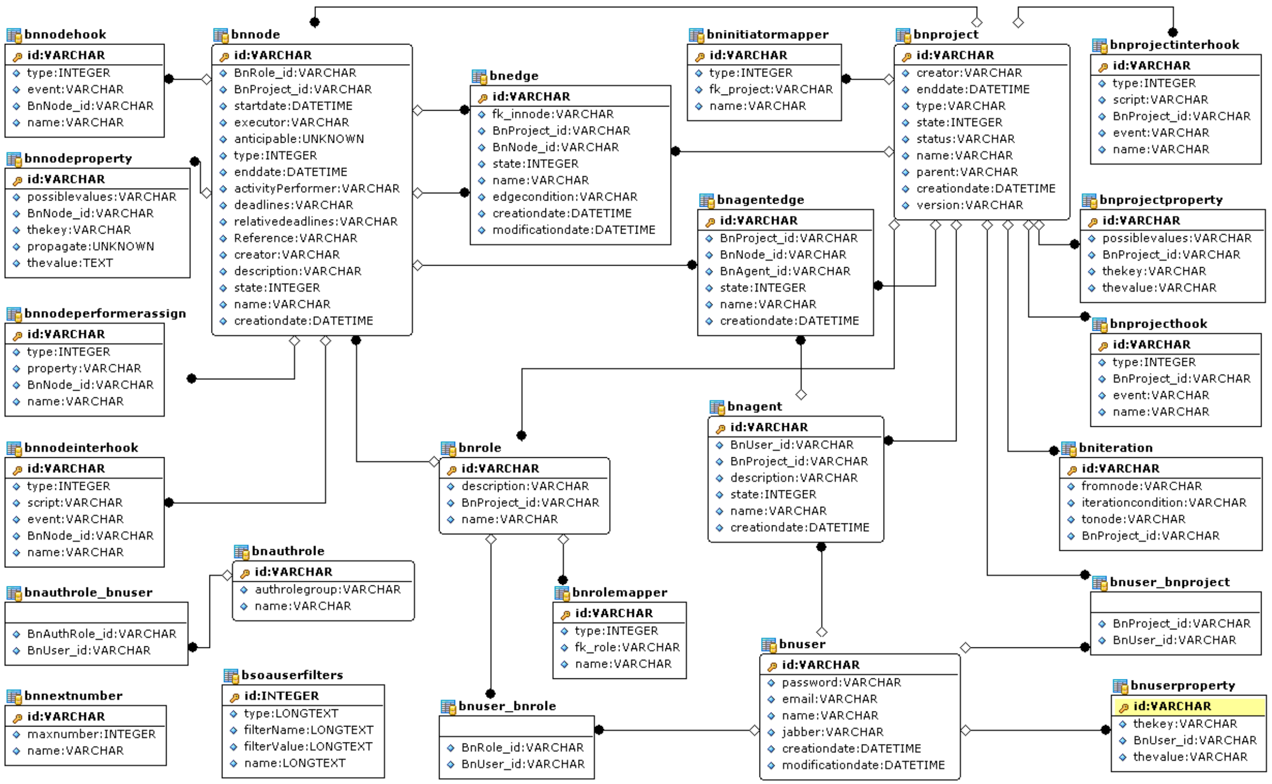


Figura 4.19 Diagrama Entidad-Relación base de datos Bonita.

Base de datos historial: El motor Bonita provee de historial, almacenando un archivo XML por cada instancia de proceso que finaliza. Para implementar el sistema se reemplazaron los archivos XML por una base de datos transaccional, debido a que manejar la información de las instancias a través de un archivo XML es mucho más complejo y requiere de mayor cantidad de tiempo para el acceso a los datos.

La base de datos para el historial replica información de las tablas más importantes de la base de datos de Bonita. Cada vez que el motor Bonita ingresa datos a su base, un trigger se encarga de duplicar ese registro en la base de datos de historial. Esta acción es realizada a fin de salvaguardar las instancias de proceso que finalizan su flujo, dado que Bonita las elimina para mantener su base de datos con pocos registros y así optimizar tiempos de respuesta. La figura 4.20 muestra el diagrama entidad-relación de ésta base de datos.

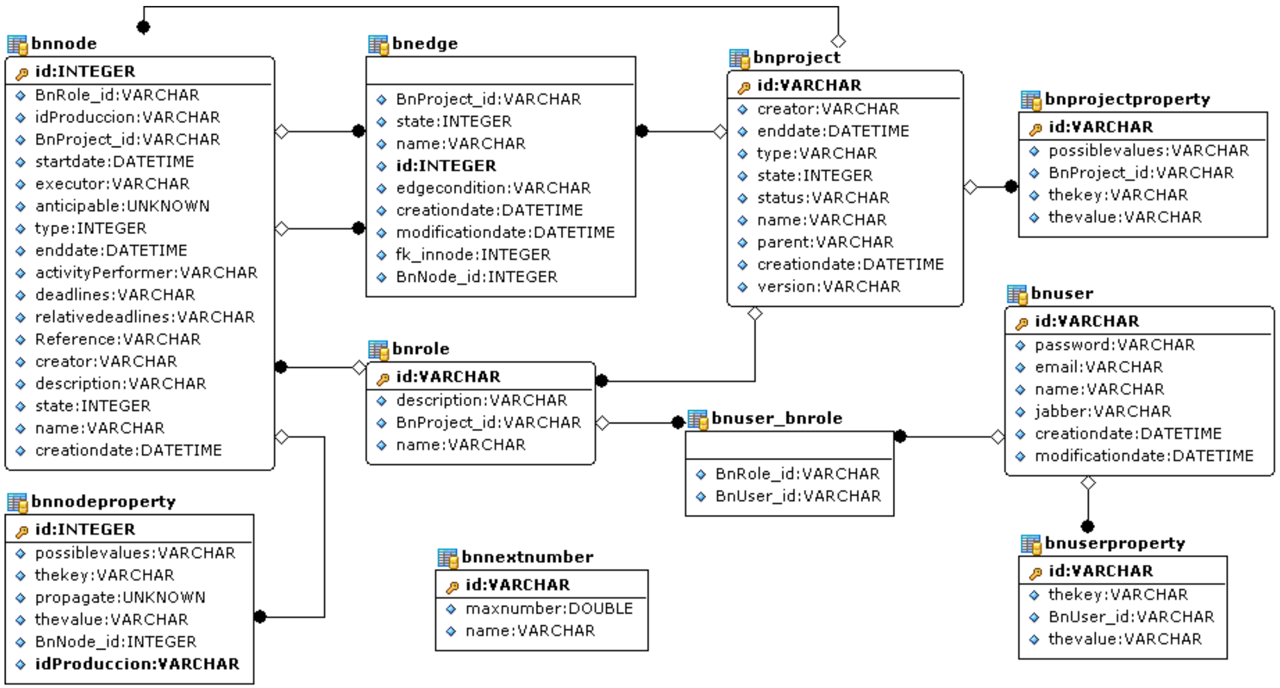


Figura 4.20 Diagrama Entidad-Relación base de datos Historial.

Base de datos solicitudes: la base de datos Bonita almacena netamente información de los procesos tales como las actividades, transiciones, condiciones, atributos, etc. Por ello fue necesario crear una base de datos para almacenar los datos granulares de las solicitudes. El modelo de esta base contiene una tabla por cada solicitud, además de los datos personales de los usuarios y los roles a los que pertenecen, y también almacena datos necesarios para iniciar las solicitudes como las materias y carreras de la facultad. En la siguiente figura se muestra el diagrama entidad-relación de esta base de datos.

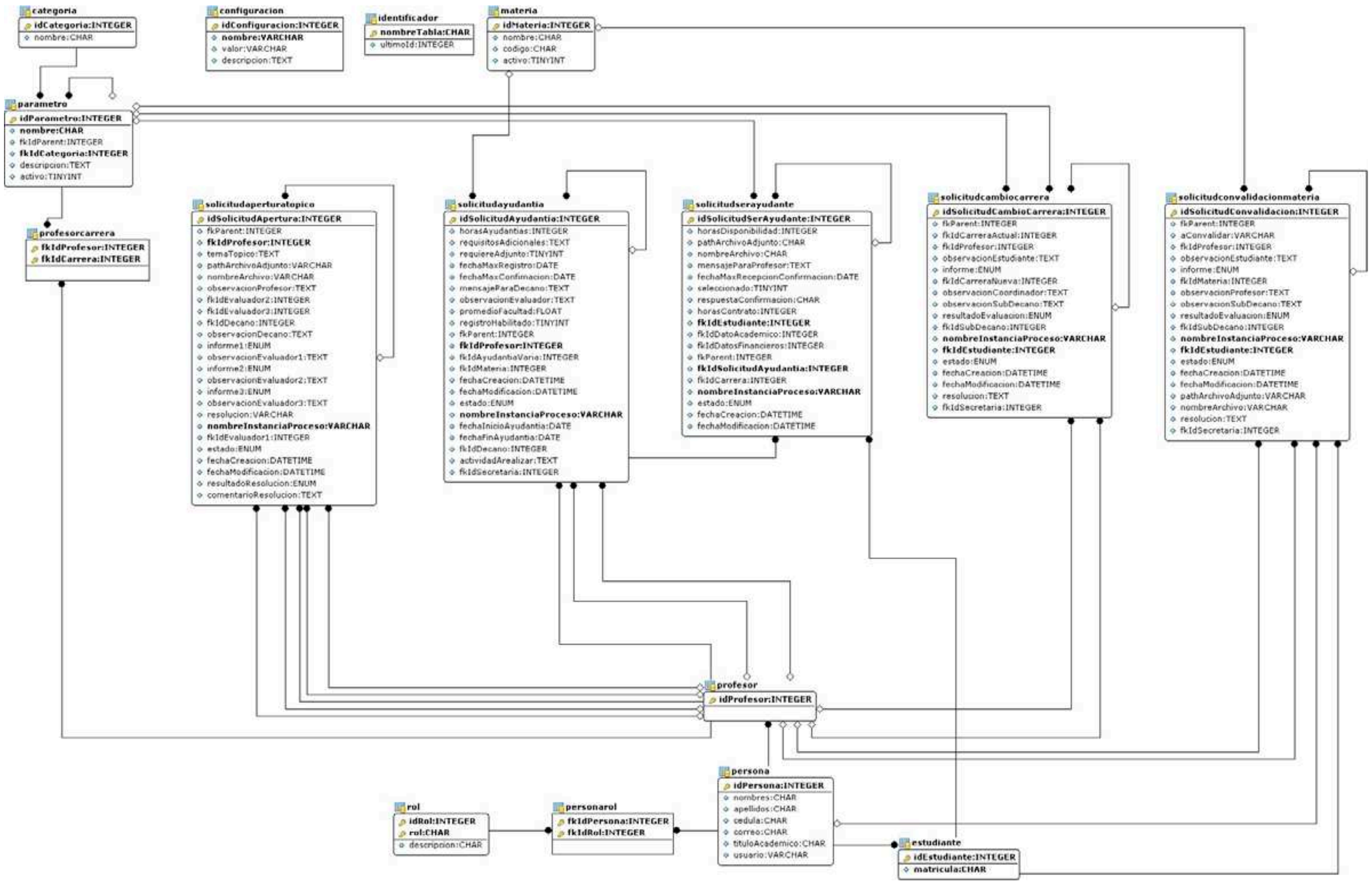


Figura 4.21 Diagrama Entidad-Relación base de datos Solicitudes FIEC.

CONCLUSIONES Y RECOMENDACIONES

Al finalizar el presente trabajo de tesis concluimos que:

- El sistema Solicitudes FIEC contribuye a la mejora de la calidad de servicio otorgada por la facultad, porque agilita notablemente los trámites comparándolo con realizarlo de manera manual. Los participantes ya no deberán acercarse a la secretaría para conocer el estado actual de su solicitud, si no que podrán hacerlo en línea a través de la aplicación Web implementada, sin necesidad de encontrarse físicamente dentro de la universidad.

- El sistema realizado facilita el acceso a la información referente a las solicitudes. Además los trámites son más transparentes debido a que todos pueden ver esta información.
- Utilizar el marco de trabajo JSF fue de gran ayuda durante la elaboración de las interfaces gráficas de los formularios web, por su amplio conjunto de controles y validadores.
- El motor de Workflow utilizado está correctamente estructurado y define claramente las partes y componentes de un proceso. De esta manera, permite que un proceso sea transformado con facilidad de la

vida real al motor de flujos de trabajo. En cuanto a la documentación de su interfaz de programación tiene que mejorar mucho, tanto en describir detallada y totalmente su API como en ofrecerla en un formato de archivo mejor que PDF, una sugerencia puede ser utilizar HTML.

- Destacamos también la importancia de utilizar una herramienta que ofrezca un buen nivel de soporte y ayuda. Durante la implementación del sistema fue necesario revisar en varias ocasiones los foros del Workflow seleccionado e incluso registrar nuevas entradas con inquietudes propias, las cuales fueran respondidas siempre y de manera oportuna.
- La automatización de un proceso sirve de gran ayuda en cuanto a disminución de errores humanos como la pérdida de documentos.
- Toda institución en la cual existan procesos que involucre manejo de documentos y que deban ser evaluados por varias personas en una secuencia determinada, deberían optar por implementarlos utilizando una herramienta de Workflow para conseguir mayor productividad y mejorar la calidad de los servicios que ofrezca.
- Una empresa podría decidir implementar sus procesos desde cero, sin utilizar una herramienta de Workflow, pero perdería las ventajas que brindan estas herramientas y la implementación requeriría de mucho más tiempo para realizarse.

Recomendamos que:

- Se recomienda crear una entidad en el modelo de base de datos para relacionar dos o más procesos porque el motor de Workflow Bonita no incluye una herramienta para hacerlo. Esta relación debe ser realizada por los desarrolladores del sistema.
- Se recomienda poblar los datos de configuración como nombre de autoridades, carreras, etc. antes de poner el sistema en producción.
- Se recomienda tener actualizado los datos de los usuarios en las bases de datos del CSI.
- Otra recomendación es que antes de que un sistema sea utilizado, exista un período de prueba donde se analicen todas las posibles situaciones que puedan ocurrir cuando el sistema esté en producción.

BIBLIOGRAFÍA

1. Universidad de la República (Uruguay) - Sistemas Computacionales, *"Workflow"*,<http://www.ccee.edu.uy/ensenian/catsistc/docs/Workflow.pdf >
2. Workflow Management Coalition., *"The Workflow Reference Model"*, Enero 19, 1995 p.6, <http://www.wfmc.org/standards/docs/tc003v11.pdf>
3. JBoss JBPM, "*Project Home*", <http://www.jboss.com/products/jbpm>
4. Netbeans Proyecto BPEL: Home, "*Project Home*", <http://enterprise.Netbeans .org/soa/>
5. YAWL: Home, "*Project Home*", < http://www.yawlfoundation.org/>
6. Object Web Bonita: Home, "*Project Home*", <http://Bonita.objectweb.org/>
7. Enhydra Shark: Home, "*Project Home*", <http://shark.enhydra.org/>
8. BONITA Workflow System., *"The BONITA Workflow System XPDL Support"*, Mayo 26, 2006 p.4, <http://wiki.Bonita.objectweb.org/xwiki/bin/download/Main/Documentation/BonitaXPDL.pdf>
9. BONITA Workflow System., *"The BONITA Workflow System Architecture"*,

<http://Bonita.objectweb.org/html/Architecture/index.html>

10. JonAs Application Server, "*JonAs Documentation, White paper*" Marzo 24 del 2006. Disponible en la sección de descargas de <http://jonas.objectweb.org/>

11. Wikipedia La Enciclopedia Libre, *"JSF"* Artículo, <http://es.wikipedia.org/wiki/JavaServer_Faces>

12. Wikipedia La Enciclopedia Libre, *"XSLT"* Artículo, <http://es.wikipedia.org/wiki/XSLT>

13. Wikipedia La Enciclopedia Libre, *"Servicios Web"* Artículo, <http://es.wikipedia.org/wiki /Servicio_Web>

ANEXO A

MANUAL DE USUARIO

A.1. Acceso al Sistema.

Para acceder al sistema debe ingresar las credenciales (usuario y contraseña) otorgados por la ESPOL.

Figura A.1. Opción para acceder al sistema

Una vez que se ingresen las credenciales de manera correcta, el sistema dará acceso a la página de inicio y el nombre del usuario aparecerá en la parte superior derecha de la ventana principal.

Figura A.2 Página inicial del sistema

Si el usuario desea desconectarse del sistema debe seleccionar la opción **Cerrar sesión** que está en la parte superior derecha de la pagina.

A.2. Opciones del sistema.

En la figura 1.2 se puede observar el menú de opciones disponibles en el sistema. Este menú se divide en 8 partes, las cuales son.

- **Inicio:** página de inicial del sistema.

- **Solicitudes a iniciar:** muestra la lista de solicitudes que el usuario logoneado puede iniciar, de acuerdo a su rol (estudiante o profesor).

Las solicitudes posibles a iniciar para un estudiante son:

- Solicitud convalidación de materia.
- Solicitud cambio de carrera o especialización.
- Solicitud para ser ayudante.

Las solicitudes que puede iniciar un profesor son:

- Solicitud de apertura postgrado.
- Solicitud de ayudantes.

- **Mis Actividades pendientes:** muestra la lista de actividades pendientes del usuario en los diferentes procesos activos.
- **Solicitudes Activas:** en esta sección el usuario podrá revisar el estado actual de las solicitudes activas en las que participa, ya sea como solicitante o como realizador de alguna de las actividades.
- **Historial:** aquí se localizan las solicitudes terminadas en las que el usuario ha participado.
- **Búsqueda:** en esta sección el sistema permite realizar búsquedas de solicitudes por diferentes criterios como: nombre de solicitante, tipo de solicitud, fecha de creación y finalización, etc.

- **Administración:** sección habilitada solo para usuarios que pertenezcan al rol administrador. Permite realizar cambios en configuraciones de ciertos datos del sistema.
- **Ayuda:** aquí el usuario podrá encontrar las guías para realizar los trámites que se manejan en el sistema, así como también un manual de usuario.

A continuación se explica cómo utilizar cada parte del menú del sistema:

Inicio

En la página de inicio el usuario puede revisar la descripción de cada proceso que se puede tramitar en el sistema. Además podrá revisar la descripción detallada de los pasos que definen el proceso, así como el diagrama de flujo de cada uno. La página de inicio se muestra en la figura A.2.

Solicitudes a iniciar

Para iniciar una solicitud diríjase al *menú del sistema, Solicitudes a iniciar* y seleccione una solicitud de la lista mostrada. Inmediatamente se mostrará en la sección de contenido el formulario correspondiente a la solicitud seleccionada. En la figura A.3 se puede ver la acción explicada, seleccionando la solicitud de convalidación de materia.

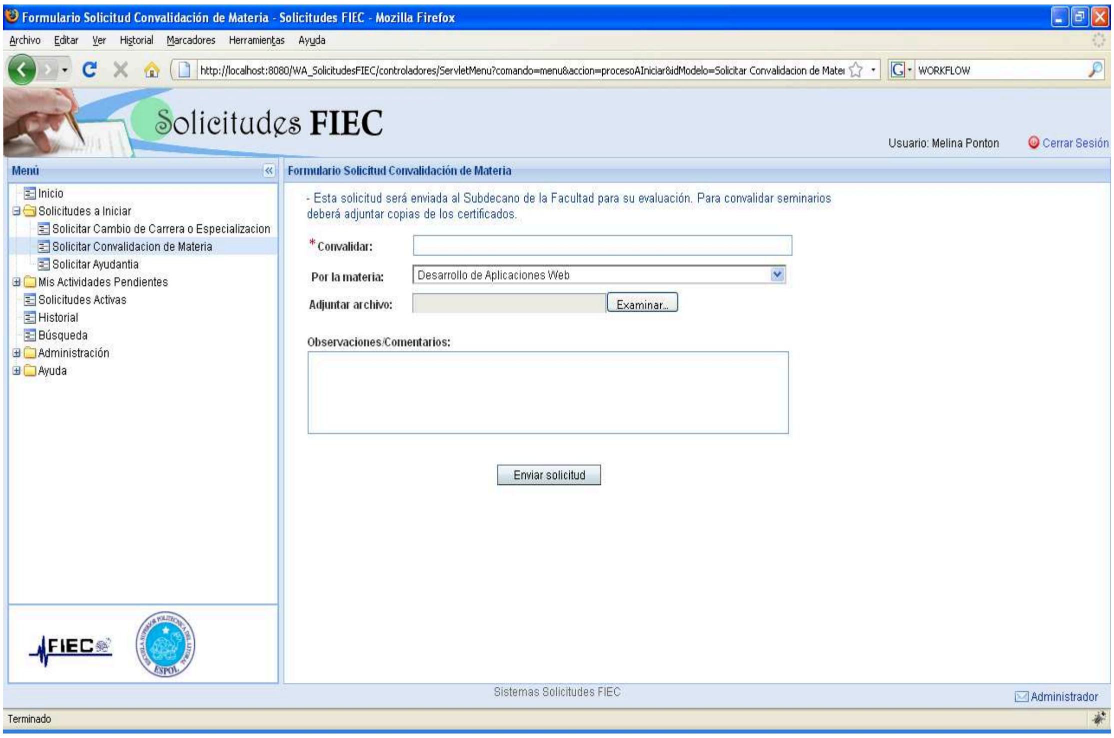


Figura A.3 Formulario para solicitar convalidación de materias.

Mis Actividades Pendientes.

Para revisar sus actividades pendientes diríjase al *menú del sistema, Mis Actividades Pendientes.*

Si usted no posee actualmente ninguna actividad pendiente verá un mensaje como el que muestra la figura A.4.

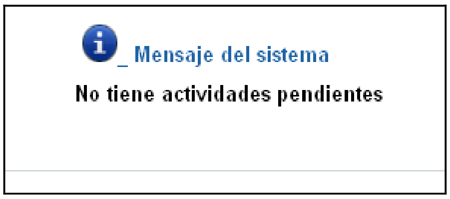


Figura A.4 Mensaje de notificación: No existen actividades pendientes.

Si usted posee actividades pendientes, seleccione un tipo de solicitud del arbol mostrado bajo el item *Mis actividades Pendientes.* Entonces el sistema mostrará una lista de las actividades pendientes del usuario logoneado, por el tipo de solicitud que seleccionó. En esta lista se encuentra los datos mas relevantes de la solicitud, como por ejmplo: el título o identificador de la solicitud, el nombre del solicitante, el nombre de la actividad pendiente y la fecha en que fue asigna. La figura A.5 muestra el resultado de realizar la acción mencionada anteriormente.

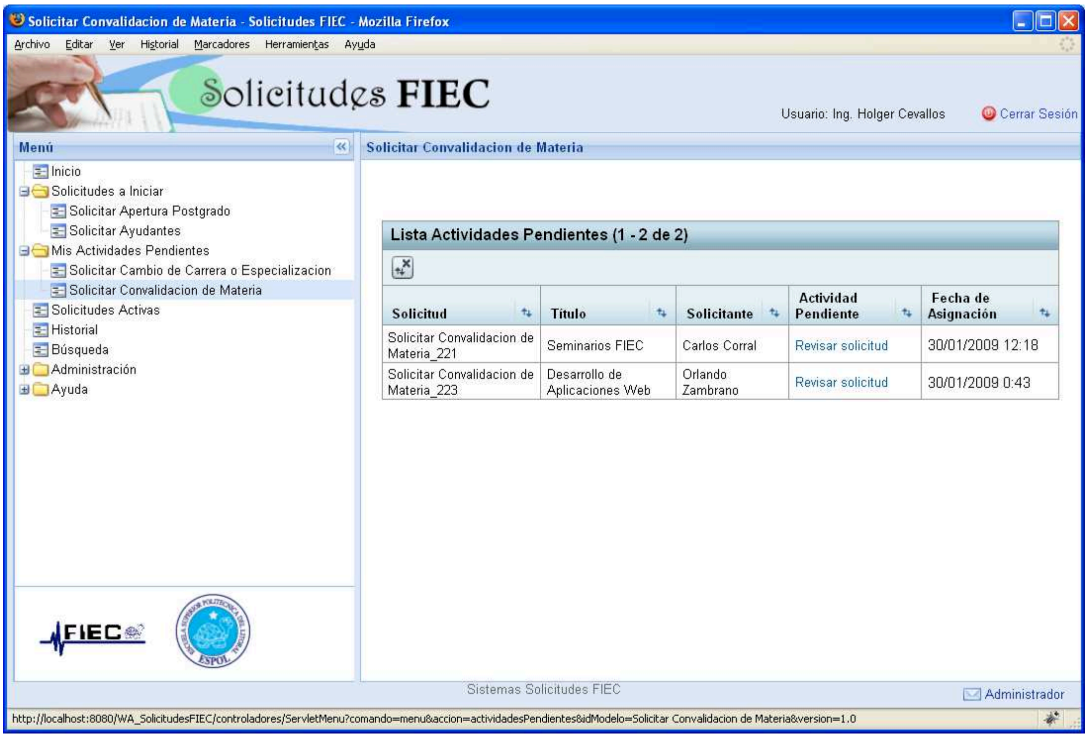


Figura A.5 Lista de actividades pendientes.

Para realizar la actividad dé clic en el nombre de la misma. De inmediato, el sistema mostrará el formulario correspondiente a la actividad para que sea realizada.

Solicitudes Activas

Para revisar sus solicitudes activas diríjase al *menú del sistema, Solicitudes Activas.* Si no posee actualmente Solicitudes Activas se mostrará un mensaje similar al visualizado en la figura A.6.

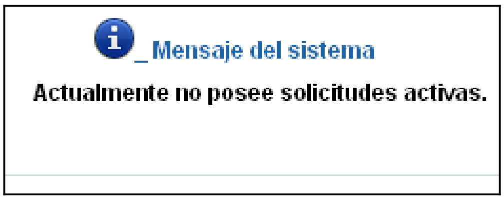
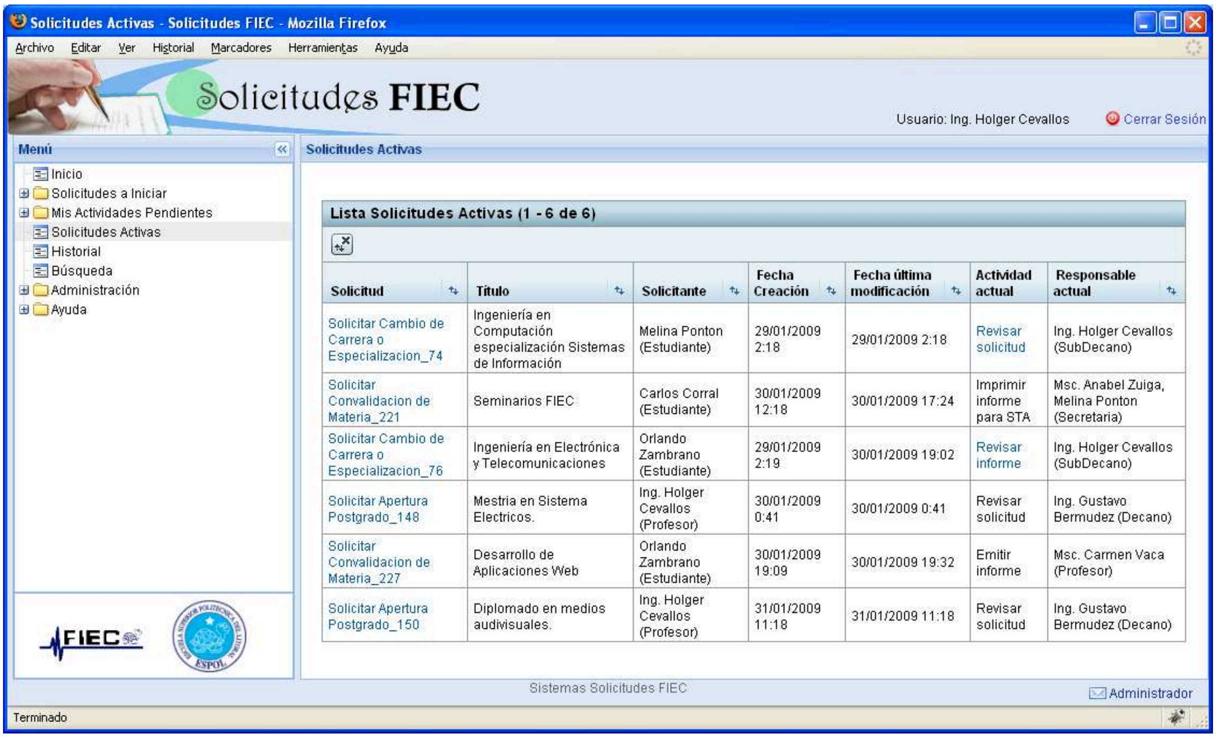


Figura A.6 Mensaje de notificación: no existen solicitudes activas.

Si usted posee Solicitudes Activas, el sistema las mostrará enlistadas. Esta lista le permitirá dar un vistazo rápido al estado actual de su solicitud, debido a que indica quien es el responsable actual (quien tiene la solicitud actualmente), el nombre de la actividad actual y otros datos relevantes de la solicitud. La figura A.7 muestra una lista de las solicitudes activas.

Si el usuario logoneado es el responsable actual de la solicitud, entonces podrá acceder al formulario de la actividad dando clic en el link del nombre de la misma, al igual que en la lista de actividades pendientes (sección 2.3).



Solicitud	Título	Solicitante	Fecha Creación	Fecha última modificación	Actividad actual	Responsable actual
Solicitar Cambio de Carrera o Especializacion_74	Ingeniería en Computación especialización Sistemas de Información	Melina Ponton (Estudiante)	29/01/2009 2:18	29/01/2009 2:18	Revisar solicitud	Ing. Holger Cevallos (SubDecano)
Solicitar Convalidacion de Materia_221	Seminarios FIEC	Carlos Corral (Estudiante)	30/01/2009 12:18	30/01/2009 17:24	Imprimir informe para STA	Msc. Anabel Zuiga, Melina Ponton (Secretaria)
Solicitar Cambio de Carrera o Especializacion_76	Ingeniería en Electrónica y Telecomunicaciones	Orlando Zambrano (Estudiante)	29/01/2009 2:19	30/01/2009 19:02	Revisar informe	Ing. Holger Cevallos (SubDecano)
Solicitar Apertura Postgrado_148	Mestria en Sistema Electricos.	Ing. Holger Cevallos (Profesor)	30/01/2009 0:41	30/01/2009 0:41	Revisar solicitud	Ing. Gustavo Bermudez (Decano)
Solicitar Convalidacion de Materia_227	Desarrollo de Aplicaciones Web	Orlando Zambrano (Estudiante)	30/01/2009 19:09	30/01/2009 19:32	Emitir informe	Msc. Carmen Vaca (Profesor)
Solicitar Apertura Postgrado_150	Diplomado en medios audivisuales.	Ing. Holger Cevallos (Profesor)	31/01/2009 11:18	31/01/2009 11:18	Revisar solicitud	Ing. Gustavo Bermudez (Decano)

Figura A.7 Lista de solicitudes activas.

Si desea visualizar las actividades por las que ha pasado su solicitud, dé clic en el nombre de la misma. El resultado de esta acción se muestra en la figura A.8. Obsérvese en esta lista que el último item corresponde al la actividad actual de la solicitud.

Figura A.8 Lista de actividades realizadas.

Adicionalmente, es posible revisar en detalle lo sucedido en cada paso (actividad) por la que transitó la solicitud. Para ello dé clic en el nombre de la actividad e inmediatamente el sistema mostrará una ventana popup con la información detallada de esa actividad (figura A.9).

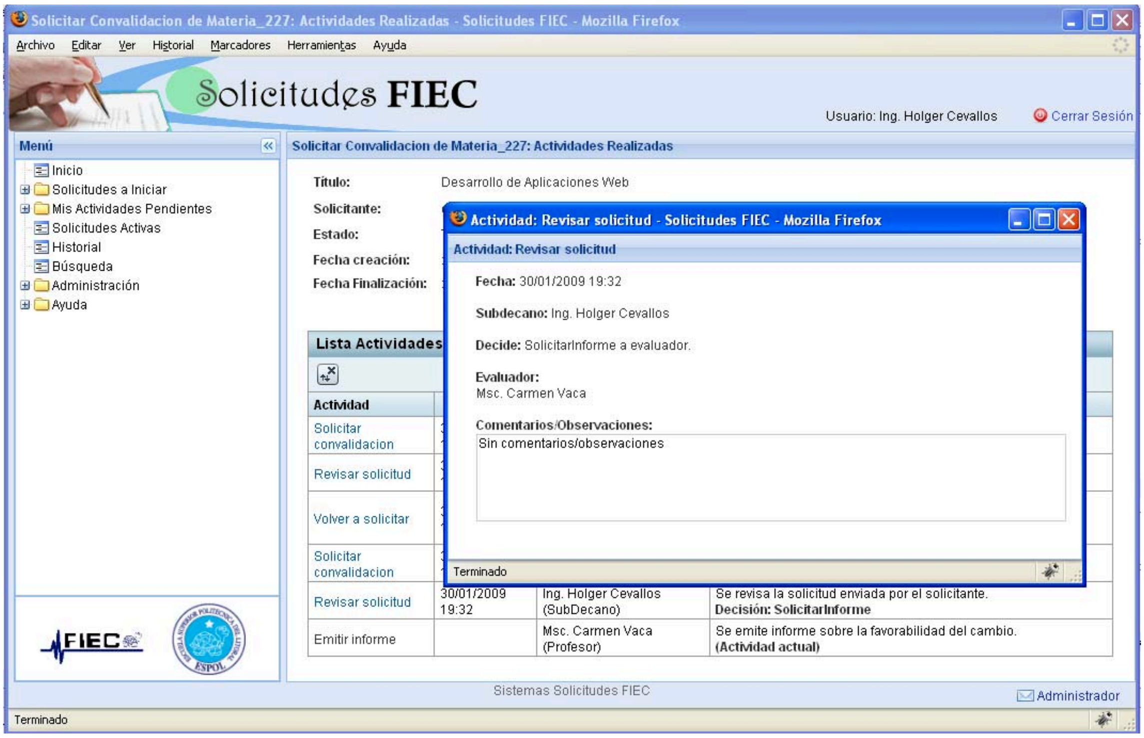


Figura A.9 Detalle de actividad realizada.

Historial

Para revisar sus solicitudes terminadas diríjase al *menú del sistema, Historial.* Si no posee actualmente solicitudes terminadas se mostrará un mensaje similar al visualizado en la figura A.10.

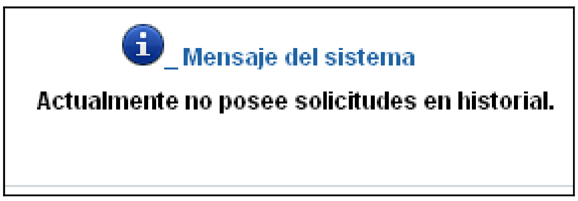


Figura A.10 Mensaje de notificación: no existen actividades en historial.

Si usted posee solicitudes terminadas, el sistema mostrará una lista de solicitudes. Como podrá ver en la figura A.11, esta lista es muy similar a la descrita en la sección 2.4, con la diferencia de que no muestra datos de actividades actuales debido a que estas solicitudes ya han finalizado. Por otro lado muestra la fecha de finalizacion de la solicitud y el estado de la misma.

Para visualizar las actividades y detalle de las mismas debe seguir los mismos pasos descritos en la sección **Solicitudes Activas**.

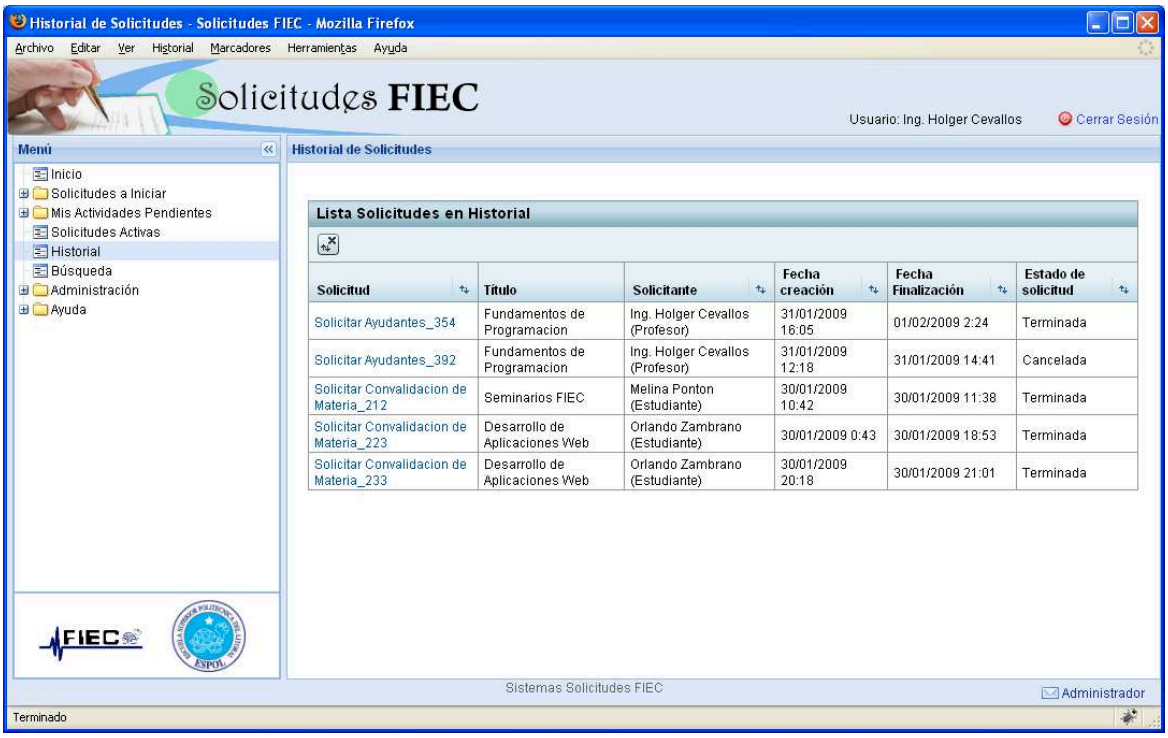


Figura A.11 Lista de Historial.

Búsqueda de solicitudes

En esta sección el usuario podrá buscar solicitudes por diferentes criterios como estado, solicitante, participantes. La figura A.12 muestra una búsqueda por tipo: solicitudes apertura de postgrado.

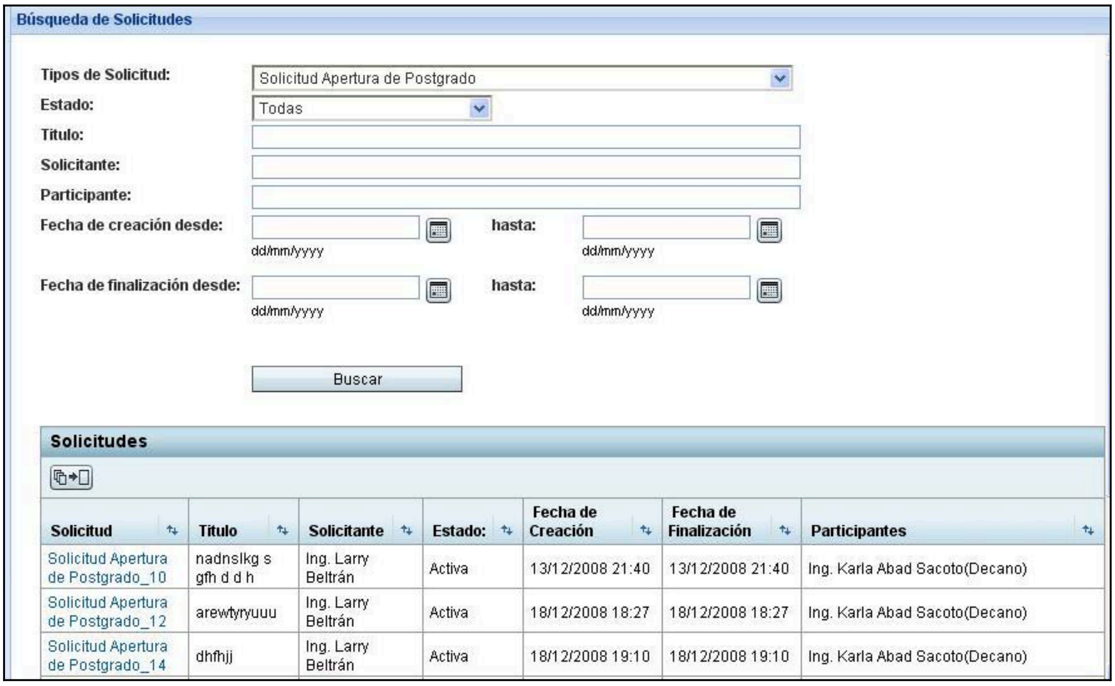


Figura A.12 Búsqueda de solicitudes.

Administración

Esta sección del sistema estará disponible solo para usuarios con rol administrador. Aquí se maneja la creación, modificación y eliminación de los siguientes elementos: usuarios, carreras, materias, ayudantías varias y parámetros de configuración.

Usuarios

Diríjase al *menú del sistema, Administración, Usuarios,* inmediatamente se visualizará una lista con los usuarios registrados en el sistema (figura A.13).

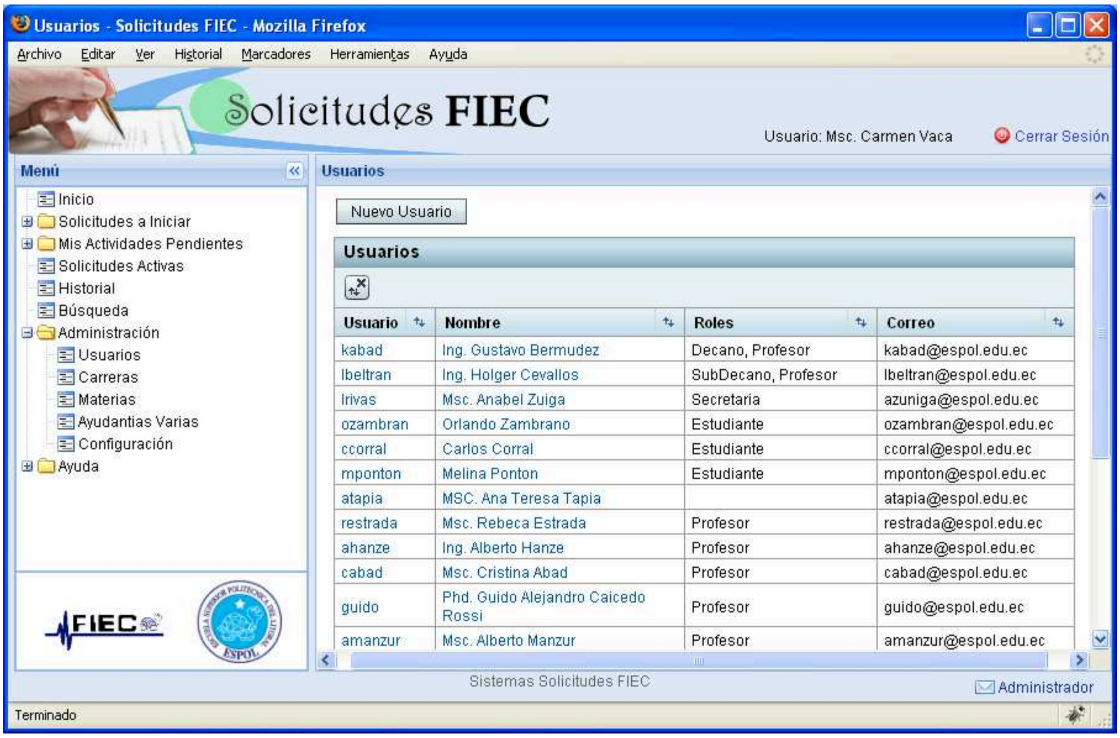


Figura A.13 Lista de usuarios del sistema.

Para crear un nuevo usuario dé clic en la opción Agregar usuario y el sistema mostrará un formulario como el de la figura A.14. En este formulario se debe ingresar los datos personales del usuario y asignar los roles al que pertenece. Además se debe seleccionar la casilla “activar” para que el usuario pueda acceder al sistema.

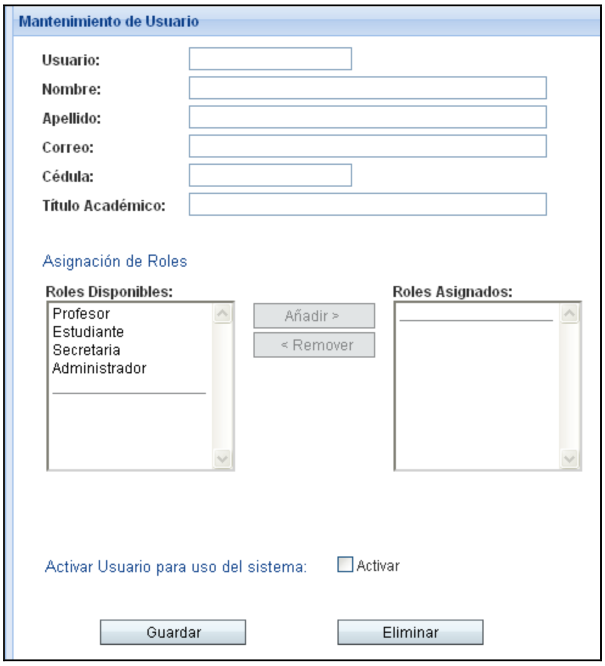


Figura A.14 Formulario creación de usuario.

Para modificar o eliminar un usuario, se debe dar clic sobre el nombre del mismo en la lista mostrada en la figura A.13. Al momento se visualizará el formulario mostrado en la figura A.15, en el cual es posible realizar modificaciones en los datos personales del usuario, en la asignación de roles y también puede activar o desactivar al usuario.

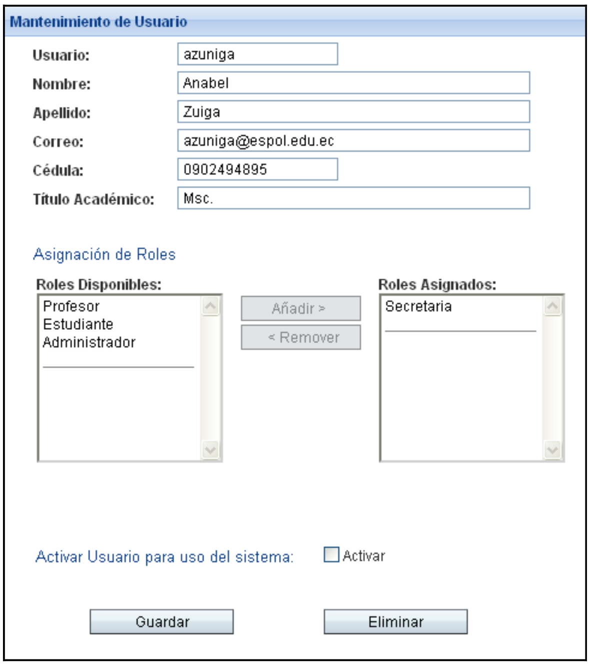


Figura A.15 Formulario modificación de usuario

Carreras

Diríjase al *menú del sistema, Administración, Carreras,* inmediatamente se visualizará una lista con los carreras registradas en el sistema (figura A.16).

Figura A.16 Lista de carreras.

Para agregar una nueva carrera dé clic en la opción Nueva carrera y el sistema mostrará un formulario como el de la figura A.17. En este formulario se debe especificar el nombre de la carrera y seleccionar el profesor designado como coordinador. Además se debe seleccionar la casilla “activar” para que la carrera esté disponible en el sistema.

Si la nueva carrera tiene especializaciones, se debe crear primero estas ingresando el nombre y coordinador de la especialización y luego seleccionar Agregar Especialización. Realizada esta acción, una nueva fila de la tabla “Especializaciones” aparecerá con los datos ingresados recientemente. Cuando haya ingresado todas las especializaciones para la nueva carrera, ingrese el nombre y coordinador de la carrera y seleccione la opción Guardar.

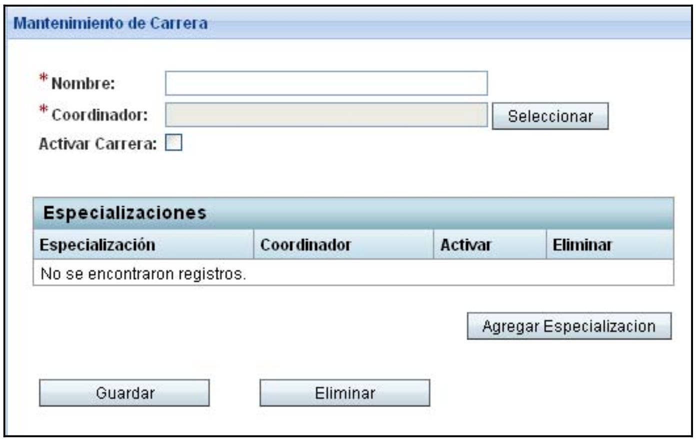


Figura A.17 Formulario creación de carrera.

Para modificar o eliminar una carrera, se debe dar clic sobre el nombre de la misma en la lista mostrada en la figura A.16. Al momento se visualizará el formulario mostrado en la figura A.18, en el cual es posible realizar modificaciones en el nombre de la carrera o cambiar el coordinador asignado y también puede activar o desactivar la carrera. En este formulario también es posible modificar o eliminar las especializaciones de una carrera.

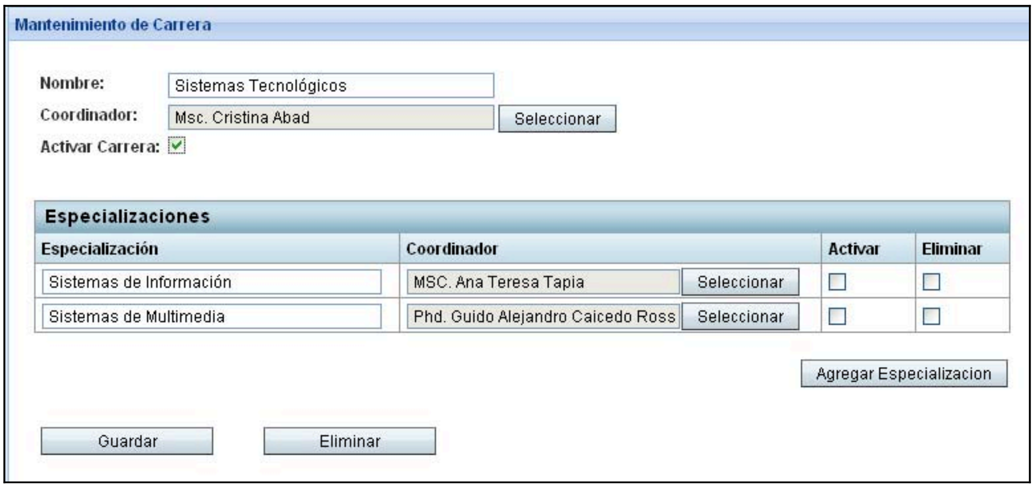


Figura A.18 Formulario modificación de carrera.

Materias

Diríjase al *menú del sistema, Administración, Materias,* inmediatamente se visualizará una lista con los carreras registradas en el sistema (figura A.19).

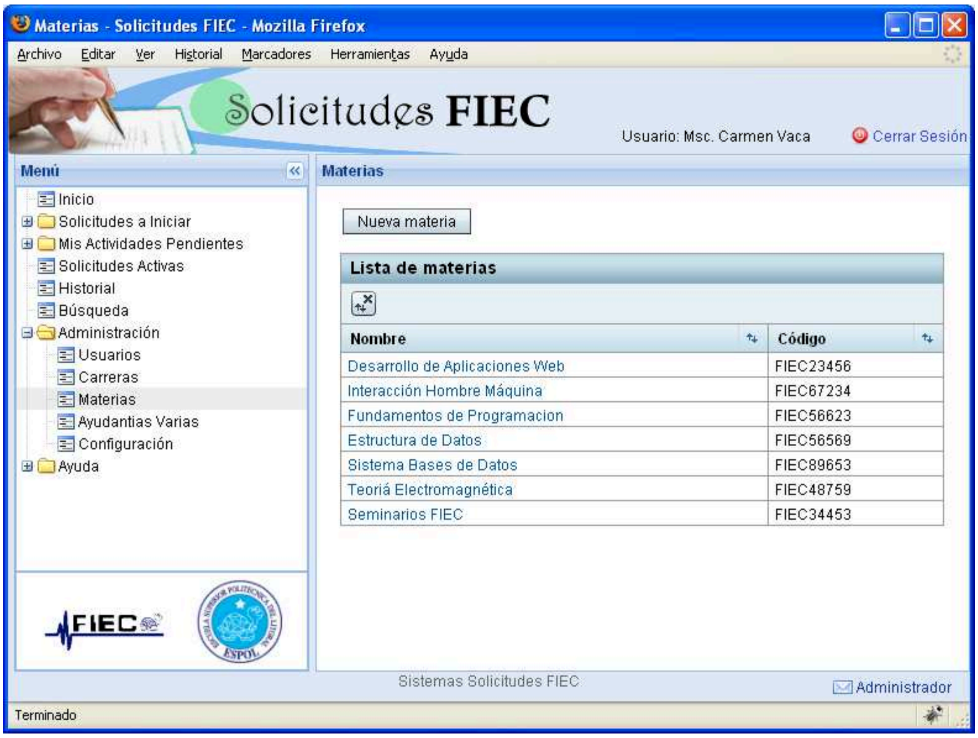


Figura A.19 Lista de materias.

Para agregar una nueva materia dé clic en la opción Nueva materia y el sistema mostrará un formulario como el de la figura A.19. En este formulario es necesario especificar el nombre y código de la materia. Además se debe seleccionar la casilla “activar” para que la materia esté disponible en el sistema.

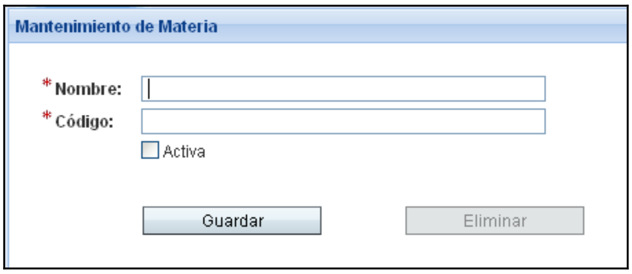


Figura A.20 Formulario creación de materia.

Para modificar o eliminar una materia, se debe dar clic sobre el nombre de la misma en la lista mostrada en la figura A.19. Al momento se visualizará el formulario mostrado en la figura A.21, en el cual es posible realizar modificaciones en el nombre o código de la materia y activarla o desactivarla.

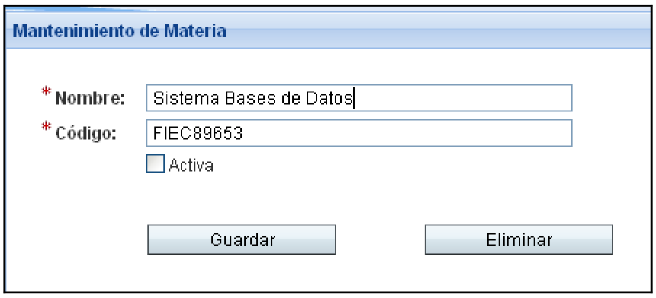


Figura A.21 Formulario modificación de materia.

Ayudantías varias

Diríjase al *menú del sistema, Administración, Ayudantías varias,* al momento se visualizará una lista con los carreras registradas en el sistema (figura A.22).

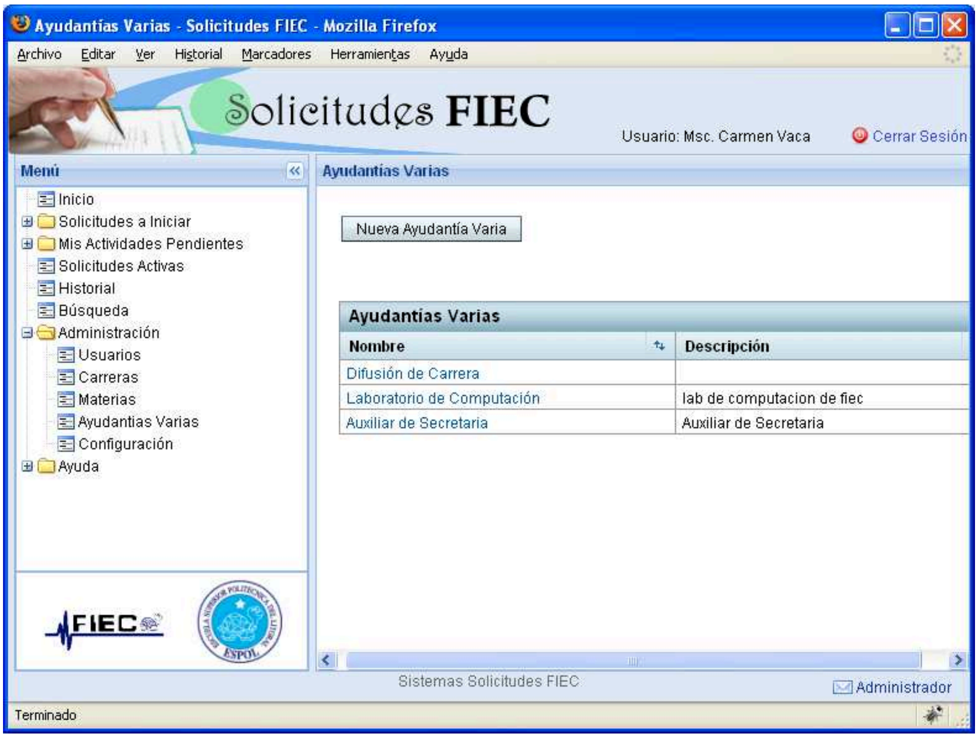


Figura A.22 Lista ayudantías varias.

Para agregar una nueva ayudantía varia dé clic en la opción Nueva Ayudantía Varia y el sistema mostrará un formulario como el de la figura A.23. En este formulario es necesario especificar el nombre y la descripción de la materia. Además se debe seleccionar la casilla “activar” para que la ayudantía varia esté disponible en el sistema.

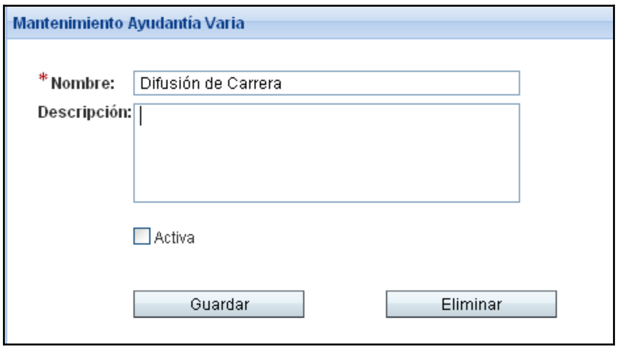


Figura A.23 Formulario creación de ayudantía varia.

Para modificar o eliminar una ayudantía varia, se debe dar clic sobre el nombre de la misma en la lista mostrada en la figura A.22. Al momento se visualizará el formulario mostrado en la figura A.24, en el cual es posible realizar modificaciones en el nombre o la en descripción de la materia. Además también es posible activarla o desactivarla.

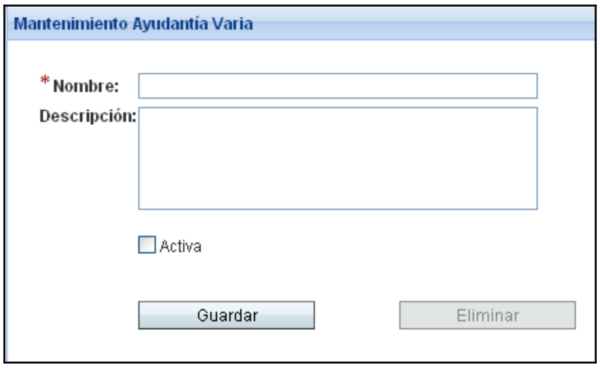


Figura A.24 Formulario modificación de ayudantía varia.

Configuraciones

Los parámetros de configuración son necesarios durante diferentes instancias dentro del sistema, por este motivo no es posible crear nuevos parámetros, sino solamente cambiar sus valores. Un ejemplo de una configuración es el reglamento de becas de la ESPOL, el cual es mostrado al usuario durante la realización de una solicitud de ayudantes. Diríjase al *menú del sistema, Administración, Configuración,* al momento se visualizará una lista con las ayudantías varias registradas en el sistema (figura A.25).

Figura A.25 Lista de configuraciones.

Para modificar un parámetro de configuración, se debe dar clic sobre el nombre de la misma en la lista mostrada en la figura A.25. Al momento se visualizará el formulario mostrado en la figura A.26, en el cual es posible modificar el valor del parámetro de configuración.

Si se desea modificar el parámetro Decano o Subdecano se debe dar clic en la opción seleccionar y buscar el profesor designado.

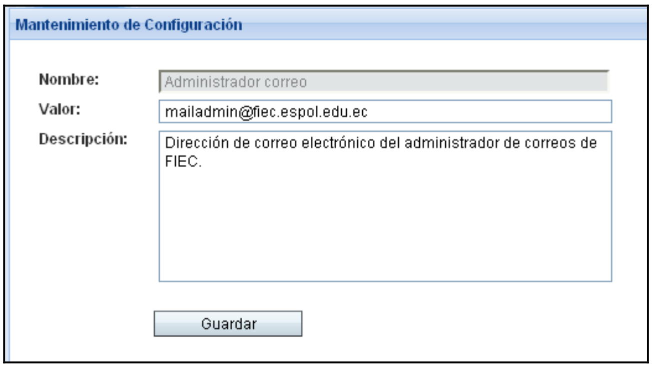


Figura A.26 Formulario modificación parámetros de configuración.

Ayuda.

En la sección de ayuda se encuentran las guías para elaborar cada solicitud que el sistema puede gestionar. Estas guías describen las rutas que puede tomar el tramite o y describe cada actividad o paso de la solicitud. La figura A.27 muestra la guía para elaborar una convalidación de materia.

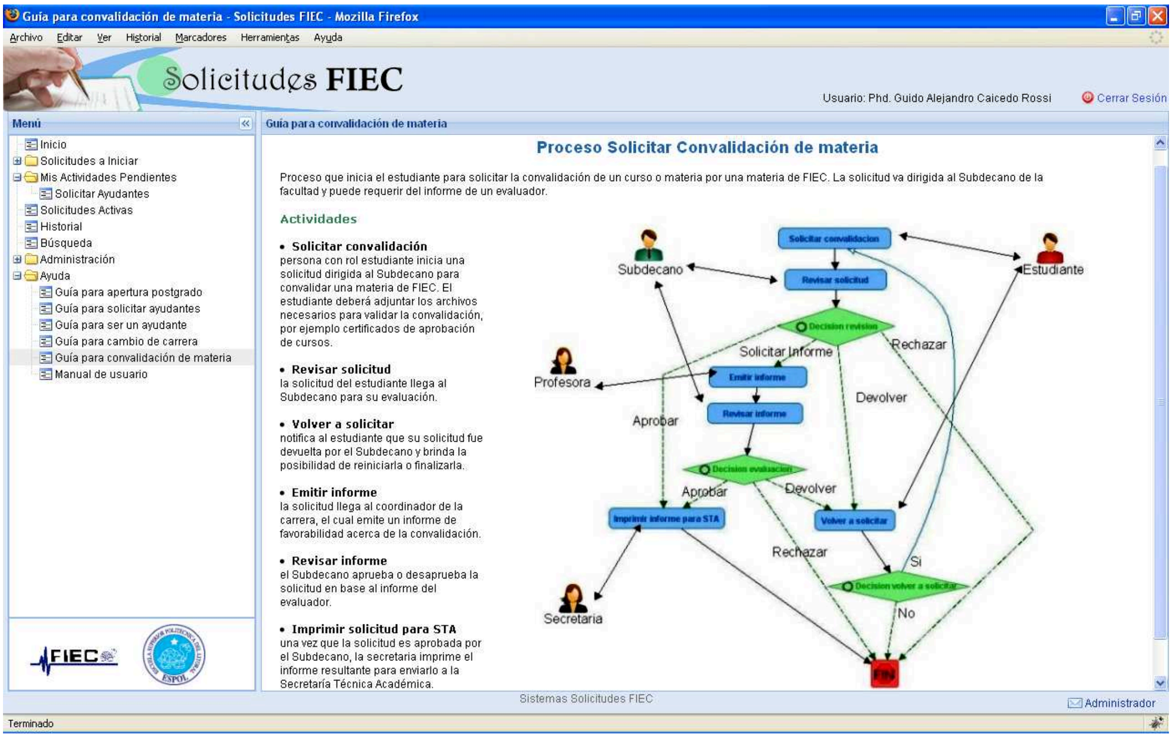


Figura A.27 Guía para elaborar una convalidación de materia.

Además en la sección de ayuda se podrá acceder al presente manual de uso del sistema en versión html.

A.3. Solicitudes

En esta sección se describirá en detalle los pasos a seguir durante la realización de un trámite (o proceso) para los diferentes tipos de solicitudes que gestiona el sistema.

Solicitud convalidación de materia

Iniciar una solicitud

Para iniciar una solicitud de convalidación de materia se debe dirigir al *menú del sistema, Solicitudes a iniciar* y seleccionar un tipo de solicitud de la lista mostrada. Al momento se visualizará el formulario de ingreso de la solicitud.

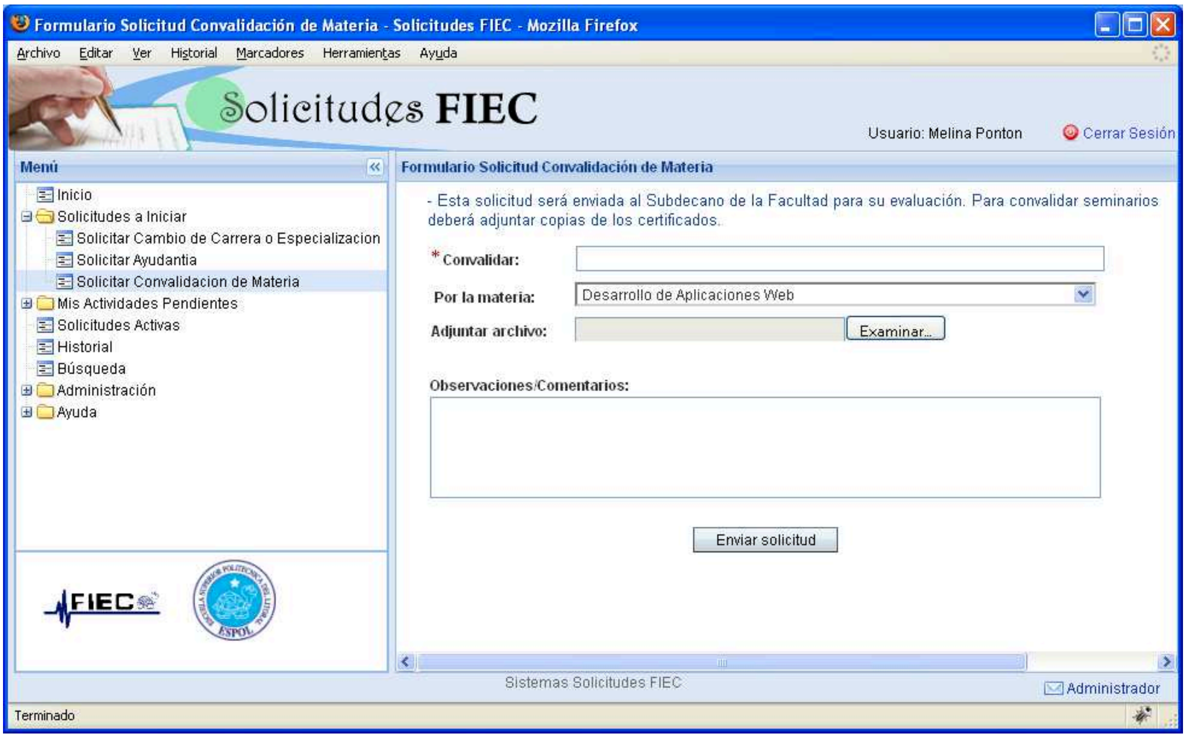


Figura A.28 Formulario de ingreso: Solicitud convalidación de materia.

En el formulario mostrado por la figura A.28 es necesario ingresar el nombre de la materia, curso o seminario a convalidar, y luego elegir la materia de FIEC equivalente. Para convalidar la materia seminarios

FIEC se debe adjuntar copias de los certificados de asistencia y aprobación de los cursos. También es posible escribir un comentario u observación acerca de su solicitud.

Una vez que los datos requeridos son llenados, se puede enviar la solicitud al Subdecano de la facultad seleccionando **Enviar Solicitud**. El sistema verificará que los datos ingresados sean correctos y en caso de no serlos mostrará mensajes alusivos al error.

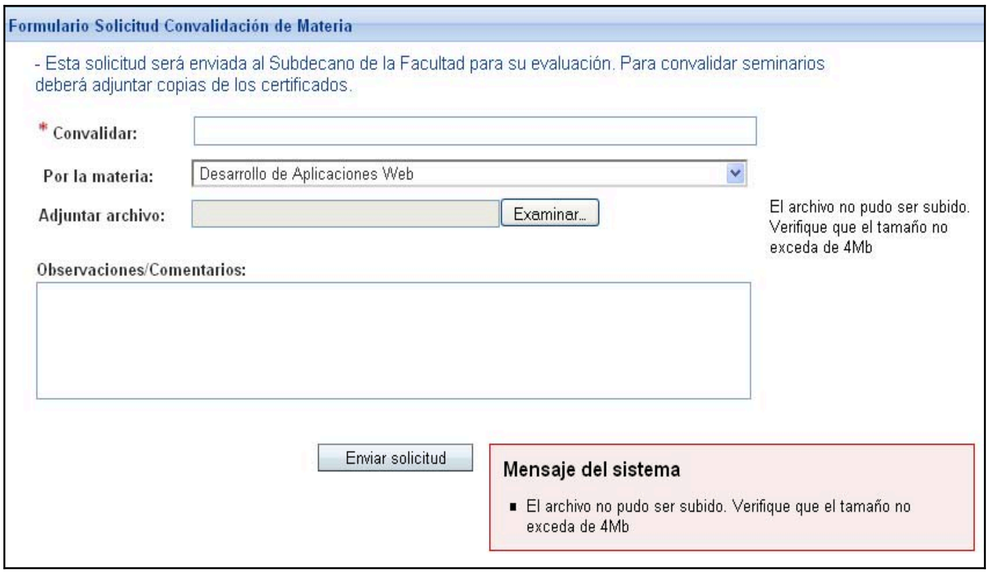


Figura A.29 Mensaje de error: Archivo adjunto mayor al máximo permitido.

En caso de que el estudiante tenga activa una solicitud de convalidación de materia para la misma materia seleccionada, el sistema no permitirá el envío de la solicitud y mostrará el siguiente mensaje de error.

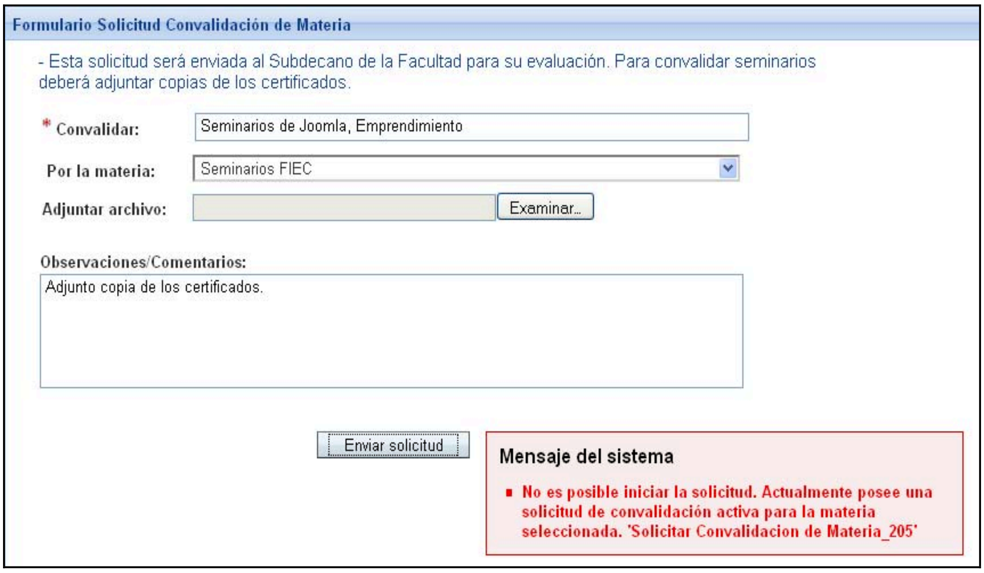


Figura A.30 Mensaje de error: solicitud de convalidación de materia existente.

Finalmente, una vez que el sistema validó el ingreso de la información obligatoria, y si no existen inconvenientes se inicia el trámite mostrando un mensaje descriptivo (figura A.31) y enviando una notificación vía correo electrónico al usuario configurado con el rol de Subdecano en el sistema.

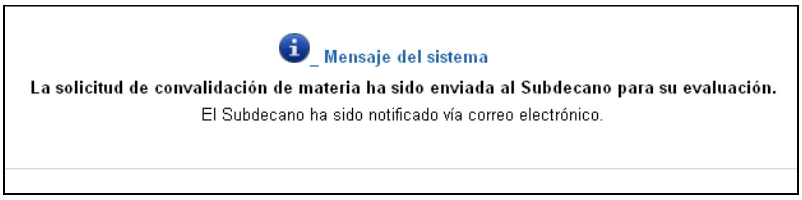


Figura A.31 Mensaje de confirmación: envío de solicitud a Subdecano.

Los solicitantes podrán seguir accediendo a su solicitud por medio del menú Solicitudes activas.

Revisar solicitud

Una vez que el estudiante ha iniciado una solicitud, esta pasa a manos del Subdecano de la facultad para su evaluación. En esta actividad el Subdecano podrá decidir entre aceptar, devolver o rechazar la solicitud. Y también puede requerir el informe de un profesor evaluador.

Para acceder a la actividad el usuario puede dirigirse al *menú del sistema, Mis Actividades Pendientes* y seleccionar un tipo de solicitud de la lista mostrada; o navegar a través de las solicitudes activas.

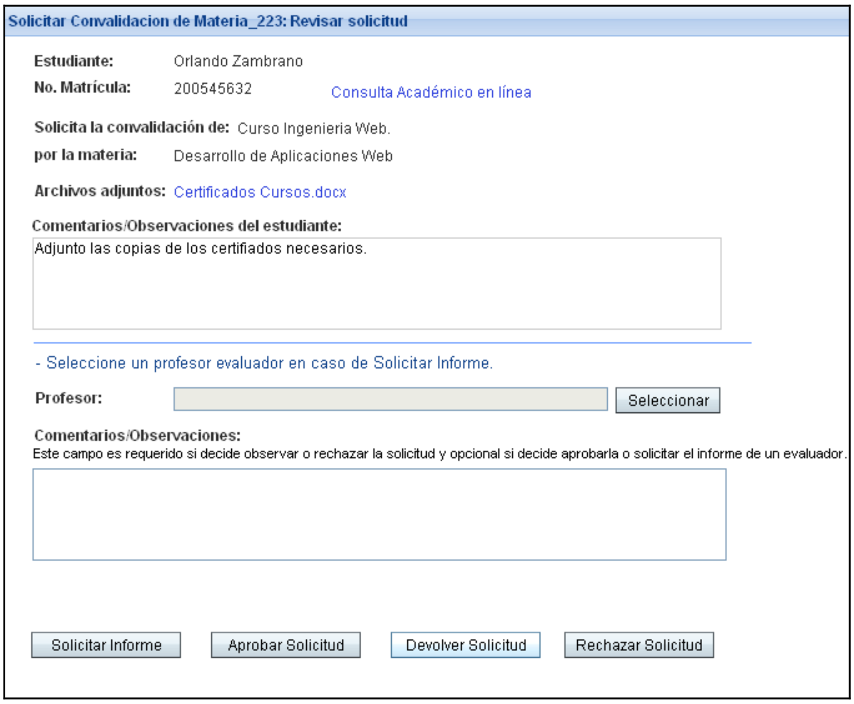


Figura A.32 Formulario revisión de solicitud.

Opción Solicitar Informe: primero se debe seleccionar un profesor evaluador, para ello se utiliza el botón **seleccionar**, que está ubicado junto al campo de texto “Profesor”.

Inmediatamente se visualizará una ventana que permite buscar un usuario con rol profesor (figura A.33). Una vez encontrado el usuario deseado, se debe dar clic sobre el nombre del mismo para seleccionarlo. Esta última acción nos lleva nuevamente al formulario *Revisión de solicitud* el cual muestra ahora el nombre del evaluador escogido (Figura A.34).

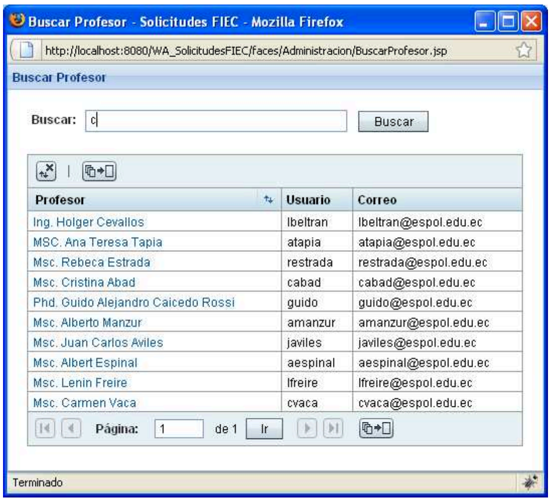


Figura A.33 Ventana Buscar profesor

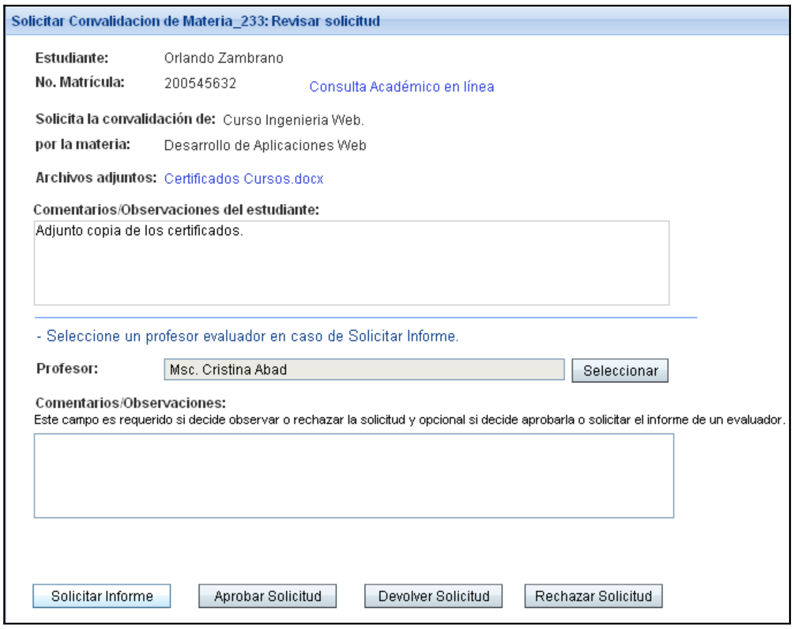


Figura A.34 Formulario revisión de solicitud: selección de evaluador.

Una vez seleccionado el evaluador, es posible seleccionar Solicitar Informe, y si no existen inconvenientes el sistema mostrará un mensaje similar al mostrado en la figura A.35. Además se enviará una notificación vía correo electrónico al evaluador seleccionado.

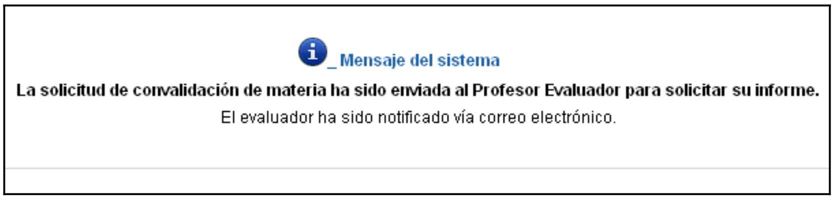


Figura A.35 Mensaje de confirmación: envío de solicitud ha evaluador.

Opciones Aprobar, Devolver y Rechazar Solicitud**:** si el Subdecano decide devolver o rechazar la solicitud, deberá ingresar la razón de su decisión en el campo de texto “Comentarios/Observaciones”. En el caso de decidir aprobarla, este campo es opcional. Si la solicitud es rechazada el proceso termina y el estudiante deberá subir una nueva solicitud si desea realizar la convalidación.

Luego de tomada la decisión aparecerá un mensaje de notificación descriptivo y se enviará una un correo electrónico al solicitante.

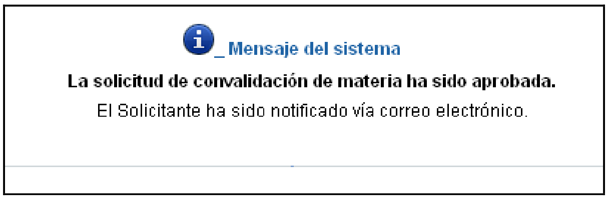


Figura A.36 Mensaje de confirmación: solicitud aprobada.

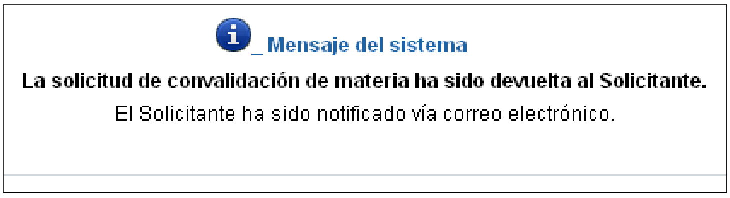


Figura A.37 Mensaje de confirmación: solicitud devuelta.

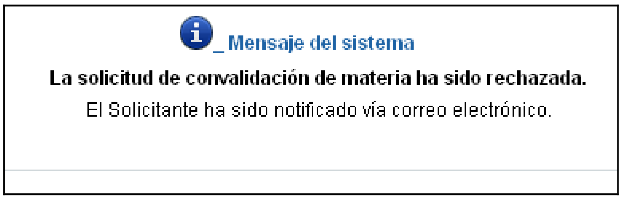


Figura A.38 Mensaje de confirmación: solicitud rechazada.

Reiniciar una solicitud

Si durante la revisión de la solicitud el Subdecano decidió devolverla, el estudiante que solicito el cambio podrá decidir si desea **reiniciar** su solicitud o finalizarla a través de la actividad *Volver a Solicitar*, cuyo formulario se muestra en la figura A.39.

Figura A.39 Formulario volver a solicitar.

Si el solicitante decide **finalizar** su solicitud, el proceso termina y el estudiante deberá subir una nueva solicitud si desea realizar la convalidación.

En caso de decidir reiniciar la solicitud, el solicitante deberá dirigirse a la sección *Mis Actividades Pendientes* para acceder nuevamente a ella, realizar los cambios necesarios y volver a enviarla. Este formulario es idéntico al descrito en la sección 3.1, con la diferencia de que los campos “Convalidar” y “Por la materia” ya no son editables pues son identificadores de la solicitud en curso.

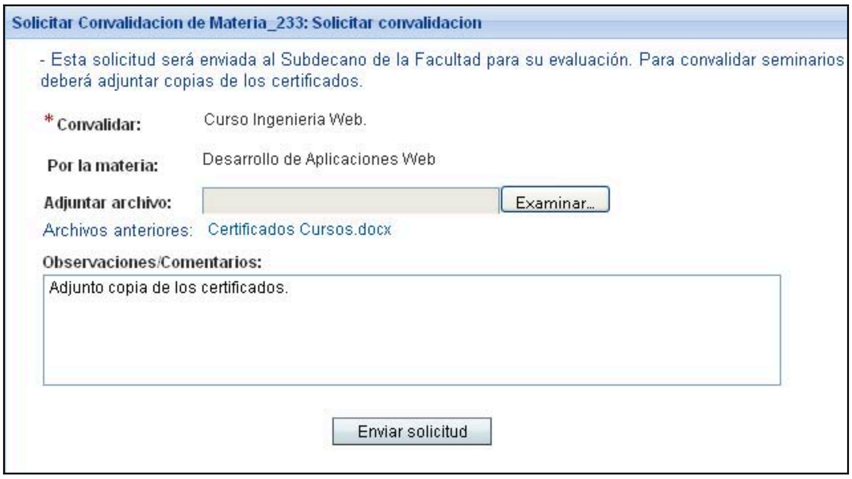
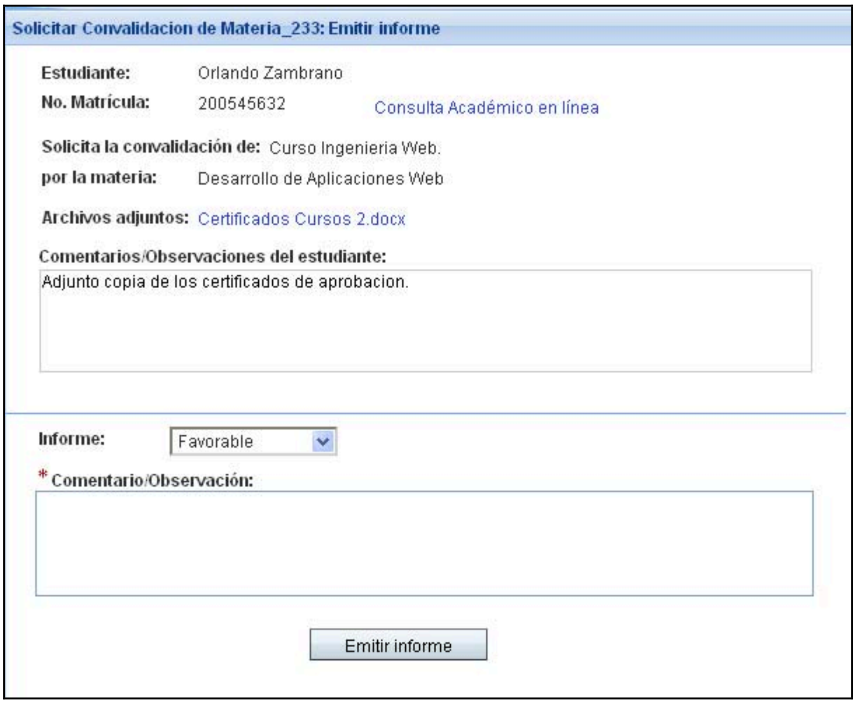


Figura A.40 Formulario de solicitud convalidación de materia, después de que la solicitud ha sido reiniciada.

Una vez realizado los cambios necesarios puede enviar nuevamente su solicitud.

Emitir Informe

Si durante la revisión de la solicitud el Subdecano decidió pedir el informe de un evaluador (profesor de la facultad), este podrá realizar la actividad dirigiéndose a la sección *Mis Actividades Pendientes* o través de la lista de *Solicitudes Activas*.



Fifura A.41 Formulario de emisión de informe.

El informe puede ser "Favorable" o "No Favorable" a discreción del evaluador, en ambos casos deberá ingresar un comentario acerca de su evaluación. Una vez emitido el informe el Subdecano es notificado vía correo electrónico.

Revisar Informe

Después de que el evaluador ha emitido su informe, la solicitud llega al Subdecano, el cual debe revisarla nuevamente y evaluarla tomando en consideración el informe del evaluador.

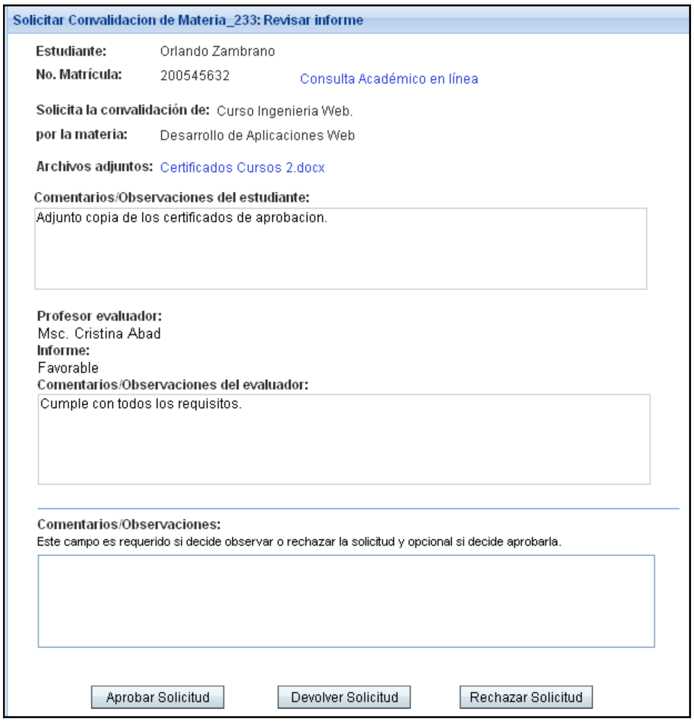


Figura A.42 Formulario de revisión de informe

Al igual que en el formulario Revisar solicitud, explicado anteriormente, el sistema obliga al usuario a ingresar la razón de su decisión en caso de que la solicitud sea devuelta o rechazada. Si la solicitud es rechazada el proceso termina y el estudiante deberá subir una nueva solicitud si desea solicitar nuevamente la convalidación.

En cualquiera de los casos el estudiante es notificado vía correo electrónico de la decisión del Subdecano. Además podrá revisar su solicitud dirigiéndose a la sección Solicitudes activas del sistema.

Imprimir informe para STA

Una vez que la solicitud ha sido aprobada, un usuario con rol secretaria podrá imprimir el informe de la solicitud para enviarlo a la Secretaría Técnica Académica de la ESPOL. Para ello deberá dirigirse a la sección *Mis Actividades Pendientes*, *Solicitar convalidación de materia* y seleccionar la actividad Imprimir informe para STA. Lo anterior ocasionará que se muestre en pantalla la siguiente ventana.

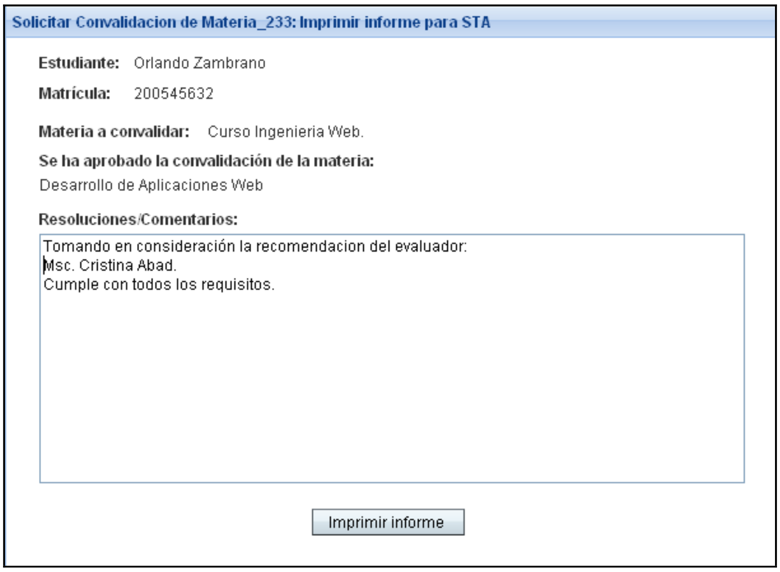
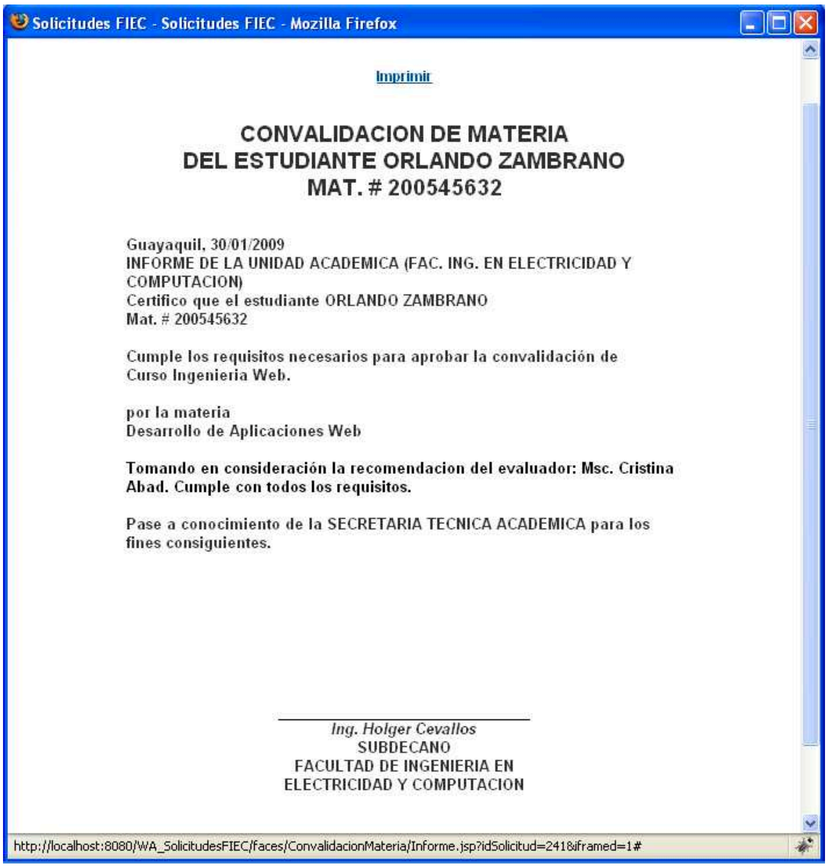


Figura A.43 Formulario de impresión de informe.

Aquí el usuario con rol secretaria verá los datos más relevantes de la solicitud. En el cuadro de texto “Resoluciones/ Comentarios” se mostrará los comentarios escritos por el profesor evaluador de la solicitud. Este campo es editable para dar la opción de adherir información al informe o cambiarla. Una vez escrito lo necesario, se debe seleccionar la opción **Imprimir informe** con lo cual se realiza la actividad y se muestra la siguiente ventana popup con el informe generado.



Solicitudes FIEC - Solicitudes FIEC - Mozilla Firefox

Imprimir

CONVALIDACION DE MATERIA
DEL ESTUDIANTE ORLANDO ZAMBRANO
MAT. # 200545632

Guayaquil, 30/01/2009
INFORME DE LA UNIDAD ACADEMICA (FAC. ING. EN ELECTRICIDAD Y COMPUTACION)
Certifico que el estudiante ORLANDO ZAMBRANO
Mat. # 200545632

Cumple los requisitos necesarios para aprobar la convalidación de Curso Ingenieria Web.

por la materia
Desarrollo de Aplicaciones Web

Tomando en consideración la recomendacion del evaluador: Msc. Cristina Abad. Cumple con todos los requisitos.

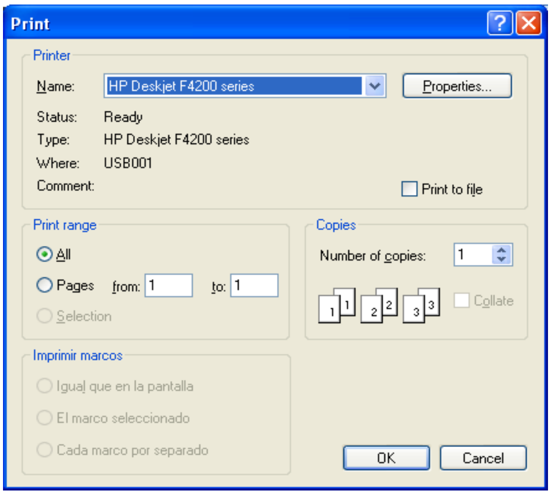
Pase a conocimiento de la SECRETARIA TECNICA ACADEMICA para los fines consiguientes.

Ing. Holger Cevallos
SUBDECANO
FACULTAD DE INGENIERIA EN
ELECTRICIDAD Y COMPUTACION

http://localhost:8080/WA_SolicitudesFIEC/faces/ConvalidacionMateria/Informe.jsp?idSolicitud=2418iframed=1#

A.44 Informe de convalidación de materia generado por el sistema.

Para imprimir el informe seleccione el link imprimir, con lo cual se mostrará el programa de impresión del sistema operativo.



A.45 Ventana impresión de documentos de Windows.

Realizada esta actividad el proceso termina, por lo cual para revisar nuevamente la solicitud debe dirigirse a la sección Historial como se describe en la sección 2.5.

Solicitud cambio de carrera

Para tramitar una solicitud de cambio de carrera siga los pasos expuestos en la sección 3.1, modificando los datos de materia por los de la carrera a la que desea cambiarse. La figura A.46 muestra el formulario de ingreso de una solicitud de cambio de carrera.

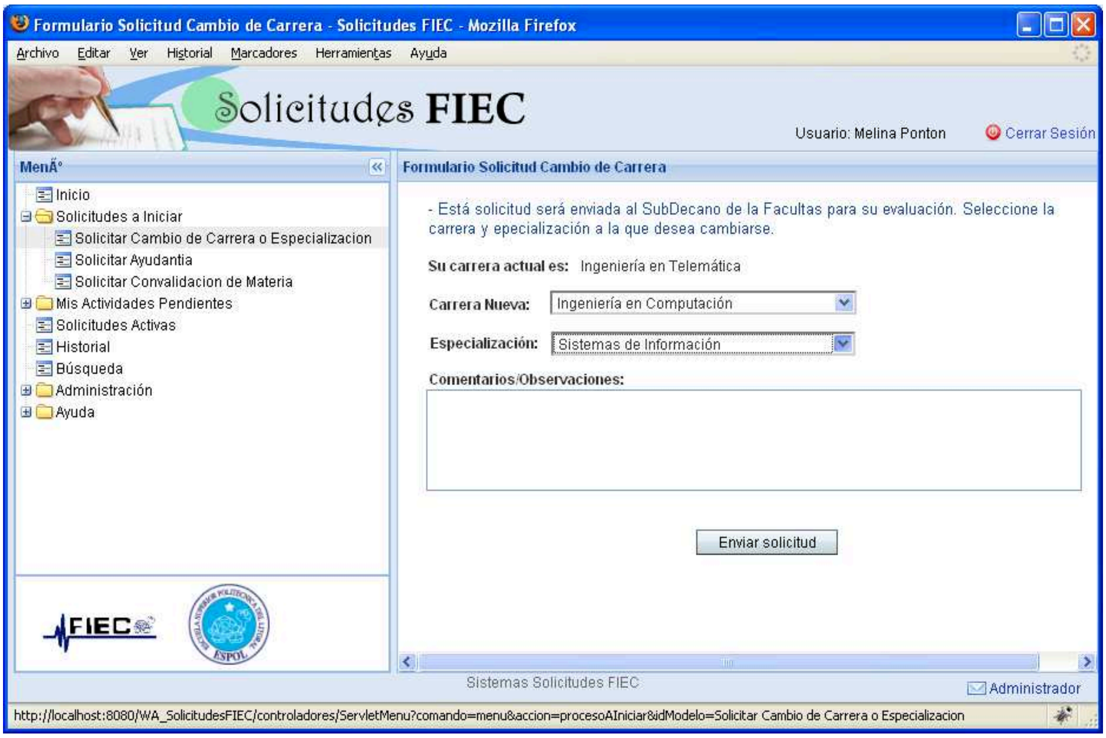


Figura A.46 Formulario de ingreso: Solicitud cambio de carrera.

Solicitud para ser ayudante

Iniciar una solicitud

Para requerir ser ayudante diríjase al *menú del sistema, Solicitudes a Iniciar, Solicitar Ayudantía*. Si existen ayudantías que estén en período de recepción de solicitudes, el sistema mostrará una lista de ellas, como se visualiza en la figura A.47. En esta lista el usuario podrá ver el tipo (académica o de actividades varias) y nombre de la ayudantía, la fecha límite para poder aplicar a ella y el nombre del profesor responsable. También podrá revisar el texto completo de la convocatoria, donde detalla la información de la ayudantía y los requisitos que debe cumplir el aspirante.

Figura A.47 Lista de ayudantías disponibles.

Para aplicar a la ayudantía el usuario debe dar clic sobre el nombre de la misma, con lo cual obtendrá el formulario mostrado en la figura A.48.

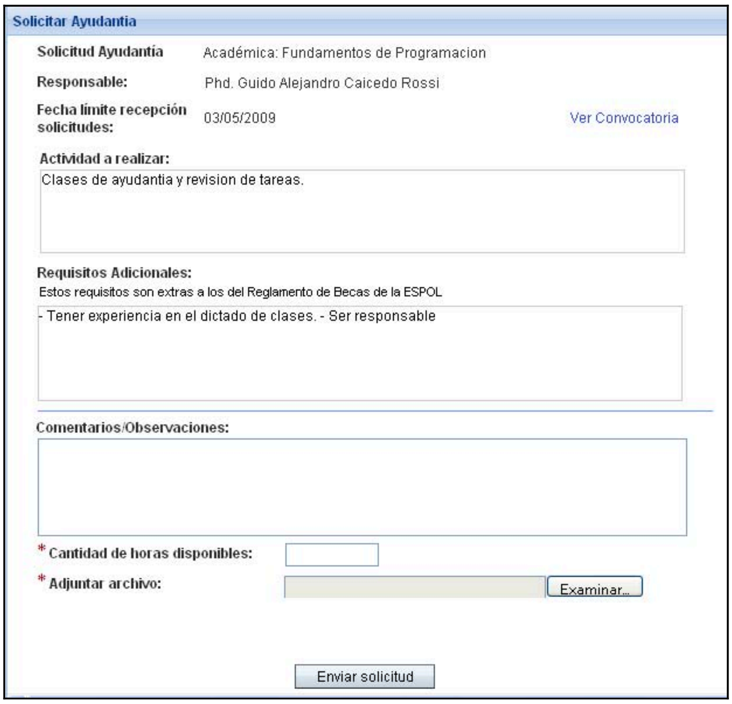


Figura A.48 Formulario solicitar ayudantía

En este formulario se debe ingresar el número de horas semanales que el estudiante tiene disponible para la ayudantía. Se recomienda que se coloque esta información en base a lo que el profesor responsable de la ayudantía solicitó (por ejemplo si se solicitaron ayudantes de 10 horas coloque 10 o 20 horas). Se debe recordar que el máximo número de horas de ayudantía semanales permitido es de 20 horas.

Además en caso de que el profesor responsable lo haya requerido, será necesario adjuntar un archivo a la solicitud. Podrá revisar qué tipo de archivo se requiere leyendo la convocatoria al dar clic en el link Ver convocatoria.

Una vez que los datos necesarios han sido ingresados y si no existen inconvenientes, es posible enviar la solicitud seleccionando la opción Enviar solicitud. El sistema mostrará un mensaje de confirmación similar al de la figura A.49.

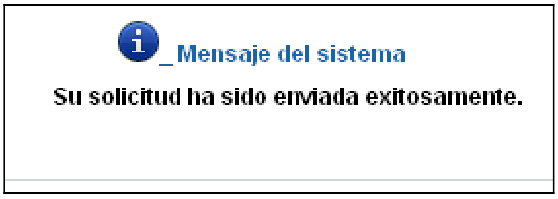


Figura A.49 Mensaje de confirmación: solicitud enviada exitosamente.

Enviada la solicitud, el estudiante deberá esperar a que se le notifique, vía correo electrónico, si fue seleccionado o no.

Confirmación de aceptación de ayudantía.

Si el estudiante fue seleccionado como ayudante, podrá realizar la actividad confirmar aceptación, en la cual debe responder si acepta o no la ayudantía. En el formulario que corresponde a esta actividad (figura A.50), es necesario ingresar los datos financieros del estudiante. Estos datos son parte del contrato de ayudantía y si el estudiante no ingresa esta información podrá responder a la confirmación, pero deberá acercarse a la secretaria de la FIEC a entregar estos datos, después de recibir un mail de notificación de que indique que fue ganador del concurso de ayudantía.

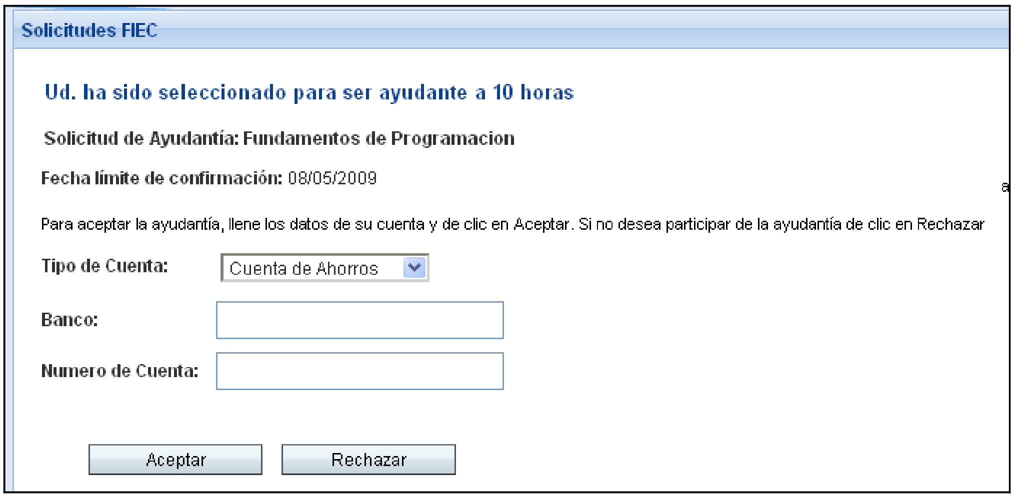


Figura A.50 Formulario confirmación de aceptación de ayudantía.

Sea cual fuera la decisión tomada el sistema mostrará un mensaje de confirmación similar al de la figura A.51.

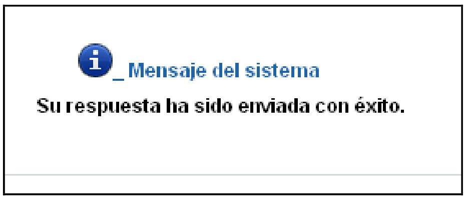


Figura A.51 Mensaje de confirmación: respuesta enviada exitosamente.

Solicitud apetura de postgrado

Para tramitar una solicitud de apertura de apertura de postgrado, puede guiarse por los pasos expuestos en la sección 3.1, tomando en cuenta las siguientes consideraciones: una solicitud de apertura postgrado puede ser iniciada solo por usuarios con rol profesor y es evaluada por el Decano de la facultad.

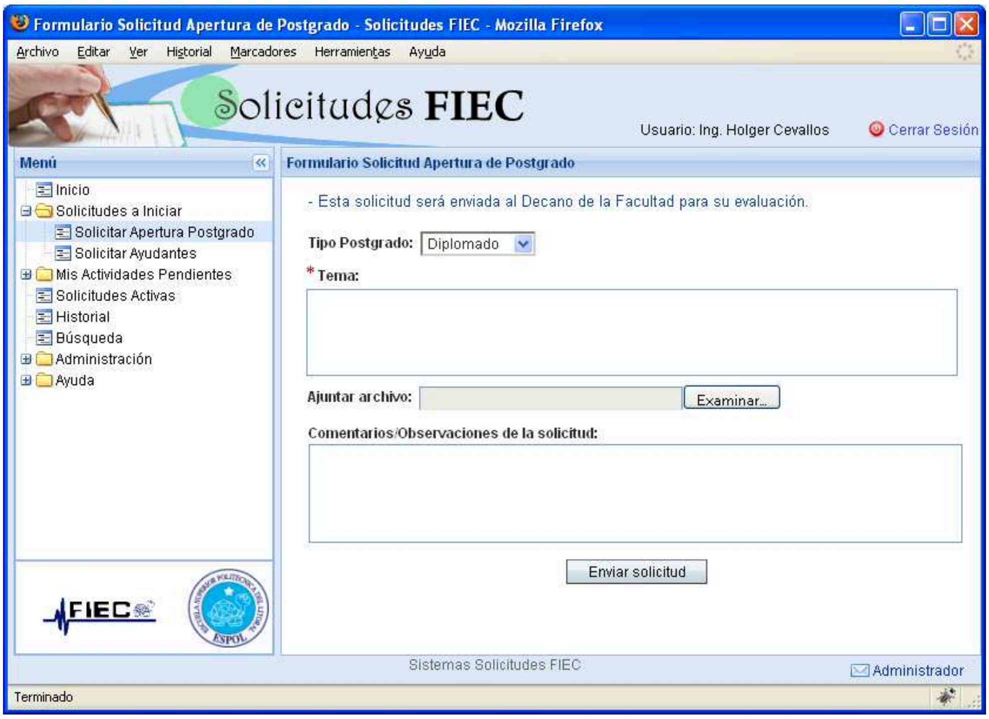


Figura A.52 Formulario de ingreso: Solicitud apertura postgrado.

Solicitud de ayudantes

Iniciar una solicitud

Para iniciar una solicitud para contratación de ayudantes se debe dirigir al *menú del sistema, Solicitudes a iniciar* y seleccionar un tipo de solicitud de la lista mostrada. Al momento se visualizará el formulario de ingreso de la solicitud.

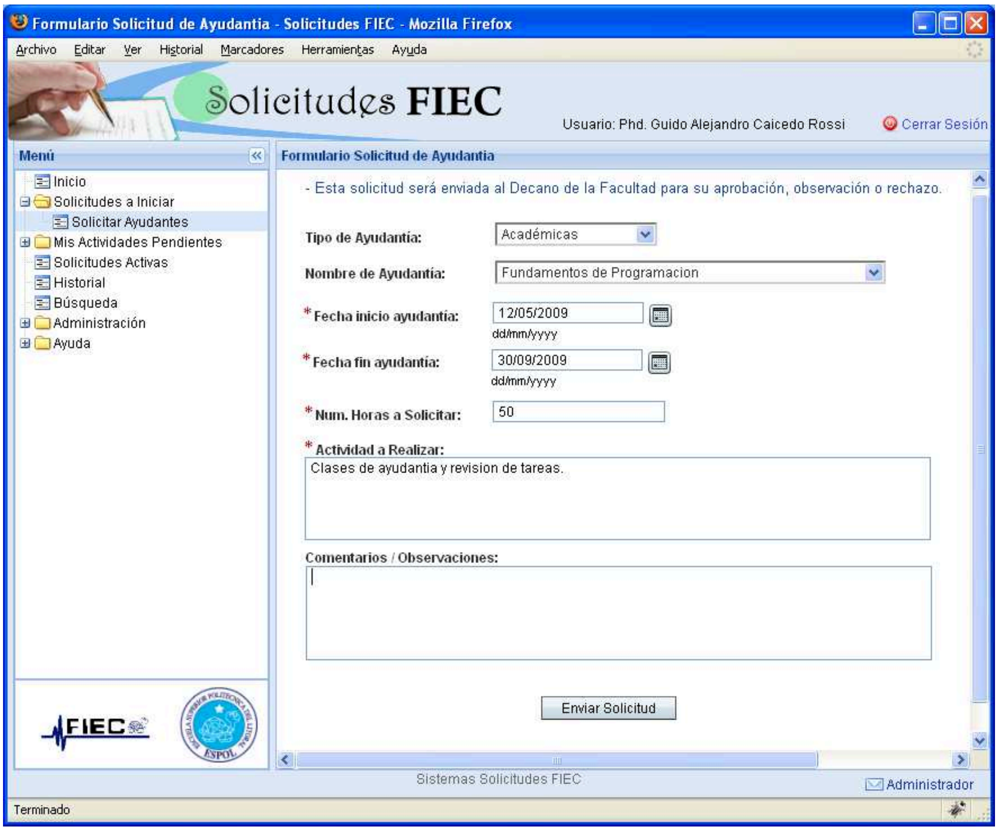


Figura A.53 Formulario de ingreso: Solicitud de ayudantes.

En el formulario mostrado por la figura A.53 es necesario ingresar la siguiente información:

Tipo de la ayudantía (académica o de actividades varias).

Nombre de la ayudantía, el cual será el nombre de una materia en caso de tratarse de una ayudantía académica.

- La fecha de inicio y de fin de la ayudantía.
- Cantidad de horas de ayudantía a solicitar.
- La actividad a realizar por los ayudantes.
- Comentarios u observaciones de la solicitud (opcional).

Una vez que los datos requeridos son llenados, se puede enviar la solicitud al Decano de la facultad seleccionando **Enviar Solicitud**. El sistema verificará que los datos ingresados sean correctos y en caso de no serlos mostrará mensajes alusivos al error.

En caso de que exista actualmente una solicitud de ayudantes activa para la misma ayudantía, el sistema no permitirá el envío de la solicitud y mostrará el siguiente mensaje de error.

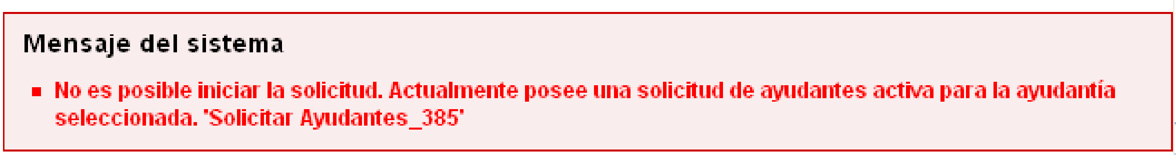


Figura A.54 Mensaje de error: Solicitud de ayudantes existente.

Finalmente, una vez que el sistema validó el ingreso de la información obligatoria, y si no existen inconvenientes se inicia el trámite mostrando un mensaje descriptivo (Figura A.55) y enviando una notificación vía correo electrónico al usuario configurado con el rol de Decano en el sistema.

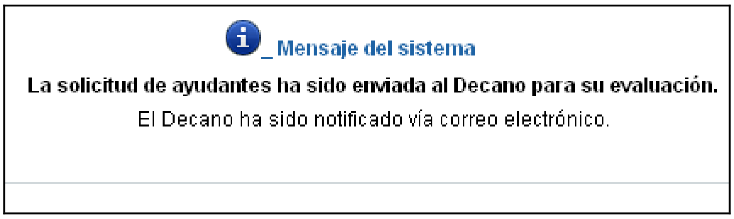


Figura A.55 Mensaje de confirmación: envío de solicitud a Decano.

Revisión de una solicitud

Una vez que el profesor ha iniciado una solicitud, esta pasa a manos del Decano de la facultad para su evaluación. En este paso el Decano podrá decidir entre aceptar, devolver o rechazar la solicitud.

Para acceder a la actividad el usuario puede dirigirse al *menú del sistema, Mis Actividades Pendientes, Solicitar Ayudantes*; o navegar a través de las solicitudes activas como se indica en la sección 2.4.

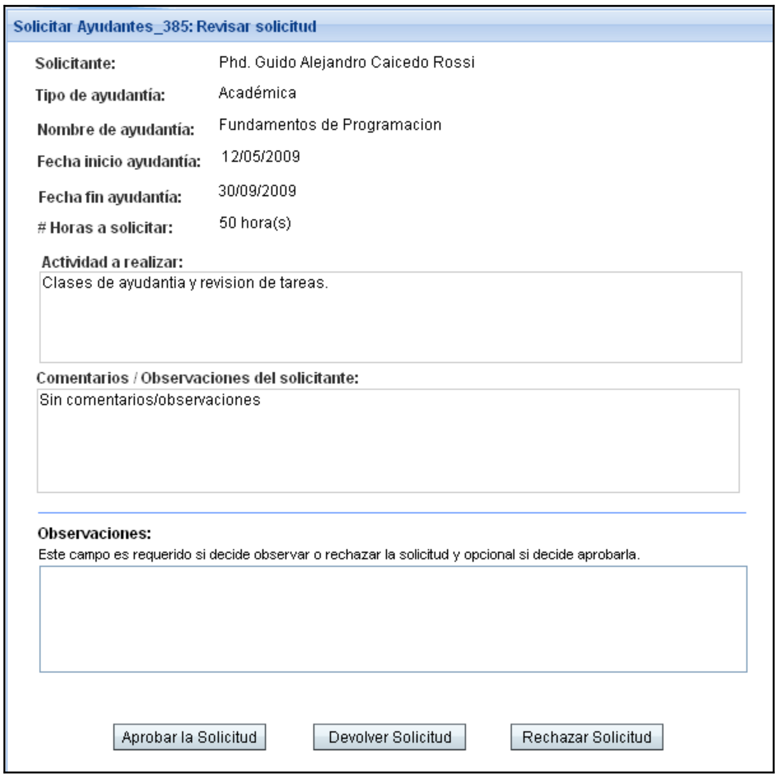


Figura A.56 Formulario Revisión de solicitud de ayudantes.

Opciones Aprobar, Devolver y Rechazar Solicitud: si el Decano decide devolver o rechazar la solicitud, deberá ingresar la razón de su decisión en el campo de texto Observaciones. En el caso de decidir aprobarla, este campo es opcional.

Si la solicitud es rechazada el proceso termina y el estudiante deberá subir una nueva solicitud si desea realizar la convalidación.

Luego de tomada la decisión aparecerá un mensaje de notificación descriptivo y se enviará una un correo electrónico al solicitante.

Reiniciar una solicitud

Si durante la revisión de la solicitud el Decano decidió devolverla, el profesor solicitante podrá decidir si desea reiniciar su solicitud o finalizarla a través de la actividad *Volver a Solicitar*, cuyo formulario se muestra en la figura A.57.

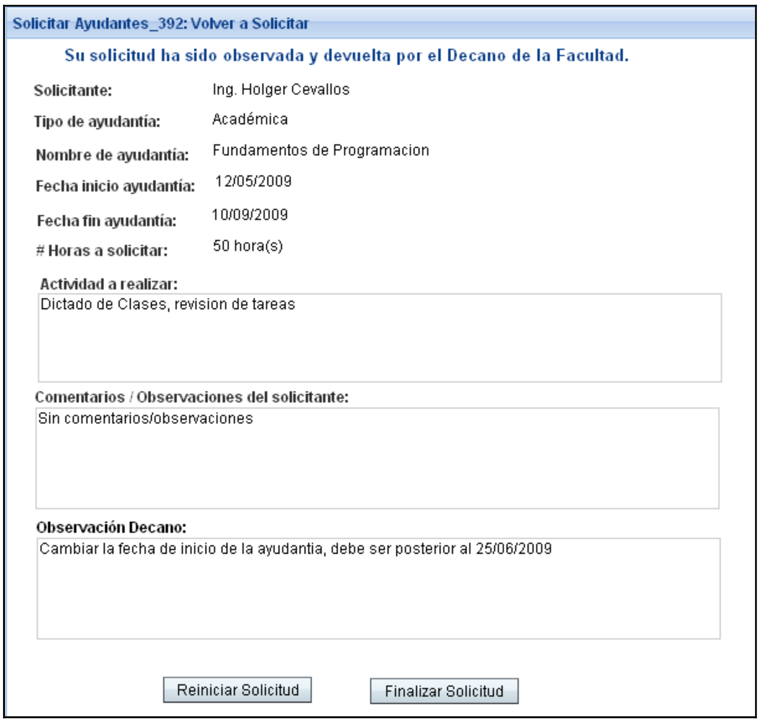


Figura A.57 Formulario volver a solicitar ayudantes.

Si el solicitante decide Finalizar su solicitud, el proceso termina y el profesor deberá subir una nueva solicitud.

En caso de decidir Reiniciar la solicitud, el solicitante deberá dirigirse a la sección *Mis Actividades Pendientes* para acceder nuevamente a ella, realizar los cambios necesarios y volver a enviarla. Este formulario es muy similar al descrito en iniciar solicitud, con la diferencia de que los campos “Tipo de Ayudantía” y “Nombre de Ayudantía” ya no son editables pues son identificadores de la solicitud en curso.

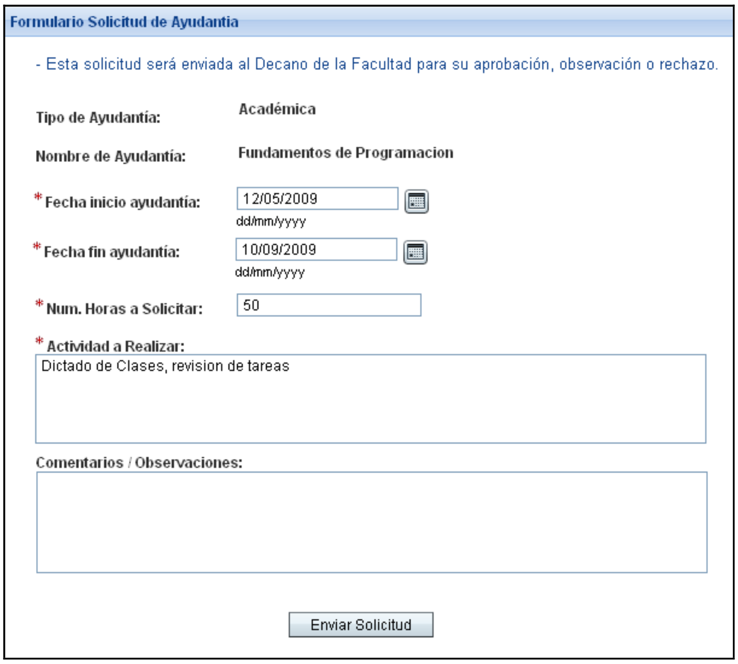


Figura A.58 Formulario de solicitud de ayudantes, después de que ha sido reiniciada.

Una vez realizado los cambios necesarios puede enviar nuevamente su solicitud.

Configurar datos para convocatoria

Luego de que el Decano ha aprobado la solicitud de ayudantes, el profesor solicitante podrá realizar la actividad *Configurar datos para convocatoria*, cuyo formulario se muestra en la figura A.59.

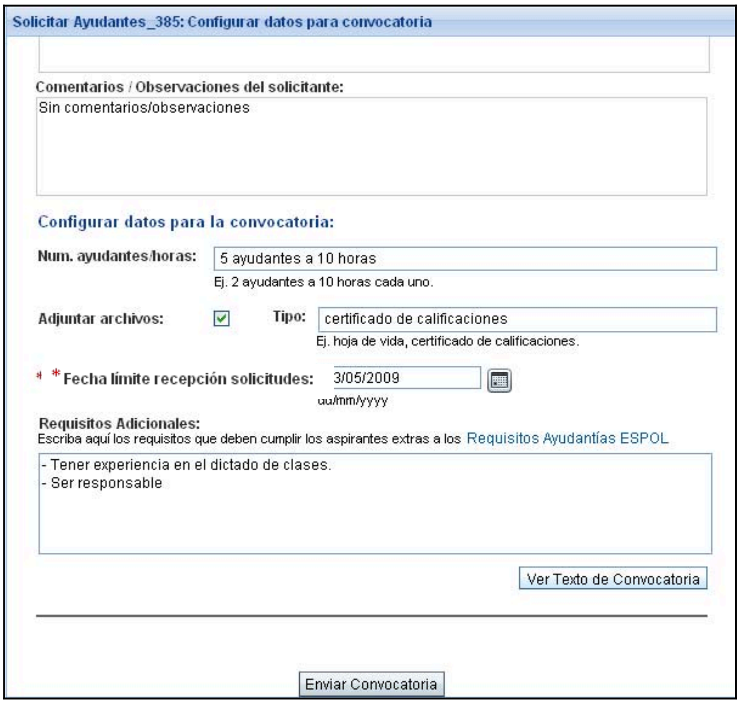


Figura A.59 Formulario configuración de datos para convocatoria.

En este formulario el solicitante podrá ingresar información necesaria para la convocatoria al concurso de ayudantía que se envía a los estudiantes.

El único campo requerido en el formulario es la “Fecha límite de recepción de solicitudes”, la cual indica la fecha tope para entregar de las solicitudes por parte de los estudiantes.

El campo “Num. ayudantes/horas” describe la división de las horas de ayudantía que el profesor solicitó de acuerdo a cuantos ayudantes a equis horas necesita. Esta información es opcional y sirve de referencia para el estudiante.

El campo “Adjuntar archivo” y “Tipo” deben ser llenados en caso de que el profesor necesite que los aspirantes adjunten algún tipo de archivo en la solicitud de ayudantía.

Una vez que los datos requeridos sean llenados puede seleccionar la opción **Ver Texto de Convocatoria** para visualizar el texto que se enviará vía e-mail a los estudiantes. Este texto contiene la información que se ingreso anteriormente en el formulario, además de los datos ingresados en el formulario de ingreso de la solicitud de ayudantes. La figura A.60 muestra un ejemplo de convocatoria.

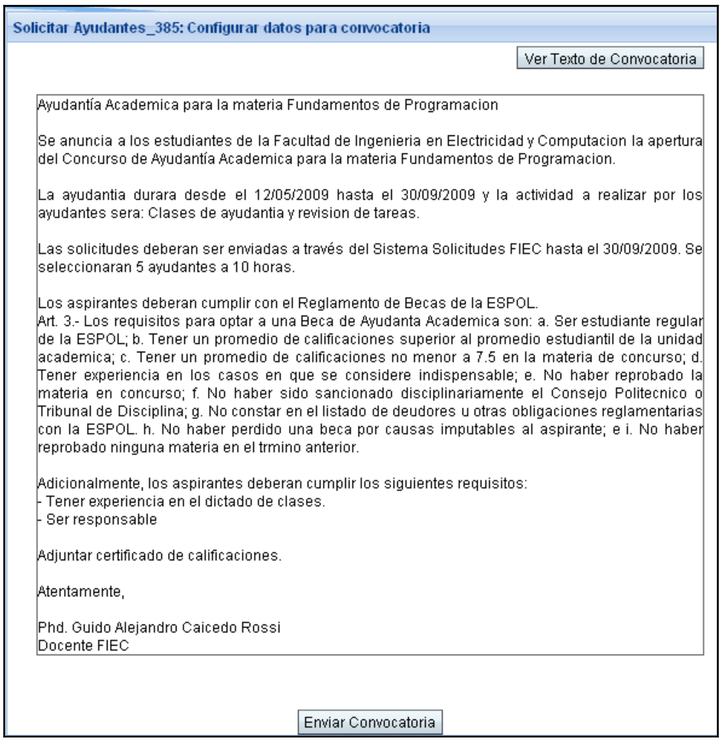


Figura A.60 Ver texto de convocatoria.

Una vez ingresado todos los datos y si no existen inconvenientes, se debe seleccionar la opción Enviar Convocatoria. Al momento el sistema mostrará un mensaje de confirmación y se enviara un mail al administrador de correo para que envíe el testo de convocatoria vía e-mail a todos los estudiantes.

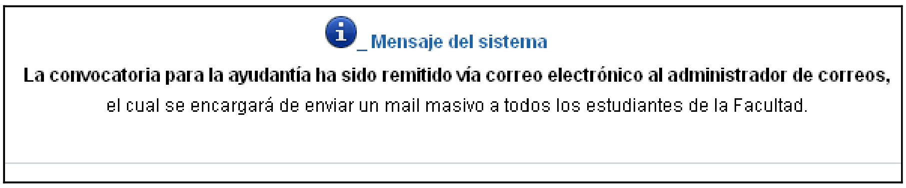
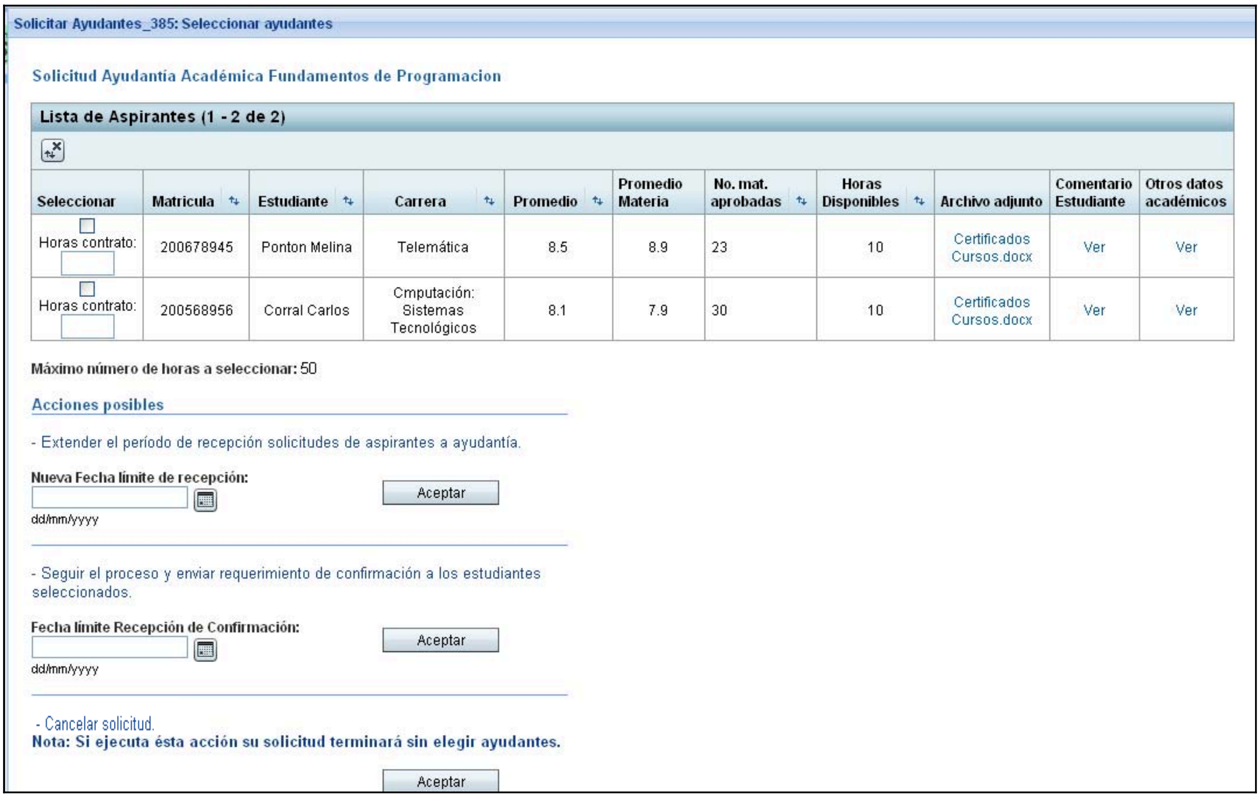


Figura A.61 Mensaje de confirmación: Convocatoria enviada.

Selección de ayudantes

Una vez concluido el tiempo para recibir las solicitudes de los aspirantes a la ayudantía, el profesor podrá re alizar la actividad *Seleccionar Ayudantes*. La figura A.62 muestra el formulario de selección de ayudantes.


Solicitar Ayudantes_385: Seleccionar ayudantes

Solicitud Ayudantía Académica Fundamentos de Programacion

Lista de Aspirantes (1 - 2 de 2)

Seleccionar	Matricula	Estudiante	Carrera	Promedio	Promedio Materia	No. mat. aprobadas	Horas Disponibles	Archivo adjunto	Comentario Estudiante	Otros datos académicos
Horas contrato:	200678945	Ponton Melina	Telemática	8.5	8.9	23	10	Certificados Cursos.docx	Ver	Ver
Horas contrato:	200568956	Corral Carlos	Cmputación: Sistemas Tecnológicos	8.1	7.9	30	10	Certificados Cursos.docx	Ver	Ver

Máximo número de horas a seleccionar: 50

Acciones posibles

- Extender el período de recepción solicitudes de aspirantes a ayudantía.

Nueva Fecha límite de recepción:

dd/mm/yyyy

Aceptar

- Seguir el proceso y enviar requerimiento de confirmación a los estudiantes seleccionados.

Fecha límite Recepción de Confirmación:

dd/mm/yyyy

Aceptar

- Cancelar solicitud.

Nota: Si ejecuta ésta acción su solicitud terminará sin elegir ayudantes.

Aceptar

Figura A.62 Formulario selección de ayudantes.

En este formulario el profesor podrá visualizar la lista de todos los estudiantes que hayan registrado una solicitud para participar del concurso de ayudantía.

En este momento de la solicitud, el profesor puede decidir entre diferentes acciones posibles. La primera es extender el período de recepción de solicitudes de los estudiantes, para lo cual deberá ingresar la nueva fecha límite de recepción y seleccionar el botón **Aceptar** que esta junto a la descripción de la acción. Realizada esta acción el sistema muestra un mensaje de confirmación como el mostrado en la figura A.63 y la ayudantía vuelve a estar disponible para que los estudiantes registren sus solicitudes.

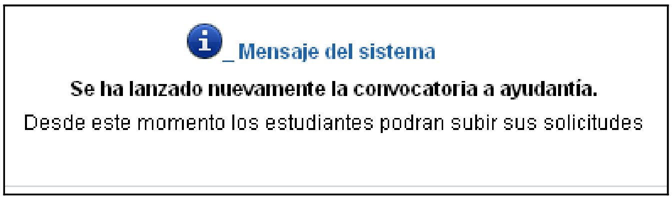


Figura A.63 Mensaje de confirmación: convocatoria enviada nuevamente.

Terminado el período de recepción de solicitudes la actividad seleccionar ayudantes se podrá realizar nuevamente.

La segunda acción posible es seleccionar a los ayudantes y envía una notificación de confirmación a los estudiantes seleccionados para que puedan responder si aceptan o no el contrato de ayudantía.

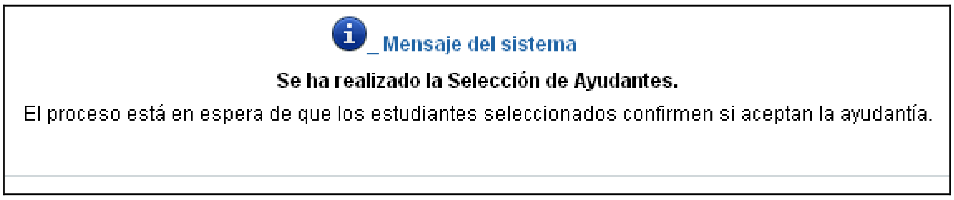


Figura A.64 Mensaje de confirmación: selección de ayudantes.

La tercera y última opción es cancelar la solicitud. Si es realizada la solicitud se cancela sin elegir ayudantes. Esta opción está disponible para los casos en los que por ejemplo se declare desierto el concurso de ayudantía a falta de aspirantes. Los estudiantes que participaron en el concurso de ayudantía serán notificados vía correo electrónico.

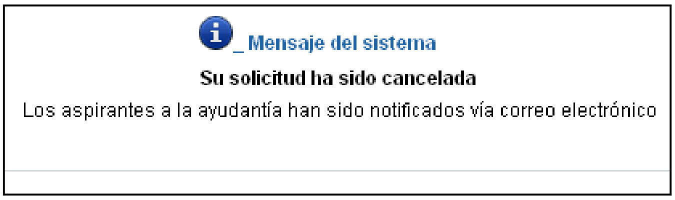
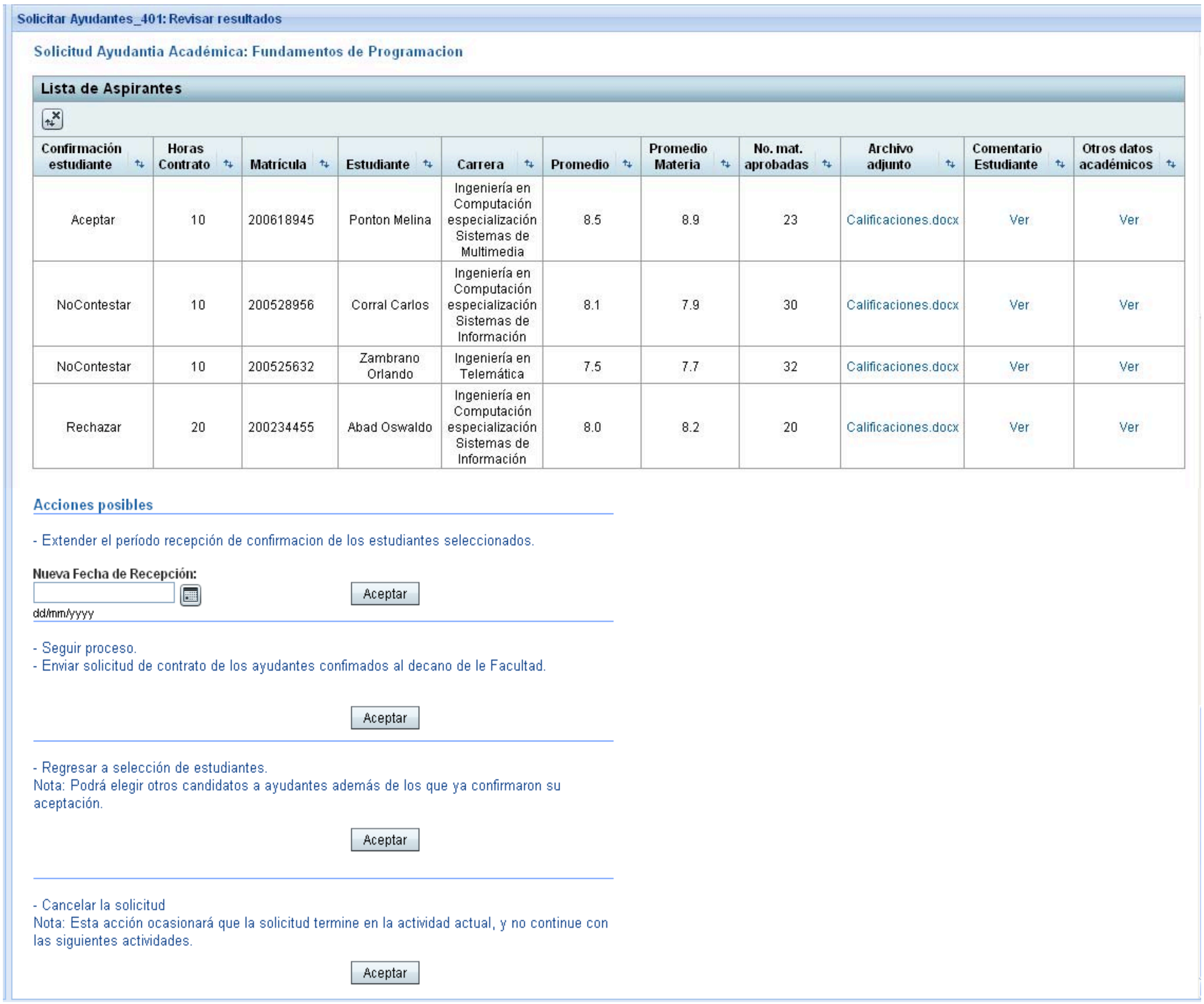


Figura A.65 Mensaje de confirmación: solicitud cancelada.

Revisión de resultados

Una vez concluido el tiempo para recibir las confirmaciones de los estudiantes seleccionados, el profesor podrá realizar la actividad *Revisión de resultados*. La figura A.66 muestra el formulario de revisión de resultados.



Solicitar Ayudantes_401: Revisar resultados

Solicitud Ayudantia Académica: Fundamentos de Programacion

Lista de Aspirantes

Confirmación estudiante	Horas Contrato	Matricula	Estudiante	Carrera	Promedio	Promedio Materia	No. mat. aprobadas	Archivo adjunto	Comentario Estudiante	Otros datos académicos
Aceptar	10	200618945	Ponton Melina	Ingeniería en Computación especialización Sistemas de Multimedia	8.5	8.9	23	Calificaciones.docx	Ver	Ver
NoContestar	10	200528956	Corral Carlos	Ingeniería en Computación especialización Sistemas de Información	8.1	7.9	30	Calificaciones.docx	Ver	Ver
NoContestar	10	200525632	Zambrano Orlando	Ingeniería en Telemática	7.5	7.7	32	Calificaciones.docx	Ver	Ver
Rechazar	20	200234455	Abad Oswaldo	Ingeniería en Computación especialización Sistemas de Información	8.0	8.2	20	Calificaciones.docx	Ver	Ver

Acciones posibles

- Extender el período recepción de confirmacion de los estudiantes seleccionados.

Nueva Fecha de Recepción:

dd/mm/yyyy

Aceptar

- Seguir proceso.
- Enviar solicitud de contrato de los ayudantes confimados al decano de le Facultad.

Aceptar

- Regresar a selección de estudiantes.

Nota: Podrá elegir otros candidatos a ayudantes además de los que ya confirmaron su aceptación.

Aceptar

- Cancelar la solicitud

Nota: Esta acción ocasionará que la solicitud termine en la actividad actual, y no continue con las siguientes actividades.

Aceptar

Figura A.66 Formulario revisión de resultados.

Este formulario muestra la lista de los estudiantes que fueron seleccionados como candidatos a ayudantes, junto a la respuesta que dieron acerca de su aceptación de la ayudantía.

En este momento de la solicitud, el profesor puede decidir entre cuatro diferentes acciones posibles. La primera es *extender el período de recepción de confirmaciones* de los estudiantes que no hayan contestado si aceptan o no la ayudantía. Para esto deberá ingresar la nueva fecha límite de recepción y seleccionar el botón **Aceptar** que esta junto a la descripción de la acción. Realizada esta acción el sistema muestra un mensaje de confirmación descriptivo y los estudiantes que no contestaron podrán hacerlo a partir de este momento.

La segunda acción posible es *enviar un informe de la selección al Decano de la facultad* y se debe realizar cuando el profesor este de acuerdo con lo mostrado en la tabla de aspirantes. Esto es debido a que el informe de selección que se envía el Decano solo contiene la lista de los estudiantes que fueron seleccionados y que contestaron favorablemente a la confirmación. Al realizarse la acción se mostrará un mensaje de confirmación y se enviará un e-mail al Decano de la facultad.

La tercera acción permite *regresar al formulario de selección de ayudantes*, esto permitirá que el profesor tenga a su disposición las acciones que se pueden realizar en esta actividad, como por ejemplo extender la recepción de solicitudes o seleccionar más estudiantes.

La cuarta y última acción es *cancelar la solicitud*. Si es realizada la solicitud se cancela sin contratar ayudantes. Esta opción está disponible para los casos en los que por ejemplo se declare desierto el concurso de ayudantía a falta de aspirantes. Los estudiantes que participaron en el concurso de ayudantía serán notificados vía correo electrónico.

En todos los casos el sistema mostrará mensajes de confirmación alusivos a la acción elegida.

Revisión de informe de selección

Luego de que el profesor ha decidido quienes serán los nuevos ayudantes, se envía un informe de la selección realizada al Decano de la Facultad. Esto es solo a manera de notificar como fue realizada la selección y quiénes son los estudiantes escogidos, dado que la contratación de ayudantes ya fue aprobada anteriormente por el Decano. Para realizar la actividad y permitir que el trámite siga su curso se debe seleccionar **Aceptar**. La figura A.67 muestra el formulario correspondiente a la actividad *Revisión Informe de Selección*.


Solicitar Ayudantes_401: Revisar informe de seleccion

Informe de selección de Ayudantes.

Solicitud Ayudantía Acaémica Fundamentos de Programacion

Lista de Ayudantes Seleccionados

Horas Contrato	Estudiante	Matrícula	Carrera	Promedio	Promedio Materia	No mat. aprobadas	Archivo adjunto	Comentario estudiante	Otros datos académicos
10	Ponton Melina	200618945	Ingeniería en Computación especialización Sistemas de Multimedia	8.5	8.9	23	Calificaciones.docx	Ver	Ver
10	Corral Carlos	200528956	Ingeniería en Computación especialización Sistemas de Información	8.1	7.9	30	Calificaciones.docx	Ver	Ver
10	Zambrano Orlando	200525632	Ingeniería en Telemática	7.5	7.7	32	Calificaciones.docx	Ver	Ver

Aceptar

Figura A.67 Formulario revisión informe de selección.

Impresión de contratos

Una vez que el Decano ha revisado el informe de selección, un usuario con rol secretaria podrá imprimir los contratos (generados por el sistema) de los nuevos ayudantes. La figura A.68 visualiza el formulario de impresión de contratos.


Solicitar Ayudantes_401: Imprimir contratos

Solicitud Ayudantía Académica Fundamentos de Programacion

Nombres	Apellidos	Matrícula	Carrera	Horas Contrato	Contrato Ayudantía
Melina	Ponton	200618945	Ingeniería en Computación especialización Sistemas de Multimedia	10	Imprimir Contrato
Carlos	Corral	200528956	Ingeniería en Computación especialización Sistemas de Información	10	Imprimir Contrato
Orlando	Zambrano	200525632	Ingeniería en Telemática	10	Imprimir Contrato

Aceptar

Figura A.68 Formulario de impresión de informe.

Aquí el usuario verá la lista de os ayudantes elegidos y podrá acceder al contrato de cada uno de ellos seleccionando *Imprimir Contrato*. Realizar esta acción dará como resultado una de las siguientes ventanas (figura A.69 y A.70) dependiendo de si la solicitud es de tipo académica o de actividades varias. Los contratos son archivos de extensión .pdf y se mostrarán utilizando el programa Adobe Reader, por lo que para la impresión del archivo puede utilizar las opciones otorgadas por el programa.

SOLICITUD PARA BECAS DE AYUDANTIA ACADEMICA E INVESTIGACION

PARA: Oficina de Personal de la ESPOL

FECHA: 31/01/2009

1.- UNIDAD SOLICITANTE: FIEC

CARRERA TRADICIONAL ☐ **CARRERA AUTOFINANCIADA** ☐

2.- INFORMACION SOBRE EL AYUDANTE SOLICITADO

APELLIDOS Ponton NOMBRES Melina

Nivel 300 Cédula No. 0735545555 Matrícula No. 200618945

Nombre de la materia Fundamentos de Programacion

Paralelo ______ Profesor Ing.HolgerCevallos

Hora/semana 10 Fecha de inicio 15/05/2009 Fecha de término 30/09/2009

3.- MODALIDAD DE LA NOMINACION:

Contratado como ganador de concurso X

Contratación directa ☐

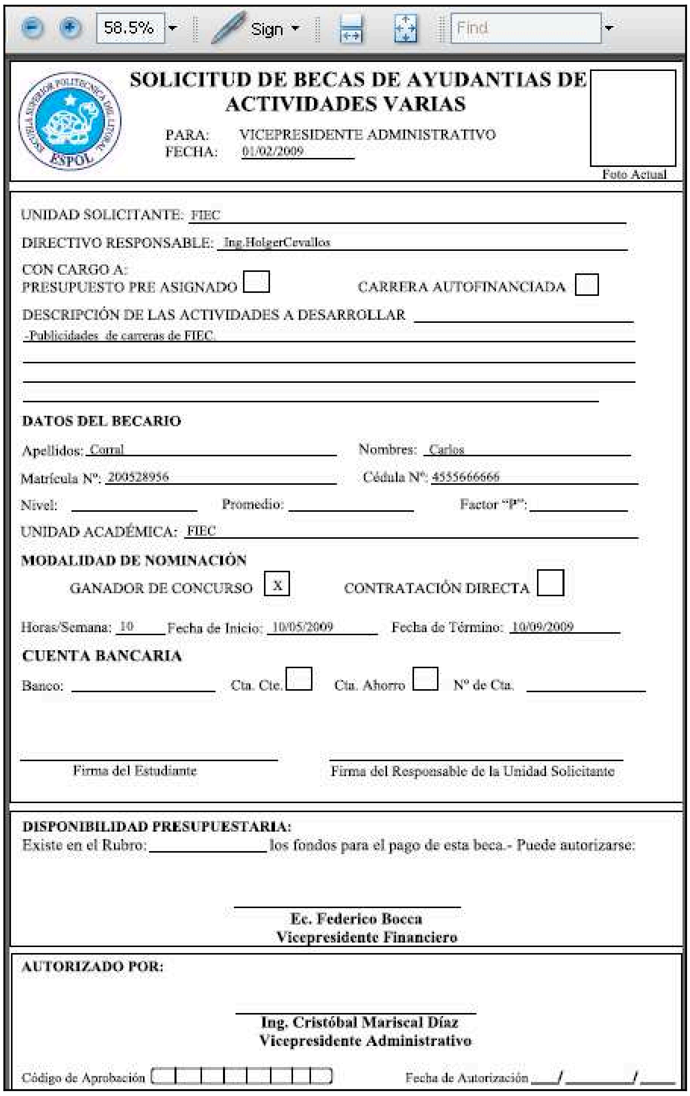
4.- TIPO DE CUENTA BANCARIA:

Banco Pacifico Cta. Cte. ☐ Ahorros X

No. 123459876-1

Firma de responsabilidad

Fifura A.69 Contrato de ayudantía académica generada por el sistema.



58.5% Sign Find

SOLICITUD DE BECAS DE AYUDANTIAS DE ACTIVIDADES VARIAS

ESCUELA SUPERIOR POLITECNICA DEL LITORAL ESPOL

PARA: VICEPRESIDENTE ADMINISTRATIVO
FECHA: 01/02/2009

Foto Actual

UNIDAD SOLICITANTE: FIEC

DIRECTIVO RESPONSABLE: Ing.HolgerCevallos

CON CARGO A:
PRESUPUESTO PRE ASIGNADO ☐ CARRERA AUTOFINANCIADA ☐

DESCRIPCIÓN DE LAS ACTIVIDADES A DESARROLLAR

-Publicidades de carreras de FIEC

DATOS DEL BECARIO

Apellidos: Corral Nombres: Carlos

Matrícula Nº: 200528956 Cédula Nº: 4555666666

Nivel: ___ Promedio: ___ Factor "P": ___

UNIDAD ACADÉMICA: FIEC

MODALIDAD DE NOMINACIÓN

GANADOR DE CONCURSO ☒ CONTRATACIÓN DIRECTA ☐

Horas/Semana: 10 Fecha de Inicio: 10/05/2009 Fecha de Término: 10/09/2009

CUENTA BANCARIA

Banco: ___ Cta. Cte. ☐ Cta. Ahorro ☐ Nº de Cta. ___

Firma del Estudiante

Firma del Responsable de la Unidad Solicitante

DISPONIBILIDAD PRESUPUESTARIA:
Existe en el Rubro: ___ los fondos para el pago de esta beca.- Puede autorizarse:

Ec. Federico Bocca
Vicepresidente Financiero

AUTORIZADO POR:

Ing. Cristóbal Mariscal Díaz
Vicepresidente Administrativo

Código de Aprobación ☐☐☐☐☐☐☐☐ Fecha de Autorización ___/___/___

Figura A.70 Contrato de ayudantía de actividades varias generado por el sistema.

ANEXO B

Manual de Configuración de aplicación Web y motor de Workflow en diferentes servidores de aplicaciones.

La idea es tener un sistema multicapa para poder aprovechar de mejor manera los recursos de los servidores. Esto permitiría tener el Workflow en un servidor (computador) dedicado completamente a la lógica de los procesos y la aplicación web en otro servidor dedicado a responder los requerimientos de los usuarios.

Este manual fue realizado utilizando el siguiente conjunto de software:

- Servidor de aplicaciones JONAS 4.8.6.
  http://wiki.jonas.objectweb.org/xwiki/bin/view/Main/WebHome
- Bonita Workflow 3.1.3
  http://wiki.bonita.objectweb.org/xwiki/bin/view/Main/WebHome
- Servidor de aplicaciones Glassfish V2.
  https://glassfish.dev.java.net/
- NetBeans 6.
  http://www.netbeans.org/

Nota: Para revisar el proceso de instalación de JONAS y Bonita Workflow puede dirigirse a http://wiki.bonita.objectweb.org/xwiki/bin/download/Main/Documentation/ Bonita_GettingStarted.pdf

Pasos:

1. Levante el servidor JONAS y compruebe que Bonita este funcionando correctamente a través de la consola de Workflow. Escriba la ruta "http://localhost:9000/jiapAdmin" e la identificación correspondiente, en este caso usuario: bsoa y clave: bsoa.

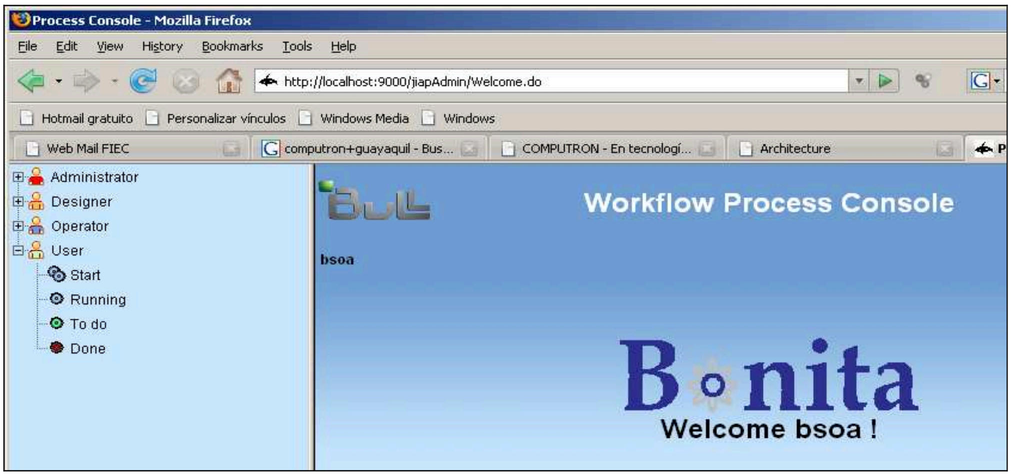


Figura B.1 Consola de administración Bonita Workflow.

2. NetBeans será la interfaz donde se desarrollará la aplicación web (Ver manual de instalación en http://www.netbeans.org/).

Abrir el programa y crear un nuevo proyecto de tipo Aplicación Web.

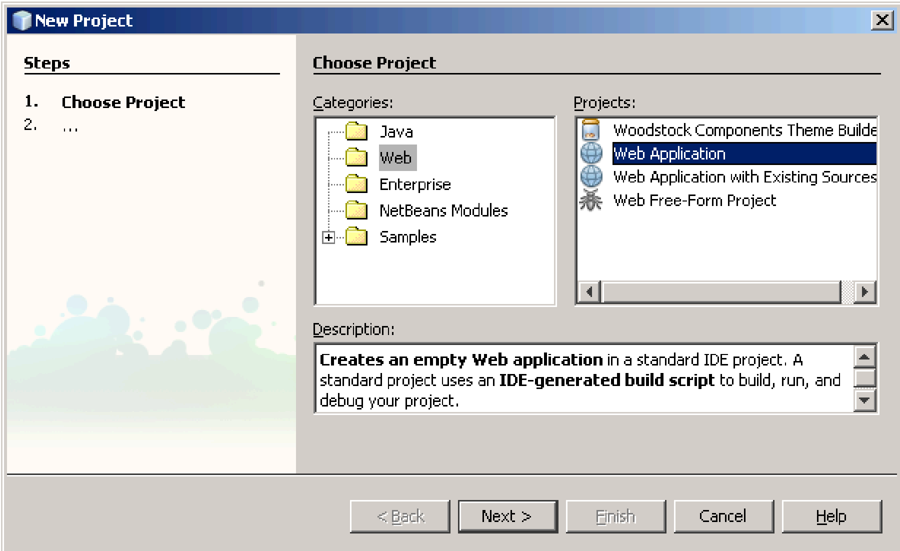


Figura B.2 NetBeans: creación de nuevo proyecto.

En la creación del nuevo proyecto se especifican los siguientes valores:

- Nombre Proyecto: PruebaBonita
- Servidor: GlassFishV2
- Versión Java EE: J2EE 1.4 (Por compatibilidad a Bonita 3.0 que fue desarrollada bajo esta plataforma).
- Quite la selección de "Set source level to 1.4" (se utilizará 1.5 que también funciona bien y trae algunas ventajas).
- En la selección de frameworks seleccione "Visual Web JSF framework"

- Lo demás tendrá los valores por defecto.

3. En la página que se crea por defecto agregue dos etiquetas (usuario y contraseña), dos cajas de texto y un botón con la etiqueta "Ingresar".

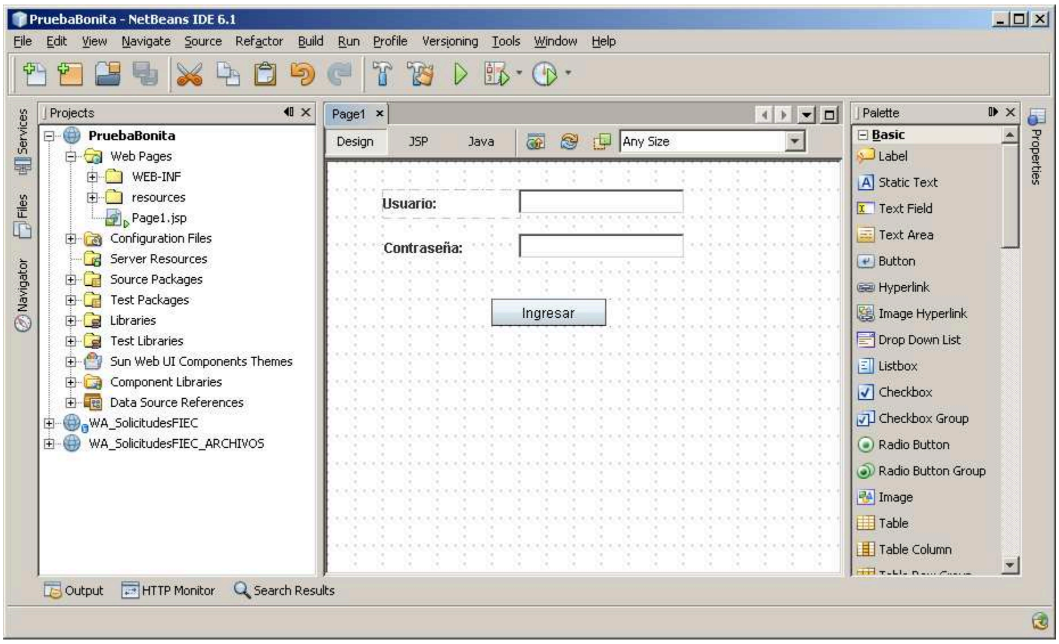


Figura B.3 Vista diseño de página.

4. En la vista proyectos, seleccione PruebaBonita y dé clic derecho en Librerías, opción "Añadir Jar". Posteriormente elija la librería bonita-client.jar, la cual puede ser encontrada en: ruta de instalación de Jonas\BSOA_base\lib\ext

5. Luego escriba el siguiente código en el archivo java relacionado a la página.

```
public String button1_action() {  //Este es el método action() del botón que agregamos hace unos
momentos
      String strUsuario = (String) getTextField1().getText();
      String strPassword=(String) getTextField2().getText();
   If (autentica(strUsuario, strPassword) ){
                  System.out.println(“Se autenticó correctamente”);
                  return “seAutentico”;
      }
            return null;
}
public boolean autentica(String strUsuario, String strContrasena) {
      ProjectSessionHome prjHome;
      // User Adminauthentication
      if(strUsuario==null)strUsuario="bsoa";
      if(strContrasena==null)strContrasena="bsoa";
      SimpleCallbackHandler handler = new
SimpleCallbackHandler(strUsuario,strContrasena.toCharArray()) ;
      LoginContext lc;
      try {
               Hashtable environment = new Hashtable();
         environment.put("java.naming.provider.url","localhost:1099");
         lc = new LoginContext("TestClient", handler);
         lc.login( ) ;
         prjHome = (ProjectSessionHome) ProjectSessionUtil.getHome(environment);
         UserRegistrationHome userRHome=
(UserRegistrationHome)UserRegistrationUtil.getHome(environment);
               UserSessionHome usrHome = (UserSessionHome)
UserSessionUtil.getHome(environment);
         ProjectSession prjSession=prjHome.create() ;
         UserRegistration usrReg = userRHome.create();
         UserSession usrSession = usrHome.create();
         return true;
      } catch (LoginException ex) {
         ex.printStackTrace();
      }catch (RemoteException ex) {
         ex.printStackTrace();
      } catch (CreateException ex) {
         ex.printStackTrace();
      }catch (NamingException ex) {
         ex.printStackTrace();
      }
```

```
    return false;
  }
```

6. Si en este momento se ejecuta la aplicación (pulse F6) y luego se da clic al botón "Ingresar" en el navegador, se imprimirá un error en el log de Glassfish que comienza como sigue:

```
javax.security.auth.login.LoginException: No se han configurado LoginModules para TestClient
    at javax.security.auth.login.LoginContext.init(LoginContext.java:256)
    at javax.security.auth.login.LoginContext.<init>(LoginContext.java:403)
    at pruebabonita2.Page1.autentica(Page1.java:274)
    at pruebabonita2.Page1.button1_action(Page1.java:260)
```

Esto se debe a que no existe un módulo de Login en el archivo "ruta de instalación de Glassfish\domains\domain1\config\login.conf". Para agregarlo, se debe copiar la configuración que tiene el archivo "ruta de instalación de Jonas\BSOA_base\conf\jaas.

```
TestClient {
  // Use the resource dsrlm_1
  org.objectweb.jonas.security.auth.spi.JResourceLoginModule required
  resourceName="dsrlm_jiap_1";
  // Use the login module to propagate security to the JOnAS server
  org.objectweb.jonas.security.auth.spi.ClientLoginModule  required
};
```

Este segmento se agrega en " ruta de instalación de Glassfish\domains\domain1\config\login.conf".

7. Inicie Jonas, reinicie Glassfish, corra la aplicación Web y dé clic en el botón "Ingresar". Esto hará que se produzca el

siguiente error:

```
javax.security.auth.login.LoginException: no se puede encontrar la clase LoginModule:
org.objectweb.jonas.security.auth.spi.JResourceLoginModule
    at javax.security.auth.login.LoginContext.invoke(LoginContext.java:808)
    at javax.security.auth.login.LoginContext.access$000(LoginContext.java:186)
    at javax.security.auth.login.LoginContext$4.run(LoginContext.java:683)
    at java.security.AccessController.doPrivileged(Native Method)
    at javax.security.auth.login.LoginContext.invokePriv(LoginContext.java:680)
    at javax.security.auth.login.LoginContext.login(LoginContext.java:579)
    at pruebabonita2.Page1.autentica(Page1.java:275)
    at pruebabonita2.Page1.button1_action(Page1.java:260)
```

Lo cual se debe a que nos hacen falta las siguientes librerías:

- Ruta de instalación de Jonas \lib\commons\jonas\ow_jonas.jar
- Ruta de instalación de Jonas \lib\commons\jonas\carol\ow_carol.jar
- Ruta de instalación de Jonas \BSOA_base\work\webapps\jonas\ear\jiapAdmin\proed\commons-logging-api.jar
- Ruta de instalación de Jonas \BSOA_base\work\webapps\jonas\ear\jiapAdmin\proed\jotm.jar

Copie estas librerías a "ruta de instalación de Glassfish\domains\domain1\lib\ext\".

8. Ejecute nuevamente la aplicación y obtendrá un error similar al siguiente:

```
javax.security.auth.login.LoginException: Error during the login phase : Cannot retrieve the resource 'jonas_remoteres'. Check that this resource is bound in the registry and that the server name is correct at
org.objectweb.jonas.security.auth.spi.JResourceLoginModule.createChainedLoginException(JResourceLoginModule.java:227)
	at
org.objectweb.jonas.security.auth.spi.JResourceLoginModule.login(JResourceLoginModule.java:214)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:39)
	at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)
	at java.lang.reflect.Method.invoke(Method.java:597)
	at javax.security.auth.login.LoginContext.invoke(LoginContext.java:769)
	at javax.security.auth.login.LoginContext.access$000(LoginContext.java:186)
	at javax.security.auth.login.LoginContext$4.run(LoginContext.java:683)
	at java.security.AccessController.doPrivileged(Native Method)
	at javax.security.auth.login.LoginContext.invokePriv(LoginContext.java:680)
	at javax.security.auth.login.LoginContext.login(LoginContext.java:579)
	at pruebabonita2.Page1.autentica(Page1.java:275)
	at pruebabonita2.Page1.button1_action(Page1.java:260)
```

Para corregirlo, debe agregar la siguiente línea:

```
 System.setProperty(Context.INITIAL_CONTEXT_FACTORY,
"com.sun.jndi.rmi.registry.RegistryContextFactory");//Linea agregada
 SimpleCallbackHandler handler = new SimpleCallbackHandler(strUsuario,
strContrasena.toCharArray());
 LoginContext lc;
...
...
```

Además es necesario copiar el archivo javaee.jar de "ruta de instalación de Glassfish\lib" y pegarlo en " ruta de instalación de Glassfish \domains\domain1\lib\ext".

En este punto ya debe funcionar bien. Vuelva e correr la aplicación, de clic en el botón “Ingresar” y verifique que la siguiente línea se imprima en el log de Glassfish: “Se autenticó correctamente”.

ANEXO C

PRUEBAS

C.1. P

lan de pruebas del sistema

Una vez terminada la implementación del sistema Solicitudes FIEC, se comenzaron las pruebas del sistema. Se decidió realizar pruebas de caja negra o funcionales, en las cuales se puede determinar el comportamiento del software a partir de las entradas o acciones y las salidas o respuestas.

Los casos de prueba que se realizarán pertenecen a los módulos de la aplicación web implementada. Cada caso es especificado mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados.

- **Plan de pruebas módulo ingreso al sistema**

Funcionalidad	Identificador	Caso de prueba
Inicio de sesión	CP-AC-001	Inicio de sesión exitoso

	CP-AC-002	Inicio de sesión fallido por ingreso de credenciales incorrectas

Tabla C.1 Plan de pruebas del módulo ingreso al sistema.

- **Plan de pruebas módulo procesamiento de solicitudes.**

Tipo de Solicitud	Funcionalidad	Identificador	Nombre
Solicitud de ayudantes	Registro de solicitud	CP-PS-001	Registro de solicitud de ayudantes exitoso
		CP-PS-002	Registro de solicitud de ayudantes fallido por existencia de otra solicitud activa para la misma materia o ayudantía varia
	Selección de ayudantes	CP-PS-003	Extensión de plazo para recepción de solicitudes de estudiantes exitosa
		CP-PS-004	Selección de ayudantes y petición de confirmaciones a estudiantes seleccionados exitosa
		CP-PS-005	Cancelación de solicitud de ayudantes exitosa
	Revisión de resultados	CP-PS-006	Extensión de plazo para recepción de confirmaciones
		CP-PS-007	Envío de informe de selección
		CP-PS-008	Regresar a seleccionar estudiantes
Solicitud para ser ayudante	Registro de solicitud	CP-PS-009	Registro de solicitud para ser ayudante exitoso
		CP-PS-010	Registro de solicitud fallido por existencia de otra aplicación a la misma ayudantía

	Registro de confirmación de aceptación de ayudantía	CP-PS-011	Aceptación exitosa
		CP-PS-012	Rechazo exitoso
Solicitud cambio de carrera/ especialización	Reinicio de solicitud	CP-PS-013	Reinicio de solicitud exitoso
		CP-PS-014	Finalización de solicitud exitoso
	Registro de informe	CP-PS-015	Registro de informe exitoso
Solicitud convalidación de materia	Revisión de solicitud	CP-PS-016	Petición de informe a profesor evaluador exitosa
		CP-PS-017	Aprobación de solicitud exitosa
		CP-PS-018	Devolución de solicitud exitosa
		CP-PS-019	Rechazo de solicitud exitosa
Solicitud apertua de postgrado	Revisión de informes	CP-PS-020	Aprobación de solicitud exitosa
		CP-PS-021	Devolución de solicitud exitosa
		CP-PS-022	Rechazo de solicitud exitosa
	Registro de resolución del consejo directivo	CP-PS-023	Registro de resolución exitoso

Tabla C.2 Plan de pruebas módulo procesamiento de solicitudes.

- **Plan de pruebas módulo administración**

Funcionalidad	Identificador	Nombre
Registro de carrera	CP-AD-001	Registro de carrera exitoso
Registro de usuarios	CP-AD-002	Registro de usuario exitoso
Registro de materias	CP-AD-003	Registro de materia exitoso
Configuración de Parámetros	CP-AD-004	Configuración de Decano exitosa
	CP-AD-005	Configuración de Subdecano exitosa

Tabla C.3 Plan de pruebas módulo administración.

C.2. Pruebas realizadas

Módulo ingreso al sistema

<table>
<tr><td colspan="2">Funcionalidad: Inicio de sesión</td></tr>
<tr><td>Código: CP-AC-001</td><td>Nombre: Inicio de sesión exitoso</td></tr>
<tr><td colspan="2">Descripción: El usuario ingresa su usuario y contraseña de ESPOL. El sistema verifica internamente las credenciales ingresadas y si son correctas se inicia la sesión del usuario y se muestra la página inicial del sistema.</td></tr>
<tr><td colspan="2">Condiciones de Ejecución: El usuario debe poseer una cuenta en ESPOL y figurar como activo en la base de datos.</td></tr>
<tr><td colspan="2">Entrada:
• Introducimos ‘kabad’ en el campo usuario
• Introducimos ‘kabad’ en el campo contraseña
• Pulsamos entrar o el botón “Ingresar” de la aplicación.</td></tr>
<tr><td colspan="2">Resultado Esperado: La sesión del usuario inicia</td></tr>
<tr><td colspan="2">Evaluación de la Prueba: Prueba superada con éxito.</td></tr>
</table>

Tabla C.4 Caso de prueba Inicio de sesión exitoso.

<table>
<tr><td colspan="2">Funcionalidad: Inicio de sesión</td></tr>
<tr><td>Código: CP-AC-002</td><td>Nombre: Inicio de sesión fallido por ingreso de credenciales incorrectas</td></tr>
<tr><td colspan="2">Descripción: El usuario ingresa su usuario y contraseña de ESPOL. El sistema verifica internamente las credenciales ingresadas, detecta un error y</td></tr>
</table>

muestra un mensaje al usuario.
Condiciones de Ejecución: El usuario debe poseer una cuenta en ESPOL y figurar como activo en la base de datos.
Entrada: • Introducimos 'kabad' en el campo usuario • Introducimos 'kaba16' en el campo contraseña • Pulsamos el botón "Ingresar" de la aplicación.
Resultado Esperado: El sistema muestra un mensaje de error advirtiendo al usuario de que sus credenciales no son correctas.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.5 Caso de prueba Inicio de sesión fallido por ingreso de credenciales incorrectas.

Módulo procesamiento de solicitudes

Funcionalidad: Registro de solicitud	
Código: CP-PS-001	**Nombre:** Registro de solicitud de ayudantes exitoso
Descripción: Un usuario con rol profesor registra una solicitud de ayudantes.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol profesor y tener iniciada una sesión en el sistema. - Las fechas ingresadas deben ser posteriores a la actual.	
Entrada: • Seleccionamos 'Académicas' de la lista tipo de ayudantía. • Seleccionamos 'Fundamentos de Programación' de la lista nombre de ayudantía. • Introducimos ´12/05/2009´ en el campo fecha inicio ayudantía. • Introducimos ´12/09/2009´ en el campo fecha fin ayudantía.	

• Introducimos 20 en el campo número de horas solicitadas. • Introducimos ´Revisión de tareas´ en el campo actividad a realizar. • Pulsamos el botón ´Enviar Solicitud´ del formulario.
Resultado Esperado: El sistema registra la nueva solicitud y un mensaje de notificación se muestra en pantalla. Además se envía un correo electrónico al Decano de la facultad.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.6 Caso de prueba Registro de solicitud de ayudantes exitoso.

Funcionalidad: Registro de solicitud	
Código: CP-PS-002	**Nombre:** Registro de solicitud de ayudantes fallido por existencia de otra solicitud activa para la misma materia o ayudantía varia
Descripción: Un usuario con rol profesor intenta registrar una solicitud de ayudantes, pero el sistema detecta un error y muestra un mensaje al usuario.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol profesor y tener iniciada una sesión en el sistema. - Las fechas ingresadas deben ser posteriores a la actual.	
Entrada: • Seleccionamos 'Académicas' de la lista tipo de ayudantía. • Seleccionamos 'Fundamentos de Programación' de la lista nombre de ayudantía. • Introducimos ´05/05/2009´ en el campo fecha inicio ayudantía. • Introducimos ´20/09/2009´ en el campo fecha fin ayudantía. • Introducimos 30 en el campo número de horas solicitadas. • Introducimos ´Revisión de tareas´ en el campo actividad a realizar. • Pulsamos el botón ´Enviar Solicitud´ del formulario.	
Resultado Esperado: El sistema muestra un mensaje de error notificando al usuario de que no es posible registrar la solicitud debido a que ya existe una	

solicitud activa para la misma ayudantía.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.7 Caso de prueba Registro de solicitud de ayudantes fallido por existencia de otra solicitud activa para la misma materia o ayudantía varia.

Funcionalidad: Selección de ayudantes	
Código: CP-PS-003	**Nombre:** Extensión de plazo para recepción de solicitudes de estudiantes exitosa
Descripción: Un usuario con rol profesor accede al formulario de Selección de ayudantes para realizar la actividad. El sistema muestra la lista de los estudiantes que han aplicado y el usuario elije la opción extender recepción de solicitudes de aspirantes.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol profesor y tener iniciada una sesión en el sistema.	
Entrada: • Ingresamos la nueva fecha límite de recepción de solicitudes. • Pulsamos el botón ´Aceptar´ que esta junto al campo fecha límite recepción de solicitudes.	
Resultado Esperado: El sistema registra la nueva fecha límite de recepción de solicitudes. Un mensaje de notificación se muestra en pantalla y se envía nuevamente la convocatoria a los estudiantes con la nueva fecha.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.8 Caso de prueba Selección de ayudantes - Extensión de plazo para recepción de solicitudes de estudiantes exitosa.

Funcionalidad: Selección de ayudantes	
Código: CP-PS-004	**Nombre:** Selección de ayudantes y petición de confirmaciones a estudiantes seleccionados exitosa

Descripción: Un usuario con rol profesor accede al formulario de Selección de ayudantes para realizar la actividad. El sistema muestra la lista de los estudiantes que han aplicado, el usuario selecciona a los nuevos ayudantes y elije la opción pedir confirmación a estudiantes seleccionados.
Condiciones de Ejecución: - El usuario debe pertenecer al rol profesor y tener iniciada una sesión en el sistema.
Entrada: • Seleccionamos a varios estudiantes de la lista de aspirantes e ingresamos el número de horas de contrato para cada uno. • Ingresamos la fecha límite de recepción de confirmaciones. • Pulsamos el botón ´Aceptar´ que esta junto al campo fecha límite recepción de confirmaciones.
Resultado Esperado: El sistema registra a los estudiantes seleccionados y la fecha límite de confirmación de solicitudes. Un mensaje de notificación se muestra en pantalla y se envía un correo electrónico a los estudiantes seleccionados.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.9 Caso de prueba Selección de ayudantes y petición de confirmaciones a estudiantes seleccionados exitosa.

Funcionalidad: Selección de ayudantes	
Código: CP-PS-005	**Nombre:** Cancelación de solicitud de ayudantes exitosa
Descripción: Un usuario con rol profesor accede al formulario de Selección de ayudantes para realizar la actividad. El sistema muestra la lista de los estudiantes que han aplicado, el usuario elije la opción cancelar solicitud.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol profesor y tener iniciada una sesión en el sistema.	
Entrada: • Pulsamos el botón ´Aceptar´ que esta junto al texto 'cancelar solicitud'.	

Resultado Esperado: El sistema registra el estado de la solicitud como cancelada. Un mensaje de notificación se muestra en pantalla y se envía un correo electrónico a los estudiantes que aplicaron a la ayudantía.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.10 Caso de prueba Selección de ayudantes - Cancelación de solicitud de ayudantes exitosa.

Funcionalidad: Revision de resultados	
Código: CP-PS-006	**Nombre:** Extensión de plazo para recepción de confirmaciones
Descripción: Un usuario con rol profesor accede al formulario de Revisión de Resultados para realizar la actividad. El sistema muestra la lista de los estudiantes que fueron seleccionados, el usuario elije la opción extender período de recepción de confirmaciones.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol profesor y tener iniciada una sesión en el sistema.	
Entrada: • Ingresamos la nueva fecha límite de recepción de confirmaciones. • Pulsamos el botón ´Aceptar´ que esta junto al texto ´extender período de recepción´.	
Resultado Esperado: El sistema registra la nueva fecha límite de recepción de confirmaciones. Un mensaje de notificación se muestra en pantalla y se envía un mail de notificación a los estudiantes que faltan por responder.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.11 Caso de prueba Selección de ayudantes - Extensión de plazo para recepción de confirmaciones.

Funcionalidad: Revisión de resultados	
Código: CP-PS-007	**Nombre:** Envío de informe de selección de ayudantes
Descripción: Un usuario con rol profesor accede al formulario de Revisión de	

Resultados para realizar la actividad. El sistema muestra la lista de los estudiantes que fueron seleccionados, el usuario elije la opción enviar informe de selección.
Condiciones de Ejecución: - El usuario debe pertenecer al rol profesor y tener iniciada una sesión en el sistema.
Entrada: • Pulsamos el botón ´Aceptar´ que esta junto al texto ´enviar informe de selección´.
Resultado Esperado: Un mensaje de notificación se muestra en pantalla y se envía un correo electrónico al Decano de la facultad notificándole que puede revisar el informe en el sistema.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.12 Caso de prueba Revisión de resultados - Envío de informe de selección de ayudantes.

Funcionalidad: Revision de resultados	
Código: CP-PS-008	**Nombre:** Regresar a seleccionar estudiantes
Descripción: Un usuario con rol profesor accede al formulario de Revisión de Resultados para realizar la actividad. El sistema muestra la lista de los estudiantes que fueron seleccionados y el usuario elije la opción regresar a seleccionar estudiantes.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol profesor y tener iniciada una sesión en el sistema.	
Entrada: • Pulsamos el botón ´Aceptar´ que esta junto al texto ´Regresar a selección de ayudantes´.	
Resultado Esperado: Un mensaje de notificación se muestra en pantalla y se habilita la actividad Seleccionar ayudantes para que pueda ser nuevamente	

realizada.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.13 Caso de prueba Revisión de resultados - Regresar a seleccionar estudiantes.

Funcionalidad: Registro de solicitud	
Código: CP-PS-009	**Nombre:** Registro de solicitud para ser ayudante exitoso
Descripción: Un usuario con rol estudiante registra una solicitud para ser ayudante.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol estudiante y tener iniciada una sesión en el sistema. - La ayudantía debe estar disponible para registro, es decir aún no se cumple la fecha límite de recepción de solicitudes.	
Entrada: • Seleccionamos la ayudantía académica ´Fundamentos de programación’ de la lista de ayudantías disponibles. • Introducimos ´20´ en el campo cantidad de horas disponibles. • Elegimos un archivo para adjuntar. • Pulsamos el botón ´Enviar Solicitud´ del formulario.	
Resultado Esperado: El sistema registra la nueva solicitud y un mensaje de notificación se muestra en pantalla.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.14 Caso de prueba Registro de solicitud para ser ayudante exitoso.

Funcionalidad: Registro de solicitud	
Código: CP-PS-010	**Nombre:** Registro de solicitud fallido por existencia de otra aplicación a la misma ayudantía

Descripción: Un usuario con rol estudiante intenta registrar una solicitud para ser ayudante, pero el sistema detecta un error y muestra un mensaje al usuario.
Condiciones de Ejecución: - El usuario debe pertenecer al rol estudiante y tener iniciada una sesión en el sistema. - La ayudantía debe estar disponible para registro, es decir aún no se cumple la fecha límite de recepción de solicitudes.
Entrada: • Seleccionamos la ayudantía académica ´Fundamentos de programación' de la lista de ayudantías disponibles. • Introducimos ´20´ en el campo cantidad de horas disponibles. • Elegimos un archivo para adjuntar. • Pulsamos el botón ´Enviar Solicitud´ del formulario.
Resultado Esperado: El sistema muestra un mensaje de error notificando al usuario de que no es posible registrar la solicitud debido a que ya envío una solicitud para la misma ayudantía.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.15 Caso de prueba Registro de solicitud fallido por existencia de otra aplicación a la misma ayudantía.

Funcionalidad: Registro de confirmación de aceptación de ayudantía	
Código: CP-PS-011	**Nombre:** Aceptación exitosa
Descripción: Un usuario con rol estudiante accede al formulario de Registro de confirmación para realizar la actividad. El usuario ingresa la información correspondiente a sus datos financieros y elije la opción Aceptar.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol estudiante y tener iniciada una sesión en el sistema. - El plazo para enviar la confirmación debe estar aún vigente.	
Entrada: • Seleccionamos cuenta de ahorros de la lista tipo de cuenta.	

• Ingresamos ´Pacifico´ en el campo banco. • Ingresamos ´123678234-9´ en el campo número de cuenta. • Pulsamos el botón ´Aceptar´.
Resultado Esperado: Un mensaje de notificación se muestra en pantalla y se registra la respuesta del estudiante.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.16 Caso de prueba Registro de confirmación de aceptación de ayudantía - Aceptación exitosa.

Funcionalidad: Registro de confirmación de aceptación de ayudantía	
Código: CP-PS-012	**Nombre:** Rechazo exitoso
Descripción: Un usuario con rol estudiante accede al formulario de Registro de confirmación para realizar la actividad. El usuario ingresa la información correspondiente a sus datos financieros y elije la opción Rechazar.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol estudiante y tener iniciada una sesión en el sistema. - El plazo para enviar la confirmación debe estar aún vigente.	
Entrada: • Pulsamos el botón ´Rechazar´.	
Resultado Esperado: Un mensaje de notificación se muestra en pantalla y se registra la respuesta del estudiante.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.17 Caso de prueba Registro de confirmación de aceptación de ayudantía - Rechazo exitoso.

Funcionalidad: Reinicio de solicitud	
Código: CP-PS-013	**Nombre:** Reinicio de solicitud exitoso
Descripción: Un usuario con rol estudiante accede al formulario de Reinicio de solicitud para realizar la actividad. El usuario elije la opción Reiniciar y la solicitud se activa para ser enviada nuevamente.	

Condiciones de Ejecución: - El usuario debe pertenecer al rol estudiante y tener iniciada una sesión en el sistema. - La solicitud fue devuelta por el Subdecano de la facultad.
Entrada: • Pulsamos el botón ´Reiniciar Solicitud´.
Resultado Esperado: Un mensaje de notificación se muestra en pantalla y se habilita la solicitud para que pueda ser enviada nuevamente.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.18 Caso de prueba Reinicio de solicitud de cambio de carrera/especialización exitoso.

Funcionalidad: Reinicio de solicitud	
Código: CP-PS-014	**Nombre:** Finalización de solicitud exitosa
Descripción: Un usuario con rol estudiante accede al formulario de Reinicio de solicitud para realizar la actividad. El usuario elije la opción Finalizar.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol estudiante y tener iniciada una sesión en el sistema. - La solicitud fue devuelta por el Subdecano de la facultad.	
Entrada: • Pulsamos el botón ´Finalizar Solicitud´.	
Resultado Esperado: Un mensaje de notificación se muestra en pantalla y se registra el estado de la solicitud a Terminada.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.19 Caso de prueba Reinicio de solicitud de cambio de carrera/especialización - Finalización de solicitud exitosa.

Funcionalidad: Registro de informe	
Código: CP-PS-015	**Nombre:** Registro de informe exitoso

Descripción: Un usuario con rol profesor accede al formulario de Registro de Informe para realizar la actividad. El usuario elije la opción Emitir informe.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol profesor y tener iniciada una sesión en el sistema. - El profesor fue seleccionado como evaluador de la solicitud.	
Entrada: • Seleccionamos ´Favorable ´ de la lista informe. • Ingresamos ´cumple con los requisitos´ en el campo de texto comentarios/observaciones. • Pulsamos el botón ´Emitir Informe´.	
Resultado Esperado: Un mensaje de notificación se muestra indicando al usuario que su informe ha sido registrado. Además se envía un correo electrónico al Subdecano de la facultad indicándole que ya puede revisar el informe del evaluador en el sistema.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.20 Caso de prueba Registro de informe exitoso.

Funcionalidad: Revisión de solicitud	
Código: CP-PS-016	**Nombre:** Petición de informe a profesor evaluador exitosa
Descripción: Un usuario configurado como Subdecano accede al formulario de Revisión de solicitud para realizar la actividad. El usuario selecciona un evaluador y elije la opción Solicitar informe.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol Subdecano y tener iniciada una sesión en el sistema.	
Entrada: • Pulsamos el botón seleccionar que esta junto al campo de texto profesor • Seleccionamos ´Ing. Jorge Aragundi´ de la lista de profesores.	

• Pulsamos el botón ´Solicitar Informe´.
Resultado Esperado: Un mensaje de notificación se muestra. Además se envía un correo electrónico al profesor evaluador indicándole que debe emitir un informe acerca de la solicitud.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.21 Caso de prueba Revisión de solicitud de cambio de carrera/especialización - Petición de informe a profesor evaluador exitosa.

Funcionalidad: Revisión de solicitud	
Código: CP-PS-017	**Nombre:** Aprobación de solicitud exitosa
Descripción: Un usuario configurado como Subdecano accede al formulario de Revisión de solicitud para realizar la actividad. El usuario elije la opción Aprobar solicitud	
Condiciones de Ejecución: - El usuario debe pertenecer al rol Subdecano y tener iniciada una sesión en el sistema.	
Entrada: • Pulsamos el botón ´Aprobar solicitud´.	
Resultado Esperado: Un mensaje de notificación se muestra. Además se envía un correo electrónico al estudiante (solicitante) indicándole que su solicitud fue aprobada.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.22 Caso de prueba Revisión de solicitud de cambio de carrera/especialización - Aprobación de solicitud exitosa.

Funcionalidad: Revisión de solicitud	
Código: CP-PS-018	**Nombre:** Devolución de solicitud exitosa
Descripción: Un usuario configurado como Subdecano accede al formulario de Revisión de solicitud para realizar la actividad. El usuario elije la opción	

Devolver solicitud
Condiciones de Ejecución: - El usuario debe pertenecer al rol Subdecano y tener iniciada una sesión en el sistema.
Entrada: • Ingresamos 'faltan prerrequisitos' en el campo de texto comentarios/observaciones. • Pulsamos el botón ´Devolver solicitud´.
Resultado Esperado: Un mensaje de notificación se muestra. Además se envía un correo electrónico al estudiante (solicitante) indicándole que su solicitud fue devuelta.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.23 Caso de prueba Revisión de solicitud de cambio de carrera/especialización - Devolución de solicitud exitosa.

Funcionalidad: Revisión de solicitud	
Código: CP-PS-019	**Nombre:** Rechazo de solicitud exitosa
Descripción: Un usuario configurado como Subdecano accede al formulario de Revisión de solicitud para realizar la actividad. El usuario elije la opción Rechazar solicitud	
Condiciones de Ejecución: - El usuario debe pertenecer al rol Subdecano y tener iniciada una sesión en el sistema.	
Entrada: • Ingresamos 'la materia no puede ser convalida' en el campo de texto comentarios/observaciones. • Pulsamos el botón ´Rechazar solicitud´.	
Resultado Esperado: Un mensaje de notificación se muestra y se cambia el estadoa de la solicitud a terminada. Además se envía un correo electrónico al	

estudiante (solicitante) indicándole que su solicitud fue rechazada.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.24 Caso de prueba Revisión de solicitud de cambio de carrera/especialización - Rechazo de solicitud exitosa.

Funcionalidad: Revisión de informes	
Código: CP-PS-020	**Nombre:** Aprobación de solicitud exitosa
Descripción: Un usuario configurado como Subdecano accede al formulario de Revisión de informes para realizar la actividad. El usuario elije la opción Aprobar solicitud	
Condiciones de Ejecución: - El usuario debe pertenecer al rol Subdecano y tener iniciada una sesión en el sistema.	
Entrada: • Pulsamos el botón ´Aprobar solicitud´.	
Resultado Esperado: Un mensaje de notificación se muestra. Además se envía un correo electrónico al estudiante (solicitante) indicándole que su solicitud fue aprobada.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.25 Caso de prueba Revisión de informes - Aprobación de solicitud de apertura de postgrado exitosa.

Funcionalidad: Revisión de informes	
Código: CP-PS-021	**Nombre:** Devolución de solicitud exitosa
Descripción: Un usuario configurado como Subdecano accede al formulario de Revisión de solicitud para realizar la actividad. El usuario elije la opción Devolver solicitud	

Condiciones de Ejecución: - El usuario debe pertenecer al rol Subdecano y tener iniciada una sesión en el sistema.
Entrada: • Ingresamos 'faltan prerrequisitos' en el campo de texto comentarios/observaciones. • Pulsamos el botón ´Devolver solicitud´.
Resultado Esperado: Un mensaje de notificación se muestra. Además se envía un correo electrónico al estudiante (solicitante) indicándole que su solicitud fue devuelta.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.26 Caso de prueba Revisión de informes - Devolución de solicitud de apertura de postgrado exitosa.

Funcionalidad: Revisión de informes	
Código: CP-PS-022	**Nombre:** Rechazo de solicitud exitosa
Descripción: Un usuario configurado como Subdecano accede al formulario de Revisión de solicitud para realizar la actividad. El usuario elije la opción Rechazar solicitud	
Condiciones de Ejecución: - El usuario debe pertenecer al rol Subdecano y tener iniciada una sesión en el sistema.	
Entrada: • Ingresamos 'la materia no puede ser convalida' en el campo de texto comentarios/observaciones. • Pulsamos el botón ´Rechazar solicitud´.	
Resultado Esperado: Un mensaje de notificación se muestra y se cambia el estado de la solicitud a terminada. Además se envía un correo electrónico al estudiante (solicitante) indicándole que su solicitud fue rechazada.	

Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.27 Caso de prueba Revisión de informes - Rechazo de solicitud de apertura de postgrado exitosa.

Funcionalidad: Registro de resolución del consejo directivo	
Código: CP-PS-023	**Nombre:** Registro de resolución exitoso
Descripción: Un usuario con rol secretaria accede al formulario de Registro de resolución del consejo directivo para realizar la actividad. El usuario ingresa la resolución acerca de la solicitud tomada por el consejo directivo y la imprime.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol Secretaria y tener iniciada una sesión en el sistema.	
Entrada: • Ingresamos '2009-102' en el campo de texto número de resolución. • Seleccionamos 'Favorable' de la lista resultado. • Pulsamos el botón ´Imprimir´.	
Resultado Esperado: Se muestra la resolución ingresada de manera que pueda ser impresa, Un mensaje de notificación se muestra y se cambia el estado de la solicitud a terminada.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.28 Caso de prueba Registro de resolución del consejo directivo exitoso.

Módulo Administración

Funcionalidad: Registro de carrera	
Código: CP-AD-001	**Nombre:** Registro de carrera exitoso
Descripción: Un usuario con rol administrador accede al formulario de Registro de carrera. El usuario ingresa la nueva carrera y sus especializaciones.	

Condiciones de Ejecución: - El usuario debe pertenecer al rol administrador y tener iniciada una sesión en el sistema.
Entrada: • En caso de que la carrera tenga especializaciones primero se debemos registrar estas. Ingresamos 'sistemas tecnológicos' en el campo de texto nombre. • Pulsamos el botón seleccionar que está junto al campo de texto profesor. • Seleccionamos ´Ing. Cristina Abad´ de la lista de profesores. • Habilitamos la casilla de verificación Activar. • Pulsamos el botón Agregar especialización. • Repetimos los pasos anteriores en caso de que existan más especializaciones. • Ingresamos el nombre y coordinador de la nueva carrera. • Pulsamos el botón ´Guardar´.
Resultado Esperado Un mensaje de notificación se muestra y se registra la nueva carrera con sus especializaciones en la base de datos.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.29 Caso de prueba Registro de carrera exitoso.

Funcionalidad: Registro de usuarios	
Código: CP-AD-002	**Nombre:** Registro de usuario exitoso
Descripción: Un usuario con rol administrador accede al formulario de Registro de usuario, ingresa los datos necesarios y lo registra en la base de datos.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol administrador y tener iniciada una sesión en el sistema.	
Entrada: • Ingresamos 'nbriones' en el campo de texto usuario	

• Ingresamos ‘Narcisa’ en el campo de texto nombre. • Ingresamos ‘Briones’ en el campo de texto apellido. • Ingresamos ‘nbriones@espol.edu.ec’. • Añadimos el rol secretaria a la lista de roles asignados. • Habilitamos la casilla de verificación Activar. • Pulsamos el botón ´Guardar´.
Resultado Esperado Un mensaje de notificación se muestra y se registra el nuevo usuario en la base de datos.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.30 Caso de prueba Registro de usuario exitoso.

Funcionalidad: Registro de materias	
Código: CP-AD-003	**Nombre:** Registro de materia exitoso
Descripción: Un usuario con rol administrador accede al formulario de Registro de una materia, ingresa los datos necesarios y la registra en la base de datos.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol administrador y tener iniciada una sesión en el sistema.	
Entrada: • Ingresamos ‘Programación Orientada a Objetos’ en el campo de texto nombre. • Ingresamos ‘FIEC04622’ en el campo de texto código. • Habilitamos la casilla de verificación Activar. • Pulsamos el botón ´Guardar´.	
Resultado Esperado Un mensaje de notificación se muestra y se registra la nueva materia en la base de datos.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.31 Caso de prueba Registro de materia exitoso.

Funcionalidad: Configuración de Parámetros	
Código: CP-AD-004	**Nombre:** Configuración de Decano exitosa
Descripción: Un usuario con rol administrador accede al formulario de Configuración de parámetros, selecciona el nombre del nuevo Decano y lo registra en la base de datos.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol administrador y tener iniciada una sesión en el sistema.	
Entrada: • Pulsamos el botón seleccionar que está junto al campo de texto Decano. • Seleccionamos ´Ing. Gustavo Bermúdez´ de la lista de profesores. • Pulsamos el botón ´Guardar´.	
Resultado Esperado Un mensaje de notificación se muestra y se registra el nuevo Decano en la base de datos.	
Evaluación de la Prueba: Prueba superada con éxito.	

Tabla C.32 Caso de prueba Configuración de Parámetros - Configuración de Decano exitosa.

Funcionalidad: Configuración de Parámetros	
Código: CP-AD-004	**Nombre:** Configuración de Subdecano exitosa
Descripción: Un usuario con rol administrador accede al formulario de Configuración de parámetros, selecciona el nombre del nuevo Subdecano y lo registra en la base de datos.	
Condiciones de Ejecución: - El usuario debe pertenecer al rol administrador y tener iniciada una sesión en el sistema.	
Entrada: • Pulsamos el botón seleccionar que está junto al campo de texto Subdecano.	

• Seleccionamos ´Ing. Holger Cevallos´ de la lista de profesores. • Pulsamos el botón ´Guardar´.
Resultado Esperado Un mensaje de notificación se muestra y se registra el nuevo Subdecano en la base de datos.
Evaluación de la Prueba: Prueba superada con éxito.

Tabla C.33 Caso de prueba Configuración de Parámetros - Configuración de Subdecano exitosa.

C.3. Análisis de los resultados obtenidos

Los resultados de las pruebas realizadas permitieron ajustar algunos detalles de la implementación como la validación y verificación de datos ingresados por el usuario. Como por ejemplo:

- Verificar el ingreso correcto de un correo electrónico durante el registro de un nuevo usuario.
- Validar que los tipos de datos ingresados sean los adecuados.
- Mejorar el texto de los errores para que sean más descriptivos y entendibles para el usuario.

En base al conjunto de pruebas que se llevaron a cabo y que se detallan en la sección anterior, se puede concluir que en general el funcionamiento del sistema Solicitudes FIEC es correcto.