

Desarrollo de un Sistema de Información para el Registro de Pasantías y Proyectos de Proyección Social en la Facultad de Ingeniería, Programa de Ingeniería de Sistemas en la Universidad de Cundinamarca sede Fusagasugá

Victor Acosta Espinoza, Mateo Rodriguez Ocampo

Abstract—The main objective of this degree project is the creation of an information system where all the internships and social projection projects that are carried out each semester in the Systems Engineering program at Fusagasugá headquarters are registered and controlled. The internships have the purpose of giving the student a practical knowledge and the current job panorama, besides being a vital requirement to choose the title in the Systems Engineering program.

In this way, it is proposed that, through the university platform, the user, whether student, teacher, administrative, etc. manage to manage, interact and access the diverse information of the internships and projects of social projection, which is important in the student's training process, thus an efficient control of the different activities performed inside and outside the University will be done.

Resumen—El presente proyecto de grado tiene como objetivo principal la creación de un sistema de información donde se registren y controlen todas las pasantías y proyectos de proyección social que se llevan a cabo cada semestre en el programa de Ingeniería de Sistemas en la sede de Fusagasugá. Las pasantías cuentan con el propósito de darle al estudiante un conocimiento práctico y el panorama laboral actual, además de ser un requisito vital para optar el título en el programa de Ingeniería de Sistemas.

De esta manera, se propone que, a través de la plataforma de la universidad, el usuario, bien sea, estudiante, profesor, administrativos, etc. logre gestionar, interactuar y acceder a la diversa información de las pasantías y proyectos de proyección social, la cual es importante en el proceso de formación del estudiante, así se hará un control eficiente de las diferentes actividades ejecutadas dentro y fuera de la Universidad.

I. INTRODUCCION

DADA la inexistencia de una tecnología que permita, en el programa de Ingeniería de Sistemas de la facultad de ingeniería de la Universidad de Cundinamarca, Sede Fusagasugá, un control y seguimiento adecuado, ordenado y sistematizado basado en las TIC's de las pasantías y proyectos

de proyección social realizadas por estudiantes, profesores o terceros, se determina la creación de un software que pueda llevar a cabo esta actividad, facilitándole al usuario la supervisión de los procesos, actividades, normativas, evaluación de las pasantías por parte de los estudiantes y las empresas, soportes, entre otras características propias de las pasantías y los proyectos de proyección social.

El ejercicio del estudiante dentro de los lineamientos de la Universidad de Cundinamarca en estos espacios (pasantías y proyectos de proyección social) es de una importancia trascendental, debido a que el desenvolvimiento adecuado equivale a la posibilidad de efectuar correctamente un puesto de trabajo.

Con la puesta en marcha del presente proyecto se busca evitar el papeleo rutinario, facilitar al usuario el corresponder a los requisitos y actividades con evidencias y/o documentos físicos, llevar un seguimiento de cada tarea, proceso, cometido, y su respectiva calificación.

II. PLANTEAMIENTO DEL PROBLEMA

Descripción del Problema:

La facultad de ingeniería, programa de ingeniería de sistemas de la Universidad de Cundinamarca, sede Fusagasugá no cuenta con un sistema de información que le permita llevar un registro adecuado de las diferentes pasantías y proyectos de proyección social que se ejecutan por parte de los estudiantes, con lo único que se cuenta es con una documentación de los diferentes proyectos que se ejecutaron en donde no se puede apreciar de forma precisa cuál fue el alcance de cada uno. Además de esto no se tiene un orden adecuado ni un lugar específico para resguardar la información, haciendo que la información sea muy vaga a la hora de consultar sobre el proceso que se lleva en cada proyecto.

Cada proyecto de pasantías que se ejecuta tiene una fecha de inicio y una de fin, pero no tiene un registro específico de las actividades que se realizaron ni del tiempo que se invierte diariamente en cada proyecto, además de esto no se puede evidenciar de forma detallada cual fue la metodología que se usó.

Como aspecto más importante es que estos pocos documentos que exigen como registro de que se realizaron estos proyectos, se encuentran expuestos a cualquier tipo de riesgo, ya que no tienen ningún tipo de seguridad al encontrarse en una carpeta en un estante como es el caso de la facultad de ingeniería, aunque también se encuentra información en los informes finales de los docentes responsables y se envía información a la oficina de interacción universitaria, ya sea en un cd o por correo.

Formulación del Problema

De acuerdo con las problemáticas mencionadas anteriormente, nace la siguiente pregunta de investigación. ¿Los módulos construidos para el registro y monitoreo de pasantías y proyectos de proyección social en el programa de ingeniería de sistemas de la facultad de ingeniería de la Universidad de Cundinamarca en la sede de Fusagasugá suplirán las necesidades establecidas?

Delimitación del Proyecto

La ejecución de este proyecto abarca la creación de un piloto el cual será un modelo de trabajo colaborativo asistido por computadora enfocado en la elaboración y control de pasantías y proyectos de proyección social, el cual es el punto de partida para el diseño y desarrollo del módulo Web.

En el diseño y desarrollo del módulo se considerarán aspectos relevantes para facilitar la coordinación y monitoreo de los diferentes proyectos de pasantía y proyección social entre los asesores, administrativos y demás personas que intervengan activamente en estos procesos, se utilizarán reportes para llevar un registro adecuado de las actividades que se encuentran activas en la Facultad de Ingeniería y del mismo modo saber que estudiantes están a cargo o participan de las mismas, toda esta recolección de información o requerimientos fue posible a través de entrevistas y consultas en el sistema de gestión de calidad donde se encuentran los formatos necesarios para los proyectos de pasantías y proyección social.

Para realizar el monitoreo y consignación de los diferentes documentos que se van creando en el transcurso del desarrollo de cada proyecto de pasantía y proyección social, se contempla que en la base de datos se creará una tabla en la cual se consignaran los diferentes archivos para su manejo, llevando consigo un control de fecha, descripción y usuario que lo adjunto o modifíco.

También se planea realizar un módulo información acerca de las diferentes pasantías y proyectos de proyección social por medio de informes en diferentes formatos tales como: html, pdf, docx, xmls y pptx. Esta información tendrá un sistema de búsqueda por medio de filtrados, para facilitar la usabilidad del aplicativo.

Para realizar el diseño y desarrollo del aplicativo utilizaremos la metodología de desarrollo XP y el motor de base

de datos Oracle, ya que estos hacen parte de los requisitos que la oficina de desarrollo de software tiene para llevar a cabo un nuevo módulo de desarrollo académico, la metodología y todo el proceso de desarrollo se encuentran consignados en el sistema de gestión de calidad, en el macroproceso de apoyo, en el proceso de sistemas y tecnología, con el código ASII008 y nombre INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL.

III. JUSTIFICACIÓN

El objeto del proyecto es crear un sistema de información para llevar los registros de cada pasantía que se realizaban en la Universidad de Cundinamarca, Facultad de Ingeniería, programa de Ingeniería de Sistemas, sede Fusagasugá, así mismo tener acceso a la información con claridad y facilidad.

Se creará una base de datos para poder almacenar toda la información pertinente de los proyectos de pasantías y proyección social que se llevan a cabo, en donde se encontraran datos de los estudiantes que realizaron cada proyecto, los profesores que intervinieron como directores de estos mismos, avances de los procesos que se realizan y el estado inicial y final de cada uno. Adicionalmente es necesario consignar toda la evidencia para poder llevar un registro detallado de las lecciones aprendidas.

También se creará un módulo que permita interactuar al usuario con la información almacenada en la base de datos, de igual manera poder tener fácil acceso a la información histórica de todos los proyectos desarrollados por año, semestre, en que se llevan a cabo los proyectos, entre otras formas de búsqueda específica de la información.

IV. OBJETIVOS

Objetivo General:

- Desarrollar un sistema de información para el registro de pasantías y proyectos de proyección social en la Universidad de Cundinamarca, facultad de Ingeniería, programa de Ingeniería de Sistemas, sede Fusagasugá.

Objetivos Específicos:

- Analizar la situación actual del sistema de registro de pasantías y proyectos de proyección social en la Universidad de Cundinamarca, facultad de Ingeniería, sede Fusagasugá.
- Proponer una herramienta para el manejo de los registros de pasantías en la Universidad de Cundinamarca, facultad de Ingeniería, programa de Ingeniería de Sistemas, sede Fusagasugá.
- Brindar una alternativa para el manejo de los registros del formato de proyectos de proyección social en la Universidad de Cundinamarca, facultad de Ingeniería, programa de Ingeniería de Sistemas, sede Fusagasugá.

V. METODOLOGIA

Xtreme Programming:

La metodología utilizada para este proyecto fue la de Xtreme

Programming la cual es un enfoque de la ingeniería de software formulado por Beck (1999), autor del primer libro sobre la materia, Extreme Programming Explained: Embrace Change.

Para el desarrollo de esta metodología se adoptó el modelo que está plasmado en el Instructivo Desarrollo de Software Institucional, el cual se encuentra en el proceso de gestión de sistemas y tecnología que hace parte del macroproceso de apoyo de la Universidad de Cundinamarca con el código de formato ASII008. Se siguió este instructivo ya que era un requisito fundamental para la Dirección de Sistemas y Tecnología, para que el software fuese compatible con la plataforma de la Universidad de Cundinamarca a la hora de su incorporación en la misma, además de lo anterior este proyecto se desarrolló en dos etapas, siendo la primera pasantías y la segunda proyección social.

A. Fase Planeación.

El primer paso que se realizó fue el levantamiento de requerimientos con los ingenieros Jorge Alfredo Mayorga Cárdenas y Diana Milena Rey Gutierrez los cuales se establecieron como funcionales:

Requerimientos Funcionales:

- El sistema debe garantizar el acceso en cualquier momento al usuario y durante el tiempo que dure la cesión.
- El sistema debe avalar la correcta importación y exportación de archivos.
- El sistema debe registrar la información adecuada de cada pasantía o proyecto de proyección social suministrada por él usuario.
- El sistema debe permitir al usuario buscar y consultar la información pertinente de cada pasantía o proyecto de proyección social.
- El sistema controlará el acceso y lo permitirá solamente a usuarios autorizados.
- El sistema hará los correspondientes registros de la evaluación del pasante y de la empresa.

En esta reunión también se evidencio que la fase de proyección social se debía delimitar solo a la sistematización de tres formatos específicos los cuales son MIUF033, MIUF015 y MIUF021, los cuales se encuentran en el macroproceso misional del proceso de gestión interacción universitaria.

Lo anterior debido a que proyección social maneja una serie de presupuestos económicos que deben ser aprobados por otras dependencias, es por esto por lo que junto a los ingenieros Alfredo Mayorga y Diana Rey se toma la decisión de no manipular las bases de datos destinadas al manejo financiero de la Universidad de Cundinamarca.

Roles del Sistema:

Tabla 1:
Roles del Sistema Funcionalidades de Pasantías

ROL	ACTIVIDAD
Director Pasantías	<ul style="list-style-type: none"> • Registro de pasantía • Consultas • Seguimiento
Coordinador interacción universitaria	<ul style="list-style-type: none"> • Consultas de pasantías y evaluaciones del pasante y empresa
Estudiante	<ul style="list-style-type: none"> • Evaluación de la empresa
Asesor Empresa	<ul style="list-style-type: none"> • Evaluación del pasante

Tabla 2:
Roles del Sistema Funcionalidades de Proyección Social

ROL	ACTIVIDAD
Docente Director de Proyecto	<ul style="list-style-type: none"> • Registro de formularios • Evaluación del proyecto

Historias de Usuario:

Se hizo uso de las historias usuario como instrumento para levantamiento de requisitos del software, ya que estas, se realizan con un lenguaje no técnico haciéndolas entendibles para cualquier persona, esto permite que tanto como el cliente como el equipo desarrollador entiendan fácilmente los requisitos implícitos en estas, además de incorporar características complementarias como los criterios mínimos de aceptación y el nivel de importancia que estas tiene dentro del proyecto, esto permite que el equipo desarrollador conozca de los criterios de aceptación plasmados por el cliente para este requisito y la importancia que este tiene a nivel de realización del proyecto, y de esta forma poder delegar los recursos necesarios para la realización óptima de este requisito.

Tabla 3:
Historias de Usuario Pasantías

HISTORIA	NOMBRE	PRIORIDAD
1	Cargar los convenios con NIT de la empresa, nombre, estado, fecha de inicio y fecha de finalización.	Alto
2	Registrar Pasantía.	Alto
3	Consulta Pasante-Pasantía.	Medio
4	Seguimiento Pasantía.	Alto
5	Consulta Convenio.	Medio
6	Consulta Empresa.	Medio
7	Consulta Pasantía.	Medio
8	Evaluación Pasantías.	Medio

Tabla 4:
Historias de Usuario Proyección Social

HISTORIA	NOMBRE	PRIORIDAD
9	Creación de propuesta campo de aprendizaje en proyección social.	Medio
10	Informe para eventos.	Medio
11	Evaluación de eventos.	Medio

Iteraciones:

Ya identificados y ordenados los requisitos continuamos con la creación del cronograma, en este se dividen las tareas a realizar llamadas iteraciones en la cuales se establecen las actividades a realizar y el tiempo estimado que están tendrán para su realización, como recomendación se expresa que el tiempo establecido no supere las cuatro semanas ya que se estima que este tiempo ya se puede presentar un incremento en la realización del proyecto debemos de tener en cuenta que con la terminación de cada iteración se debe de obtener una parte del proyecto ya funcional la cual se presentará al cliente (facultad de ingeniería de universidad de Cundinamarca sede Fusagasugá) como avance en el desarrollo del proyecto.

Tabla 16:
Velocidad del Proyecto

ITERACIONES	DÍAS	HORAS DIARIAS	HISTORIAS DE USUARIO
Iteración 1, 2 y 3	8	8	3
Iteración 4, 5 y 6	8	8	3
Iteración 7, 8 y 9	8	8	3
Iteración 10 y 11	8	8	2

Las reuniones con el cliente fueron en promedio cada dos semanas, estas con fin de mostrar al cliente (facultad de ingeniería de universidad de Cundinamarca sede Fusagasugá) los avances incrementales que ha tenido el software y a su vez retroalimentarse de las opiniones del cliente en base al avance presentado del proyecto.

B. Fase de Diseño

Lo primero fue realizar el diseño del modelo de la base de datos por medio de la información levantada a través de la historia de usuario y juntamente con los documentos proporcionados por el cliente (facultad de ingeniería de universidad de Cundinamarca sede Fusagasugá), entre los cuales se encuentran los formatos con código EPIF001, MEXF037 y MEXF038 para los procesos de pasantías, por otro lado, en la fase de proyección social, como ya se había mencionado en la etapa de planeación se usaron los formatos MIUF033, MIUF015 y MIUF021.

Iniciamos creando el modelo entidad relación de la base de datos en el cual se identifica cada una de las entidades que tendrá el modelo con los correspondientes atributos de cada entidad y las relaciones entre las entidades. (Ver Manual Técnico).

Durante la elaboración del modelo de entidad relación se realizó una reunión con los ingenieros Alfredo Mayorga y Diana Rey en donde se nos comunicó que gracias a que la Universidad de Cundinamarca está realizando el proceso de acreditación de alta calidad, muchos de sus procesos están siendo actualizados entre los que se encuentran los procesos de proyección social, educación continuada y convenios, es por esto que los formatos que existen actualmente están en proceso de modificación, por esto se recurrió a la realización de entrevistas con los diferentes directores que están actualizando los procesos, dichas entrevistas fueron realizadas con la compañía del ingeniero Alfredo Mayorga quien posteriormente sería el encargado de dar el aval al diseño que se presentó de la base de datos.

Como caso particular los procesos correspondientes a la realización de los convenios con la universidad tuvo un gran cambio, puesto que en un principio cada facultad tenía el manejo de sus convenios y los controlaba con total autonomía, con la actualización de este proceso, los convenios ya no serían distribuidos por facultad sino por sede, seccional o extensión lo cual implica que un convenio puede ser usado por cualquier facultad de la sede, seccional o extensión de la Universidad de Cundinamarca en donde se esté celebrando actualmente, es decir que los convenios de la sede Fusagasugá no los podrá usar la sede de Girardot ni ninguna otra sede, seccional o extensión.

Adicional a los procesos anteriormente mencionados que están en proceso de actualización hay que resaltar de la Dirección de Sistemas y Tecnología está actualizando sus procesos de igual manera, empezando con el cambio de formato de instructivo de desarrollo a la delimitación de políticas de desarrollo de software académico, el cual es el que se usó para el desarrollo de este proyecto.

Ya realizado el modelo entidad relación continuamos con la construcción del modelo relacional en el cual identificamos las llaves primarias y foranas de las tablas y las tablas que son

resultado de la relación de las entidades.

Para la creación de la base de datos se utilizó el asistente de interfaz gráfica SQL Developer, asignando como nombre a la base de datos “PASANTIAS”

Creación de interfaces de usuario:

En la creación de las diferentes funcionalidades o vistas se uso una plantilla que el área de desarrollo de software nos facilitó, la cual estaba en blanco y lista para empezar a crear cada funcionalidad, estas vista se trabajan por medio de jsp en los cuales se incorpora el código html para la creación de todos los aspectos gráficos, los cuales fueron trabajos con el framework de diseño Bootstrap, este framework divide la pantalla en 12 columnas y a su vez cada columna se podía dividir en 12 nuevamente, esto ya que Bootstrap posiciona los elementos dependiendo la cantidad de columnas que se desee usar, además que Bootstrap tiene un diseño gráfico de elementos que facilita su uso ya que solo hay que llamarlos y ellos ya tienen su estilo definido y no hay que configurarlos desde cero, como cuenta ya con su propia configuración facilitan el trabajo aún más puesto que se adaptan a cualquier pantalla de forma automática.

Figura 1. Interfaz Gráfica Funcionalidades

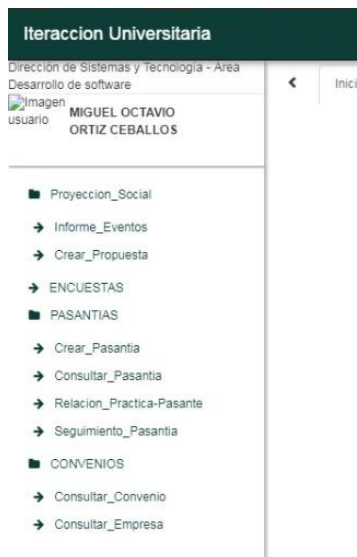
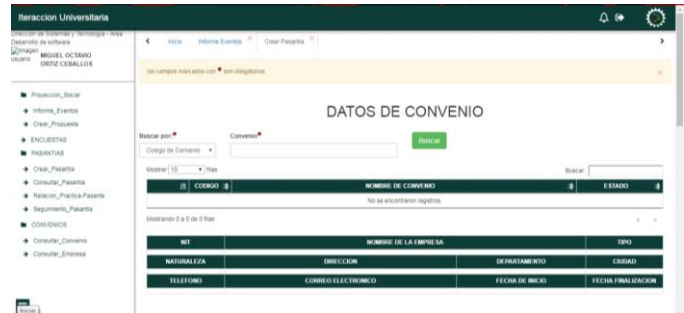


Figura 2. Ejemplo Interfaz Gráfica Proyección Social



Figura 3. Ejemplo Interfaz Gráfica Pasantías



El equipo desarrollador solo se enfocó en la solución de los requisitos actuales por iteración esto debido a que la metodología nos propone solucionar solo los inconvenientes que se van presentando y que son directamente relacionados con el requisito ya que el pensar en solucionar error o problemas que no han surgido en el momento provoca el gasto de tiempo y recursos en algo que no ha sucedido y que probablemente no se vaya a utilizar.

C. Fase Desarrollo

Aunque la metodología sugiere que el cliente este siempre el desarrollo del proyecto en este caso se modificó un poco este aspecto de la metodología debido a que el cliente no contaba con el tiempo, para estar siempre presente en el desarrollo se acordó el mostrar avances en periodos de tiempo de 2 semanas en donde ya se tendrían resultado de las iteraciones programadas para este periodo de tiempo.

Tras la terminación de cada historia de usuario esta iba siendo integrada al proyecto cumpliendo con una integración frecuente y un avance incremental de este.

Como también se trabajó apoyándose en frameworks como JSF y BOOTSTRAP y conjuntamente implementando patrones de diseño como MVC (modelo – vista - controlador), DAO estos aportaron a la estandarización del código desde su arquitectura.

Utilización de patrones de diseño:

Para la creación de la aplicación se hizo uso de los siguientes patrones de diseño esto con el fin de tener una arquitectura de software lo más sólida posible.

Patrón MVC (modelo visto controlador):

Como mencionamos anteriormente para la construcción del software se hizo uso del framework para JAVA JSF el cual trabaja con el patrón MVC (modelo visto controlador), este nos permite separar los componentes del aplicativo desde las vistas del usuario es decir las interfaces con las cuales este interactúa con el sistema, pasando por los controladores del aplicativo los cuales actúan como intermediarios entre la vista y los modelos quienes son los que se comunican con nuestra base de datos implementado la lógica de negocio.

- Vista = interfaces de usuarios
- Controladores = implementa lógica para comunicar las vistas con los modelos de datos
- Modelos = capa a donde se trabaja con los datos (base de datos select, insert, update, delete)

Este patrón lo podemos ver reflejado en la estructura del proyecto como se muestra en la siguiente imagen.

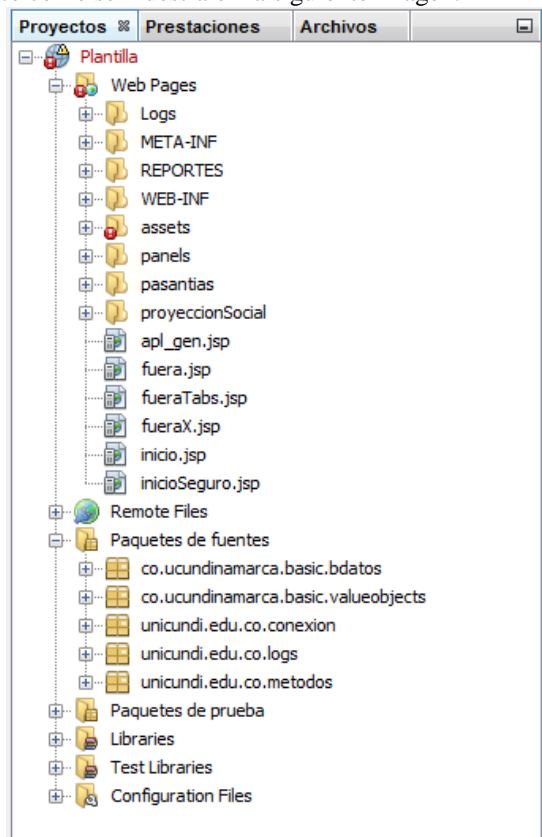


Figura 5. Estructura del Proyecto

Como podemos observar en el proyecto se encuentran separados los componentes, las vistas conformadas por la interfaz de usuarios creadas en el html, los controladores en los paquetes conexión, logs y métodos programados en java y los modelos y lógica del negocio contenida en los paquetes bdatos y valueobjects los cuales realizan las operaciones sobre la base de datos.

Patrón value object:

Este patrón consiste en mapear las tablas correspondientes de nuestra base de datos en clases para la creación de objetos sobre

estas, esto con el fin de facilitar la creación y mantenimiento de las funciones de transacción que tenga el aplicativo sobre la base de datos, puesto que ya no se trabajarían con valores serados e independientemente si no en cambio la función podrá recibir todo el objeto el cual ya contiene la información necesaria para realizar la operación correspondientes sobre las tablas.

Este patrón lo vemos reflejado en el proyecto específicamente en los modelos estos se encuentran en el paquete bdatos, en este están el mapeo de cada una de las tablas utilizadas en el aplicativo convertidas en clases, como ejemplo tenemos la tabla convenios y su correspondiente clase en donde podemos observar que la clase contiene los atributos correspondientes a su tabla juntamente con sus métodos get y set de cada uno de los atributos para el acceso a estos.

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 EMPR_ID	NUMBER(30,0)	No	(null)	1	(null)
2 CONV_ID	NUMBER(30,0)	No	(null)	2	(null)
3 CONV_NOMBRE	VARCHAR2(100 CHAR)	No	(null)	3	(null)
4 CONV_ESTADO	VARCHAR2(15 CHAR)	No	(null)	4	(null)
5 CONV_FECHAINICIO	DATE	No	(null)	5	(null)
6 CONV_FECHAFIN	DATE	No	(null)	6	(null)
7 CONV_REGISTRADOPOR	VARCHAR2(30 CHAR)	No	(null)	7	(null)
8 CONV_FECHACAMBIO	DATE	No	(null)	8	(null)
9 CONV_CODIGO	NUMBER(30,0)	No	(null)	9	(null)

Figura 6. Tabla de Convenios

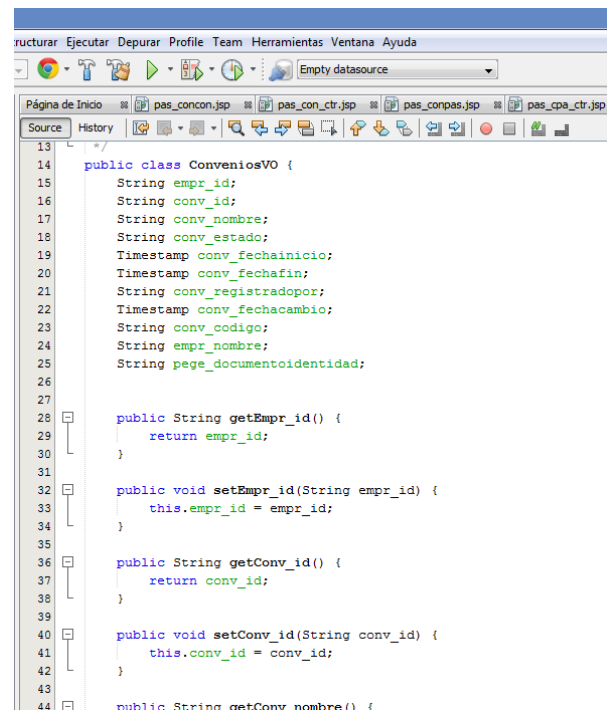


Figura 7. Clase ConveniosVO

Patrón DAO (Data Access object):

Este patrón trabaja sobre la accesibilidad de los datos en este caso como el aplicativo se conecta con la base de datos, su uso lo podemos ver reflejado en la creación del archivo DAO el cual es el encargado de realizar las funciones de abrir la conexión y cerrar la conexión con la base de datos especificando los parámetros para realizar estas tareas

El objetivo principal de la utilización del patrón DAO es

brindar al aplicativo la flexibilidad de desacoplar la conectividad de los datos de la lógica del negocio

Si cambiamos el nombre de base de datos o alguno de los atributos necesarios para la conexión como usuario o contraseña del motor base de datos solo con hacer los cambios respectivos en el archivo DAO el aplicativo seguirá con su funcionamiento normal, en caso contrario que este no estuviese aplicado se tendría que realizar este cambio en cada uno de los códigos que realizan procesos de transacción con la base de datos volviendo más engorroso el proceso de creación y mantenimiento del sistema.

D. Fase Pruebas

Pruebas de Aceptación:

La primera fase de pruebas fue realizada entre los codificadores, para esto se organizaron de forma tal, que cuando se terminaba una funcionalidad del sistema de información se evaluaba con la inserción de datos y si la funcionalidad no presentaba ningún problema, a su vez se revisaba que los datos se guardaran de forma adecuada en el sistema y que a su vez la funcionalidad cargara los datos de forma correcta a través del aplicativo en la base de datos.

Estas pruebas de aceptación son realizadas con el usuario en las cuales, este evalúa el software según los criterios de aceptación estipulada para cada historia de usuario.

Para la realización de estas pruebas el tester junto con usuario crean casos de prueba en los cuales se estipulan las entradas y salidas que deberá de tener el módulo evaluado.

A continuación, observaremos la aplicación de la prueba de aceptación sobre el módulo de Crear Pasantía con su correspondiente caso de prueba.

Tabla 17:
Ejemplo Caso de Prueba

Caso de prueba	
Nombre de la prueba	Crear Pasantía
Numero de referencia	1
Historia de usuario	2
Iteración	1
Funcionalidad crítica	Si
¿Qué se testeará?	Prueba interfaz de la creación de pasantía, validación de información e inserción en la base de datos
Categoría	Gestión de pasantías
Prerrequisitos	Convenio, estudiante y asesores vigentes y registrados.

Tabla 18:
Pasos y acciones de los casos de prueba Crear Pasantía

D Pasantía	D Pasante	D Asesor Interno	D Asesor Externo	resultado
Vacío	Lleno	Lleno	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Vacío	Lleno	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Vacío	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Lleno	Vacío	No permite la inserción y señala que falta completar o diligenciar
Vacío un solo campo obligatorio	Lleno	Lleno	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Vacío un solo campo obligatorio	Lleno	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Vacío un solo campo obligatorio	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Lleno	Vacío un solo campo obligatorio	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Lleno	Lleno	Permite la inserción en la base de datos

Luego se realizaron pruebas unitarias, las cuales son pruebas realizadas por los desarrolladores sobre el código estas pruebas se realizan con el fin de obtener un código lo más limpio posible de errores.

Pruebas Unitarias

Las pruebas unitarias en proyecto fueron aplicadas a través de la utilización del framework de pruebas JUNIT este es una herramienta que permite la realización de pruebas del código sobre proyecto creados en java.

A continuación, mostraremos los pasos correspondientes para la realización de estas pruebas sobre la conexión de la aplicación a la base de datos.

1. Lo primero es agregar al proyecto la librería JUNIT la cual nos permitirá la aplicación de las pruebas. En este caso utilizaremos JUNIT en su versión 4.12 este archivo .JAR lo podremos encontrar en la página oficial del proyecto JUNIT.
2. Crearemos una nueva clase de tipo JUnit Test con el nombre "Pruebas_conexion_base_de_datos" y especificamos el paquete con el nombre pruebas.
3. Escribiremos el código correspondiente a la prueba, como la prueba se realizará sobre la conexión de la base de datos esta la realizaremos sobre el código escrito en la clase ConexionExtencionUdec la cual es la encargada de definir los parámetros de conexión sobre la base de datos como el controlador de conexión, el nombre, la base de datos, usuario y contraseña definidos. La prueba específicamente se realizará sobre los métodos de SetCon, GetCon y Close entre la aplicación y la base de datos.

```

45  @Test
46  public void testSetCon() throws Exception {
47      System.out.println("testCon");
48      Connection con = null;
49      ConexionExtencionUdec instance = new ConexionExtencionUdec();
50      instance.setCon(con);
51      // TODO review the generated test code and remove the default call to fail.
52      fail("The test case is a prototype.");
53  }
54
55  /**
56   * Test of getCon method, of class ConexionExtencionUdec.
57   * @throws java.lang.Exception
58   */
59  @Test
60  public void testGetCon() throws Exception {
61      System.out.println("getCon");
62      ConexionExtencionUdec instance = new ConexionExtencionUdec();
63      Connection expectedResult = null;
64      Connection result = instance.getCon();
65      assertEquals(expectedResult, result);
66      // TODO review the generated test code and remove the default call to fail.
67      fail("The test case is a prototype.");
68  }
69
70  /**
71   * Test of close method, of class ConexionExtencionUdec.
72   * @throws java.lang.Exception
73   */
74  @Test
75  public void testClose() throws Exception {
76      System.out.println("close");
77      ConexionExtencionUdec instance = new ConexionExtencionUdec();
78      instance.close();
79      // TODO review the generated test code and remove the default call to fail.
80      fail("The test case is a prototype.");
81  }
82
83  }

```

Figura 11. Código de Prueba Unitaria

Como observamos los métodos de prueba se colocan anticipados de la anotación @test esto indica al framework que el método a continuación es la ejecución de una prueba en el cual dentro de este indicamos el método a ejecutar de la clase a la que realizaremos la prueba y el resultado esperado, de esta manera sabremos si la prueba realizada finalizo correctamente o si el código evaluado contiene un error y no devuelve el valor esperado.

4. Ejecutamos la prueba, esta se realiza corriendo el archivo que ejecuta la prueba
5. Una vez ejecutado el archivo este mostrara los resultados obtenidos de la ejecución de la prueba.

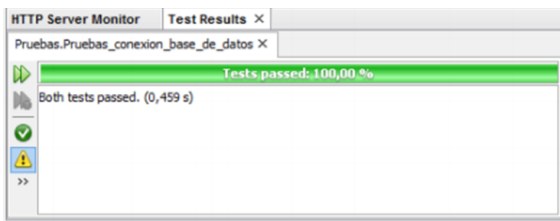


Figura 12. Resultado de la Prueba

Como podemos observar la prueba fue realizada correctamente ya que nos muestra un porcentaje de 100% esto indica que el código ejecutado de los métodos de la clase ConexionExtencionUdec en este caso retornaron los valores esperados, es decir el código cumple correctamente con su propósito.

La segunda fase de pruebas fue realizada por la Dirección de Sistemas y Tecnología de la Universidad de Cundinamarca, quien basado en el instructivo de desarrollo realizo pruebas a la base de datos, pruebas de diseño, pruebas de navegabilidad y por último pruebas con el usuario, todas estas pruebas tienen su respectiva metodología, la cual se puede consultar en el formato ASII008.

Implementación del Sistema de Información:

Para la implementación del sistema, se realizó entrega a el área Sistema de información un CD con la información necesaria para que el sistema fuese publicado en el portal web de la Universidad de Cundinamarca, además de esto el equipo desarrollador se comprometió a realizar cualquier tipo de modificación necesaria durante su proceso de implantación, para así de esta forma garantizar el correcto funcionamiento del módulo entregado.

VI. CONCLUSIONES

En este trabajo se ha descrito un procedimiento de modelación que ha consistido en la captura, análisis y representación de los aspectos y características más significativas de los procesos de pasantías y proyección social.

Sin duda los factores relacionados con la organización son los más trascendentes para el éxito o fracaso de un diseño y desarrollos de un aplicativo. El profundo conocimiento de la organización y del entendimiento de cuán adecuada es la implantación de un aplicativo para apoyar los procesos.

Los factores planteados en la presente investigación son el resultado del análisis del estudio exploratorio y recopilaciones bibliográficas. Contrastadas en el diseño y desarrollo de un Aplicativo para los proyectos de proyección social y pasantías de la universidad de Cundinamarca. Sin embargo, debe tenerse en cuenta que se ofrece como una reseña para abordar proyectos con características similares.

Es por lo anterior que este proyecto presenta un trabajo conjunto con la Dirección de Sistemas y Tecnología en todo el proceso de desarrollo en donde se acogió a los integrantes del proyecto como si fueran un colaborador más en el desarrollo de software para la Universidad de Cundinamarca, esto logro que se delimitaran herramientas que se podrán compartir con cualquier estudiante que desee realizar un proyecto con la Universidad de Cundinamarca. Como recurso más importante es una plantilla para el desarrollo de módulos de software que permite que se pueda trabajar desde cualquier locación de trabajo por fuera de las instalaciones de la Universidad de Cundinamarca, esta plantilla permite que cualquier persona interesada pueda generar un software que sea compatible con el software usado por la Universidad de Cundinamarca, para que así de esta manera no se generen inconvenientes de compatibilidad como los que se evidenciaron en la implantación del software de CASA UDEC (Revisar Antecedentes).

Como parte del levantamiento de información, es fundamental la participación de los actores de los grupos de trabajo. Con el fin de conocer la forma en que se realizan las tareas antes de diseñar soluciones asistidas por computadora. De igual forma son los actores quienes ofrecen la mejor retroalimentación a fin de lograr un entendimiento de los procesos grupales.

Gracias a la participación activa de todos los actores de los grupos de trabajo durante el desarrollo de las diferentes entrevistas en la fase de planeación, se vio la necesidad de la creación de módulos adicionales para que el sistema fuera más

eficiente, ya que los módulos de pasantías y proyección social hacen parte de un macroproceso que se llama interacción universitaria, el cual comprende temas adicionales como lo son los convenios que la Universidad de Cundinamarca realiza con las diferentes empresas para la realización de pasantías.

Es por lo anterior que se recomienda desarrollar un módulo para la creación y gestión de los convenios donde se pueda tener un control adecuado de estos procesos, y a su vez ayude a que los módulos de pasantías y proyección social sean más eficientes, para esto se presenta el diagrama de flujo del proceso actual correspondiente al proceso de legalización de convenios en la Universidad de Cundinamarca.

REFERENCIAS

- [1] ACM SIGCHI. (1992). ACM SIGCHI Curricula for Human Computer Interaction. Curriculum Development Group, Pag: 200-203.
- [2] ALAN, S. R. (2002). *Workgroup computing: workflow, groupware and messaging*. London: Publicaciones Adventure Works.
- [3] BECK, K. (02 de 03 de 1999). *Extreme Programming Explained., de computerworld: www.computerworld.com/softwaretopics/software/appdev/story/0,10801,66192,00.html*
- [4] Boman, M., Bubenko, J. A., Johannesson, P., & Wangler, B. (1997). *Conceptual Modelling. USA: Prentice Hall Series in Computer Science*.
- [5] CHAFFNEY, D. (1998). Groupware, workflow and Intranets: Reengineering the Enterprise with Collaborative Software. Digital Press.
- [6] Checkland, P. (1998). *Sistemas, Información y Sistemas de Información*. Publicaciones Adventure Works.
- [7] DESANCTIS, G., & GALLUPE, B. (1987). A Foundation for the Study of Group Decision Support Systems. *Management Science*, 589-609.
- [8] DILLENBORUG, P. (1995). *The Evolution of Research on Collaborative Learning*. London: Spada.
- [9] DIX, A., FINLAY, J., ABOARD, G., & BEALE, R. (1998). *Human Computer Interaction*. Prentice Hall .
- [10] ELLIS, C. A., GIBBS, S. J., & REIN, G. L. (1991). *Groupware: Some Issues and Experiences*.
- [11] Communications of the ACM.
- [12] GARRIDO, J. L. (2003). *AMENITIES: Una Metodología para el desarrollo de Sistemas Cooperativos basada en Modelos de Comportamiento y Tarea*. (Tesis Doctoral). Granada: Publicaciones Universidad de Granada.
- [13] GONCALVES, A. (2010). *Beginning Java EE 6 Platform with GlassFish 3*. Apress.
- [14] GRUDIN, J. (1988). Why CSCW applications fail: problems in the design and evaluation of organization of organizational interfaces. *ACM Conference on Computer-Supported Cooperative Work* (págs. 85-95). ACM Press.
- [15] ISLA MONTES, J. L. (2007). *Modelado Conceptual de Sistemas Cooperativos en base a Patrones en AMENITIES* (Tesis doctoral). Granada: Publicaciones UNIVERSIDAD DE GRANADA.
- [16] JAKOB, N. (2000). *Usabilidad: Diseño de sitios Web*. Madrid: Prentice Hall.
- [17] JOHANSEN, R. (1988). *Groupware: Computer Support for Business Teams*. The Free Press. KAYE, B., & ROGERS, I. (1995). *Trabajo de Grupo en las Escuelas Secundaria*. México – Buenos Aires: Publicaciones "El Ateneo".
- [18] LAUDON, K. C., & LAUDON, J. P. (2002). *Sistemas de Información Gerencial. Organización y tecnología de la empresa conectada en red*. Prentice Hall: Pearson Educación.
- [19] LEE, C. (01 de 03 de 2005). Traducción The New Methodology ., de programacionextrema: <http://www.programacionextrema.org/>
- [20] LIU, K., SUN, L., & DIX, A. (2001). Norm-based agency for designing collaborative information systems. *Information Systems Journal* .
- [21] LYNCH, K. J., SNYDER, J. M., & VOLGEL, D. R. (1990). *The Arizona Analyst Information System: Supporting Collaborative Research on international Technological Trends*.
- [22] Conference on Multi-User Interfaces and Applications (págs. 40-50). Crete: Greece.
- [23] MABEL SOSA, R. Z. (2006). Modelando aspectos de grupo en entornos colaborativos para proyectos de investigación. *Revista de Informática Educativa y Medios Audiovisuales*, Pag: 22-31.
- [24] MALON, T. W., & CROWSTON, K. (1990). What is Coordination Theory and How Can It Help Design Cooperative Work Systems. *Proceedings of the Conference on Computer Supported Cooperative Work (CSCW'90)* (págs. 357-370). New York: ACM Press.
- [25] MÁRQUEZ, J. (01 de 04 de 2003). *Arquitectura de Información*, de Jmarquez.com: <http://www.jmarquez.com/documentos.html>
- [26] MCGRATH, J. (1993). *Time, Interaction and Performance: a theory of groups* In: *Readings in Groupware and Computer-Supported Cooperative Work*. Baecker.
- [27] MUNKVONLD, B. E. (2002). *Organisational implementation of collaboration technologies: an integrative review*. Kristiansand: University College.

Desarrollo de un Sistema de Información para el Registro de Pasantías y Proyectos de Proyección Social en la Facultad de Ingeniería, Programa de Ingeniería de Sistemas en la Universidad de Cundinamarca, Sede

Fusagasugá

Víctor Acosta y Mateo Rodríguez

Universidad de Cundinamarca



Notas del autor

Víctor Acosta y Mateo Rodríguez, Facultad de Ingeniería, Universidad de Cundinamarca

Este proyecto ha sido financiado por los propios estudiantes

La correspondencia relacionada con este proyecto de grado debe ser dirigido a Ana Esperanza Merchán,

Universidad de Cundinamarca, Fusagasugá

Contacto: andres_acosta0810@hotmail.com, mateo_9_@hotmail.com

Dedicatorias

A mi madre, ejemplo de vida, esfuerzo y admiración, por ser el pilar fundamental en toda mi educación, tanto académicamente, como de la vida, por su incondicional apoyo perfectamente mantenido a través del tiempo.

Todo este trabajo ha sido gracias a ellos.

Víctor Acosta.

A mis padres por ser mi base esencial en todo, por brindarme mi educación, tanto académica, como de la vida, por su apoyo absoluto durante todos estos años. A mis amigos, Que nos apoyamos mutuamente en nuestra formación profesional y que hasta ahora, seguimos siendo amigos. Finalmente, a los maestros, aquellos que marcaron cada etapa de mi camino universitario, y que me ayudaron en asesorías y dudas presentadas en la elaboración de la tesis.

Mateo Rodríguez.

Tabla De Contenido

Resumen	9
Introducción	10
Planteamiento Del Problema	11
Descripción del Problema	11
Formulación del Problema	12
Antecedentes.	12
Delimitación Del Proyecto	17
Justificación	19
Objetivos.....	20
Objetivo General.	20
Objetivos Específicos	20
Marco Teórico.....	21
Sistema de información.	21
Apache Tomcat.	22
JasperReports.	23
JavaServer Pages	23
HTML	23
CSS	24
Jquery.....	24
Alertify.....	24
Ajax	25
Api.....	25
Bootstrap.....	25

Java.....	26
Xtreme Programming.....	26
Valores.....	27
Principios Básicos.....	29
Proceso Continuo En Lugar De Por Bloques.....	30
Oracle.....	31
Tecnologías Utilizadas.....	33
Metodología Del Proyecto.....	34
Fase Planeación.....	35
Requerimientos Funcionales:.....	35
Roles del Sistema.....	36
Historias de Usuario.....	37
Iteraciones.....	51
Fase de Diseño.....	53
Diseño base de datos.....	53
Creación de interfaces de usuario.....	55
Fase de Codificación.....	58
Patrón MVC (modelo vista controlador).....	58
Patrón value object.....	60
Patron DAO (Data Access object).....	62
Fase de Pruebas.....	64
Pruebas de Aceptación.....	64
Pruebas Unitarias.....	66
Implementación del Sistema de Información.....	71
Conclusiones y Recomendaciones.....	72

Referencias75

Anexos.....79

Lista de Tablas

Tabla 1:	36
Tabla 2:	36
Tabla 3:	37
Tabla 4:	38
Tabla 5:	39
Tabla 6:	40
Tabla 7:	41
Tabla 8:	42
Tabla 9:	43
Tabla 10:	44
Tabla 11:	45
Tabla 12:	47
Tabla 13:	48
Tabla 14:	50
Tabla 15:	51
Tabla 16:	52
Tabla 17:	64
Tabla 18:	65

Tabla de Figuras

Figura 1. Ejemplo de Interfaz Grafica	55
Figura 2. Modelo Vista Controlador	59
Figura 3. Estructura del Proyecto	60
Figura 4. Tabla de Convenios	61
Figura 5. Clase ConveniosVO.....	62
Figura 6. Librería JUnit 4.12.....	66
Figura 7. Selección de Archivo JUnit Test	67
Figura 8. Asignación de Nombre a la Prueba	67
Figura 9. Código de Prueba Unitaria	68
Figura 10. Resultado de la Prueba.....	69
Figura 11. Proceso de Convenios	74

Resumen

El presente proyecto de grado tiene como objetivo principal la creación de un sistema de información donde se registren y controlen todas las pasantías y proyectos de proyección social que se llevan a cabo cada semestre en el programa de Ingeniería de Sistemas en la sede de Fusagasugá. Las pasantías cuentan con el propósito de darle al estudiante un conocimiento práctico y el panorama laboral actual, además de ser un requisito vital para optar el título en el programa de Ingeniería de Sistemas.

De esta manera, se propone que, a través de la plataforma de la Universidad de Cundinamarca, el usuario, bien sea, estudiante, profesor, administrativos, etc. logre gestionar, interactuar y acceder a los módulos de las pasantías y proyectos de proyección social, la cual es importante en el proceso de formación del estudiante, así se hará un control eficiente de las diferentes actividades ejecutadas dentro y fuera de la Universidad.

Introducción

Dada la inexistencia de una tecnología que permita, en el programa de Ingeniería de Sistemas de la facultad de ingeniería de la Universidad de Cundinamarca, Sede Fusagasugá, un control y seguimiento adecuado, ordenado y sistematizado basado en las TIC's de las pasantías y proyectos de proyección social realizadas por estudiantes, profesores o terceros, se determina la creación de un software que pueda llevar a cabo esta actividad, facilitándole al usuario la supervisión de los procesos, actividades, normativas, evaluación de las pasantías por parte de los estudiantes y las empresas, soportes, entre otras características propias de las pasantías y los proyectos de proyección social.

El ejercicio del estudiante dentro de los lineamientos de la Universidad de Cundinamarca en estos espacios (pasantías y proyectos de proyección social) es de una importancia trascendental, debido a que el desenvolvimiento adecuado equivale a la posibilidad de efectuar correctamente un puesto de trabajo.

Con la puesta en marcha del presente proyecto se busca evitar el papeleo rutinario, facilitar al usuario el corresponder a los requisitos y actividades con evidencias y/o documentos físicos, llevar un seguimiento de cada tarea, proceso, cometido, y su respectiva calificación.

Planteamiento Del Problema

Descripción del Problema

La facultad de ingeniería, programa de ingeniería de sistemas de la Universidad de Cundinamarca, sede Fusagasugá no cuenta con un sistema de información que le permita llevar un registro adecuado de las diferentes pasantías y proyectos de proyección social que se ejecutan por parte de los estudiantes, con lo único que se cuenta es con una documentación de los diferentes proyectos que se ejecutaron en donde no se puede apreciar de forma precisa cuál fue el alcance de cada uno. Además de esto no se tiene un orden adecuado ni un lugar específico para resguardar la información, haciendo que la información sea muy vaga a la hora de consultar sobre el proceso que se lleva en cada proyecto.

Cada proyecto de pasantías que se ejecuta tiene una fecha de inicio y una de fin, pero no tiene un registro específico de las actividades que se realizaron ni del tiempo que se invierte diariamente en cada proyecto, además de esto no se puede evidenciar de forma detallada cual fue la metodología que se usó.

Como aspecto más importante es que estos pocos documentos que exigen como registro de que se realizaron estos proyectos, se encuentran expuestos a cualquier tipo de riesgo, ya que no tienen ningún tipo de seguridad al encontrarse en una carpeta en un estante como es el caso de la facultad de ingeniería, aunque también se encuentra información en los informes finales de los docentes responsables y se envía información a la oficina de interacción universitaria, ya sea en un cd o por correo.

Formulación del Problema

De acuerdo con las problemáticas mencionadas anteriormente, nace la siguiente pregunta de investigación. ¿Los módulos construidos para el registro y monitoreo de pasantías y proyectos de proyección social en el programa de ingeniería de sistemas de la facultad de ingeniería de la Universidad de Cundinamarca en la sede de Fusagasugá suplirán las necesidades establecidas?

Antecedentes

La Universidad Técnica Estatal de Quevedo en cumplimiento de la Ley Orgánica de Educación superior en lo que respecta a los requisitos previos de la obtención del título, dispone que los estudiantes de pregrado cumplan con el desarrollo de prácticas pre profesionales con el objetivo de vincular a los estudiantes con la realidad laboral en las empresas u organizaciones de la región o el país, acorde a su perfil profesional demostrando en la práctica las capacidades adquiridas en el transcurso de la formación. Para el control de este proceso se asigna a un docente que realice funciones de: planificación, evaluación y seguimiento, del periodo de prácticas preprofesionales. (Mendoza Coello & Vicuña Pino, 2017).

En ese orden, se constituye el desarrollo de un sistema de gestión para la coordinación de prácticas preprofesionales de la UTEQ. La necesidad de crear un sistema de gestión para la coordinación de prácticas preprofesionales surge debido a que estas tareas el docente coordinador las lleva a cabo mediante procesos tradicionales, asistiendo a los cursos o a secretaría en busca de información de los

estudiantes que no han realizado prácticas, llevando documentación física de los tramites, es decir de manera manual. Con el desarrollo de la aplicación se garantiza un mejor control de todas las actividades relacionadas al cumplimiento de este requisito de graduación por parte de los coordinadores de las prácticas preprofesionales. (Mendoza Coello & Vicuña Pino, 2017).

De esta manera, se solucionan inconvenientes como: la no disponibilidad de la información requerida por parte de los coordinadores de manera oportuna, las dificultades en el seguimiento de procesos involucrados en el desarrollo de prácticas, la mala organización y disponibilidad de las evidencias, el incumplimiento de tareas programadas, entre otros. (Mendoza Coello & Vicuña Pino, 2017).

La aplicación informática permitirá que los coordinadores de prácticas preprofesionales dispongan de una correcta, oportuna y actualizada información del proceso de prácticas preprofesionales de los estudiantes, así como también permitirá efectuar un correcto seguimiento de las evidencias necesarias tales como trámites reglamentarios, planes de prácticas, informes mensuales, e informes finales. Otro de los beneficios de contar con esta aplicación es que mediante la disponibilidad de la información relacionada con los procesos la Universidad Técnica Estatal de Quevedo de prácticas, se asegura el cumplimiento de las tareas programadas por los estudiantes y coordinadores. (Mendoza Coello & Vicuña Pino, 2017).

En la Universidad Estatal Península de Santa Elena existen las prácticas preprofesionales, las cuales le permiten al estudiante explorar el ámbito práctico-

laboral y son un requisito para obtener el título de tercer nivel, dada la importancia de estas prácticas, diferentes universidades evalúan y controlan a través de una plataforma tecnológica que facilita esta tarea. Para tal propósito se diseña una plataforma funcional para gestionar las actividades que el estudiante realiza dentro de la empresa, incluyendo la calificación, soporte y consulta de información. (Orozco Iguasnia & Flores Constante, 2016).

La implementación de tal sistema permite un mejor control en la calificación del estudiante durante sus prácticas en la empresa, además de que la documentación necesaria estará disponible en cualquier momento, evitando el archivo físico. Para tal proyecto se utilizó el IDE de desarrollo de Java en conjunto con el framework ZK, el gestor de librerías Maven, SQL server como el gestor de bases de datos, Apache Tomcat y Evolus Pencil. (Orozco Iguasnia & Flores Constante, 2016).

Además de todo lo anterior la Universidad de Cundinamarca en su área de desarrollo en los últimos años ha desarrollado los diferentes módulos de gestión para mejorar el acceso y manejo de la información dentro de la institución, entre los cuales tenemos:

- ACADEMUSOFT
- GESTASOFT
- HERMESOFT

Estos tres módulos fueron comprados para la gestión de los procesos Académicos, Colaborativos, Administrativos, Servicios, Virtual y Snies de la Universidad de Cundinamarca, son un completo sistema de información en Web

que facilita la administración de la información de forma segura y eficiente, ayudando a la Universidad de Cundinamarca, a llevar a cabo de forma exitosa, y rápida todos sus procesos misionales, estas plataformas son de un fácil manejo, intuitivo, y con la suficiente sintonía con los objetivos institucionales.

Además de esto el área de desarrollo cuenta con un software de enfermería para el control de la información de los pacientes y una plataforma Web interactiva para los mismos y sus cuidadores llamado Casa UDEC, el cual fue desarrollado por estudiantes de la Universidad de Cundinamarca mediante la figura de pasantías, aunque el software soluciono los problemas que se presentaban antes de su implantación, luego de cierto periodo de tiempo comenzó a generar una serie de conflictos, ya que la arquitectura de desarrollo que se utilizó no es la que usa la Universidad de Cundinamarca, evidenciándose problemas a la hora de usar ciertas librerías que no son compatibles con las que usa la Universidad de Cundinamarca, es por esto que al software no se le ha dado el uso adecuado generando así el desaprovechamiento del mismo.

La oficina de sistemas y tecnología se encuentra a cargo del ingeniero Edilson Martínez Clavijo, el cual determina que aplicativos deben tener prioridad para su desarrollo y coordina estos nuevos módulos con el ingeniero Alfredo Mayorga, quien es el encargado de la oficina de desarrollo de software, este a su vez tiene una serie de colaboradores los cuales se encargan de monitorear, codificar y actualizar los diferentes módulos que existen y a su vez son los encargados de la creación y puesta en marcha de nuevos módulos.

Anexo 1 - Estructura Organizacional de la Universidad de Cundinamarca



Delimitación Del Proyecto

La ejecución de este proyecto abarca la creación de un piloto el cual será un modelo de trabajo colaborativo asistido por computadora enfocado en la elaboración y control de pasantías y proyectos de proyección social, el cual es el punto de partida para el diseño y desarrollo del módulo Web.

En el diseño y desarrollo del módulo se considerarán aspectos relevantes para facilitar la coordinación y monitoreo de los diferentes proyectos de pasantía y proyección social entre los asesores, administrativos y demás personas que intervengan activamente en estos procesos, se utilizarán reportes para llevar un registro adecuado de las actividades que se encuentran activas en la Facultad de Ingeniería y del mismo modo saber que estudiantes están a cargo o participan de las mismas, toda esta recolección de información o requerimientos fue posible a través de entrevistas y consultas en el sistema de gestión de calidad donde se encuentran los formatos necesarios para los proyectos de pasantías y proyección social.

Para realizar el monitoreo y consignación de los diferentes documentos que se van creando en el transcurso del desarrollo de cada proyecto de pasantía y proyección social, se contempla que en la base de datos se creará una tabla en la cual se consignarán los diferentes archivos para su manejo, llevando consigo un control de fecha, descripción y usuario que lo adjunto o modifiqué.

También se planea realizar un módulo información acerca de las diferentes

pasantías y proyectos de proyección social por medio de informes en diferentes formatos tales como: html, pdf, docx, xmls y pptx. Esta información tendrá un sistema de búsqueda por medio de filtrados, para facilitar la usabilidad del aplicativo.

Para realizar el diseño y desarrollo del aplicativo utilizaremos la metodología de desarrollo XP y el motor de base de datos Oracle, ya que estos hacen parte de los requisitos que la oficina de desarrollo de software tiene para llevar a cabo un nuevo módulo de desarrollo académico, la metodología y todo el proceso de desarrollo se encuentran consignados en el sistema de gestión de calidad, en el macroproceso de apoyo, en el proceso de sistemas y tecnología, con el código ASII008 y nombre INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL.

Justificación

El objeto del proyecto es crear un sistema de información para llevar los registros de cada pasantía que se realizaban en la Universidad de Cundinamarca, Facultad de Ingeniería, programa de Ingeniería de Sistemas, sede Fusagasugá, así mismo tener acceso a la información con claridad y facilidad.

Se creará una base de datos para poder almacenar toda la información pertinente de los proyectos de pasantías y proyección social que se llevan a cabo, en donde se encontraran datos de los estudiantes que realizaron cada proyecto, los profesores que intervinieron como directores de estos mismos, avances de los procesos que se realizan y el estado inicial y final de cada uno. Adicionalmente es necesario consignar toda la evidencia para poder llevar un registro detallado de las lecciones aprendidas.

También se creará un módulo que permita interactuar al usuario con la información almacenada en la base de datos, de igual manera poder tener fácil acceso a la información histórica de todos los proyectos desarrollados por año, semestre, en que se llevan a cabo los proyectos, entre otras formas de búsqueda específica de la información.

Objetivos

Objetivo General.

- Desarrollar un sistema de información para el registro de pasantías y proyectos de proyección social en la Universidad de Cundinamarca, facultad de Ingeniería, programa de Ingeniería de Sistemas, sede Fusagasugá.

Objetivos Específicos

- Analizar la situación actual del sistema de registro de pasantías y proyectos de proyección social en la Universidad de Cundinamarca, facultad de Ingeniería, sede Fusagasugá.
- Proponer una herramienta para el manejo de los registros de pasantías en la Universidad de Cundinamarca, facultad de Ingeniería, programa de Ingeniería de Sistemas, sede Fusagasugá.
- Brindar una alternativa para el manejo de los registros del formato de proyectos de proyección social en la Universidad de Cundinamarca, facultad de Ingeniería, programa de Ingeniería de Sistemas, sede Fusagasugá.

Marco Teórico

Para el entendimiento de los sistemas de trabajo, la presente sección presenta una completa recopilación teórica de referencia para todo el proyecto. Comenzando con una ligera definición de los sistemas de información, después se examina el tema central de este documento, el “Desarrollo de software”, del cual se presentan definiciones, características, taxonomía, etc.

Sistema de información.

Puede definirse como un conjunto de componentes interrelacionados que permiten capturar, procesar, almacenar y distribuir la información para apoyar la toma de decisiones y el control en una institución.

Los sistemas de información sirven para:

- Un acceso rápido a determinada información y por ende mejora tanto en tiempos como en resultados el servicio a los usuarios.
- Motivar a todo tipo de funcionario de las compañías para requerimientos de cualquier índole con excelentes resultados.
- Generar información e indicadores los cuales permiten analizar, comparar estudiar para detectar fallas y así mismo tener el control del sistema.
- Da la posibilidad de planear, idear proyectos los cuales van a estar generados de un sistema de información que tiene unos elementos claros y en dado caso sustentados para prever cualquier tipo de requerimientos.

- Evita la pérdida de tiempo en la organización de la información ya que realizándola de forma manual se corre el riesgo de no dar la investigación correcta.
- Hay mayor interés en la creación de nuevos procesos de trabajo debido a la facilidad que brinda para la obtención y el procesamiento de información.
- Se hace más efectiva la comunicación entre procesos y por lo tanto entre grupos de trabajo, una comunicación de diferentes instancias con los mismos resultados ágiles y confiables.
- Organización de archivo automatizado, clasificado de interés general y particular, entre otras.

Un sistema de información es tan importante que además de llevar un control adecuado de todas las operaciones que allí se registren, conlleva a buen funcionamiento de cualquier proceso y por ende de una entidad por lo que genera buenos beneficios tanto para la empresa como para sus usuarios, sean estos internos o externos.

Apache Tomcat.

“Es un servidor web multiplataforma que funciona como contenedor de servlets y que se desarrolla bajo el proyecto denominado Jakarta perteneciente a la Apache Software Foundation bajo la licencia Apache 2.0 y que implementa las especificaciones de los servlets y de JavaServer Pages o JSP de Sun Microsystem. Dicho servidor es mantenido y desarrollado por miembros de la fundación y voluntarios independientes.” (APACHE SOFTWARE FOUNDATION, 2018).

JasperReports.

“La Biblioteca JasperReports es el motor de informes de código abierto más popular del mundo. Está totalmente escrito en Java y es capaz de utilizar datos procedentes de cualquier tipo de fuente de datos y producir documentos de píxeles perfectos que pueden ser vistos, impresos o exportados en una variedad de formatos de documento incluyendo HTML, PDF, Excel, OpenOffice y Word. JasperReports Library es una librería Java que ofrece una interfaz para el motor de informes de la librería JasperReports. Esta biblioteca es el motor en el centro del iReport Designer, Jaspersoft Studio y JasperReports Server. Además, está integrado en miles de aplicaciones en todo el mundo.” (TIBCO Software Inc., 2018).

JavaServer Pages

Permite desarrollar rápidamente y mantener fácilmente páginas web dinámicas y ricas en información que aprovechan los sistemas comerciales existentes. La tecnología JSP permite el desarrollo rápido de aplicaciones basadas en web que son independientes de la plataforma. La tecnología JSP separa la interfaz de usuario de la generación de contenido, lo que permite a los diseñadores cambiar el diseño general de la página sin alterar el contenido dinámico subyacente.

HTML

“HTML, que significa Lenguaje de Marcado para Hipertextos (HyperText Markup Language) es el elemento de construcción más básico de una página web y se usa para crear y representar visualmente una página web. Determina el

contenido de la página web, pero no su funcionalidad. Otras tecnologías distintas de HTML son usadas generalmente para describir la apariencia/presentación de una página web (CSS) o su funcionalidad (JavaScript). (Mozilla, 2018)

HTML le da "valor añadido" a un texto estándar en español. Hiper Texto se refiere a enlaces que conectan una página Web con otra, ya sea dentro de una página web o entre diferentes sitios web. los vínculos son un aspecto fundamental de la Web. HTML usa "markup" o marcado para anotar textos, imágenes, y otros contenidos que se muestran en el Navegador Web.” (Mozilla, 2018).

CSS

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

Jquery

jQuery es una biblioteca de JavaScript rápida, pequeña y rica en funciones. Hace cosas como el recorrido y manipulación de documentos HTML, manejo de eventos, animación, y Ajax mucho más simple con una API fácil de usar que funciona en una multitud de navegadores.

Alertify

Un marco de JavaScript para desarrollar diálogos y notificaciones del navegador.

Ajax

No es una tecnología por sí misma, es un término que describe un nuevo modo de utilizar conjuntamente varias tecnologías existentes. Esto incluye: HTML o XHTML, CSS, JavaScript, DOM, XML, XSLT, y el objeto XMLHttpRequest. Cuando estas tecnologías se combinan en un modelo AJAX, es posible lograr aplicaciones web capaces de actualizarse continuamente sin tener que volver a cargar la página completa. Esto crea aplicaciones más rápidas y con mejor respuesta a las acciones del usuario.

Api

La API de Java proporciona una interfaz de Java con un conjunto de métodos y clases de utilidad que se puede utilizar para escribir código Java.

Bootstrap.

“Creado originalmente por un diseñador y desarrollador en Twitter, Bootstrap se ha convertido en uno de los frameworks front-end y proyectos de código abierto más populares del mundo. (Equipo central Bootstrap, 2018).

Bootstrap fue creado en Twitter a mediados de 2010 por @mdo y @fat. Antes de ser un framework de código abierto, Bootstrap era conocido como Twitter Blueprint. A los pocos meses de desarrollo, Twitter celebró su primera Semana del Hack y el proyecto explotó cuando los desarrolladores de todos los niveles de habilidad saltaron sin ninguna orientación externa. Sirvió como guía de estilo para el desarrollo de herramientas internas en la compañía durante más de

un año antes de su lanzamiento al público, y continúa haciéndolo hoy en día.”
(Equipo central Bootstrap, 2018).

Java.

“Java es un lenguaje orientado a objetos de propósito general. Aunque Java comenzará a ser conocido como un lenguaje de programación de applets que se ejecutan en el entorno de un navegador web, se puede utilizar para construir cualquier tipo de proyecto. Su sintaxis es muy parecida a la de C y C++, pero hasta ahí llega el parecido. Java no es una evolución ni de C++ ni un C++ mejorado.”
(Belmonte Fernández, 2004).

Xtreme Programming.

La programación extrema o Xtreme Programming (XP) es un enfoque de la ingeniería de software formulado por (BECK, 1999). Es el más destacado de los procesos ágiles de desarrollo de software. Al igual que éstos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos (LEE,2005).

Se puede considerar la programación extrema como la adopción de las mejores metodologías de desarrollo de acuerdo con lo que se pretende llevar a

cabo con el proyecto, y aplicarlo de manera dinámica durante el ciclo de vida del software.

Valores.

Los Valores originales de la programación extrema son: simplicidad, comunicación, retroalimentación (*feedback*) y coraje (BECK, 1999) Los valores se detallan a continuación:

Simplicidad:

La simplicidad es la base de la programación extrema. Se simplifica el diseño para agilizar el desarrollo y facilitar el mantenimiento. Un diseño complejo del código junto a sucesivas modificaciones por parte de diferentes desarrolladores hace que la complejidad aumente exponencialmente. Para mantener la simplicidad es necesaria la refactorización del código, ésta es la manera de mantener el código simple a medida que crece. También se aplica la simplicidad en la documentación, de esta manera el código debe comentarse en su justa medida, intentando eso sí que el código esté autodocumentado. Para ello se deben elegir adecuadamente los nombres de las variables, métodos y clases. Los nombres largos no decrementan la eficiencia del código ni el tiempo de desarrollo gracias a las herramientas de autocompletado y refactorización que existen actualmente. Aplicando la simplicidad junto con la autoría colectiva del código y la programación por parejas se asegura que cuanto más grande se haga el proyecto, todo el equipo conocerá más y mejor el sistema completo.

Comunicación:

La comunicación se realiza de diferentes formas. Para los programadores el código comunica mejor cuanto más simple sea. Si el código es complejo hay

que esforzarse para hacerlo inteligible. El código autocomentado es más fiable que los comentarios ya que éstos últimos pronto quedan desfasados con el código a medida que es modificado. Debe comentarse sólo aquello que no va a variar, por ejemplo, el objetivo de una clase o la funcionalidad de un método. Las pruebas unitarias son otra forma de comunicación ya que describen el diseño de las clases y los métodos al mostrar ejemplos concretos de cómo utilizar su funcionalidad. Los programadores se comunican constantemente gracias a la programación por parejas. La comunicación con el cliente es fluida ya que el cliente forma parte del equipo de desarrollo. El cliente decide qué características tienen prioridad y siempre debe estar disponible para solucionar dudas.

Retroalimentación (feedback):

Al estar el cliente integrado en el proyecto, su opinión sobre el estado del proyecto se conoce en tiempo real. Al realizarse ciclos muy cortos tras los cuales se muestran resultados, se minimiza el tener que rehacer partes que no cumplen con los requisitos y ayuda a los programadores a centrarse en lo que es más importante. Considérense los problemas que derivan de tener ciclos muy largos. Meses de trabajo pueden tirarse por la borda debido a cambios en los criterios del cliente o malentendidos por parte del equipo de desarrollo. El código también es una fuente de retroalimentación gracias a las herramientas de desarrollo. Por ejemplo, las pruebas unitarias informan sobre el estado de salud del código. Ejecutar las pruebas unitarias frecuentemente permite descubrir fallos debidos a cambios recientes en el código.

Coraje o valentía:

Muchas de las prácticas implican valentía. Una de ellas es siempre diseñar

y programar para hoy y no para mañana. Esto es un esfuerzo para evitar empantanarse en el diseño y requerir demasiado tiempo y trabajo para implementar todo lo demás del proyecto. La valentía les permite a los desarrolladores que se sientan cómodos con reconstruir su código cuando sea necesario. Esto significa revisar el sistema existente y modificarlo si con ello los cambios futuros se implementaran más fácilmente.

Principios Básicos

Tenemos 12 principios básicos que se agrupan en 4 categorías distintas:

- Retroalimentación.
 - ✓ Principio de pruebas: lo primero que se debe hacer es establecer un período de pruebas de aceptación del programa, en el cual se definirán las entradas y salidas del sistema. Básicamente se define lo que debe hacer el software desarrollado. Como si fuese una caja negra.
 - ✓ Planificación: el cliente (o su representante) escribirá sus necesidades para definir concretamente las actividades que el sistema debe realizar. En esta fase se creará un documento que contendrá historias de usuario que forman el plan de liberación, el cual define los tiempos de entrega de la aplicación para poder recibir feedback por parte del cliente.
 - ✓ Cliente in-situ: el cliente (o su representante) deberá formar parte del equipo de desarrollo. Se le dará poder para determinar los requisitos de la aplicación, definir la funcionalidad y dar prioridad a determinadas cosas. Gracias a esto, habrá una fuerte interacción con los programadores, disminuyendo así el tiempo de comunicación y la

cantidad de documentación a redactar. El cliente estará con el equipo durante todo el proceso de desarrollo del proyecto.

- ✓ Pair-programming: este punto junto con el anterior son los más radicales de esta metodología. Consiste en escribir código en parejas compartiendo una sola máquina. Según los experimentos ya realizados sobre este método, se producen mejores y más consistentes aplicaciones a igual o menor coste.

Proceso Continuo En Lugar De Por Bloques

- ✓ Integración continua: consiste en implementar progresivamente las nuevas características del software. En lugar de crear versiones estables en función de una planificación previamente realizada, los programadores reúnen su código y reconstruyen el proyecto varias veces al día si hace falta.
- ✓ Refactorización: mediante la constante eliminación de código duplicado y/o ineficiente los equipos de programación mejoran el diseño del sistema. El código se evalúa continuamente para ofrecer la mayor calidad posible.
- ✓ Entregas pequeñas: el producto es evaluado en un ambiente real mediante la colocación de un sistema sencillo en producción el cual se actualizará rápidamente, es decir, cada 2 semanas (3 como máximo) el software será puesto en producción.

- Proceso continuo en lugar de por bloques.
- Propiedad intelectual compartida.

- Entendimiento compartido.
 - ✓ Diseño simple: el mejor programa será aquel que cumpla con los requisitos y sea más simple. Es importante proporcionar un software que cubra las necesidades de un cliente. Ni más ni menos.
 - ✓ Metáfora: expresa la visión evolutiva del proyecto y define los objetivos del sistema mediante una historia.
 - ✓ Propiedad colectiva del código: el código tiene propiedad compartida. Nadie es propietario de nada, ni siquiera de lo que ha desarrollado. Todos los programadores son "dueños" de todo el código. Según esta metodología, cuantos más programadores haya trabajado en una parte de código, menos errores tendrá.
 - ✓ Estándar de programación: define las reglas para escribir y documentar código, además de cómo se comunican las diferentes piezas de código desarrolladas por diferentes equipos. El objetivo de esto es que parezca que el código ha sido escrito por una única persona.

Oracle.

Oracle es básicamente una herramienta cliente/servidor para la gestión de base de datos la gran potencia que tiene y su elevado precio hace que solo se vea en empresas muy grandes y multinacionales, por norma general.

Características de Oracle

Desarrollado sobre Oracle Database, Oracle Content Database ha sido diseñada para que las organizaciones puedan controlar y gestionar grandes volúmenes de contenidos

no estructurados en un único repositorio con el objetivo de reducir los costes y los riesgos asociados a la pérdida de información.

Estructuras de Oracle

Una BD Oracle tiene una estructura física y una estructura lógica:

La estructura física se corresponde a los ficheros del sistema operativo.

La estructura lógica está formada por los tablespaces y los objetos de un esquema de BD

Estructura lógica.

Se divide en unidades de almacenamiento lógicas: Tablespaces.

Cada BD estará formado por uno o más tablespaces (al menos existe el tablespace SYSTEM catálogo del sistema) Cada tablespace se corresponde con uno o más ficheros de datos.

Objetos: tablas, vistas, índices asociados a una tabla, clusters.

Estructura física

Una B.D. tiene uno o más ficheros de datos. Estos ficheros son de tamaño fijo y se establecen en el momento en que se crea la base de datos o en el momento en el que se crean tablespaces.

Los datos del fichero de datos son leídos cuando se necesitan y situados en una caché de memoria compartida para que el próximo acceso a los mismos sea más rápido.

Tecnologías Utilizadas

Algunas de las nuevas tecnologías que han surgido son: *JavaServer Pages (JSP)* proporciona una forma simplificada y rápida de crear contenido web dinámico. La tecnología JSP permite el desarrollo rápido de aplicaciones basadas en web que son independientes del servidor y de la plataforma y que permite integrar otras tecnologías como las hojas de estilo en cascada (*Cascade Style Sheet, CSS*) que describen como se va a mostrar un documento, Ajax (*Asynchronous JavaScript And XML*) y el API (*Application Programming Interface*) de Java. También se hace uso del Servidor Apache Tomcat y del motor de Base de Datos Oracle 12.2.0.1.0.

- JavaServer Pages
- Css
- Ajax
- Api
- Tomcat
- Oracle
- Bootstrap
- JQuery
- Alertify
- Jasper Report

Metodología Del Proyecto

La metodología utilizada para este proyecto fue la de Xtreme Programming la cual es un enfoque de la ingeniería de software formulado por Beck (1999), autor del primer libro sobre la materia, Extreme Programming Explained: Embrace Change.

Para el desarrollo de esta metodología se adoptó el modelo que está plasmado en el Instructivo Desarrollo de Software Institucional, el cual se encuentra en el proceso de gestión de sistemas y tecnología que hace parte del macroproceso de apoyo de la Universidad de Cundinamarca con el código de formato ASII008. Se siguió este instructivo ya que era un requisito fundamental para la Dirección de Sistemas y Tecnología, para que el software fuese compatible con la plataforma de la Universidad de Cundinamarca a la hora de su incorporación en la misma, además de lo anterior este proyecto se desarrolló en dos etapas, siendo la primera pasantías y la segunda proyección social.

Fase Planeación

El primer paso que se realizó fue el levantamiento de requerimientos con los ingenieros Jorge Alfredo Mayorga Cárdenas y Diana Milena Rey Gutierrez los cuales se establecieron como funcionales:

Requerimientos Funcionales:

- El sistema debe garantizar el acceso en cualquier momento al usuario y durante el tiempo que dure la cesión.
- El sistema debe avalar la correcta importación y exportación de archivos.
- El sistema debe registrar la información adecuada de cada pasantía o proyecto de proyección social suministrada por él usuario.
- El sistema debe permitir al usuario buscar y consultar la información pertinente de cada pasantía o proyecto de proyección social.
- El sistema controlará el acceso y lo permitirá solamente a usuarios autorizados.
- El sistema hará los correspondientes registros de la evaluación del pasante y de la empresa.

En esta reunión también se evidencio que la fase de proyección social se debía delimitar solo a la sistematización de tres formatos específicos los cuales son MIUF033, MIUF015 y MIUF021, los cuales se encuentran en el macroproceso misional del proceso de gestión interacción universitaria.

Lo anterior debido a que proyección social maneja una serie de presupuestos económicos que deben ser aprobados por otras dependencias, es por esto por lo que junto a los ingenieros Alfredo Mayorga y Diana Rey se toma la decisión de no manipular las

bases de datos destinadas al manejo financiero de la Universidad de Cundinamarca.

Roles del Sistema

Tabla 1:

Roles del Sistema Funcionalidades de Pasantías

ROL	ACTIVIDAD
Director Pasantías	<ul style="list-style-type: none"> • Registro de pasantía • Consultas • Seguimiento
Coordinador interacción universitaria	<ul style="list-style-type: none"> • Consultas de pasantías y evaluaciones del pasante y empresa
Estudiante	<ul style="list-style-type: none"> • Evaluación de la empresa
Asesor Empresa	<ul style="list-style-type: none"> • Evaluación del pasante

Tabla 2:

Roles del Sistema Funcionalidades de Proyección Social

ROL	ACTIVIDAD
Docente Director de Proyecto	<ul style="list-style-type: none"> • Registro de formularios • Evaluación del proyecto

Historias de Usuario

Se hizo uso de las historias usuario como instrumento para levantamiento de requisitos del software, ya que estas, se realizan con un lenguaje no técnico haciéndolas entendibles para cualquier persona, esto permite que tanto como el cliente como el equipo desarrollador entiendan fácilmente los requisitos implícitos en estas, además de incorporar características complementarias como los criterios mínimos de aceptación y el nivel de importancia que estas tiene dentro del proyecto , esto permite que el equipo desarrollador conozca de los criterios de aceptación plasmados por el cliente para este requisito y la importancia que este tiene a nivel de realización del proyecto, y de esta forma poder delegar los recursos necesarios para la realización optima de este requisito.

Tabla 3:

Historias de Usuario Pasantías

HISTORIA	NOMBRE	PRIORIDAD
1	Cargar los convenios con NIT de la empresa, nombre, estado, fecha de inicio y fecha de finalización.	Alto
2	Registrar Pasantía.	Alto
3	Consulta Pasante-Pasantía.	Medio
4	Seguimiento Pasantía.	Alto
5	Consulta Convenio.	Medio

6	Consulta Empresa.	Medio
7	Consulta Pasantía.	Medio
8	Evaluación Pasantías.	Medio

Tabla 4:

Historias de Usuario Proyección Social

HISTORIA	NOMBRE	PRIORIDAD
9	Creación de propuesta campo de aprendizaje en proyección social.	Medio
10	Informe para eventos.	Medio
11	Evaluación de eventos.	Medio

A comparación de su equivalente en las metodologías tradicionales que sería los casos de uso, las historias de usuario expresan lo más claro posible y en lenguaje muy coloquial para ambas partes lo que el cliente quiere de esta parte de proyecto permitiendo poder obtener el resultado correcto con lo que el cliente desea. Por estos motivos se implementó este instrumento para el levantamiento de información en la realización del proyecto a través del siguiente formato:

Tabla 5:

Carga de Convenio

		HISTORIA DE USUARIO
NUMERO: 1	USUARIO: ADMINISTRADOR	
NOMBRE DE HISTORIA: CARGA CONVENIO		
PRIORIDAD: ALTA		RIESGO: ALTA
ESTIMACION:15 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p><i>DESCRIPCION: los responsables de administrar la base de datos deben cargar los convenios aprobados y establecidos para el programa de ingeniería de sistemas de la facultad de ingeniería de la sede de Fusagasugá, los cuales son provistos por la oficina de interacción universitaria. Se cargará directamente en la base de datos el convenio de acuerdo con los datos pertinentes de este, los cuales son: NIT de la empresa, nombre, fecha de inicio y fecha de finalización.</i></p>		

Tabla 6:

Creación Pasantía

		<i>HISTORIA DE USUARIO</i>
NUMERO: 2	USUARIO: DIRECTOR PASANTIAS	
NOMBRE DE HISTORIA: CREACION PASANTIA		
PRIORIDAD: ALTA		RIESGO: ALTO
ESTIMACION:30 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p><i>DESCRIPCION: los responsables del proyecto deben establecer una funcionalidad que permita la creación de una pasantía con los siguientes parámetros: Se instaura una primera consulta para obtener los datos y elección del convenio mediante una búsqueda determinada por cualquiera de los siguientes tres parámetros: código del convenio, nombre de la empresa y numero NIT de la empresa, luego se establecen los datos de la pasantía como lo son, el nombre, acta de comité, fecha de inicio, fecha de finalización, sede, facultad y programa al que corresponda dicha pasantía, seguidamente se establecerá la información de los pasantes en la cual se ingresaran mediante una búsqueda del número de documento de identificación que mostrara datos personales del pasante logrando una selección exacta del pasante solicitado, este mismo mecanismo de consulta por número de documento de identificación será establecido para la asignación de asesores internos de la pasantía y permitiendo también elegir el tipo de este (Metodológico o Disciplinar) y por último se establecerá la información de asesor externo que se realizara mediante el ingreso de datos personales como lo son: documento de identificación, nombres, apellidos, teléfono correo y cargo.</i></p>		

Tabla 7:

Seguimiento de Pasantía

		HISTORIA DE USUARIO
NUMERO: 3	USUARIO: DIRECTOR PASANTIAS	
NOMBRE DE HISTORIA: SEGUIMIENTO PASANTIA		
PRIORIDAD: ALTA		RIESGO: ALTO
ESTIMACION:10 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p>DESCRIPCION: <i>los responsables del proyecto deben establecer una funcionalidad que permita consultar los documentos que hagan falta a cualquier pasantía, además de esto se debe permitir la carga de dichos documentos faltantes. Se instaura una primera consulta para obtener los datos y elección del convenio mediante una búsqueda determinada por cualquiera de los siguientes tres parámetros: código del convenio, nombre de la empresa y numero NIT de la empresa, generando las pasantías constituidas en dicha empresa, Luego de esto se selecciona la pasantía solicitada y admitirá la carga de alguna acta o evaluación que haga falta, adicional a esto, se decreta que solo se podrá subir un documento en lo que corresponde a acta inicial, acta final y evaluación de la pasantía mientras que para las actas de seguimiento se podrán cargar el número de documentos deseados.</i></p>		

Tabla 8:

Consulta Pasante Pasantía

		HISTORIA DE USUARIO
NUMERO: 4	USUARIO: DIRECTOR PASANTIAS, COORDINADOR INTERACCION UNIVERSITARIA	
NOMBRE DE HISTORIA: CONSULTA PASANTE-PASANTIA		
PRIORIDAD: MEDIA		RIESGO: MEDIA
ESTIMACION:5 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p>DESCRIPCION: <i>los responsables del proyecto deben establecer una funcionalidad que permita generar una búsqueda de acuerdo con los filtros y parámetros establecidos en el formato MXF038 asentado en el SGC. Se insta una consulta de acuerdo al formato MEXF038 establecido en el SGC (Sistema de Gestión y Calidad) de la universidad que permita realizar un informe que cimiente una relación entre prácticas y pasantes generando así como resultado la muestra de los siguientes datos: nombre, teléfono, empresa o entidad, ciudad, tipo de entidad, convenio, periodo académico, pasantía y estado y todo esto brindado de acuerdo a cualquiera de los siguientes filtros de búsqueda fijados: fecha, facultad, programa académico, sede y estado de la pasantía.</i></p>		

Tabla 9:

Consulta Convenio

		HISTORIA DE USUARIO
NUMERO: 5	USUARIO: DIRECTOR PASANTIAS, COORDINADOR DE INTERACCION UNIVERSITARIA	
NOMBRE DE HISTORIA: CONSULTA CONVENIO		
PRIORIDAD: MEDIA		RIESGO: MEDIA
ESTIMACION: 15 DIAS/HOMBRE		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p><i>DESCRIPCION: los responsables del proyecto deben establecer una funcionalidad que permita consultar los datos que competen a los convenios estipulados entre las empresas y la universidad. Se instaura una búsqueda del convenio de acuerdo con el código de este o al número NIT de la empresa permitiendo visualizar los datos pertinentes a este, los cuales son: NIT de la empresa, nombre, estado, fecha de inicio y fecha de finalización.</i></p>		

Tabla 10:

Consulta Empresa

		HISTORIA DE USUARIO
NUMERO: 6	USUARIO: DIRECTOR PASANTIAS, COORDINADOR INTERACCION UNIVERSITARIA	
NOMBRE DE HISTORIA: CONSULTA EMPRESA		
PRIORIDAD: MEDIA		RIESGO: MEDIA
ESTIMACION:5 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p><i>DESCRIPCION: los responsables del proyecto deben establecer una funcionalidad que permita consultar los datos que conciernen con las empresas consignadas en los convenios con la universidad. Se instaura una búsqueda de la empresa de acuerdo con el nombre o número NIT de está permitiendo visualizar los datos pertinentes a está, los cuales son: NIT, nombre, dirección, departamento, ciudad, tipo, teléfono y correo.</i></p>		

Tabla 11:

Consulta Pasantía

		HISTORIA DE USUARIO
NUMERO: 7	USUARIO: DIRECTOR PASANTIAS	
NOMBRE DE HISTORIA: CONSULTAR PASANTIA		
PRIORIDAD: MEDIA		RIESGO: MEDIA
ESTIMACION:10 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p>DESCRIPCION: <i>los responsables del proyecto deben establecer una funcionalidad que permita consultar y modificar la información requerida de cualquier pasantía. Se instaura una primera consulta para obtener los datos y elección del convenio mediante una búsqueda determinada por cualquiera de los siguientes tres parámetros: código del convenio, nombre de la empresa y numero NIT de la empresa, visualizando datos relevantes tanto del convenio como de la empresa y a su vez las pasantías constituidas mediante el convenio en dicha empresa, también percibiendo datos importantes de estas como lo son: acta de comité, nombre, estado, facultad, programa, sede, cantidad de pasantes, fecha de inicio y fecha finalización, permitiendo seleccionar cualquier pasantía si se requiere realizar una modificación, al solicitar la opción de modificación se visualizan los datos de la pasantía como lo son acta de comité, nombre, estado, facultad, programa, sede, cantidad de pasantes, fecha de inicio y fecha de finalización, permitiendo</i></p>		

mediante dos opciones, la modificación del estado y la cancelación de la pasantía, seguidamente se muestran los datos pertinentes al convenio establecido para la pasantía, los cuales son código, nombre, estado, fecha inicio, fecha finalización, NIT empresa, nombre empresa, tipo empresa, naturaleza empresa, dirección, departamento, ciudad, teléfono y correo, además de esto se podrá concebir los datos de los pasantes e información de los asesores, tenido para los primeros las opciones de modificar el pasante y otra para prescindir de este en lo que respecta a la pasantía, para los asesores se contara con las siguientes opciones: la primera será para modificar el asesor, la segunda para excluir un asesor y la última para agregar un nuevo asesor si se es necesario, por último se establecerán los documentos referentes que fueron cargados a dicha pasantía y se tendrá una opción para realizar una modificación a cualquier documento que sea seleccionado por el usuario.

Tabla 12:

Evaluación de Pasantía

		HISTORIA DE USUARIO
NUMERO: 8	USUARIO: ESTUDIANTE, ASESOR EMPRESA	
NOMBRE DE HISTORIA: EVALUACION DE PASANTIA		
PRIORIDAD: MEDIA		RIESGO: MEDIA
ESTIMACION:5 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p><i>DESCRIPCION: los responsables del proyecto deben establecer una funcionalidad que permita la realización de la evaluación de la pasantía por parte de los estudiantes y empresas para así de esta manera poder realizar las estadísticas de esta, para esto se designa una ventana en donde encontraran dos links los cuales dirigir al estudiante o a la empresa al formato de evaluación para ser diligenciado.</i></p> <p><i>Los formatos mencionados anteriormente para la evaluación, estudiante (MIUF065_VI) y empresa (MIUF037_V3), son tomados del sistema de gestión de calidad que establece la Universidad de Cundinamarca.</i></p>		

Tabla 13:

Creación de propuesta campo de aprendizaje en proyección social

		HISTORIA DE USUARIO
NUMERO: 9	USUARIO: DOCENTE DIRECTOR DE PROYECTO	
NOMBRE DE HISTORIA: CREACIÓN DE PROPUESTA CAMPO DE APRENDIZAJE EN PROYECCIÓN SOCIAL.		
PRIORIDAD: MEDIA		RIESGO: MEDIA
ESTIMACION:5 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p>DESCRIPCION: <i>los responsables del proyecto deben establecer una funcionalidad que permita generar digitar los parámetros establecidos en el formato MIUF033 asentado en el SGC. Se instaura un formato en el cual el usuario digitara el título, autor(es), propuesta campo de aprendizaje en proyección social, nombre de la oficina o programa que avala el campo de aprendizaje, sede, seccional, extensión, ciudad, información general del campo de aprendizaje que comprende(título, duración(meses),líneas de acción en las que se inscribe el campo, nombre del docente responsable del campo(correo, electrónico, teléfono, cargo), nombre del co-responsable del campo(correo, electrónico, teléfono, cargo), cofinanciación, información de entidad cofinanciadora(entidad, representante legal, dirección, correo electrónico, teléfono, fax, ciudad, departamento, tipo de entidad), personal de apoyo en el proyecto(nombres y apellidos, cargo, horas dedicadas, funciones en el proyecto), conferencistas(nombres y apellidos, perfil del docente, cedula, celular,</i></p>		

correo electrónico)), generalidades del campo de aprendizaje que comprende (introducción, contexto, antecedentes, justificación, beneficiarios directos, definición del problema y alternativa de solución, descripción del campo, objetivos general y específicos, resultados esperados, bibliografía, palabras claves), programación de actividades académicas que comprende (clase evento, docente asignado, lugar de ejecución, horas asignadas, fecha de inicio y fecha de finalización), proyección presupuestal que comprende (vinculación profesional académico(a que evento académico relacionado en el cronograma será vinculado, nombres y apellidos, cedula, número de horas, valor hora, valor total), organización del evento(concepto, cantidad, valor unitario, total, rubro, valor total), por último se ingresaran datos de la facultad o dependencia solicitante que comprende facultad, programa académico, nombre del decano, celular, correo electrónico, fecha y número de acta de aprobación consejo de facultad.

Además de esto, tendrá un botón al final del formato el cual generará un documento en formato de PDF para su respectiva impresión.

Tabla 14:

Informe para Eventos de Proyección Social

		HISTORIA DE USUARIO
NUMERO: 10	USUARIO: DOCENTE DIRECTOR DE PROYECTO	
NOMBRE DE HISTORIA: INFORME PARA EVENTOS DE PROYECCION SOCIAL		
PRIORIDAD: MEDIA		RIESGO: MEDIA
ESTIMACION:5 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p><i>DESCRIPCION: los responsables del proyecto deben establecer una funcionalidad que permita generar digitar los parámetros establecidos en el formato MIUF021 asentado en el SGC. Se instaura un formato en el cual el usuario digitara la información general, la cual se compone de: nombre de evento, elección de si el evento es para educación continuada o proyección social, programa académico, facultad, sede/seccional/extensión, docente responsable, cedula, correo electrónico, de este, a quien va dirigido, intensidad horaria, fecha de inicio y de finalización, lugar de ejecución y su respectiva sede/seccional/extensión, número de personas a certificar y número de participantes, luego de esto ingresara en una breve descripción la justificación, descripción de las actividades desarrolladas y tareas realizadas, a continuación de esto, el usuario deberá digitar los resultados cuantitativos, cualitativos, la caracterización de la población beneficiada que estará segmentada en una tabla con los campos de estudiante, administrativo, docente, comunidad y total, por ultimo deberá ingresar la descripción de la población beneficiada, las conclusiones. Además de esto, tendrá un botón al final del formato el cual generará un documento en formato de PDF para su respectiva impresión.</i></p>		

Tabla 15:

Evaluación de Eventos

		HISTORIA DE USUARIO
NUMERO: 11	USUARIO: DOCENTE DIRECTOR DE PROYECTO	
NOMBRE DE HISTORIA: EVALUACIÓN DE EVENTOS		
PRIORIDAD: MEDIA		RIESGO: MEDIA
ESTIMACION:5 DIAS		
RESPONSABLE: VICTOR ACOSTA/ MATEO RODRIGUEZ		
<p>DESCRIPCION: <i>los responsables del proyecto deben establecer una funcionalidad que permita la realización de la evaluación del evento por parte del usuario para así de esta manera poder realizar las estadísticas de esta, para esto se designa una ventana en donde encontrará un link el cual dirigirá al usuario al formato de evaluación para ser diligenciado.</i></p> <p><i>El formato (MIUF015) mencionado anteriormente para la evaluación es tomado del sistema de gestión de calidad que establece la Universidad de Cundinamarca.</i></p>		

Iteraciones

Ya identificados y ordenados los requisitos continuamos con la creación del cronograma, en este se dividen las tareas a realizar llamadas iteraciones en la cuales se establecen las actividades a realizar y el tiempo estimado que están tendrán para su realización, como recomendación se expresa que el tiempo

establecido no supere las cuatro semanas ya que se estima que este tiempo ya se puede presentar un incremento en la realización del proyecto debemos de tener en cuenta que con la terminación de cada iteración se debe de obtener una parte del proyecto ya funcional la cual se presentará al cliente (facultad de ingeniería de universidad de Cundinamarca sede Fusagasugá) como avance en el desarrollo del proyecto.

Tabla 16:

Velocidad del Proyecto

ITERACIONES	DÍAS	HORAS DIARIAS	HISTORIAS DE USUARIO
Iteración 1, 2 y 3	8	8	3
Iteración 4, 5 y 6	8	8	3
Iteración 7, 8 y 9	8	8	3
Iteración 10 y 11	8	8	2

Las reuniones con el cliente fueron en promedio cada dos semanas, estas con fin de mostrar al cliente (facultad de ingeniería de universidad de Cundinamarca sede Fusagasugá) los avances incrementales que ha tenido el software y a su vez retroalimentarse de las opiniones del cliente en base al avance presentado del proyecto.

Fase de Diseño

Diseño base de datos

Lo primero fue realizar el diseño del modelo de la base datos por medio de la información levantada a través de la historia de usuario y juntamente con los documentos proporcionados por el cliente (facultad de ingeniería de universidad de Cundinamarca sede Fusagasugá), entre los cuales se encuentran los formatos con código EPIF001, MEXF037 y MEXF038 para los procesos de pasantías, por otro lado, en la fase de proyección social, como ya se había mencionado en la etapa de planeación se usaron los formatos MIUF033, MIUF015 y MIUF021.

Iniciamos creando el modelo entidad relación de la base datos en el cual se identifica cada una de las entidades que tendrá el modelo con los correspondientes atributos de cada entidad y las relaciones entre las entidades. (Ver Manual Técnico).

Durante la elaboración del modelo de entidad relación se realizó una reunión con los ingenieros Alfredo Mayorga y Diana Rey en donde se nos comunicó que gracias a que la Universidad de Cundinamarca está realizando el proceso de acreditación de alta calidad, muchos de sus procesos están siendo actualizados entre los que se encuentran los procesos de proyección social, educación continuada y convenios, es por esto que los formatos que existen actualmente están en proceso de modificación, por esto se recurrió a la realización de entrevistas con los diferentes directores que están actualizando los procesos, dichas entrevistas fueron realizadas con la compañía del ingeniero Alfredo Mayorga quien posteriormente sería el encargado de dar el aval al diseño que se

presentó de la base de datos.

Como caso particular los procesos correspondientes a la realización de los convenios con la universidad tuvo un gran cambio, puesto que en un principio cada facultad tenía el manejo de sus convenios y los controlaba con total autonomía, con la actualización de este proceso, los convenios ya no serían distribuidos por facultad sino por sede, seccional o extensión lo cual implica que un convenio puede ser usado por cualquier facultad de la sede, seccional o extensión de la Universidad de Cundinamarca en donde se esté celebrando actualmente, es decir que los convenios de la sede Fusagasugá no los podrá usar las sede de Girardot ni ninguna otra sede, seccional o extensión.

Adicional a los procesos anteriormente mencionados que están en proceso de actualización hay que resaltar de la Dirección de Sistemas y Tecnología está actualizando sus procesos de igual manera, empezando con el cambio de formato de instructivo de desarrollo a la delimitación de políticas de desarrollo de software académico, el cual es el que se usó para el desarrollo de este proyecto.

Ya realizado el modelo entidad relación continuamos con la construcción del modelo relacional en el cual identificamos las llaves primarias y foranas de las tablas y las tablas que son resultado de la relación de las entidades.

Para la creación de la base de datos se utilizó el asistente de interfaz gráfica SQL Developer, asignando como nombre a la base de datos "PASANTIAS"

Creación de interfaces de usuario

En la creación de las diferentes funcionalidades o vistas se uso una plantilla que el área de desarrollo de software nos facilitó, la cual estaba en blanco y lista para empezar a crear cada funcionalidad, estas vista se trabajan por medio de jsp en los cuales se incorpora el código html para la creación de todos los aspectos gráficos, los cuales fueron trabajos con el framework de diseño Bootstrap, este framework divide la pantalla en 12 columnas y a su vez cada columna se podía dividir en 12 nuevamente, esto ya que Bootstrap posiciona los elementos dependiendo la cantidad de columnas que se desee usar, además que Bootstrap tiene un diseño gráfico de elementos que facilita su uso ya que solo hay que llamarlos y ellos ya tienen su estilo definido y no hay que configurarlos desde cero, como cuenta ya con su propia configuración facilitan el trabajo aún más puesto que se adaptan a cualquier pantalla de forma automática.



Figura 1. Interfaz Gráfica Funcionalidades



Figura 2. Ejemplo Interfaz Gráfica Proyección Social

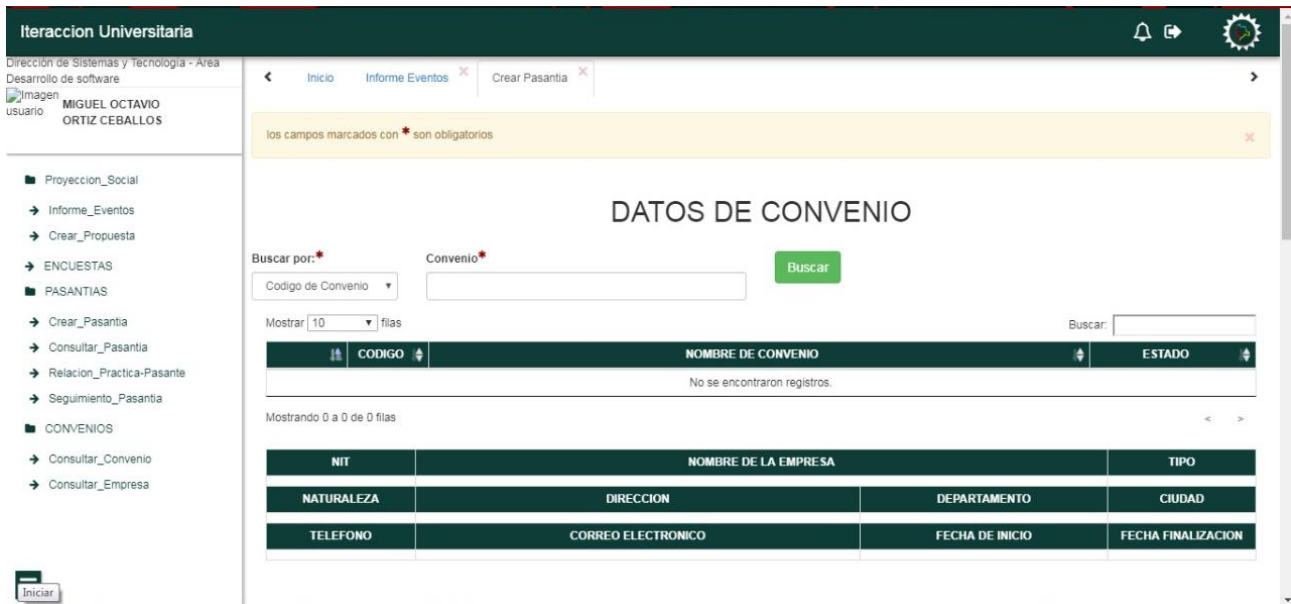


Figura 3. Ejemplo Interfaz Gráfica Pasantías

El equipo desarrollador solo se enfocó en la solución de los requisitos actuales por iteración esto debido a que la metodología nos propone solucionar

solo los inconvenientes que se van presentando y que son directamente relacionados con el requisito ya que el pensar en solucionar error o problemas que no han surgido en el momento provoca el gasto de tiempo y recursos en algo que no ha sucedido y que probablemente no se vaya a utilizar.

Fase de Codificación.

Aunque la metodología sugiere que el cliente este siempre en el desarrollo del proyecto en este caso se modificó un poco este aspecto de la metodología debido a que el cliente no contaba con el tiempo, para estar siempre presente en el desarrollo se acordó el mostrar avances en periodos de tiempo de dos semanas en donde ya se tendrían resultado de las iteraciones programadas para este periodo de tiempo.

Tras la terminación de cada historia de usuario esta iba siendo integrada al proyecto cumpliendo con una integración frecuente y un avance incremental de este.

Como también se trabajó apoyándose en frameworks como JSF y BOOTSTRAP y conjuntamente implementando patrones de diseño como MVC (modelo – vista - controlador), DAO estos aportaron a la estandarización del código desde su arquitectura.

Patrón MVC (modelo vista controlador)

Como mencionamos anteriormente para la construcción del software se hizo uso del framework para JAVA JSF el cual trabaja con el patrón MVC (modelo vista controlador), este nos permite separar los componentes del aplicativo desde las vistas del usuario es decir las interfaces con las cuales este interactúa con el sistema, pasando por los controladores del aplicativo los cuales actúan como intermediarios entre la vista y los modelos quienes son los que se comunican con nuestra base de datos implementado la lógica de negocio.

Vista = interfaces de usuarios

Controladores = implementa lógica para comunicar las vistas con los modelos de datos.

Modelos = capa a donde se trabaja con los datos (base de datos select, insert, update, delete).

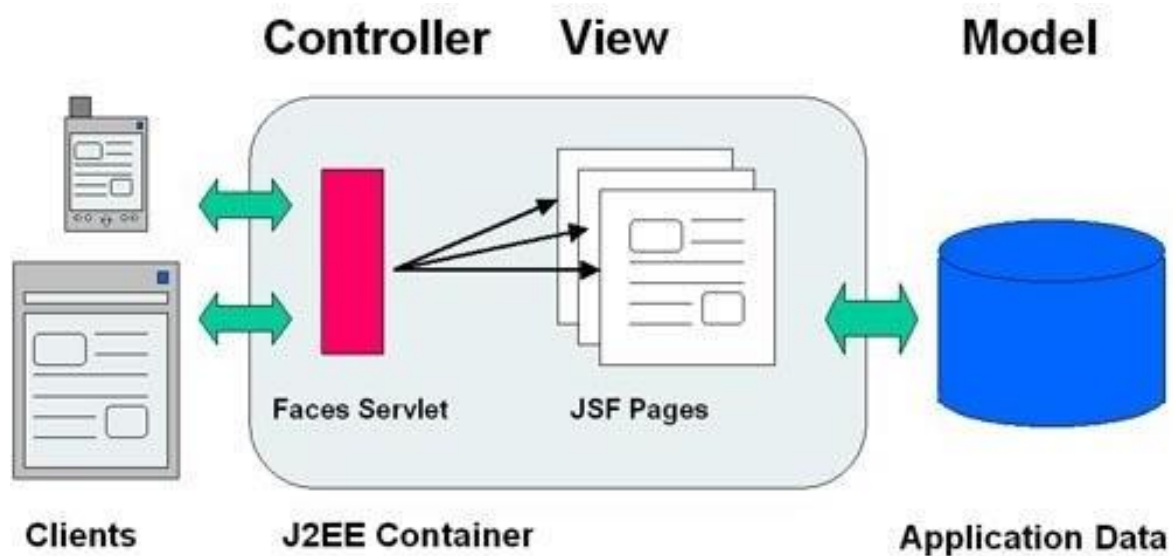


Figura 4. Modelo Vista Controlador

Este patrón lo podemos ver reflejado en la estructura del proyecto como se muestra en la siguiente imagen.

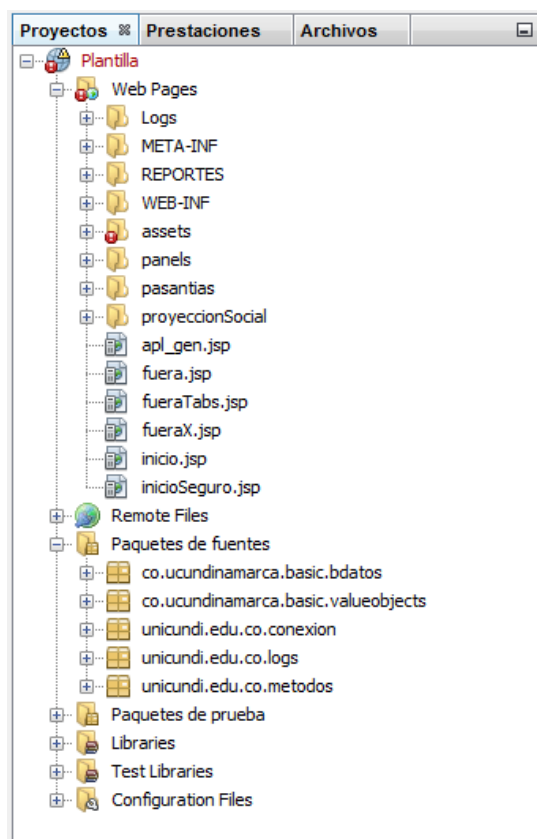


Figura 5. Estructura del Proyecto

Como podemos observar en el proyecto se encuentran separados los componentes, las vistas conformadas por la interfaz de usuarios creadas en el html, los controladores en los paquetes conexión, logs y métodos programados en java y los modelos y lógica del negocio contenida en los paquetes bdatos y valueobjects los cuales realizan las operaciones sobre la base de datos.

Patrón value object

Este patrón consiste en mapear las tablas correspondientes de nuestra base de datos en clases para la creación de objetos sobre estas, esto con el fin de facilitar la creación y mantenimiento de las funciones de transacción que tenga el aplicativo sobre la base de datos, puesto que ya no se trabajarían con valores

serados e independientemente si no en cambio la función podrá recibir todo el objeto el cual ya contiene la información necesaria para realizar la operación correspondientes sobre las tablas.

Este patrón lo vemos reflejado en el proyecto específicamente en los modelos estos se encuentran en el paquete bdatos, en este están el mapeo de cada una de las tablas utilizadas en el aplicativo convertidas en clases, como ejemplo tenemos la tabla convenios y su correspondiente clase en donde podemos observar que la clase contiene los atributos correspondientes a su tabla juntamente con sus métodos get y set de cada uno de los atributos para el acceso a estos.

	◇ COLUMN_NAME	◇ DATA_TYPE	◇ NULLABLE	DATA_DEFAULT	◇ COLUMN_ID	◇ COMMENTS
1	EMPR_ID	NUMBER(30,0)	No	(null)	1	(null)
2	CONV_ID	NUMBER(30,0)	No	(null)	2	(null)
3	CONV_NOMBRE	VARCHAR2(100 CHAR)	No	(null)	3	(null)
4	CONV_ESTADO	VARCHAR2(15 CHAR)	No	(null)	4	(null)
5	CONV_FECHAINICIO	DATE	No	(null)	5	(null)
6	CONV_FECHAFIN	DATE	No	(null)	6	(null)
7	CONV_REGISTRADOPOR	VARCHAR2(30 CHAR)	No	(null)	7	(null)
8	CONV_FECHACAMBIO	DATE	No	(null)	8	(null)
9	CONV_CODIGO	NUMBER(30,0)	No	(null)	9	(null)

Figura 6. Tabla de Convenios

```

13  */
14  public class ConveniosVO {
15      String empr_id;
16      String conv_id;
17      String conv_nombre;
18      String conv_estado;
19      Timestamp conv_fechaInicio;
20      Timestamp conv_fechaFin;
21      String conv_registradoPor;
22      Timestamp conv_fechaCambio;
23      String conv_codigo;
24      String empr_nombre;
25      String pege_documentoIdentidad;
26
27
28  public String getEmpr_id() {
29      return empr_id;
30  }
31
32  public void setEmpr_id(String empr_id) {
33      this.empr_id = empr_id;
34  }
35
36  public String getConv_id() {
37      return conv_id;
38  }
39
40  public void setConv_id(String conv_id) {
41      this.conv_id = conv_id;
42  }
43
44  public String getConv nombre() {

```

Figura 7. Clase ConveniosVO

Patron DAO (Data Access object)

Este patrón trabaja sobre la accesibilidad de los datos en este caso como el aplicativo se conecta con la base de datos, su uso lo podemos ver reflejado en la creación del archivo DAO el cual es el encargado de realizar las funciones de abrir la conexión y cerrar la conexión con la base de datos especificando los parámetros para realizar estas tareas.

El objetivo principal de la utilización del patrón DAO es brindar al aplicativo la flexibilidad de desacoplar la conectividad de los datos de la lógica del negocio.

Si cambiamos el nombre de base de datos o alguno de los atributos necesarios para la conexión como usuario o contraseña del motor base de datos solo con hacer los cambios respectivos en el archivo DAO el aplicativo seguirá con su funcionamiento normal, en caso contrario que este no estuviese aplicado se tendría que realizar este cambio en cada uno de los códigos que realizan procesos de transacción con la base de datos volviendo más engorroso el proceso de creación y mantenimiento del sistema.

Fase de Pruebas.

Pruebas de Aceptación

La primera fase de pruebas fue realizada entre los codificadores, para esto se organizaron de forma tal, que cuando se terminaba una funcionalidad del sistema de información se evaluaba con la inserción de datos y si la funcionalidad no presentaba ningún problema, a su vez se revisaba que los datos se guardaran de forma adecuada en el sistema y que a su vez la funcionalidad cargara los datos de forma correcta a través del aplicativo en la base de datos.

Estas pruebas de aceptación son realizadas con el usuario en las cuales, este evalúa el software según los criterios de aceptación estipulada para cada historia de usuario.

Para la realización de estas pruebas el tester junto con usuario crean casos de prueba en los cuales se estipulan las entradas y salidas que deberá de tener el módulo evaluado.

A continuación, observaremos la aplicación de la prueba de aceptación sobre el módulo de Crear Pasantía con su correspondiente caso de prueba.

Tabla 17:

Ejemplo Caso de Prueba

Caso de prueba	
Nombre de la prueba	Crear Pasantía
Numero de referencia	1
Historia de usuario	2
Iteración	1
Funcionalidad crítica	Si

¿Qué se testeará?	Prueba interfaz de la creación de pasantía, validación de información e inserción en la base de datos
Categoría	Gestión de pasantías
Prerrequisitos	Convenio, estudiante y asesores vigentes y registrados.

Tabla 18:

Pasos y acciones de los casos de prueba Crear Pasantía

D Pasantía	D Pasante	D Asesor Interno	D Asesor Externo	resultado
Vacío	Lleno	Lleno	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Vacío	Lleno	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Vacío	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Lleno	Vacío	No permite la inserción y señala que falta completar o diligenciar
Vacío un solo campo obligatorio	Lleno	Lleno	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Vacío un solo campo obligatorio	Lleno	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Vacío un solo campo obligatorio	Lleno	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Lleno	Vacío un solo campo obligatorio	No permite la inserción y señala que falta completar o diligenciar
Lleno	Lleno	Lleno	Lleno	Permite la inserción en la base de datos

Luego se realizaron pruebas unitarias, las cuales son pruebas realizadas por los desarrolladores sobre el código estas pruebas se realizan con el fin de obtener un código lo más limpio posible de errores.

Pruebas Unitarias

Las pruebas unitarias en proyecto fueron aplicadas a través de la utilización del framework de pruebas JUNIT este es una herramienta que permite la realización de pruebas del código sobre proyecto creados en java.

A continuación, mostraremos los pasos correspondientes para la realización de estas pruebas sobre la conexión de la aplicación a la base de datos.

1. Lo primero es agregar al proyecto la librería JUNIT la cual nos permitirá la aplicación de las pruebas. En este caso utilizaremos JUNIT en su versión 4.12 este archivo .JAR lo podremos encontrar en la página oficial del proyecto JUNIT.

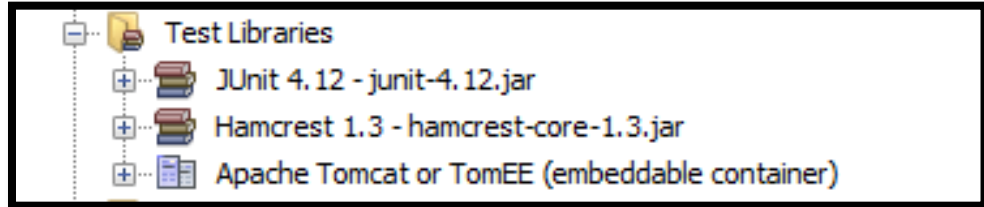


Figura 8. Librería JUnit 4.12

2. Crearemos una nueva clase de tipo JUnit Test con el nombre "Pruebas_conexion_base_de_datos" y especificamos el paquete con el nombre pruebas.

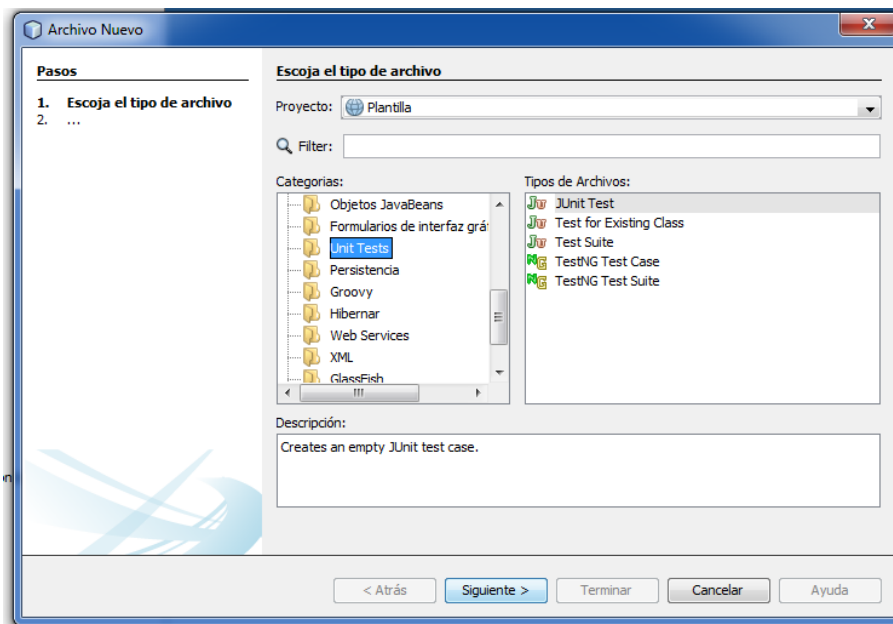


Figura 9. Selección de Archivo JUnit Test

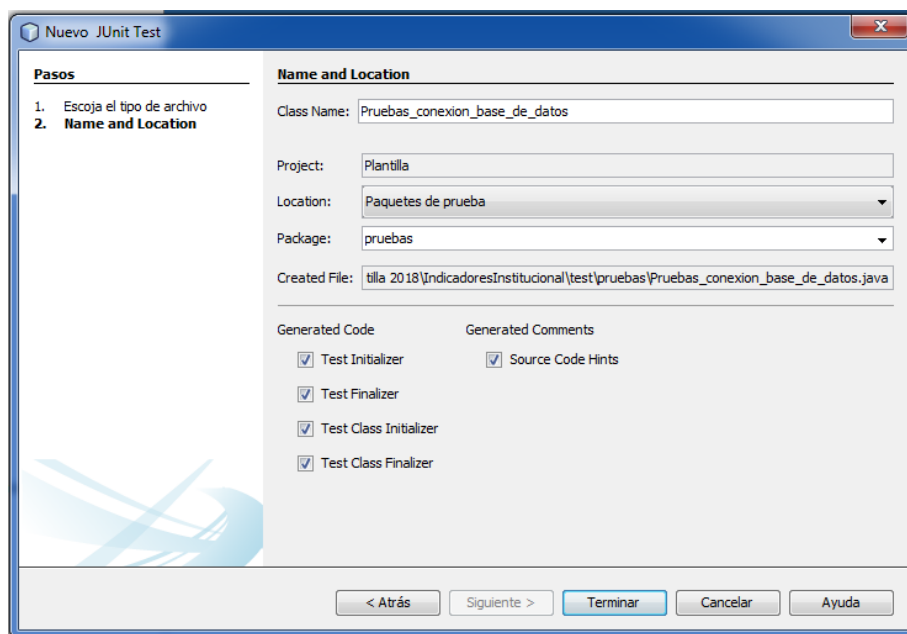


Figura 10. Asignación de Nombre a la Prueba

3. Escribiremos el código correspondiente a la prueba, como la prueba se realizará sobre la conexión de la base de datos esta la realizaremos sobre el código escrito en la clase ConexionExtencionUdec la cual es la encargada de definir los parámetros de conexión sobre la base de datos como el controlador de conexión, el nombre, la base de datos, usuario y contraseña definidos. La prueba específicamente se realizará sobre los métodos de SetCon, GetCon y Close entre la aplicación y la base de datos.

```

45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
}

@Test
public void testSetCon() throws Exception {
    System.out.println("setCon");
    Connection con = null;
    ConexionExtencionUdec instance = new ConexionExtencionUdec();
    instance.setCon(con);
    // TODO review the generated test code and remove the default call to fail.
    fail("The test case is a prototype.");
}

/**
 * Test of getCon method, of class ConexionExtencionUdec.
 * @throws java.lang.Exception
 */
@Test
public void testGetCon() throws Exception {
    System.out.println("getCon");
    ConexionExtencionUdec instance = new ConexionExtencionUdec();
    Connection expResult = null;
    Connection result = instance.getCon();
    assertEquals(expResult, result);
    // TODO review the generated test code and remove the default call to fail.
    fail("The test case is a prototype.");
}

/**
 * Test of close method, of class ConexionExtencionUdec.
 * @throws java.lang.Exception
 */
@Test
public void testClose() throws Exception {
    System.out.println("close");
    ConexionExtencionUdec instance = new ConexionExtencionUdec();
    instance.close();
    // TODO review the generated test code and remove the default call to fail.
    fail("The test case is a prototype.");
}
}

```

Figura 11. Código de Prueba Unitaria

Como observamos los métodos de prueba se colocan anticipados de la anotación `@test` esto indica al framework que el método a continuación es la ejecución de una prueba en el cual dentro de este indicamos el método a ejecutar de la clase a la que realizaremos la prueba y el resultado esperado, de esta manera sabremos si la prueba realizada finalizo correctamente o si el código evaluado contiene un error y no devuelve el valor esperado.

4. Ejecutamos la prueba, esta se realiza corriendo el archivo que ejecuta la prueba
5. una vez ejecutado el archivo este mostrara los resultados obtenidos de la ejecución de la prueba.

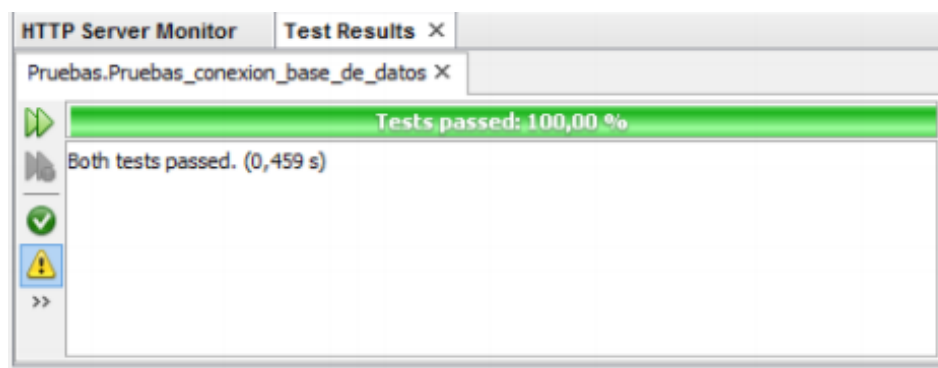


Figura 12. Resultado de la Prueba

Como podemos observar la prueba fue realizada correctamente ya que nos muestra un porcentaje de 100% esto indica que el código ejecutado de los métodos de la clase `ConexionExtencionUdec` en este caso retornaron los valores esperados, es decir el código cumple correctamente con su propósito.

La segunda fase de pruebas fue realizada por la Dirección de Sistemas y Tecnología de la Universidad de Cundinamarca, quien basado en el instructivo de desarrollo realizó pruebas a la base de datos, pruebas de diseño, pruebas de navegabilidad y por último pruebas con el usuario, todas estas pruebas tienen su respectiva metodología, la cual se puede consultar en el formato ASII008.

Implementación del Sistema de Información

Para la implementación del sistema, se realizó entrega a el área Sistema de información un CD con la información necesaria para que el sistema fuese publicado en el portal web de la Universidad de Cundinamarca, además de esto el equipo desarrollador se comprometió a realizar cualquier tipo de modificación necesaria durante su proceso de implantación, para así de esta forma garantizar el correcto funcionamiento del módulo entregado.

Conclusiones y Recomendaciones

En este trabajo se ha descrito un procedimiento de modelación que ha consistido en la captura, análisis y representación de los aspectos y características más significativas de los procesos de pasantías y proyección social.

Sin duda los factores relacionados con la organización son los más trascendentes para el éxito o fracaso de un diseño y desarrollos de un aplicativo. El profundo conocimiento de la organización y del entendimiento de cuán adecuada es la implantación de un aplicativo para apoyar los procesos.

Los factores planteados en el presente proyecto son el resultado del análisis del estudio exploratorio y recopilaciones bibliográficas. Contrastadas en el diseño y desarrollo de un Aplicativo para los proyectos de proyección social y pasantías de la universidad de Cundinamarca. Sin embargo, debe tenerse en cuenta que se ofrece como una reseña para abordar proyectos con características similares.

Es por lo anterior que este proyecto presenta un trabajo conjunto con la Dirección de Sistemas y Tecnología en todo el proceso de desarrollo en donde se acogió a los integrantes del proyecto como si fueran un colaborador más en el desarrollo de software para la Universidad de Cundinamarca, esto logro que se delimitaran herramientas que se podrán compartir con cualquier estudiante que desee realizar un proyecto con la Universidad de Cundinamarca. Como recurso más importante es una plantilla para el desarrollo de módulos de software que permite que se pueda trabajar desde cualquier locación de trabajo por fuera de las instalaciones de la Universidad de Cundinamarca, esta plantilla permite que cualquier persona interesada pueda generar un software que sea

compatible con el software usado por la Universidad de Cundinamarca, para que así de esta manera no se generen inconvenientes de compatibilidad como los que se evidenciaron en la implantación del software de CASA UDEC (Revisar Antecedentes).

Como parte del levantamiento de información, es fundamental la participación de los actores de los grupos de trabajo. Con el fin de conocer la forma en que se realizan las tareas antes de diseñar soluciones asistidas por computadora. De igual forma son los actores quienes ofrecen la mejor retroalimentación a fin de lograr un entendimiento de los procesos grupales.

Gracias a la participación activa de todos los actores de los grupos de trabajo durante el desarrollo de las diferentes entrevistas en la fase de planeación, se vio la necesidad de la creación de módulos adicionales para que el sistema fuera más eficiente, ya que los módulos de pasantías y proyección social hacen parte de un macroproceso que se llama interacción universitaria, el cual comprende temas adicionales como lo son los convenios que la Universidad de Cundinamarca realiza con las diferentes empresas para la realización de pasantías.

Es por lo anterior que se recomienda desarrollar un módulo para la creación y gestión de los convenios donde se pueda tener un control adecuado de estos procesos, y a su vez ayude a que los módulos de pasantías y proyección social sean más eficientes, para esto se presenta el diagrama de flujo del proceso actual correspondiente al proceso de legalización de convenios en la Universidad de Cundinamarca.

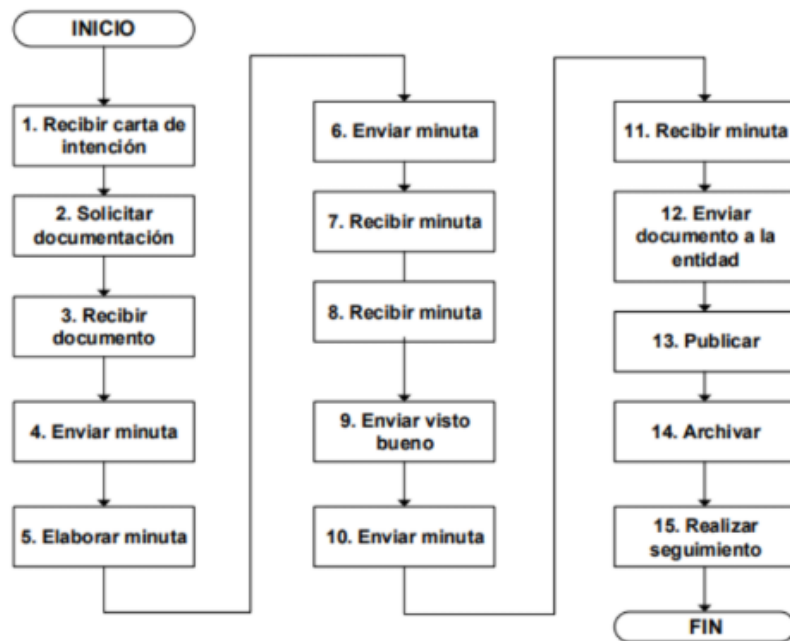


Figura 13. Proceso de Convenios

Referencias

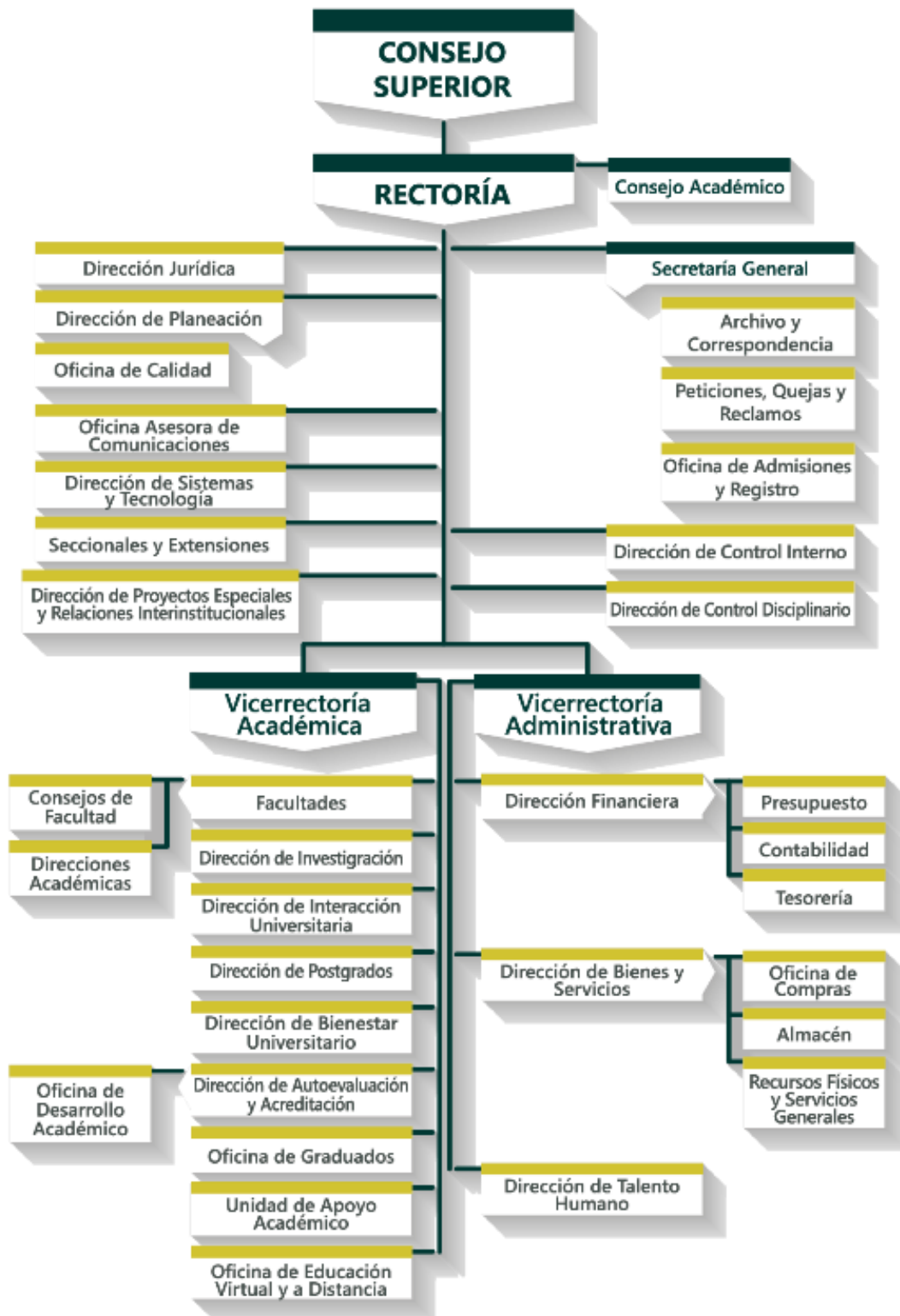
- ACM SIGCHI. (1992). ACM SIGCHI Curricula for Human Computer Interaction. Curriculum Development Group, Pag: 200-203.
- ALAN, S. R. (2002). Workgroup computing: workflow, groupware and messaging. London: Publicaciones Adventure Works.
- BECK, K. (02 de 03 de 1999). Extreme Programming Explained. Recuperado el 20 de 04 de 2012, de computerworld: www.computerworld.com/softwaretopics/software/appdev/story/0,10801,66192,00.html
- Boman, M., Bubenko, J. A., Johannesson, P., & Wangler, B. (1997). Conceptual Modelling. USA: Prentice Hall Series in Computer Science.
- CHAFFNEY, D. (1998). Groupware, workflow and Intranets: Reengineering the Enterprise with Collaborative Software. Digital Press.
- Checkland, P. (1998). Sistemas, Información y Sistemas de Información. Publicaciones Adventure Works.
- DESANCTIS, G., & GALLUPE, B. (1987). A Foundation for the Study of Group Decision Support Systems. Management Science, 589-609.
- DILLENBORUG, P. (1995). The Evolution of Research on Collaborative Learning. London: Spada.
- DIX, A., FINLAY, J., ABOWD, G., & BEALE, R. (1998). Human Computer Interaction. Prentice Hall.
- ELLIS, C. A., GIBBS, S. J., & REIN, G. L. (1991). Groupware: Some Issues and Experiences. Communications of the ACM.
- GARRIDO, J. L. (2003). AMENITIES: Una Metodología para el desarrollo de Sistemas Cooperativos basada en Modelos de Comportamiento y Tarea. (Tesis Doctoral). Granada: Publicaciones Universidad de Granada.
- GONCALVES, A. (2010). Beginning Java EE 6 Platform with GlassFish 3. Apress.
- GRUDIN, J. (1988). Why CSCW applications fail: problems in the design and evaluation of organization of organizational interfaces. ACM Conference on Computer-Supported Cooperative Work (págs. 85-95). ACM Press.

- ISLA MONTES, J. L. (2007). Modelado Conceptual de Sistemas Cooperativos en base a Patrones en AMENITIES (Tesis doctoral). Granada: Publicaciones UNIVERSIDAD DE GRANADA.
- JAKOB, N. (2000). Usabilidad: Diseño de sitios Web. Madrid: Prentice Hall.
- JOHANSEN, R. (1988). Groupware: Computer Support for Business Teams. The Free Press. KAYE, B., & ROGERS, I. (1995). Trabajo de Grupo en las Escuelas Secundaria. México – Buenos Aires: Publicaciones “El Ateneo”.
- LAUDON, K. C., & LAUDON, J. P. (2002). Sistemas de Información Gerencial. Organización y tecnología de la empresa conectada en red. Prentice Hall: Pearson Educación.
- LEE, C. (01 de 03 de 2005). Traducción The New Methodology, de programacionextrema: <http://www.programacionextrema.org/>
- LIU, K., SUN, L., & DIX, A. (2001). Norm-based agency for designing collaborative information systems. Information Systems Journal.
- LYNCH, K. J., SNYDER, J. M., & VOLGEL, D. R. (1990). The Arizona Analyst Information System: Supporting Collaborative Research on international Technological Trends.
- Conference on Multi-User Interfaces and Applications (págs. 40-50). Crete: Greece.
- MABEL SOSA, R. Z. (2006). Modelando aspectos de grupo en entornos colaborativos para proyectos de investigación. Revista de Informática Educativa y Medios Audiovisuales, Pag: 22-31.
- MALON, T. W., & CROWSTON, K. (1990). What is Coordination Theory and How Can It Help Design Cooperative Work Systems. Proceedings of the Conference on Computer Supported Cooperative Work (CSCW'90) (págs. 357-370). New York: ACM Press.
- MÁRQUEZ, J. (01 de 04 de 2003). Arquitectura de Información. de Jmarquez.com: <http://www.jmarquez.com/documentos.html>
- MCGRATH, J. (1993). Time, Interaction and Performance: a theory of groups In: Readings in Groupware and Computer-Supported Cooperative Work. Baecker.
- MUNKVONLD, B. E. (2002). Organisational implementation of collaboration technologies: an integrative review. Kristiansand: University College.
- NORUEGA GARCIA, M. (2009). MODELADO Y ANÁLISIS DE SISTEMAS CSCW: SIGUIENDO UN ENFOQUE DE INGENIERÍA DIRIGIDA POR ONTOLOGÍAS. Granada: Publicaciones UNIVERSIDAD DE GRANADA.

- Ortega, M. (1998). Ambiente de Trabajo Cooperativo en Internet para Management Game (Tesis Licenciatura). Puebla: Publicaciones Universidad de las Américas.
- Porras, H. (2007). TECNOLOGÍAS MODERNAS EN TELEMÁTICA (Tesis Maestría en Informática). Santander: publicaciones Universidad Industrial de Santander.
- RISING, L., & JANOFF, N. (02 de 10 de 2000). The Scrum Software Development Process for Small Teams. de members.cox: <http://members.cox.net/rising11/Articles/IEEEScrum.pdf>
- RODRÍGUEZ GONZÁLEZ, P. (2008). ESTUDIO DE LA APLICACIÓN DE METODOLOGÍAS ÁGILES PARA LA EVOLUCIÓN DE PRODUCTOS SOFTWARE. Madrid: Publicaciones UNIVERSIDAD POLITÉCNICA DE MADRID.
- ROMERO PÉREZ, F. (1997). Sistema Colaborativo Para el Apoyo Electrónico a Reuniones (Tesis de maestría). Chile: Publicaciones Pontificia Universidad Católica de Chile.
- SCHLICHTER, G., WIELINGA, B., & AKKERMANS, H. (1994). CML: The CommonKADS Conceptual Modelling Language. Berlin: European Knowledge Acquisition Workshop.
- SCHWABER, K. (2004). Agile Project Management with Scrum. Microsoft Press.
- SCHWABER, K. (03 de 02 de 2010). Advanced Development Methods. SCRUM Development Process. de jeffsutherland: <http://jeffsutherland.com/scrumpapers.pdf>.
- SERSERGEY, (10 de 03 de 2000). The anatomy of a large-scale hypertextual Web search engine. de <http://infolab.stanford.edu/>: <http://infolab.stanford.edu/~backrub/google.html>
- SUÁREZ, P., & MENDOZA FORERO, C. (2011). RETROSPECTIVA DE LA INVESTIGACION EN LA UNIVERSIDAD DE CUNDINAMARCA 2005 – 2011. Fusagasuga: Publicaciones UDEC.
- TAKEUCHI, H., & NONAKA, I. (1986). The New New Product Development Game. Harvard Business Review.
- Terveen. (1997). An Overview of Human-Computer Collaboration. In Knowledge-Based Systems Journal. Special Issue on Human-Computer Collaboration.
- Univerisdad de Cundinamarca. (2002). Proyecto de acuerdo “Por el cual se establece el Sistema de Investigación en la Universidad de Cundinamarca”. Fusagasuga: Publicaciones UDEC.
- APACHE SOFTWARE FOUNDATION. (2018). APACHE SOFTWARE FOUNDATION. Obtenido de <https://apachefoundation.wikispaces.com/Apache+Tomcat>

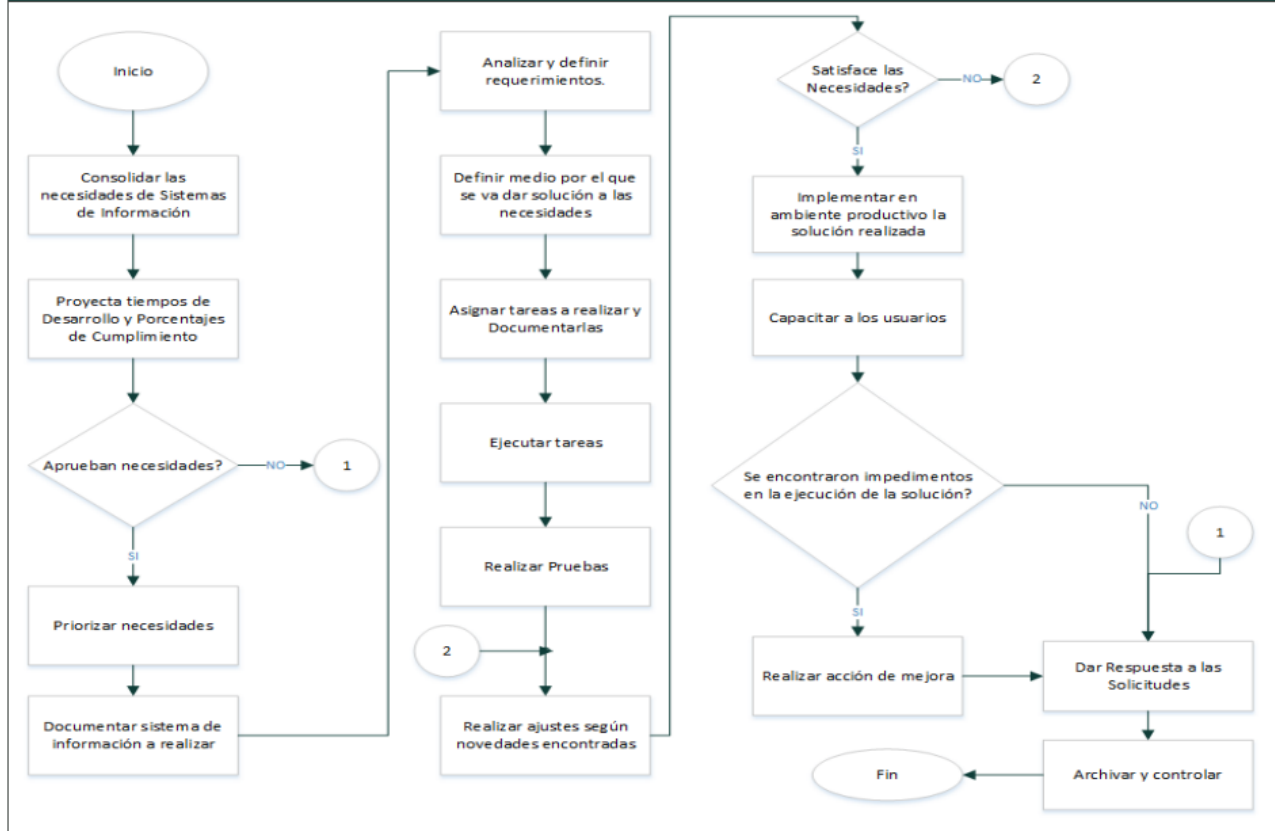
- Belmonte Fernández, O. (2004). Introducción al lenguaje de programación Java. Una guía básica.
- Equipo central Bootstrap. (2018). About. Obtenido de <https://getbootstrap.com/docs/4.1/about/overview/>
- Mendoza Coello, M. I., & Vicuña Pino, A. E. (2017). Diseño de un sistema de gestión para la coordinación de prácticas pre profesionales de la Universidad Técnica Estatal de Quevedo. Quevedo, Ecuador: UTEQ.
- Microsoft. (2018). Microsoft SQL Server. Obtenido de <https://msdn.microsoft.com/es-es/library/bb545450.aspx>
- Mozilla. (2018). HTML. Obtenido de <https://developer.mozilla.org/es/docs/Web/HTML>
- Orozco Iguasnia, J., & Flores Constante, J. M. (2016). Sistema de Control de Actividades de Prácticas Pre-Profesionales para la Universidad Estatal Península de Santa Elena. Trabajo de Titulación, 118. La Libertad, Ecuador: La Libertad: Universidad Estatal Península de Santa Elena. de <http://repositorio.upse.edu.ec/handle/46000/3737>
- TIBCO Software Inc. (2018). Getting Started with JasperReports Library. Obtenido de <https://community.jaspersoft.com/wiki/getting-started-jasperreports-librar>

Anexos



Anexo 1. Estructura Organizacional de la Universidad de Cundinamarca

DIAGRAMA DE FLUJO



Anexo 2. Procedimiento de Sistemas de Información

ACTIVIDAD	RESPONSABLE	CONTROL	DOCUMENTO DE REFERENCIA	REGISTRO RESULTANTE
Consolidar las necesidades de Sistemas de Información. (Solicitudes pendientes y solicitudes nuevas).	Gestor sistemas y tecnología.	Las necesidades deben obedecer por lo menos a un Procedimiento Documentado.	ASII008 ASIr002	ADOr010
Proyectar tiempos de desarrollo y porcentajes de cumplimiento según capacidad instalada.	Gestor de sistemas y tecnología.	Esta actividad está sujeta según el alcance de cada proyecto.	No Aplica	ADOr010 EPIr001
Priorizar y aprobar Necesidades de Sistemas de Información	Comité SAC	No aplica	ADOr010 EPIr001	ADOr010
Documentar sistema de información a realizar o atender, según las necesidades planteadas.	Jefe de área o Gestor Responsable de Área solicitante y Gestor de Sistemas y Tecnología.	Se debe diferenciar en el cronograma las etapas de Investigación y Desarrollo.	ADOr010	ASIr024 EPIr001
Analizar y definir los requerimientos funcionales y no funcionales.	Gestor de Sistemas y Tecnología	No aplica	ASII008	ASIr025
Definir medio por el que se va a dar solución a las necesidades de Sistema de Información. (Desarrollo nuevo, actualización, adquisición o soporte a aplicación).	Gestor de Sistemas y Tecnología	No aplica	No aplica	ASIr025
Asignar tareas a realizar y documentarlas.	Gestor de Sistemas y Tecnología	No aplica	No aplica	ASIr025
Ejecutar tareas bajo metodología adoptada por la Dirección de Sistemas y Tecnología y Documentar eventualidades.	Gestor de realizar la solución	No aplica	ASII008	ASIr025 Códigos Fuente, Objetos de Base de Datos.
Realizar las pruebas necesarias para garantizar un producto de Calidad y que satisfaga las necesidades de software.	Gestor de Sistemas y Tecnología	No aplica	ASII008	ASIr019
Realizar ajustes según novedades de software encontradas.	Gestor de realizar la solución	No aplica	ASII008	Códigos Fuente, Objetos de Base de Datos.
Implementar en ambiente productivo la solución realizada, Entregar, Documentar y realizar el Control de Cambios.	Gestor de Sistemas y Tecnología	No aplica	ASII008	ASIr020 ASIr012 ADOr010 Sistema de Información Implementado
Capacitar a los usuarios de los sistemas de información implementados.	Gestor de realizar la solución	No aplica	ASIr020 ATHPXX	ACAr015 Usuarios capacitados
Dar Respuesta a las solicitudes, Archivar y controlar documentación del proceso realizado.	Gestor de Sistemas y Tecnología	No aplica	ASIM001 ACAP02 ADOP03	ASIr007
Realizar acción de mejora a los impedimentos encontrados al cierre de vigencia.	Gestor Responsable Gestión Sistemas y Tecnología	No aplica	SCIP02	SCIr048

Anexo 3 Tablas de Responsables y Documentos de Procedimientos de Sistemas de Información



ACTA No. 37 DEL 2017-11-28

CLASE SE REUNIÓN: Términos de Análisis, diseño, desarrollo e implementación de software institucional.

CIUDAD Y FECHA: Fusagasugá, 2017-11-22
HORA: 09: 00 a.m.
LUGAR: Tecnología en Desarrollo de Software

ASISTENTES:

MILTON HERNANDO BUITRAGO TORRES
Director de Programa Ingeniería de Sistemas

VICTOR ANDRES ACOSTA ESPINOSA
Estudiante

MATEO RODRIGUEZ OCAMPO
Estudiante

ANA ESPERANZA MERCHÁN HERNÁNDEZ
Director de trabajo de grado

JORGE ALFREDO MAYORGA CARDENAS
Profesional III

ORDEN DEL DIA:

1. Términos para adelantar análisis, diseño, desarrollo e implementación de software institucional.
2. Observaciones.
3. Entregas.
4. Sistema de información para el registro de pasantías y proyectos de proyección social en la Facultad de Ingeniería, Programa de Ingeniería de Sistemas, de la Universidad de Cundinamarca.

DESARROLLO DE LA SESIÓN:

1. Se realiza reunión en la sede Fusagasugá para establecer los términos de trabajo en equipo para el desarrollo de software con el objetivo de implementar en la institución dejando las siguientes premisas para tener en cuenta para llevar a cabo la implementación del software en la plataforma institucional:
 - Los proyectos de software se deben realizar respetando el contexto organizacional y el contexto tecnológico de la institución.
 - Para el desarrollo de software institucional, se cuenta con el procedimiento ASIP05 – DESARROLLO DE SOFTWARE.
 - Se debe realizar un acta de inicio para la ejecución de cualquier proyecto de software, firmados por la dependencia beneficiada con el desarrollo, el coordinador del programa, docente a cargo del proyecto, los desarrolladores y la Dirección de Sistemas y Tecnología.
 - Se deben entregar cronograma de trabajo por parte de los desarrolladores del proyecto.
 - Se debe validar los diseños de datos (Bases de datos), por parte de la Dirección de Sistemas y Tecnología.



- Se debe brindar los accesos necesarios por parte de la Dirección de Sistemas y Tecnología para la ejecución del proyecto de software.
 - Se deben programar reuniones periódicas para validación de avances por parte de la dependencia beneficiada y la Dirección de Sistemas y Tecnología.
 - Las personas que intervienen en el proyecto deberán manejar la información suministrada con un grado de confidencialidad alto.
2. Todos los proyectos de software a ejecutar dentro de la Universidad de Cundinamarca deberán ser solicitados formalmente por coordinador del programa de Ingeniería de Sistemas a la Dirección de Sistemas y Tecnología, nombrando el Docente en cargado del proyecto y los desarrolladores a ejecutarlo.

Las solicitudes serán validadas por la Dirección de Sistemas y Tecnología según las necesidades institucionales de desarrollo de software.

La documentación de las versiones de los desarrollos de software los llevara a cabo la Dirección de Sistemas y Tecnología.

3. Al término de la ejecución del proyecto de software los desarrolladores y docente en cargado entregaran a la Dirección de Sistemas y Tecnología los siguientes entregables:

- Documentación del Desarrollo de Software.
- Modelo de Datos utilizado.
- Cronograma de trabajo utilizado.
- Manuales de Usuario.
- Manual Técnico de implantación.
- Acta de Capacitación a Usuario Final.
- Códigos Fuentes
- Carta de sesión de Derechos patrimoniales de los desarrolladores.

4. Requisitos Funcionales:

- El sistema debe avalar la correcta importación y exportación de archivos en .PDF que son las evidencias de las pasantías y proyectos de proyección social.
- El sistema debe registrar toda la información adecuada de cada pasantía o proyecto de proyección social suministrada por el usuario.
- Las evidencias y documentos esenciales de las pasantías y proyectos de Proyección social deberán ser cargados por el docente a cargo.
- La creación de las pasantías y proyectos de proyección social estará a cargo del Docente a cargo.
- El sistema debe permitir al usuario buscar y consultar la información pertinente de cada pasantía o proyecto de proyección social por parte un usuario designado por el administrador.
- El sistema controlará el acceso y lo permitirá solamente a usuarios autorizados.
- La aplicación debe aplicar para todos los docentes de la Universidad sin importar el programa Académico.



VICTOR ANDRES ACOSTA ESPINOSA
Estudiante Ingeniería de Sistemas
Programa de Ingeniería de Sistemas

MATEO RODRIGUEZ CAMPO
Estudiante Ingeniería de Sistemas
Programa de Ingeniería de Sistemas

ANA ESPERANZA MERCHÁN HERNÁNDEZ
Director de Proyecto de Grado
Programa de Ingeniería de Sistema

JORGE ALFREDO MAYORGA CARDENAS
PROFESIONAL I
Dirección de Sistemas y Tecnología

MILTON HERNANDO BUITRAGO TORRES
Director de Ingeniería de Sistemas
Programa de Ingeniería de Sistemas

15.

ACTA No. 49

CLASE SE REUNIÓN: Fase de Pruebas, Proyecto de Grado
CIUDAD Y FECHA: Fusagasugá, 2018 - 08 - 08
HORA: 9: 00 a.m.
LUGAR: UNIVERSIDAD DE CUNDINAMARCA.
ASISTENTES: JORGE ALFREDO MAYORGA CARDENAS
Director
ANA ESPERANZA MERCHAN HERNANDEZ
Director de trabajo de grado
VICTOR ACOSTA
Estudiante
MATEO RODRIGUEZ
Estudiante

AUSENTES CON EXCUSA:

INVITADOS:

ORDEN DEL DIA:

1. LLAMADO A LISTA Y VERIFICACIÓN DEL QUÓRUM.
2. LECTURA Y APROBACIÓN DEL ORDEN DEL DÍA.
3. ENTREGA DE SISTEMA DE INFORMACION PARA PRUEBAS

DESARROLLO DE LA SESIÓN:

1. Verificación del quorum
2. Siendo las 9:00 am se inició la reunión con la lectura del orden del día, el cual es aprobado por unanimidad por los miembros.
3. Se realiza la entrega de los archivos (script DLL B.D, archivo .war, plantilla como archivo zip, manual técnico y manual de usuario), solicitados por el ingeniero Alfredo Mayorga para que la oficina de Sistemas y Tecnología realice las pruebas oportunas al sistema de información.



15.

ACTA No. 49

COMPROMISOS Y RESPONSABLES:

1. Los estudiantes Víctor Acosta y Mateo Rodríguez se hacen responsables ante cualquier cambio que surja luego de la realización de pruebas por parte de la oficina de Sistemas y Tecnología al sistema de información.
2. El ingeniero Alfredo Mayorga se compromete a realizar la correspondiente carga del sistema de información a un repositorio de pruebas para que los estudiantes puedan cumplir con su respectiva sustentación de trabajo de grado.

No siendo más el orden del día se da por terminada la reunión a las 10:00 a. m.

JORGE ALFREDO MAYORGA CARDENAS
Profesional Director Área

ANA ESPERANZA MERCHAN HERNANDEZ
Director de trabajo de grado

VICTOR ANDRES ACOSTA ESPINOSA
Estudiante

MATEO RODRIGUEZ OCAMPO
Estudiante

Elaboró: Mateo Rodríguez
Revisó: Jorge Alfredo Mayorga Cárdenas

15-52.13



UDECC
UNIVERSIDAD DE
CUNDINAMARCA

ACTA No. 4 DEL 2018-07-10

CLASE SE REUNIÓN:

Asesoría Trabajo de Grado "DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA EL REGISTRO DE PASANTÍAS Y PROYECTOS DE PROYECCIÓN SOCIAL EN LA FACULTAD DE INGENIERÍA, PROGRAMA DE INGENIERÍA DE SISTEMAS, DE LA UNIVERSIDAD DE CUNDINAMARCA

CIUDAD Y FECHA:

Fusagasugá, 2018-07-12

HORA:

09: 00 a.m.

LUGAR:

Interacción Universitaria

ASISTENTES:

VICTOR ANDRES ACOSTA ESPINOSA
Estudiante

MATEO RODRIGUEZ OCAMPO

Estudiante

OMAR FERNANDO MUÑOZ DIMATE

Director oficina interacción Universitaria

ANA ESPERANZA MERCHÁN HERNÁNDEZ

Director de trabajo de grado

ORDEN DEL DIA:

1. LLAMADO A LISTA Y VERIFICACIÓN DEL QUÓRUM.
2. REVISION SISTEMA DE INFORMACIÓN

DESARROLLO DE LA SESIÓN:

1. Verificación de Quorum

- Siendo las 9:00 am se inició la reunión con la lectura del orden del día, el cual es aprobado por unanimidad por los miembros.

2. Revisión sistema de información.

- Se realiza una reunión en la oficina de interacción universitaria de la sede de Fusagasugá para verificar el desarrollo del sistema de información con el objetivo de una aceptación por parte del director de Interacción Universitaria Omar Fernando Muñoz Dimate.



UDECC
UNIVERSIDAD DE
CUNDINAMARCA

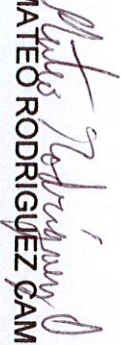
Página 2 de 2

- Se hace una descripción por parte del director de interacción universitaria mediante un flujograma sobre los procedimientos que se tienen establecidos para los convenios y las pasantías.
- Se hace una verificación y explicación de cada una de las funcionalidades del sistema de información.
- Se establecen unas sugerencias por parte del director de Interacción Universitaria, Omar Fernando Muñoz Dimate respecto a unos cambios básicos en una funcionalidad y la incorporación de una nueva en el sistema de información, además de esto incluye, que quede consignado como recomendación en el trabajo de grado el establecimiento de unos módulos faltantes para mejorar el sistema de información con las necesidades trascendentales para los convenios.

COMPROMISOS Y RESPONSABLES:

- El acuerdo para terminar el proceso de creación de convenios con pasantes, para lo cual la ingeniera Esperanza Merchán se compromete en colaborar en la búsqueda de estudiantes de ingeniería que deseen desarrollar esta actividad.
- Los estudiantes Víctor Acosta y Mateo Rodríguez se comprometen a incorporar una funcionalidad para la evaluación tanto del estudiante hacia la empresa como viceversa de acuerdo a las observaciones que realiza el director de Interacción Universitaria, el señor Omar Fernando Muñoz Dimate, quien, además, sugiere, se realice con encuestas online como e-encuesta.


VICTOR ANDRES ACOSTA ESPINOSA
Estudiante Ingeniería de Sistemas
Programa de Ingeniería de Sistemas


MATEO RODRIGUEZ CAMPO
Estudiante Ingeniería de Sistemas
Programa de Ingeniería de Sistemas


ANA ESPERANZA MERCHÁN HERNÁNDEZ
Director de Proyecto de Grado
Programa de Ingeniería de Sistema



MACROPROCESO DE APOYO
 PROCESO GESTIÓN CALIDAD
 REGISTRO DE ASISTENCIA

CÓDIGO: ACA015
 VERSIÓN: 6
 VIGENCIA: 2017-08-10
 PAGINA: 1 de 1

Código de la dependencia.

ACTIVIDAD: Reunión de Avances de Proyecto de Cero de Proyección Social y Esp. Fecha: AÑO 2018 MES 02 DÍA 16

No.	NOMBRES Y APELLIDOS	NÚMERO DE IDENTIFICACIÓN	PROCESO	SEDE/ SECCIONAL/EXTENSIÓN	CORREO ELECTRÓNICO	FIRMA
1	Victor Andres Acosta Espinosa	106975277		Fusagasuga	vandresacosta@ucundinamarca.edu.co	
2	Mateo Rodriguez Casapio	1018443259		"	mateo.rodri@ucundinamarca.edu.co	
3	Daye Alfredo Mejia S.C.	1069725915	Sistemas y Tecnología	figueroa	daye.alfredo@ucundinamarca.edu.co	
4	Ana Esperanza Lopez	39.619.173	Ph.D. en Matemáticas	figueroa	esperanza.lopez@ucundinamarca.edu.co	
5	Diana Tello de Quiroz	20.392.076	Sistemas y Tecnología	Fusagasuga	dianatello@ucundinamarca.edu.co	
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						

Código Serie Documental (Ver Tabla de Retención Documental).

Diagonal 18 No. 20-29 Fusagasuga - Cundinamarca
 Teléfono (091) 8281483 Línea Gratuita 018000976000
 E-mail: info@ucundinamarca.edu.co
 www.ucundinamarca.edu.co
 NIT: 890.680.062-2

Documento controlado por el Sistema de Gestión de la Calidad
 Asegúrese que corresponde a la última versión consultando el Portal Institucional



UDEC
UNIVERSIDAD DE
CUNDINAMARCA

ACTA No. 3 DEL 2018-07-10

CLASE SE REUNIÓN:

Asesoría Trabajo de Grado "DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA EL REGISTRO DE PASANTÍAS Y PROYECTOS DE PROYECCIÓN SOCIAL EN LA FACULTAD DE INGENIERÍA, PROGRAMA DE INGENIERÍA DE SISTEMAS, DE LA UNIVERSIDAD DE CUNDINAMARCA

CIUDAD Y FECHA:

Fusagasugá, 2018-07-10

HORA:

09: 00 a.m.

LUGAR:

Interacción Universitaria

ASISTENTES:

VICTOR ANDRES ACOSTA ESPINOSA
Estudiante

MATEO RODRIGUEZ OCCAMPO
Estudiante

ANA ESPERANZA MERCHÁN HERNÁNDEZ
Director de trabajo de grado

ORDEN DEL DIA:

1. LLAMADO A LISTA Y VERIFICACIÓN DEL QUÓRUM.
2. REVISIÓN SISTEMA DE INFORMACIÓN

DESARROLLO DE LA SESIÓN:

1. Verificación de Quorum

- Siendo las 9:00 am se inició la reunión con la lectura del orden del día, el cual es aprobado por unanimidad por los miembros.

2. Revisión sistema de información.

- Se realiza una reunión en la oficina de interacción universitaria de la sede de Fusagasugá para verificar el desarrollo del sistema de información con el objetivo de una aceptación por parte de la encargada de pasantías.
- Se hace una verificación y explicación de cada una de las funcionalidades del sistema de información, para lo cual la funcionaria Laura Camila Pesellin Silva considera que es muy funcional y le sería útil a los encargados de pasantías, a su vez, recomienda que el director de interacción universitaria lo observe y de sus propios conceptos; para lo cual sugiere reunión.



UDECC
UNIVERSIDAD DE
CUNDINAMARCA

Página 2 de 2

COMPROMISOS Y RESPONSABLES:

- Los estudiantes Victor Acosta y Mateo Rodriguez se comprometen a realizar una reunión con el director de Interacción Universitaria, el señor Omar Fernando Muñoz Dimate para una revisión completa del sistema de información y así obtener observaciones o sugerencias para implementárselas al sistema.

VICTOR ANDRES ACCOSTA ESPINOSA
Estudiante Ingeniería de Sistemas
Programa de Ingeniería de Sistemas

MATEO RODRIGUEZ CAMPO
Estudiante Ingeniería de Sistemas
Programa de Ingeniería de Sistemas

ANA ESPERANZA MERCHÁN HERNÁNDEZ
Director de Proyecto de Grado
Programa de Ingeniería de Sistema



UDECC
UNIVERSIDAD DE
CUNDINAMARCA

ACTA No. 5 DEL 2018-07-30

CLASE SE REUNIÓN:

Asesoría Trabajo de Grado “DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA EL REGISTRO DE PASANTÍAS Y PROYECTOS DE PROYECCIÓN SOCIAL EN LA FACULTAD DE INGENIERÍA, PROGRAMA DE INGENIERÍA DE SISTEMAS, DE LA UNIVERSIDAD DE CUNDINAMARCA

CIUDAD Y FECHA:

Fusagasugá, 2018-07-30

HORA:

09: 00 a.m.

LUGAR:

Interacción Universitaria

ASISTENTES:

VICTOR ANDRES ACOSTA ESPINOSA
Estudiante

MATEO RODRIGUEZ OCAMPO
Estudiante

LAURA CAMILA PESELLIN SILVA
Funcionaria interacción Universitaria

ANA ESPERANZA MERCHÁN HERNÁNDEZ
Director de trabajo de grado

ORDEN DEL DIA:

1. LLAMADO A LISTA Y VERIFICACIÓN DEL QUÓRUM.
2. REALIZACION DE PRUEBAS DE SISTEMA DE INFORMACIÓN

DESARROLLO DE LA SESIÓN:

1. Verificación de Quorum

- Siendo las 9:00 am se inició la reunión con la lectura del orden del día, el cual es aprobado por unanimidad por los miembros.

2. Realización de pruebas de sistema de información.

- Se realiza una reunión en la oficina de interacción universitaria de la sede de Fusagasugá para la ejecución de pruebas del sistema de

Diagonal 18 No. 20-29 Fusagasugá – Cundinamarca

Teléfono (091) 8281483 Telefax: 8732554

Línea Gratuita 018000976000

www.unicundi.edu.co E-mail: unicundi@mail.unicundi.edu.co

NTT: 890.680.062-2




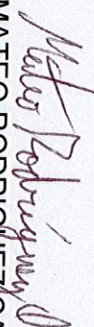
UDEEC
UNIVERSIDAD DE
CUNDINAMARCA


Página 2 de 2

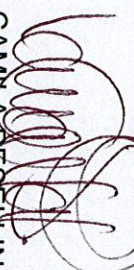
información con el objetivo de una aceptación por parte de la funcionaria de Interacción Universitaria Laura Camila Pesellin Silva.

- Se hace una verificación y explicación de cada una de las funcionalidades del sistema de información y se procede a la práctica de las pruebas correspondientes a cada una de estas.
- La funcionaria luego de practicar las pruebas decide que el sistema de información es funcional y que este cumple con las necesidades requeridas.


VICTOR ANDRES ACOSTA ESPINOSA
Estudiante Ingeniería de Sistemas
Programa de Ingeniería de Sistemas


MATEO RODRIGUEZ CAMPO
Estudiante Ingeniería de Sistemas
Programa de Ingeniería de Sistemas


ANA ESPERANZA MERCHÁN HERNÁNDEZ
Director de Proyecto de Grado
Programa de Ingeniería de Sistema


LAURA CAMILA PESELLIN SILVA
Funcionaria
Oficina de Interacción Universitaria



ACTA No. 2 DEL 2018-04-30

CLASE SE REUNIÓN:

Asesoría Trabajo de Grado "DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA EL REGISTRO DE PASANTÍAS Y PROYECTOS DE PROYECCIÓN SOCIAL EN LA FACULTAD DE INGENIERÍA, PROGRAMA DE INGENIERÍA DE SISTEMAS, DE LA UNIVERSIDAD DE CUNDINAMARCA

CIUDAD Y FECHA:

Fusagasugá, 2018 – 04 30

HORA:

09: 00 a.m.

LUGAR:

Sala de profesores – Bloque F – segundo piso

ASISTENTES:

MATEO RODRIGUEZ OCAMPO
Estudiante
VICTOR ANDRES ACOSTA ESPINOSA
Estudiante
Ing. ANA ESPERANZA MERCHAN HERNANDEZ
Docente Tiempo Completo Ocasional

ORDEN DEL DIA:

1. LLAMADO A LISTA Y VERIFICACIÓN DEL QUÓRUM.
2. REVISION AVANCES DOCUMENTO

DESARROLLO DE LA SESIÓN:

1. Verificación de Quorum

Siendo las 9:00 am se inició la reunión con la lectura del orden del día, el cual es aprobado por unanimidad por los miembros.

2. Revisión avances documento y software.

Se revisa documente y software, para lo cual la docente les indica que esta muy incompleto y de igual forma el software, los estudiantes indican que se debio a que en la oficina de desarrollo de la universidad les entregaron muy tarde la plantilla en la cual deben trabajar y respecto al documento que no le habian dedicado suficiente tiempo, razón por la cual se decide por unanimidad dejar la entrega para el próximo semestre,

No siendo más el orden del día se da por terminada la reunión a las 10:00a.m.


MATEO RODRIGUEZ OCAMPO
Estudiante


VICTOR ANDRES ACOSTA ESPINOSA
Estudiante


Ing. ANA ESPERANZA MERCHAN HERNANDEZ
Docente Tiempo Completo Ocasional



ACTA No. 1 DEL 2018-02-13

CLASE SE REUNIÓN:

Asesoría Trabajo de Grado “DESARROLLO DE UN SISTEMA DE INFORMACIÓN PARA EL REGISTRO DE PASANTÍAS Y PROYECTOS DE PROYECCIÓN SOCIAL EN LA FACULTAD DE INGENIERÍA, PROGRAMA DE INGENIERÍA DE SISTEMAS, DE LA UNIVERSIDAD DE CUNDINAMARCA

CIUDAD Y FECHA:

Fusagasugá, 2018 – 02 13

HORA:

09: 00 a.m.

LUGAR:

Sala de profesores – Bloque F – segundo piso

ASISTENTES:

MATEO RODRIGUEZ OCAMPO
Estudiante
VICTOR ANDRES ACOSTA ESPINOSA
Estudiante
Ing. ANA ESPERANZA MERCHAN HERNANDEZ
Docente Tiempo Completo Ocasional

ORDEN DEL DIA:

1. LLAMADO A LISTA Y VERIFICACIÓN DEL QUÓRUM.
2. REVISION CRONOGRAMA

DESARROLLO DE LA SESIÓN:

1. Verificación de Quorum


Siendo las 9:00 am se inició la reunión con la lectura del orden del día, el cual es aprobado por unanimidad por los miembros.

2. Revision Cronograma


Se revisa cronograma y se ajusta con la metodología expuesta en la plataforma de la universidad sección SISTEMA GESTION DE CALIDAD – SISTEMAS DE INFORMACION, información suministrada por el ingeniero ALFREDO MAYORGA, quien coordina la oficina de desarrollo. Se anexa nuevo cronograma, siguiendo la metodología de programación extrema (XP).

No siendo más el orden del día se da por terminada la reunión a las 10:00 a.m.



MATEO RODRIGUEZ OCAMPO
Estudiante


VICTOR ANDRES ACOSTA ESPINOSA
Estudiante


Ing. ANA ESPERANZA MERCHAN HERNANDEZ
Docente Tiempo Completo Ocasional

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 1 de 22

	CONTENIDO	Pág.
INSTRUCTIVO DESARROLLO DE SOFTWARE	1
1 RECOLECCIÓN DE REQUERIMIENTOS	1
1.1 OBTENER INFORMACIÓN DE LOS USUARIOS	1
1.2 ESPECIFICAR LOS REQUERIMIENTOS TÉCNICOS Y TECNOLÓGICOS	3
2 TIPOS DE SOLUCIÓN INFORMÁTICA	3
2.1 ACTUALIZACIÓN	3
2.2 MÓDULO	3
2.3 APLICACIÓN	3
2.4 SITIO WEB	3
3 DESARROLLO DE SOFTWARE	4
3.1 OBJETIVOS DE XP	4
3.2 ESPECIFICAR LA METÁFORA	6
3.3 RECOLECCIÓN DE LAS HISTORIAS DE USUARIO	7
3.4 REALIZAR DIAGRAMA ENTIDAD-RELACIÓN	7
3.5 PROGRAMACIÓN	7
3.5.1 Nomenclatura De Objetos	7
3.5.2 Manejo De Los Corchetes	8
3.5.3 Indentación	9
3.5.4 Objetos Web	9
3.5.5 Comentarios	10
3.5.6 Patrones para el desarrollo de Software	10
3.5.7 Estructura del directorio de Archivos	11
3.5.8 Lenguajes de Programación	11
3.6 BASES DE DATOS	11
3.7 PRUEBAS AL APLICATIVO	12
3.8 ESPECIFICACIONES ENTREGABLES DEL PROYECTO	18
4 SEGURIDAD DE LA APLICACIÓN	19
4.1 IDENTIFICACIÓN	19
4.2 AUTENTICACIÓN	19
4.3 AUTORIZACIÓN	19
5 VIABILIDAD DE REQUERIMIENTOS O SOLICITUDES DE SOFTWARE	20

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 2 de 22

INSTRUCTIVO DESARROLLO DE SOFTWARE

1. RECOLECCIÓN DE REQUERIMIENTOS

1.1 OBTENER INFORMACIÓN DE LOS USUARIOS


Se realiza una entrevista inicial con el solicitante, de manera que se pueda establecer el tipo de solución informática que pueda satisfacer las necesidades específicas del usuario.

1.2 ESPECIFICAR LOS REQUERIMIENTOS TÉCNICOS Y TECNOLÓGICOS

Se hace un listado de cada requerimiento funcional, no funcional o externo, definidos así:


- Un requerimiento funcional puede ser una descripción de lo que un sistema debe hacer. Este tipo de requerimiento especifica algo que el sistema entregado debe ser capaz de realizar.
- Un requerimiento no funcional: de rendimiento, de calidad, etc; especifica algo sobre el propio sistema, y cómo debe realizar sus funciones. Algunos ejemplos son la disponibilidad, el testeo, el mantenimiento, la facilidad de uso, etc.
- Otros tipos de limitaciones externas, que afectan en una forma indirecta al producto. Estas pueden ir desde la compatibilidad con cierto sistema operativo hasta la adecuación a leyes o regulaciones aplicables al producto.

Posteriormente se revisa que satisfagan las siguientes características, en caso contrario, deben ser reformulados:

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 3 de 22

- Necesario: Lo que pida un requerimiento debe ser necesario para el producto.
- No ambiguo: El texto debe ser claro, preciso y tener una única interpretación posible.
- Conciso: Debe redactarse en un lenguaje comprensible por los inversores en lugar de uno de tipo técnico y especializado, aunque aun así debe referenciar los aspectos importantes
- Consistente: Ningún requerimiento debe entrar en conflicto con otro requerimiento diferente, ni con parte de otro. Asimismo, el lenguaje empleado entre los distintos requerimientos debe ser consistente también.
- Completo: Los requerimientos deben contener en sí mismos toda la información necesaria, y no remitir a otras fuentes externas que los expliquen con más detalle.
- Alcanzable: Un requerimiento debe ser un objetivo realista, posible de ser alcanzado con el dinero, el tiempo y los recursos disponibles.
- Verificable: Se debe poder verificar con absoluta certeza, si el requerimiento fue satisfecho o no. Esta verificación puede lograrse mediante inspección, análisis, demostración o testeo.¹

¹ Especificación de Requisitos según el estándar de IEEE 830

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 4 de 22

2. TIPOS DE SOLUCIÓN INFORMÁTICA

2.1 ACTUALIZACIÓN

Modificación al código existente que no implica la creación de nuevos archivos (.JSP) o manipulación de campos o tablas de las Bases de Datos.

2.2 MÓDULO


Parte de un aplicativo que cumple al menos una de las tareas necesarias para satisfacer la necesidad del usuario, alimentándose de la salida de otro módulo o de los datos de entrada del sistema y proporcionando una salida que alimentará a otro módulo o que será la salida final del sistema.

2.3 APLICACIÓN

Programa informático diseñado para facilitar al usuario la realización de un trabajo determinado. Puede ir integrada con otras aplicaciones o aislada, pero, en cualquier caso, satisface por sí misma una necesidad del usuario.

2.4 SITIO WEB

Conjunto de documentos (páginas web) organizados jerárquicamente. Cada documento (página web) contiene texto y/o gráficos que aparecen como información digital en la pantalla de un ordenador. Un sitio puede contener una combinación de gráficos, texto, audio, vídeo, y otros materiales dinámicos o estáticos.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 5 de 22

3. DESARROLLO DE SOFTWARE


El software se desarrollará mediante la metodología XP (extreme programming), utilizando las prácticas de desarrollo.

3.1 OBJETIVOS DE XP

- ✓ La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor. Un proceso es ágil si a las pocas semanas de empezar ya entrega software que funcione aunque sea rudimentario. El cliente decide si pone en marcha dicho software con la funcionalidad que ahora le proporciona o simplemente lo revisa e informa de posibles cambios a realizar.
- ✓ Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva. Este principio es una actitud que deben adoptar los miembros del equipo de desarrollo. Los cambios en los requisitos deben verse como algo positivo. Les va a permitir aprender más, a la vez que logran una mayor satisfacción del cliente. Este principio implica además que la estructura del software debe ser flexible para poder incorporar los cambios sin demasiado coste añadido. El paradigma orientado a objetos puede ayudar a conseguir esta flexibilidad.


Luego existen una serie de principios que tienen que ver directamente con el proceso de desarrollo de software a seguir.

- ✓ Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas. Las entregas al

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 6 de 22

cliente se insisten en que sean software, no planificaciones, ni documentación de análisis o de diseño.

- ✓ La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto. El proceso de desarrollo necesita ser guiado por el cliente, por lo que la interacción con el equipo es muy frecuente.
- ✓ Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo. La gente es el principal factor de éxito, todo los demás (proceso, entorno, gestión, etc.) queda en segundo plano. Si cualquiera de ellos tiene un efecto negativo sobre los individuos debe ser cambiado.
- ✓ El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo. Los miembros de equipo deben hablar entre ellos, éste es el principal modo de comunicación. Se pueden crear documentos pero no todo estará en ellos, no es lo que el equipo espera.
- ✓ El software que funciona es la medida principal de progreso. El estado de un proyecto no viene dado por la documentación generada o la fase en la que se encuentre, sino por el código generado y en funcionamiento. Por ejemplo, un proyecto se encuentra al 50% si el 50% de los requisitos ya están en funcionamiento.
- ✓ Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante. No se trata de desarrollar lo más rápido posible, sino de mantener el ritmo de desarrollo durante toda la duración del proyecto, asegurando en todo momento que la calidad de lo producido es máxima.


	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 7 de 22

Finalmente los últimos principios están más directamente relacionados con el equipo de desarrollo, en cuanto metas a seguir y organización del mismo.

- ✓ La atención continua a la calidad técnica y al buen diseño mejora la agilidad. Producir código claro y robusto es la clave para avanzar más rápidamente en el proyecto.
- ✓ La simplicidad es esencial. Tomar los caminos más simples que sean consistentes con los objetivos perseguidos. Si el código producido es simple y de alta calidad será más sencillo adaptarlo a los cambios que puedan surgir.
- ✓ Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos. Todo el equipo es informado de las responsabilidades y éstas recaen sobre todos sus miembros. Es el propio equipo el que decide la mejor forma de organizarse, de acuerdo a los objetivos que se persigan.
- ✓ En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento. Puesto que el entorno está cambiando continuamente, el equipo también debe ajustarse al nuevo escenario de forma continua. Puede cambiar su organización, sus reglas, sus convenciones, sus relaciones, etc., para seguir siendo ágil.

3.2 ESPECIFICAR LA METÁFORA

Se desarrolla al inicio del proyecto, definiendo una historia de cómo funcionara el sistema completo. Las historias son breves descripciones de un trabajo de un sistema y la metáfora expresa la visión evolutiva del proyecto que define el alcance y propósito del sistema.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 8 de 22

3.3 RECOLECCIÓN DE LAS HISTORIAS DE USUARIO.

Se interactúa con el usuario para la recolección de las historias que son la descripción de la funcionalidad que se le quiere dar al aplicativo, se clasifica cada historia por la complejidad de llevarla a desarrollo y la prioridad para el usuario sobre el aplicativo.

3.4 REALIZAR DIAGRAMA ENTIDAD-RELACIÓN


Diagrama de los esquemas, bases de datos y tablas, con los respectivos campos y relaciones existentes entre ellos.

3.5 PROGRAMAR

Para el propósito de implementación del software se deben respetar los siguientes estándares de desarrollo:

3.5.1 Nomenclatura De Objetos

- Ningún nombre de objeto llevará tilde o caracteres especiales salvo la línea baja (underline), que se usará para separar palabras o contracciones diferentes.
- Ningún nombre de objeto superará los 20 caracteres.
- Los nombres de tablas irán en **MAYÚSCULA**. Los nombres de los campos y llaves irán en **minúscula**.
- Los nombres de los objetos irán en singular, ya que el nombre representa la entidad que almacena. Los nombres de las tablas serán en plural, por ser más intuitivo para el programador.
- Los nombres de los objetos serán descriptivos, no crípticos.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 9 de 22

- Se usarán abreviaturas cuando sean necesarias y serán suficientemente claras.
- Se usarán abreviaturas estándar siempre que sea posible.
- Los nombres de claves primarias irán precedidas por las siglas PK y los de clave foránea por FK.
- Cada nombre de campo tendrá un prefijo consistente de cuatro letras que identifiquen la tabla en la que se encuentra, seguidas de una línea baja. El nombre de un campo no debería superar los 32 caracteres.
- Tipos de Identificadores: FuncionUno(), CLASE, subrutina
- Variables locales de CLASES: l_nombre_variable.
- Los nombres de las variables que lleven o traigan datos de campos de la Base de Datos tendrán el mismo nombre que el campo siempre que sea posible: unid_nombre contiene la información traída del campo UNID_NOMBRE de la tabla UNIDAD.


3.5.2. Manejo De Los Corchetes

Los corchetes irán justo bajo la instrucción o función que los genere.

```

if (condición1)
{
    If (condición2)
    {
        FuncionUno();
    }
}

```


	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 10 de 22

3.5.3 Indentación

- Se obtiene consistencia si todas las instrucciones se comienzan en una misma columna.
- En los lenguajes en que se hagan declaraciones sobre las variables a utilizar, la información quedará más claramente representada si los atributos declarados se alinean en forma vertical.
- El uso de líneas en blanco ayuda a mostrar con más claridad las relaciones existentes entre distintos ítems agrupados en las declaraciones
- Las instrucciones son mucho más fáciles de localizar y de cambiar si no se escribe más de una instrucción por línea.
- La visión de control de las estructuras lógicas o de los bloques se clarifica si las identaciones controladas son idénticas por alguna cantidad constante. Se sugiere una indentación de tres espacios.
- Máximo 4 columnas de indentación.
- El espacio entre indentaciones será de 1 tabulación.

3.5.4 Objetos Web

- Las Variables de Sesión tendrán el sufijo “_ses”: unidad_ses.
- Los nombres de los objetos de las páginas tendrán sufijos de la siguiente forma:
 - ✓ Cajas de texto: _txt
 - ✓ Select: _sel

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 11 de 22

- ✓ Grupo de opciones: `_opc`
- ✓ Campos de texto ocultos: `_ocu`
- Los nombres de los forms empezarán siempre con la palabra form.


3.5.5 Comentarios

Se utilizarán comentarios en los siguientes casos.

- En los inicios de las clases y funciones cuyo objetivo no sea evidente: una clase llamada “COMUN _ MULTIPLO” no necesita comentario, mientras una llamada “REPORTE_COMPLETO” sí.
- Como etiquetas en el inicio y fin de los bloques más grandes, de manera que sea fácil identificar el grupo de instrucciones dentro de una condición o ciclo.
- Todos los otros casos en los que el programador considere que el código no puede ser entendido al leerse o que haya algún dato que necesite recordar.

3.5.6 Patrones para el desarrollo de Software

- DAO (Data Access Objetc) : Cuando la capa de lógica de negocio necesite interactuar con la base de datos, va a hacerlo a través de la API que le ofrece DAO. Generalmente esta API consiste en métodos CRUD (Insertar, Listar, Actualizar y Borrar). En una aplicación, hay tantos DAOs como modelos. Es decir, en una base de datos relacional, por cada tabla, habría un DAO.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 12 de 22

- VO (Value Object) API que permite el almacenamiento y manipulación de información, API encargada de transportar la información recopilada en formularios o subrutinas al API DAO la cual se encargara de su mediación con la base de datos.

3.5.7 Estructura del directorio de Archivos

Se debe de tener una estructura de archivos para la aplicación que consta de:


- un directorio de imágenes
- un directorio para hojas de estilos.
- Un directorio para archivos java script o plugins.

3.5.8 Lenguajes de Programación

- Lenguaje de Servidor: JSP.
- Objetos Web: HTML, java script, CSS.
- Clases: java.


3.6 BASES DE DATOS

El gestor de base de datos para el desarrollo de software a ser utilizado a nivel institucional debe ser Oracle 11g ya que la universidad cuenta con la infraestructura y el licenciamiento necesario para su uso.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 13 de 22

3.7 PRUEBAS AL APLICATIVO

PRUEBAS DE BASES DE DATOS
PRUEBAS DE INTEGRIDAD DE DATOS
<p>OBJETIVO: Verificar la calidad de los datos, mediante inspección en cada una de las bases de datos de todos los sistemas, para evitar incongruencia en los datos</p>
<p>METODOLOGÍA</p> <ol style="list-style-type: none"> 1. Identificar las herramientas y recursos, el entorno físico que incluye hardware, software y configuraciones de red, incluyendo Herramientas para backup y restore de la base de datos. 2. Contar con un listado de Scripts de carga de información y utilidades para ejecutarlos. 3. Definir los datos de las pruebas. 4. Invocar cada método de acceso y procesos, alimentándolos con datos válidos e inválidos o consultando información. 5. Inspeccionar la base de datos para asegurar que los datos han sido cargados como se pretendía, todos los eventos ocurrieron correctamente, o que la información fue recabada correctamente 6. Aplicar el diseño de la prueba. Desarrollar las pruebas de integridad de acuerdo con el diseño del plan. 7. Monitorizar las pruebas. Validar las pruebas, los datos de las pruebas, y recoger los resultados 8. Analizar los resultados; Volver a priorizar el resto de las pruebas y volver a ejecutarlas de ser necesario.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 14 de 22

PRUEBAS DE DESEMPEÑO

OBJETIVO: Analizar y evaluar las características del software relacionadas con el desempeño como el tiempo de respuesta, throughput y capacidad.


Tiempo de respuesta: Es el intervalo de tiempo que transcurre entre la solicitud de un usuario al sistema y la respuesta de este último

Throughput: Es la tasa a la cual el sistema puede atender las peticiones

Capacidad: máxima cantidad de trabajo útil que se puede realizar por unidad de tiempo.

METODOLOGÍA

1. Identificar las herramientas y recursos, el entorno físico que incluye hardware, software y configuraciones de red. Tener conocimiento sobre esto desde el inicio permite identificar problemas en las pruebas en fases tempranas del proyecto.
2. Identificar los criterios de aceptación de rendimiento, determinado el tiempo de respuesta, el rendimiento y la utilización de los recursos
 - Tiempo de respuesta le corresponde al usuario
 - El rendimiento al negocio
 - Y la utilización de recursos al sistema
3. Planificar y diseñar las pruebas, determinado la variabilidad de los usuarios y definiendo los datos de las pruebas.
4. Aplicar el diseño de la prueba. Desarrollar las pruebas de rendimiento de acuerdo con el diseño del plan.
5. Monitorizar las pruebas. Validar las pruebas, los datos de las pruebas, y recoger los resultados
6. Analizar los resultados; Volver a priorizar el resto de las pruebas y volver a ejecutarlas de ser necesario.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 15 de 22

PRUEBAS DE STRESS

OBJETIVO: Investigar el comportamiento del sistema bajo condiciones que sobrecargan sus recursos.

METODOLOGÍA


1. Utilizar scripts que puedan ser ejecutados en una máquina, deben ser repetidos con múltiples clientes (virtuales o actuales).
2. Para probar recursos limitados, las pruebas se deben correr en un servidor con configuración reducida (o limitada).
3. Para las pruebas de stress restantes, deben utilizarse múltiples clientes, ya sea corriendo los mismos scripts o scripts complementarios para producir el peor caso de volumen de transacciones.
4. Aplicar el diseño de la prueba. Desarrollar las pruebas de stress de acuerdo con el diseño del plan.
5. Monitorizar las pruebas. Validar las pruebas, los datos de las pruebas, y recoger los resultados
6. Analizar los resultados; Volver a priorizar el resto de las pruebas y volver a ejecutarlas de ser necesario.

PRUEBAS DE VOLUMEN

OBJETIVO: someter al sistema a grandes volúmenes de datos para determinar si el mismo puede manejar el volumen de datos especificado en sus requisitos.

METODOLOGÍA

1. Utilizar scripts que puedan ser ejecutados en una máquina, deben ser repetidos con múltiples clientes (virtuales o actuales).
2. Deben usarse múltiples clientes, ya sea corriendo las mismas pruebas o pruebas complementarias para producir el peor caso de volumen

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 16 de 22

3. Se utiliza un tamaño máximo de Base de datos. (actual, escalado o con datos representativos) y múltiples clientes para correr consultas simultáneamente para períodos extendidos.
4. Aplicar el diseño de la prueba. Desarrollar las pruebas de volumen de acuerdo con el diseño del plan.
5. Monitorizar las pruebas. Validar las pruebas, los datos de las pruebas, y recoger los resultados
6. Analizar los resultados; Volver a priorizar el resto de las pruebas y volver a ejecutarlas de ser necesario.


PRUEBAS DE DISEÑO

PRUEBA DE INTERFAZ DE USUARIO

OBJETIVO: Verificar que La navegación a través de las aplicaciones reflejen las funciones pedidas por el usuario en la captura de los requerimientos, incluyendo la navegación de ventana a ventana, de campo a campo, y la utilización de métodos de acceso y además que este diseñado conforme a los estándares de la universidad

METODOLOGÍA

1. Identificar las herramientas y recursos, el entorno físico que incluye hardware, software y configuraciones de red. Tener conocimiento sobre esto desde el inicio permite identificar problemas en las pruebas en fases tempranas del proyecto.
2. Tener conocimiento con respecto a los requerimientos del usuario y el conocer el diseño estandarizado.
3. Realizar una verificación de ortografía
4. Verificar que la para la creación de pantallas (input boxes, combo boxes, list boxes y check buttons, etc) no cambiar radicalmente sus atributos de despliegue (colores, formas) y estándares tradicionales, para lograr que el usuario sepa intuitivamente como usarlo
5. Verificar que los botones de la funciones básicas siempre se muestren del mismo modo
6. Aplicar el diseño de la prueba. Desarrollar las pruebas de interfaz de usuario de acuerdo con el diseño del plan.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 17 de 22

7. Monitorizar las pruebas. Validar las pruebas, los datos de las pruebas, y recoger los resultados
8. Analizar los resultados; Volver a priorizar el resto de las pruebas y volver a ejecutarlas de ser necesario.


PRUEBAS DE NAVEGABILIDAD

PRUEBAS DE FUNCIONALIDAD

OBJETIVO: Asegurar la correcta funcionalidad de la aplicación, incluyendo la navegación, la entrada de datos, el procesamiento, la consulta de información, la actualización y la búsqueda.

METODOLOGÍA

1. Identificar las herramientas y recursos, el entorno físico que incluye hardware, software y configuraciones de red. Tener conocimiento sobre esto desde el inicio permite identificar problemas en las pruebas en fases tempranas del proyecto.
2. Realizar una verificación de formularios validando el ingreso de datos para asegurar que sean bien ingresados y guardados.
3. Validación de campos obligatorios, que sean ingresados los campos requeridos de manera obligatoria y también que estén marcados de alguna manera para para dar a conocer a los usuarios la obligatoriedad de ingresar la información en estos campos adicional a esto se debe explicar esta condición claramente
4. Se debe hacer una sintaxis de ingreso, validando que en los campos sean ingresados con datos validos
5. Confirmar en la tabla de destino que efectivamente se estén enviando los datos de la manera en que se ha previsto.
6. Comprobar que los diferentes formularios funciones en diferentes versiones de navegadores web, sistemas operativos y de tipos de conexiona internet.
7. Confrontar que los botones interactivos que permiten imprimir, enviar email, regresar a la página anterior, inicio, etc. estén realizando correctamente la acción indicada.
8. En los sistemas de búsqueda se debe validar que efectivamente permitan encontrar documentos existentes en el sitio.
9. Aplicar el diseño de la prueba. Desarrollar las pruebas de funcionalidad de acuerdo con el diseño del plan.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 18 de 22

10. Monitorizar las pruebas. Validar las pruebas, los datos de las pruebas, y recoger los resultados
11. Analizar los resultados; Volver a priorizar el resto de las pruebas y volver a ejecutarlas de ser necesario

PRUEBAS DE SEGURIDAD Y CONTROL DE ACCESO

OBJETIVO: Verifica que un actor solo pueda acceder a las funciones y datos que su usuario tiene permitido y que solo los actores con acceso al sistema y a la aplicación están habilitados para accederla.

METODOLOGÍA:

1. Identificar cada tipo de usuario, las funciones y los datos que tiene autorizados
2. Crear pruebas para cada tipo de usuario y verificar cada permiso, creando transacciones para cada tipo de usuario
3. Modificar tipo de usuarios y volver a ejecutar las pruebas y verificar en cada caso si los datos o funciones adicionales quedan correctamente permitidos o denegados
4. Aplicar el diseño de la prueba. Desarrollar las pruebas de seguridad y control de acuerdo con el diseño del plan.
5. Monitorizar las pruebas. Validar las pruebas, los datos de las pruebas, y recoger los resultados
6. Analizar los resultados; Volver a priorizar el resto de las pruebas y volver a ejecutarlas de ser necesario


PRUEBAS CON EL USUARIO

PRUEBAS DE ACEPTACIÓN

OBJETIVO: Determinar si el usuario final acepta o rechaza el aplicativo web desarrollado

METODOLOGÍA


1. Haber ejecutado las pruebas mencionadas anteriormente, corregido fallas encontradas y haber realizado cambios sugeridos

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 19 de 22

2. Al ejecutar las pruebas de aceptación el usuario final comprueba si la aplicación sigue con los estándares establecidos
3. El usuario realiza una navegación por la aplicación verificando los estándares de GUI del cliente
4. Valida objetos gráficos contra los estándares establecidos
5. Aplicar el diseño de la prueba. Desarrollar las pruebas de aceptación de acuerdo con el diseño del plan.
6. Monitorizar las pruebas. Validar las pruebas, los datos de las pruebas, y recoger los resultados
7. Analizar los resultados y realizar un informe; Volver a priorizar el resto de las pruebas y volver a ejecutarlas de ser necesario

3.8 ESPECIFICACIONES ENTREGABLES DEL PROYECTO

- Soportes del desarrollo del proyecto: Esto incluirá historias de usuario, copia de formato de solicitud, control de cambios.
- Documento de Requerimientos: Un documento con la especificación de requerimientos clasificados por funcionales y no funcionales.
- Diagramas: Documento con el diagrama de clases, entidad relación, interfaz.
- Manual de usuario: Documento con especificaciones de uso flujo de información y navegabilidad de la aplicación.
- Manual técnico: En este documento se especificara las condiciones técnicas necesarias para el buen funcionamiento de la aplicación, deberá incluir diagrama entidad relación, diccionario de datos, versiones de exploradores (Internet Explorer, Mozilla, Opera), Servidor de base de datos, Servidor web.
- Paquete código fuente: archivos con código, clases, archivos de conexión a bases de datos necesarios para el funcionamiento de la aplicación.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 20 de 22

4. SEGURIDAD DE LA APLICACIÓN

4.1 IDENTIFICACIÓN


Un sistema de información posee las características de identificación cuando .es capaz de reconocer usuarios individuales. La identificación es el primer paso en conseguir el acceso para el material seguro, y sirve como soporte para las subsiguientes autenticación y autorización. La autenticación e identificación son esenciales para estandarizar el nivel de acceso o autorización que se le concede a un individuo. Por lo general el nivel de identificación es dado mediante un nombre de usuario o ID; su manejo confidencial, así como su permanente actualización son claves a la hora de la implementación.

4.2 AUTENTICACIÓN

La autenticación ocurre cuando un control da pruebas de que el usuario posee la identidad que suministra. Existen algunos hardware en criptografía que facilitan este mecanismo.

4.3 AUTORIZACIÓN

Después de que la identidad del usuario es autenticada, Un proceso llamado autorización da la seguridad de que un usuario ha sido específicamente y explícitamente autorizado para acceder, actualizar o eliminar los contenidos de un archivo, un ejemplo de este control es la activación o uso de listas de control de acceso y grupos de autorización en un ambiente de trabajo en red. Es una característica bien importante cuando se utilizan varias terminales

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 21 de 22


donde las personas pueden adicionar o cambiar datos a la información contenida en bases de datos.

Un programa de seguridad exitoso en información, combina estos y otros elementos. El arte de reducir y administrar el riesgo requiere comunicación y cooperación entre todos los niveles de la organización. En otras palabras el aseguramiento de los activos de información puede ser alcanzado solo a través de la administración cuidadosa y la concientización de todos los funcionarios, lo cual genera cultura en torno al tema de la seguridad.

5. VIABILIDAD DE REQUERIMIENTOS O SOLICITUDES DE SOFTWARE

Se considera viable una solicitud de desarrollo de software cuando cumpla las siguientes condiciones:

- ✓ No se encuentre una aplicación funcional que cumpla con la actividad u tarea solicitada.
- ✓ Lo solicitado pueda ser implantado en la infraestructura tecnológica con la cual cuenta la institución.
- ✓ El tiempo disponible para ejecutar el desarrollo se encuentre en los periodos establecidos en el procedimiento de Desarrollo de Software.
- ✓ El solicitante disponga del tiempo y atención suficiente para atender las visitas de recolección de requerimientos.
- ✓ La solicitud sea coherente con las soluciones informáticas que brinda el procedimiento de Desarrollo de Software.

	MACROPROCESO DE APOYO	CODIGO:ASII008
	PROCESO GESTION SISTEMAS Y TECNOLOGIA	VERSION: 2
	INSTRUCTIVO DESARROLLO DE SOFTWARE INSTITUCIONAL	PAGINA: 22 de 22

NOTA: Estas políticas deberán seguir un proceso de actualización periódica sujeto a los cambios organizacionales relevantes: crecimiento de la planta de personal, cambio en la infraestructura computacional, desarrollo de nuevos servicios, entre otros.

CONTROL DE CAMBIOS				
VERSION	FECHA			DESCRIPCION DEL CAMBIO
1	2014	05	08	Emisión Documento
2	2014	07	18	Se ingresan documentación de las pruebas de Calidad de Software ASIP12 que se realizan a las aplicaciones institucionales

ELABORO					
NOMBRES Y APELLIDOS		CARGO			
Edilson Martínez Clavijo		Gestor de Proceso			
Daniel Leonardo Guerrero Forero		Gestor de Proceso			
Jorge Alfredo Mayorga Cárdenas		Gestor de Proceso			
Diana Mabel Padilla Candela		Gestor de Proceso			
APROBO (GESTOR RESPONSABLE DEL PROCESO)					
NOMBRES Y APELLIDOS		CARGO	FECHA		
			AAAA	MM	DD
José Omar Luna Carvajal		Director de Sistemas y Tecnología	2014	07	18

Sistema de información para el Registro de
Pasantías y Proyectos de Proyección Social en la
Facultad de Ingeniería de Sistemas en la
Universidad de Cundinamarca sede Fusagasugá

MANUAL TECNICO

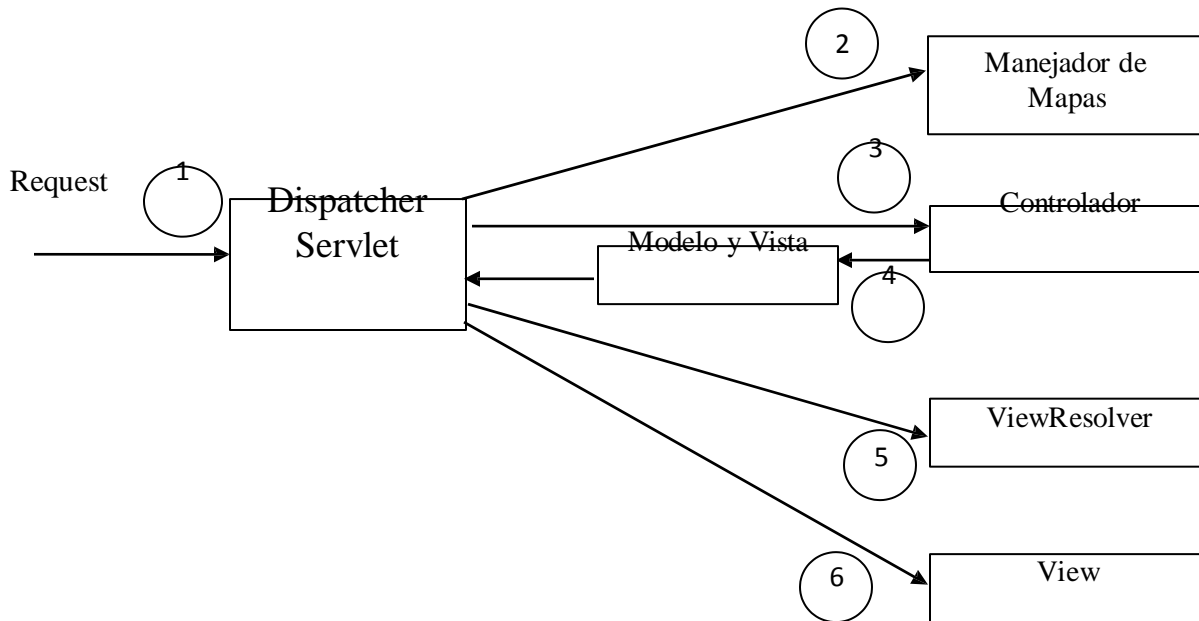


UDEC
UNIVERSIDAD DE
CUNDINAMARCA

TABLA DE CONTENIDO

ARQUITECTURA	3
MODELO VISTA CONTROLADOR.....	4
HERRAMIENTAS EMPLEADAS EN EL DESARROLLO.....	5
DISEÑO DE LA APLICACIÓN.....	6
DICCIONARIO DE DATOS	9
RECURSOS	23
ARCHIVOS DE CONFIGURACIÓN	24
CREDITOS.....	25

ARQUITECTURA



Framework Spring MVC

El marco de trabajo de Spring Web model-view-controller (MVC) está diseñado en torno a un *DispatcherServlet* que maneja todas las solicitudes y respuestas HTTP. El flujo de trabajo de procesamiento de solicitudes del Spring Web MVC *DispatcherServlet*.

- ▮ Después de recibir una solicitud HTTP, *DispatcherServlet* consulta *HandlerMapping* para llamar al *controlador* apropiado.
- ▮ El *controlador* toma la solicitud y llama a los métodos de servicio apropiados en función del método GET o POST utilizado. El método de servicio establecerá los datos del modelo en función de la lógica empresarial definida y devolverá el nombre de la vista al *DispatcherServlet*.
- ▮ El *DispatcherServlet* tomará la ayuda de *ViewResolver* para recoger la vista definida para la solicitud.
- ▮ Una vez finalizada la vista, The *DispatcherServlet* pasa los datos del modelo a la vista que finalmente se procesa en el navegador.

Todos los componentes mencionados anteriormente, es decir, *HandlerMapping*, *Controller* y *ViewResolver* son partes de *WebApplicationContext* w, que es una extensión del *ApplicationContext* simple con algunas características adicionales necesarias para las aplicaciones web.

MODELO VISTA CONTROLADOR

La arquitectura de la aplicación Spa Udec basa sus fundamentos en el MODELO VISTA CONTROLADOR (MVC) quien se define como una propuesta de diseño de software utilizada para implementar sistemas donde se requiere el uso de interfaces de usuario. Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.

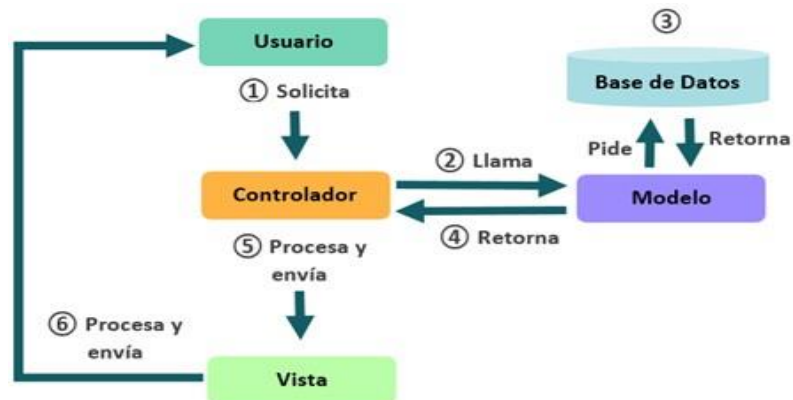


Figura 2
VISTA

CONTROLADOR

MODELO

HERRAMIENTAS EMPLEADAS EN EL DESARROLLO

IDE

NetBeans 8.2.

SERVIDOR DE BASE DE DATOS

Oracle.

SERVIDOR WEB

Apache Tomcat

DISEÑO DE LA APLICACIÓN

Para el diseño del aplicativo primero se realizó la conexión local con la base de datos ORACLE como se muestra a continuación:

```
package unicundi.edu.co.conexion;

import java.sql.Connection;
import java.sql.SQLException;
import javax.naming.Context;
import javax.naming.InitialContext;
import javax.sql.DataSource;

public class ConexionExtencionUdec {

    private Context initCtx;
    private Context envCtx;
    private DataSource ds;
    private Connection con;

    public ConexionExtencionUdec() throws SQLException, Exception {
        initCtx = null;
        envCtx = null;
        ds = null;
        con = null;
        try {
            initCtx = new InitialContext();
            envCtx = (Context) initCtx.lookup("java:comp/env");
            ds = (DataSource) envCtx.lookup("jdbc/ExtencionUdec");
            con = ds.getConnection();
        } catch (SQLException e) {
            System.out.println("Err no se puedo cargar el driver en ExtencionUdec --> DefinitionCon"
                + "nnectionVortal "
                + e.getMessage());
            con = null;
            throw e;
        } catch (Exception e) {
            System.out.println("Err general en ExtencionUdec --> ExtencionUdec " + e.getMessage());
            con = null;
            throw e;
        }
    }

    public void setCon(Connection con) {
        this.con = con;
    }
}
```

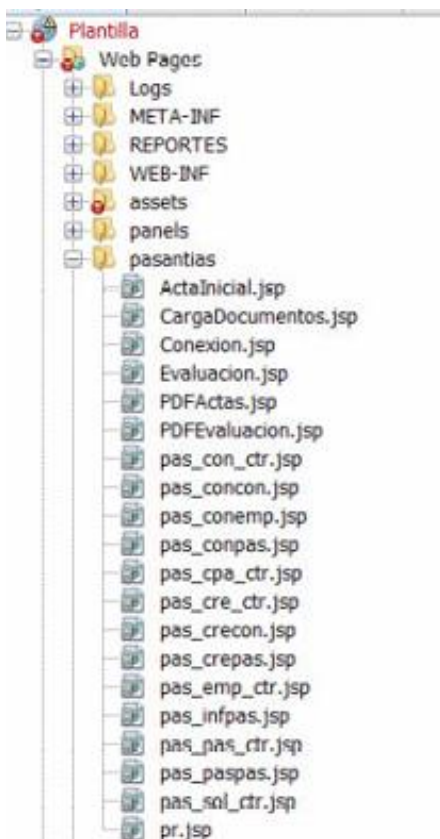
```

public Connection getCon() {
    return con;
}

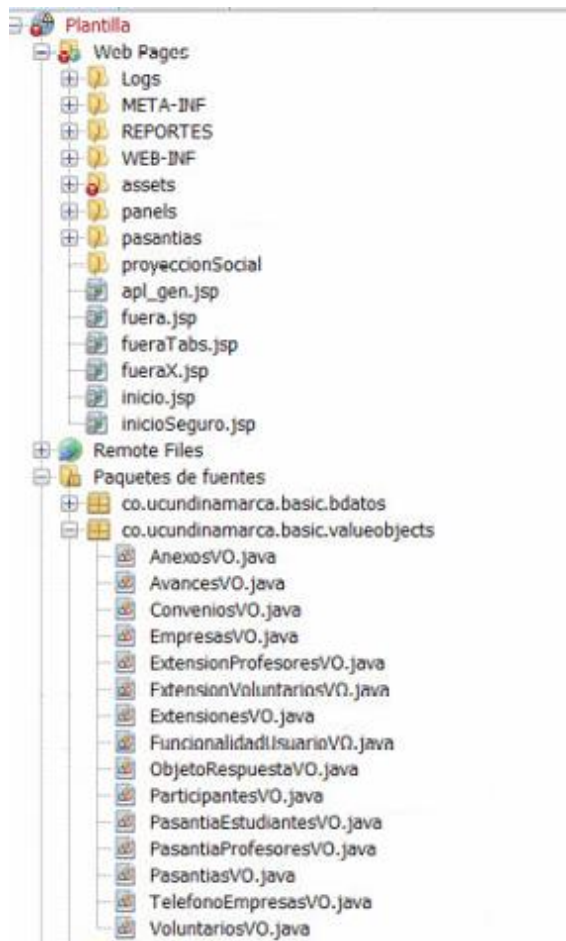
public void close() throws SQLException, Exception {
    try {
        if (this.con != null) {
            this.con.close();
            this.con = null;
        }
    } catch (SQLException e) {
        System.out.println("Err -->CerticadoNotas ::> ExtencionUdec ::> ExtencionUdec ExtencionUdec::>
ExtencionUdec ::> close()" + e.getMessage());
        e.printStackTrace();
        throw e;
    }
}
}
}

```

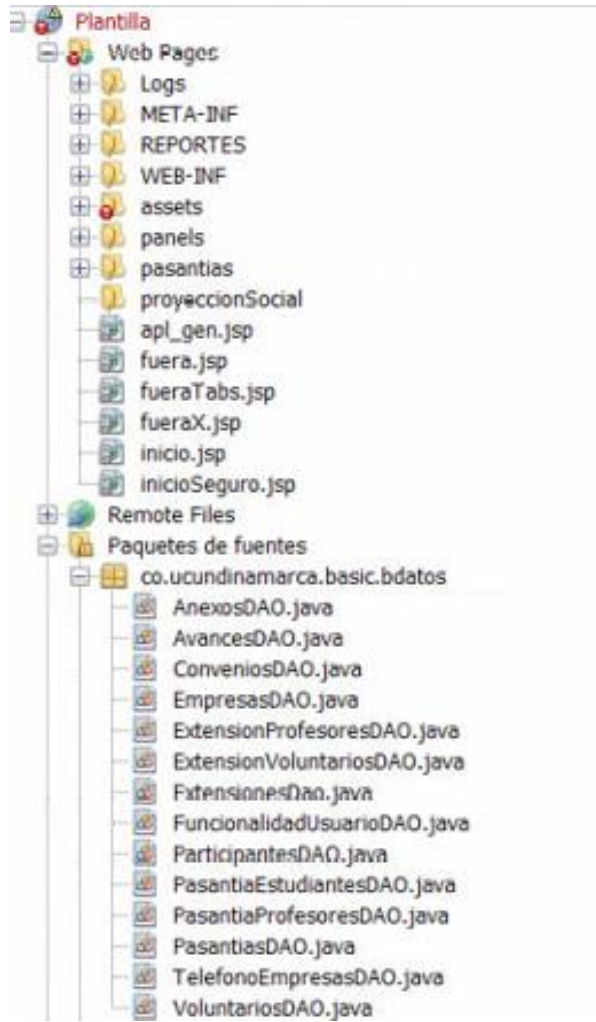
Una vez esta conexión está comprobada se diseñaron las vistas que están ubicadas en el fichero WEB-PAGES JSP quienes ayudadas por una plantilla en Bootstrap para usar de forma generalizada en cada una de las vistas.



Los métodos funciones y clases que permiten ejecutar los Scripts en el aplicativo están diseñados en la capa controlador por eso cada vista tiene métodos propios, pero a su vez interactúan entre sí para facilidad y reutilización de código se diseñó para cada vista un controlador, que es donde finalmente está el filtro de las peticiones hechas por el cliente a la base de datos.



Para finalizar la fase de diseño del aplicativo se genera el pool de conexiones en donde llegan las solicitudes del controlador y se encarga de realizar la consulta en la BD y retornar un resultado bien sea de carga o de consulta y como en el controlador cada tabla de la BD tiene su fichero de conexión.



DICcionario DE DATOS

Table Name	1. GENERAL.ACTIVIDADECONOMICA
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	ACEC_ID	P		Y	NUMERIC (30)	LT				
2	ACEC_DESCRIPCION				VARCHAR (500)	LT				
3	ACEC_CODIGO				VARCHAR (30)	LT				
4	ACEC_FEHACAMBIO			Y	Date	LT				
5	ACEC_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
ACTIVIDADECONOMICA_PK	PK				ACEC_ID	ASC

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
PERSONAJURIDICAGEN_ACTIECO_FK	PERSONAJURIDICAGENERAL	Y	Y		ACEC_ID

Table Name	2. EXTENSIONUDEC.ANEXOS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	ANEX_ID	P		Y	NUMERIC (30)	LT				
2	EXTE_ID		F	Y	NUMERIC (30)	LT				
3	ANEX_FECHA			Y	Date	LT				
4	ANEX_DESCRIPCION			Y	VARCHAR (250 CHAR)	LT				
5	ANEX_REGISTRADOPOR				VARCHAR (30 CHAR)	LT				
6	ANEX_FECHACAMBIO			Y	Date	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
ANEXOS_PK	PK				ANEX_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
ANEXOS_EXTENCIONES_FK	EXTENCIONES	Y	Y		EXTE_ID

Table Name	3. GENERALUDEC.AREAS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	AREA_ID	P		Y	NUMERIC (30)	LT				
2	AREA_DESCRIPCION				VARCHAR (500)	LT				
3	AREA_ESTADO				NUMERIC (1)	LT				
4	PROC_ID			Y	NUMERIC (30)	LT				
5	AREA_REGISTRADOPOR				VARCHAR (30 BYTE)	LT				
6	AREA_FECHACAMBIO				Date (7)	LT				
7	AREA_INTERNA			Y	NUMERIC (1)	LT		0		
8	AREA_FEHACAMBIO			Y	Date	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
AREAS_PK	PK				AREA_ID	ASC
					AREA_ID	ASC

Table Name	4. GENERALUDEC.AREAS_UNIDADES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	AREA_ID	P		Y	NUMERIC (30)	LT				
2	UNID_ID	P		Y	NUMERIC (30)	LT				
3	ARUN_ESTADO				NUMERIC (1)	LT				
4	ARUN_REGISTRADOPOR				VARCHAR (30 BYTE)	LT				
5	ARUN_FECHACAMBIO				Date (7)	LT				
6	ARUN_CORREO				VARCHAR (250 BYTE)	LT				
7	ARUN_CONTRATA				NUMERIC (1)	LT				
8	ARUN_CARGOJEFE				VARCHAR (200 BYTE)	LT				
9	PROG_ID				NUMERIC (30)	LT				
10	ARUN_FEHACAMBIO			Y	Date	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
AREASUNIDADES_PK	PK				AREA_ID	ASC
					UNID_ID	ASC
					AREA_ID	ASC
					UNID_ID	ASC

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
EXTENCIONES_AREAS_UNIDADES_FK	EXTENCIONES	Y	Y		AREA_ID
					UNID_ID
PASANTIAS_AREAS_UNIDADES_FK	PASANTIAS	Y	Y		AREA_ID
					UNID_ID

Table Name	5. EXTENSIONUDEC.AVANCES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	AVAN_ID	P		Y	NUMERIC (30)	LT				
2	PAPR_ID		F	Y	NUMERIC (30)	LT				
3	PASA_ID		F	Y	NUMERIC (30)	LT				
4	AVAN_FECHA			Y	Date	LT				
5	AVAN_DESCRIPCION			Y	VARCHAR (250 CHAR)	LT				
6	AVAN_ACTA			Y	BFile	LT				
7	AVAN_FEHACAMBIO			Y	Date	LT				
8	AVAN_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
AVANCES_PK	PK				AVAN_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
AVANCES_PASANTIAPROFESORES_FK	PASANTIAPROFESORES	Y	Y		PAPR_ID
AVANCES_PASANTIAS_FK	PASANTIAS	Y	Y		PASA_ID

Table Name	6. GENERAL.CIUDADGENERAL
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	CIGE_ID	P		Y	NUMERIC (30)	LT				
2	DEGE_ID		F	Y	NUMERIC (30)	LT				
3	CIGE_NOMBRE				VARCHAR (50)	LT				
4	CIGE_FEHACAMBIO			Y	Date	LT				
5	CIGE_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
CIUDADGENERAL_PK	PK				CIGE_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
CIUDADGENERAL_DEPARTAGENE_FK	DEPARTAMENTOGENERAL	Y	Y		DEGE_ID

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
PERSONAGENERAL_CIUDADGENE_FK	PERSONAGENERAL	Y	Y		CIGE_ID

Table Name	7. EXTENSIONUDEC.CONVENIOS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	EMPR_ID		F	Y	NUMERIC (30)	LT				
2	CONV_ID	P		Y	NUMERIC (30)	LT				
3	CONV_NOMBRE			Y	VARCHAR (100 CHAR)	LT				
4	CONV_ESTADO			Y	VARCHAR (15 CHAR)	LT				
5	CONV_FECHAINICIO			Y	Date	LT				
6	CONV_FECHAFIN			Y	Date	LT				
7	CONV_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				
8	CONV_FECHACAMBIO			Y	Date	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
CONVENIOS_PK	PK				CONV_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
CONVENIOS_EMPRESAS_FK	EMPRESAS	Y	Y		EMPR_ID

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
PASANTIAS_CONVENIOS_FK	PASANTIAS	Y	Y		CONV_ID

Table Name	8. GENERAL.DEPARTAMENTOGENERAL
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	DEGE_ID	P		Y	NUMERIC (30)	LT				
2	DEGE_NOMBRE				VARCHAR (80)	LT				
3	DEGE_FEHACAMBIO			Y	Date	LT				
4	DEGE_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
DEPARTAMENTOGENERAL_PK	PK				DEGE_ID	ASC

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
CIUDADGENERAL_DEPARTAGENE_FK	CIUDADGENERAL	Y	Y		DEGE_ID

Table Name	9. EXTENSIONUDEC.EMPRESAS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	EMPR_ID	P		Y	NUMERIC (30)	LT				
2	PEGE_ID		F	Y	NUMERIC (30)	LT				
3	EMPR_NOMBRE			Y	VARCHAR (100 CHAR)	LT				
4	EMPR_TIPO				VARCHAR (50)	LT				
5	EMPR_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				
6	EMPR_FECHACAMBIO			Y	Date	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
EMPRESAS_PK	PK				EMPR_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
EMPRESAS_PERSONAGENERAL_FK	PERSONAGENERAL	Y	Y		PEGE_ID

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
CONVENIOS_EMPRESAS_FK	CONVENIOS	Y	Y		EMPR_ID
TELEFONOEMPRESAS_EMPRESAS_FK	TELEFONOEMPRESAS	Y	Y		EMPR_ID

Table Name	10. EXTENSIONUDEC.EXTENCIONES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	EXTE_ID	P		Y	NUMERIC (30)	LT				
2	EXTE_TIPOEVENTO			Y	VARCHAR (25 CHAR)	LT				
3	EXTE_FACULTAD			Y	VARCHAR (100 CHAR)	LT				
4	AREA_ID		F	Y	NUMERIC (30)	LT				
5	UNID_ID		F	Y	NUMERIC (30)	LT				
6	EXTE_DIRIGIDOA			Y	VARCHAR (100 CHAR)	LT				
7	EXTE_INTENSIDADH			Y	VARCHAR (5 CHAR)	LT				
8	EXTE_FECHAINICIO			Y	Date	LT				
9	EXTE_FECHAFIN			Y	Date	LT				
10	UNID_IDSEDE		F	Y	NUMERIC (30)	LT				
11	EXTE_ESTADO			Y	VARCHAR (15 CHAR)	LT				
12	EXTE_PROPUESTA			Y	BFile	LT				
13	EXTE_INFOMEFINAL			Y	BFile	LT				
14	EXTE_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				
15	EXTE_FECHACAMBIO			Y	Date	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
EXTENCIONES_PK	PK				EXTE_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
EXTENCIONES_UNIDAD_FK	UNIDAD	Y	Y		UNID_ID
EXTENCIONES_AREAS_UNIDADES_FK	AREAS_UNIDADES	Y	Y		AREA_ID
					UNID_ID

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
ANEXOS_EXTENCIONES_FK	ANEXOS	Y	Y		EXTE_ID
EXTENSIONPROFESORES_EXTEN_FK	EXTENSIONPROFESORES	Y	Y		EXTE_ID
EXTENSIONVOLUNTARIOS_EXTEN_FK	EXTENSIONVOLUNTARIOS	Y	Y		EXTE_ID
PARTICIPANTES_EXTENCIONES_FK	PARTICIPANTES	Y	Y		EXTE_ID

Table Name	11. EXTENSIONUDEC.EXTENSIONPROFESORES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	EXTE_ID		F	Y	NUMERIC (30)	LT				
2	PEGE_ID		F	Y	NUMERIC (30)	LT				
3	EXPR_ESTADO				NUMERIC (1)	LT				
4	EXPR_ID	P		Y	NUMERIC (30)	LT				
5	EXPR_FEHACAMBIO			Y	Date	LT				
6	EXPR_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
PASANTIAPROFESORESv1_PK	PK				EXPR_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
EXTENSIONPROFESORES_EXTEN_FK	EXTENCIONES	Y	Y		EXTE_ID
EXTENSIONPROFESORES_PERGEN_FK	PERSONAGENERAL	Y	Y		PEGE_ID

Table Name	12. EXTENSIONUDEC.EXTENSIONVOLUNTARIOS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	EXVO_ID	P		Y	NUMERIC (30)	LT				
2	VOLU_ID		F	Y	NUMERIC (30)	LT				
3	EXTE_ID		F	Y	NUMERIC (30)	LT				
4	EXVO_FECHA				Date	LT				
5	EXVO_FEHACAMBIO			Y	Date	LT				
6	EXVO_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
EXTENSIONVOLUNTARIOS_PK	PK				EXVO_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
EXTENSIONVOLUNTARIOS_EXTEN_FK	EXTENCIONES	Y	Y		EXTE_ID
EXTENSIONVOLUNTARIOS_VOLUN_FK	VOLUNTARIOS	Y	Y		VOLU_ID

Table Name	13. GENERALUDEC.FUNCIONALIDADES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	FUNC_ID	P		Y	NUMERIC (30)	LT				
2	APLI_ID			Y	NUMERIC (30)	LT				
3	FUNC_CODIGO				VARCHAR (20)	LT				
4	FUNC_TITULO				VARCHAR (100)	LT				
5	FUNC_DESCRIPCION				VARCHAR (100)	LT				
6	FUNC_RUTA				VARCHAR (100)	LT				
7	FUNC_ICONO				VARCHAR (100)	LT				
8	FUNC_REGISTRADOPOR			Y	VARCHAR (30)	LT				
9	FUNC_FECHACAMBIO			Y	Date	LT				

Columns Comments

No	Column Name	Description	Notes
1	FUNC_ID	FUNC_ID es una llave principal de tipo numerico de treinta (30) caracteres, obligatoria, guarda el codigo de las funcionalidades	
2	APLI_ID	APLI_ID campo de tipo numerico de treinta (30) caracteres, obligatoria, guarda el codigo de las aplicaciones	
3	FUNC_CODIGO	FUNC_CODIGO campo de tipo VARCHAR de veinte (20) caracteres, obligatoria, guarda el codigo de las funcionalidades	

No	Column Name	Description	Notes
4	FUNC_TITULO	FUNC_TITULO campo de tipo VACHAR de cien (100) caracteres, obligatorio, guarda el titulo de la funcionalidad	
5	FUNC_DESCRIPCION	FUNC_DESCRIPCION Campo de tipo VARCHAR de cien (100) caracteres que guarda la descripcion de la funcionalidad	
6	FUNC_RUTA	FUNC_RUTA campo de tipo VARCHAR de cien (100) caracteres que guarda la ruta de esta funcionalidad	
7	FUNC_ICONO	FUNC_ICONO campo de tipo VARCHAR de cien (100) caracteres que guarda el icono de la funcionalidad	
8	FUNC_REGISTRADOPOR	FUNC_REGISTRADOPOR campo tipo VARCHAR de treinta (30) caracteres, obligatorio, asigna el nombre completo de quien registra la funcionalidad	
9	FUNC_FECHACAMBIO	FUNC_FECHACAMBIO campo de tipo Date, obligatorio, asigna la fecha en que se realiza el cambio a la funcionalidad	

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
FUNCIONALIDADES_PK	PK				FUNC_ID	ASC
					FUNC_ID	ASC

Table Name	14. EXTENSIONUDEC.PARTICIPANTES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	PART_ID	P		Y	NUMERIC (30)	LT				
2	EXTE_ID		F	Y	NUMERIC (30)	LT				
3	PEGE_ID		F		NUMERIC (30)	LT				
4	PEEX_ID		F		NUMERIC	LT				
5	PART_ESTADO				VARCHAR (50)	LT				
6	PART_CERTIFICADO				NUMERIC (1)	LT				
7	PART_FEHACAMBIO			Y	Date	LT				
8	PART_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
PARTICIPANTES_PK	PK				PART_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
PARTICIPANTES_PERSOGENE_FK	PERSONAGENERAL		Y		PEGE_ID
PARTICIPANTES_PERSOEXTER_FK	PERSONAEXTERNA		Y		PEEX_ID
PARTICIPANTES_EXTENCIONES_FK	EXTENCIONES	Y	Y		EXTE_ID

Table Name	15. EXTENSIONUDEC.PASANTIAPROFESORES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	PAPR_ID	P		Y	NUMERIC (30)	LT				
2	PASA_ID		F	Y	NUMERIC (30)	LT				
3	PEGE_ID		F	Y	NUMERIC (30)	LT				
4	PAPR_ESTADO				NUMERIC (1)	LT				
5	PAPR_FEHACAMBIO			Y	Date	LT				
6	PAPR_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
Relation_7_PK	PK				PAPR_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
PASANTIAPROFESORES_PERGENE_FK	PERSONAGENERAL	Y	Y		PEGE_ID
PASANTIAPROFESORES_PASANTI_FK	PASANTIAS	Y	Y		PASA_ID

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
AVANCES_PASANTIAPROFESORES_FK	AVANCES	Y	Y		PAPR_ID

Table Name	16. EXTENSIONUDEC.PASANTIAS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	PASA_ID	P		Y	NUMERIC (30)	LT				
2	CONV_ID		F	Y	NUMERIC (30)	LT				
3	PASA_ACTACOMITE			Y	VARCHAR (20 CHAR)	LT				
4	PASA_NOMBRE			Y	VARCHAR (100 CHAR)	LT				
5	AREA_ID		F	Y	NUMERIC (30)	LT				
6	UNID_ID		F	Y	NUMERIC (30)	LT				
7	PASA_FECHAINICIO			Y	Date	LT				
8	PASA_FECHAFIN			Y	Date	LT				
9	UNID_IDSEDE		F	Y	NUMERIC (30)	LT				
10	PASA_ESTADO			Y	VARCHAR (15 CHAR)	LT				
11	PASA_EVALUACION			Y	BFile	LT				
12	PASA_ACTACOMPROMISO			Y	BFile	LT				
13	PASA_ACTAINICIO			Y	BFile	LT				
14	PASA_ACTAFIN			Y	BFile	LT				
15	PASA_INFORMEFINAL				BFile	LT				
16	EMPR_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				
17	EMPR_FECHACAMBIO			Y	Date	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
PASANTIAS_PK	PK				PASA_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
PASANTIAS_UNIDAD_FK	UNIDAD	Y	Y		UNID_ID
PASANTIAS_AREAS_UNIDADES_FK	AREAS_UNIDADES	Y	Y		AREA_ID
					UNID_ID
PASANTIAS_CONVENIOS_FK	CONVENIOS	Y	Y		CONV_ID

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
AVANCES_PASANTIAS_FK	AVANCES	Y	Y		PASA_ID
ESTUDIANTES_PASANTIAS_FK	PASANTIASESTUDIANTES	Y	Y		PASA_ID
PASANTIAPROFESORES_PASANTI_FK	PASANTIAPROFESORES	Y	Y		PASA_ID

Table Name	17. EXTENSIONUDEC.PASANTIASESTUDIANTES
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	PAES_ID	P		Y	NUMERIC (30)	LT				
2	PASA_ID		F	Y	NUMERIC (30)	LT				
3	PEGE_ID		F	Y	NUMERIC (30)	LT				
4	ESTU_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				
5	ESTU_FECHACAMBIO			Y	Date	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
ESTUDIANTES_PK	PK				PAES_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
PASANTIASESTUDIANTES_PERGEN_FK	PERSONAGENERAL	Y	Y		PEGE_ID
ESTUDIANTES_PASANTIAS_FK	PASANTIAS	Y	Y		PASA_ID

Table Name	18. GENERALUDEC.PERSONAEXTERNA
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	PEEX_ID	P		Y	NUMERIC	LT				
2	PEEX_PRIMERAPELLIDO				VARCHAR (300)	LT				
3	PEEX_SEGUNDOAPELLIDO				VARCHAR (300)	LT				
4	PEEX_PRIMERNOMBRE				VARCHAR (300)	LT				
5	PEEX_SEGUNDONOMBRE				VARCHAR (300)	LT				
6	PEEX_DOCUMENTOIDENTIDAD				VARCHAR (15)	LT				
7	TIDG_ID				NUMERIC (30)	LT				
8	PEEX_DIRECCIONRESIDE				VARCHAR (500 BYTE)	LT				
9	PEEX_TELEFONOFIJO				VARCHAR (15)	LT				
10	PEEX_TELEFONOCELULAR				VARCHAR (15)	LT				
11	PEEX_CORREOUNO				VARCHAR (500)	LT				
12	PEEX_CORREODOS				VARCHAR (500)	LT				
13	CIGE_ID				NUMERIC (30)	LT				
14	PEEX_FECHACAMBIO				Date (7)	LT				
15	PEEX_REGISTRADOPOR				VARCHAR (30 BYTE)	LT				
16	PEEX_RH				VARCHAR (4 BYTE)	LT				
17	PEEX_GENERO				VARCHAR (1 BYTE)	LT		'F'		
18	PEEX_FECHANACIMIENTO				Date (7)	LT				

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
19	CIGE_IDLUGARNACIMIENTO				NUMERIC (15)	LT		0		
20	ESCG_ID				VARCHAR (10 BYTE)	LT		0		
21	REGL_ID				NUMERIC (5)	LT				
22	PEEX_FOTO				CLOB (4000)	LT				
23	PEEX_FEHACAMBIO			Y	Date	LT				

Columns Comments

No	Column Name	Description	Notes
14	PEEX_FECHACAMBIO	CAMPO DE AUDITORIA DE LA TABLA PERSONAEXTERNA	
15	PEEX_REGISTRADOPOR	CAMPO DE AUDITORIA DE LA TABLA PERSONAEXTERNA	

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
PERSONAEXTENSION_PK	PK				PEEX_ID	ASC
					PEEX_ID	ASC

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
PARTICIPANTES_PERSOEXTER_FK	PARTICIPANTES		Y		PEEX_ID

Table Name	19. GENERAL.PERSONAGENERAL
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	PEGE_ID	P		Y	NUMERIC (30)	LT				
2	PEGE_DOCUMENTOIDENTIDAD				NUMERIC (30)	LT				
3	PEGE_DIRECCION				VARCHAR (100)	LT				
4	PEGE_TELEFONO				VARCHAR (50)	LT				
5	CIGE_IDRESIDENCIA		F	Y	NUMERIC (30)	LT				
6	PEGE_MAIL				VARCHAR (50)	LT				
7	PEGE_FEHACAMBIO			Y	Date	LT				
8	PEGE_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
PERSONAGENERAL_PK	PK				PEGE_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
PERSONAGENERAL_CIUADGENE_FK	CIUADGENERAL	Y	Y		CIGE_ID

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
EMPRESAS_PERSONAGENERAL_FK	EMPRESAS	Y	Y		PEGE_ID
EXTENSIONPROFESORES_PERGEN_FK	EXTENSIONPROFESORES	Y	Y		PEGE_ID
PARTICIPANTES_PERSOGENE_FK	PARTICIPANTES		Y		PEGE_ID
PASANTIAPROFESORES_PERGENE_FK	PASANTIAPROFESORES	Y	Y		PEGE_ID
PASANTIASESTUDIANTES_PERGEN_FK	PASANTIASESTUDIANTES	Y	Y		PEGE_ID
VOLUNTARIOS_PERSONAGENERAL_FK	VOLUNTARIOS	Y	Y		PEGE_ID

Table Name	20. GENERAL.PERSONAJURIDICAGENERAL
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	PEGE_ID	P		Y	NUMERIC (30)	LT				
2	PEJG_RAZONSOCIAL				VARCHAR (100)	LT				
3	ACEC_ID		F	Y	NUMERIC (30)	LT				
4	PEJG_NATURALEZA				VARCHAR (7)	LT				
5	PEJG_REPRESENTANTELEGAL				VARCHAR (80)	LT				
6	PEJG_FEHACAMBIO			Y	Date	LT				
7	PEJG_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
PERSONAJURIDICAGENERAL_PK	PK				PEGE_ID	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
PERSONAJURIDICAGEN_ACTIECO_FK	ACTIVIDADECONOMICA	Y	Y		ACEC_ID

Table Name	21. GENERAL.PERSONANATURALGENERAL
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	PEGE_ID	P		Y	NUMERIC (30)	LT				
2	PENG_PRIMERAPELLIDO				VARCHAR (50)	LT				
3	PENG_SEGUNDOAPELLIDO				VARCHAR (50)	LT				
4	PENG_PRIMERNOMBRE				VARCHAR (50)	LT				
5	PENG_SEGUNDONOMBRE				VARCHAR (50)	LT				
6	PENG_FEHACAMBIO			Y	Date	LT				
7	PENG_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
PERSONANATURALGENERAL_PK	PK				PEGE_ID	ASC

Table Name	22. EXTENSIONUDEC.TELEFONOEMPRESAS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	TELE_TELEFONO	P		Y	VARCHAR (100 CHAR)	LT				
2	EMPR_ID		F	Y	NUMERIC (30)	LT				
3	TELE_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				
4	TELE_FECHACAMBIO			Y	Date	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
TELEFONOEMPRESAS_PK	PK				TELE_TELEFONO	ASC

Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
TELEFONOEMPRESAS_EMPRESAS_FK	EMPRESAS	Y	Y		EMPR_ID

Table Name	23. ACADEMICO.UNIDAD
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	UNID_ID	P		Y	NUMERIC (30)	LT				
2	UNID_NOMBRE				VARCHAR (100)	LT				
3	TIUN_ID				NUMERIC (30)	LT				
4	UNID_FEHACAMBIO			Y	Date	LT				
5	UNID_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
UNIDAD_PK	PK				UNID_ID	ASC

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
EXTENCIONES_UNIDAD_FK	EXTENCIONES	Y	Y		UNID_ID
PASANTIAS_UNIDAD_FK	PASANTIAS	Y	Y		UNID_ID

Table Name	24. EXTENSIONUDEC.VOLUNTARIOS
Functional Name	
Abbreviation	
Classification Type Name	
Object Type Name	

Columns

No	Column Name	PK	FK	M	Data Type	DT kind	Domain Name	Formula (Default Value)	Security	Abbreviation
1	VOLU_ID	P		Y	NUMERIC (30)	LT				
2	PEGE_ID		F	Y	NUMERIC (30)	LT				
3	VOLU_FECHAINSCRIPCION				Date	LT				
4	VOLU_FEHACAMBIO			Y	Date	LT				
5	VOLU_REGISTRADOPOR			Y	VARCHAR (30 CHAR)	LT				

Indexes

Index Name	State	Functional	Spatial	Expression	Column Name	Sort Order
ESTUEXTENCIONES_PK	PK				VOLU_ID	ASC

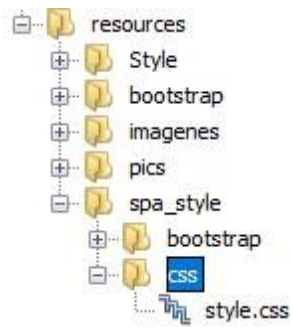
Foreign Keys (referring to)

Name	Referring To	Mandatory	Transferable	In Arc	Column Name
VOLUNTARIOS_PERSONAGENERAL_FK	PERSONAGENERAL	Y	Y		PEGE_ID

Foreign Keys (referred from)

Name	Referred From	Mandatory	Transferable	In Arc	Column Name
EXTENSIONVOLUNTARIOS_VOLUN_FK	EXTENSIONVOLUNTARIOS	Y	Y		VOLU_ID

RECURSOS



Nombre del Recurso

Css

Imagen

Bootstrap

vistas.

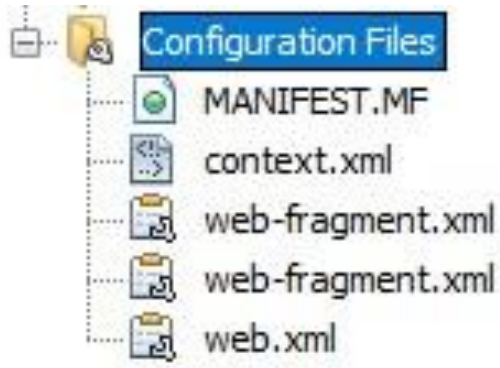
Descripción

Se almacenan los estilos usados.

Se almacena las imágenes.

Plantilla y framework para

ARCHIVOS DE CONFIGURACIÓN



Nombre del Recurso

Web-fragment.xml

Web.xml

Descripción

Guarda la configuración de tomcat.

Guarda la configuración del Spring.

CREDITOS

UNIVERSIDAD DE CUNDINAMARCA

Ing. Ana Esperanza Merchán

Contacto. 312.337.7408

ASESORA METODOLOGICA

Víctor Andres Acosta Espinosa

Contacto. 320.220.2096

ESTUDIANTE

Mateo Rodríguez Ocampo

Contacto. 301.354.8528

ESTUDIANTE



UDECA
UNIVERSIDAD DE
CUNDINAMARCA

Sistema de Información para el Registro de Pasantías y Proyectos de Proyección Social en la Facultad de Ingeniería de Sistemas en la Universidad de Cundinamarca sede Fusagasugá



MANUAL DE USUARIO DESARROLLO DE SOFTWARE UNIVERSIDAD DE CUNDINAMARCA 2018

www.unicundi.edu.co
unicundi@mail.unicundi.edu.co
Línea gratuita 018000 976000



Dirección de Sistemas y Tecnología
sistemasytecnologia@mail.unicundi.edu.co
PBX: 828 14 83 Ext. 110-170
Sede Fusagasugá



TABLA DE CONTENIDO

INTRODUCCIÓN	4
1. USUARIOS	5
1.1. DIRECTOR DE PASANTIAS	5
1.2. COORDINADOR INTERACCION UNIVERSITARIA	5
1.3. ESTUDIANTE	5
1.4. ASESOR EMPRESA	5
2. REQUISITOS DE SOFTWARE	6
3. APLICATIVO EXTENSION UNIVERSITARIA	7
3.1. INGRESO AL APLICATIVO	7
3.2. ENCUESTA EVALUACION ESTUDIANTE	8
3.3. ENCUESTA EVALUACION EMPRESA	8
3.4. CREAR PASANTIA	9
3.5. CONSULTAR PASANTIA	11
3.6. CONSULTA PRACTICA-PASANTE	13
3.7. SEGUIMIENTO PASANTIA	14
3.8. CONSULTAR CONVENIO	16
3.9. CONSULTAR EMPRESA	17
4. CONTROL DE CAMBIOS DEL MANUAL	20



TABLA DE ILUSTRACIONES

Ilustración 1.....	7
Ilustración 2.....	8
Ilustración 3.....	8
Ilustración 4.....	9
Ilustración 5.....	9
Ilustración 6.....	11
Ilustración 7.....	11
Ilustración 8.....	13
Ilustración 9.....	14
Ilustración 10.....	14
Ilustración 11.....	16
Ilustración 12.....	17



INTRODUCCIÓN

Dada la inexistencia de una tecnología que permita, en el programa de Ingeniería de Sistemas de la facultad de ingeniería de la Universidad de Cundinamarca de Fusagasugá, un control y seguimiento adecuado, ordenado y sistematizado basado en las TIC's de las pasantías y proyectos de proyección social realizadas por estudiantes, profesores o terceros, se determina la creación de un software que pueda llevar a cabo esta actividad, facilitándole al usuario la supervisión de los procesos, actividades, normativas, evaluación de las pasantías por parte de los estudiantes y las empresas, soportes, entre otras características propias de las pasantías y los proyectos de proyección social.

El ejercicio del estudiante dentro de los lineamientos de la Universidad de Cundinamarca en estos espacios (pasantías y proyectos de proyección social) es de una importancia trascendental, debido a que el desenvolvimiento adecuado equivale a la posibilidad de efectuar correctamente un puesto de trabajo.

Con la puesta en marcha del presente proyecto se busca evitar el papeleo rutinario, facilitar al usuario el corresponder a los requisitos y actividades con evidencias y/o documentos físicos, llevar un seguimiento de cada tarea, proceso, cometido, y su respectiva calificación.



1. USUARIOS

1.1. DIRECTOR DE PASANTIAS

Registro de pasantías, consultas, seguimiento.

1.2. COORDINADOR INTERACCION UNIVERSITARIA

Consultas.

1.3 ESTUDIANTE

Evaluación de la empresa.

1.4 ASESOR EMPRESA

Evaluación del pasante.



2. REQUISITOS DE SOFTWARE

- El sistema debe garantizar el acceso en cualquier momento al usuario y durante el tiempo que dure la cesión.
- El sistema debe avalar la correcta importación y exportación de archivos.
- El sistema debe registrar la información adecuada de cada pasantía o proyecto de proyección social suministrada por él usuario.
- El sistema debe permitir al usuario buscar y consultar la información pertinente de cada pasantía o proyecto de proyección social.
- El sistema controlará el acceso y lo permitirá solamente a usuarios autorizados.
- El sistema hará los correspondientes registros de la evaluación del pasante y de la empresa.



3. APLICATIVO EXTENSION UNIVERSITARIA

3.1. INGRESO AL APLICATIVO

El acceso general del aplicativo se encontrará en el módulo principal y llevará el nombre de extensión universitaria



Ilustración 1



3.2. ENCUESTA EVALUACION ESTUDIANTE

PROCESO GESTIÓN INTERACCIÓN UNIVERSITARIA

1. EVALUACIÓN DEL PASANTE O PRACTICANTE HACIA LA EMPRESA O INSTITUCIÓN

ACTIVIDAD ACADÉMICA DESARROLLADA (*)

PASANTÍA

PRÁCTICA

DATOS DEL ESTUDIANTE (*)

Nombre:

Apellidos:

Cedula:

Celular:

Correo Electronico

Ilustración 2

Este será el formato de la evaluación hacia la empresa que realizará el estudiante o practicante que efectuó la pasantía.

3.3. ENCUESTA EVALUACION EMPRESA

PROCESO GESTIÓN INTERACCIÓN UNIVERSITARIA

1. EVALUACIÓN DE LA EMPRESA O INSTITUCIÓN HACIA EL PASANTE O PRACTICANTE

ACTIVIDAD ACADÉMICA DESARROLLADA (*)

PASANTÍA

PRÁCTICA

DATOS DEL ESTUDIANTE (*)

Nombre:

Apellidos:

Cedula:

Celular:

Correo Electronico

Ilustración 3

Este será el formato de la evaluación hacia el pasante que realizará el asesor de la empresa donde se efectuó la pasantía.



3.4. CREAR PASANTIA

Iteración Universitaria

Dirección de Sistemas y Tecnología - Área Desarrollo de software
usuario MIGUEL OCTAVIO ORTIZ CEBALLOS

Inicio Crear Pasantía

DATOS DE CONVENIO

Buscar por: * Convenio* (1)

Codigo de Convenio

Mostrar 10 filas

CODIGO	NOMBRE DE CONVENIO	ESTADO
No se encontraron registros.		

Mostrando 0 a 0 de 0 filas

NIT	NOMBRE DE LA EMPRESA	TIPO
NATURALEZA	DIRECCION	DEPARTAMENTO
TELEFONO	CORREO ELECTRONICO	FECHA DE INICIO
		FECHA FINALIZACION

DATOS DE LA PASANTIA

Nombre de la pasantía*

Ilustración 4

Iteración Universitaria

Dirección de Sistemas y Tecnología - Área Desarrollo de software
usuario MIGUEL OCTAVIO ORTIZ CEBALLOS

Inicio Crear Pasantía

DATOS DE LA PASANTIA

Nombre de la pasantía* (4)

Ata de comite*

Fecha de inicio*

Fecha de finalizacion*

Sede*

Facultad de*

Programa*

INFORMACION DE PASANTES

Documento de identificación* (6)

Mostrar 10 filas

IDENTIFICACION	NOMBRES	APELLIDOS	TELEFONO	CORREO	DEPARTAMENTO	CIUDAD
No se encontraron registros.						

Mostrando 0 a 0 de 0 filas

Ilustración 5

Descripción:

1. Se establece una opción para la búsqueda de un convenio por medio de tres filtros que podrá seleccionar el usuario.
2. Se muestran los datos pertinentes al convenio buscado



3. Se muestran los datos apropiados a la empresa con el que se tiene el convenio.
4. Se realiza el registro de los datos adecuados de la pasantía.
5. Se establece una opción de búsqueda del estudiante mediante el número del documento de identificación.
6. Se muestran los datos oportunos al estudiante consultado para realizar su respectiva selección.



3.5. CONSULTAR PASANTIA

Iteración Universitaria

Dirección de Sistemas y Tecnología - Área Desarrollo de software
usuario MIGUEL OCTAVIO ORTIZ CEBALLOS

DATOS DE CONVENIO

Buscar por: * Convenio*

Codigo de Convenio

Mostrar 10 filas

CODIGO	NOMBRE DE CONVENIO	ESTADO
No se encontraron registros.		

Mostrando 0 a 0 de 0 filas

NIT	NOMBRE DE LA EMPRESA	TIPO	
NATURALEZA	DIRECCION	DEPARTAMENTO	CIUDAD
TELEFONO	CORREO ELECTRONICO	FECHA DE INICIO	FECHA FINALIZACION

1 2 3

Ilustración 6

Iteración Universitaria

Dirección de Sistemas y Tecnología - Área Desarrollo de software
usuario MIGUEL OCTAVIO ORTIZ CEBALLOS

DATOS DE PASANTIAS

No se encontraron registros.

Mostrando 0 a 0 de 0 filas

ACTA DE COMITE	NOMBRE DE LA PASANTIA	ESTADO	FACULTAD	PROGRAMA	SEDE/SECCIONAL/EXTENSION	PASANTES	FECHA DE INICIO	FECHA FINALIZACION
No se encontraron registros.								

Mostrando 0 a 0 de 0 filas

4 5

Ilustración 7

Descripción:

1. Se establece una opción para la búsqueda de un convenio por medio de tres filtros que podrá seleccionar el usuario.
2. Se muestran los datos pertinentes al convenio buscado
3. Se muestran los datos apropiados a la empresa con el que se tiene el convenio.



4. Se muestran todas las pasantías que se adecuadas al convenio solicitado
5. Se selecciona la pasantía deseada y se le da al botón de cargar la información para agregar o modificar datos de esta



3.6. CONSULTA PRACTICA-PASANTE

The screenshot shows the 'Iteración Universitaria' web application interface. The top navigation bar includes the title 'Iteración Universitaria' and a user profile for 'MIGUEL OCTAVIO ORTIZ CEBALLOS'. The main content area is titled 'Consulta Pasantía' and contains a search form with the following fields:

- Fecha***: Input field with placeholder 'dd/mm/aaaa'.
- Facultad***: Dropdown menu with 'Facultad de Ingeniería' selected.
- Programa Académico***: Dropdown menu with 'Ingeniería de Sistemas' selected.
- Sede/Seccional/Extensió***: Dropdown menu with 'Sede Pasagasuga' selected.
- Estado Pasantía***: Dropdown menu with 'ACTIVA' selected.
- Cordinador de Programa**: Input field with 'ING. MILTON'.
- Telefono**: Input field with '1234'.
- Correo**: Input field with 'jefesistemas@ucundinamarca.edu.co'.

A green 'Buscar' button is located below the form. A yellow banner above the form states 'los campos marcados con * son obligatorios'. Below the form, the title 'INFORME PASANTIA - PASANTE' is displayed above a table with the following columns:

NOMBRE	TELEFONO	EMPRESA O ENTIDAD	CIUDAD	TIPO DE ENTIDAD	CONVENIO	PASANTIA	ESTADO
--------	----------	-------------------	--------	-----------------	----------	----------	--------

Two blue callout boxes with numbers '1' and '2' are present. Box '1' points to the search filters, and box '2' points to the table header.

Ilustración 8

Descripción:

1. Se establecen los filtros de consulta para que el usuario pueda escoger a su deseo
2. Se muestran los datos relevantes de las pasantías y el pasante de acuerdo al filtro de búsqueda elegido por el usuario



3.7. SEGUIMIENTO PASANTIA

Iteración Universitaria

Dirección de Sistemas y Tecnología - Área Desarrollo de software
usuario MIGUEL OCTAVIO ORTIZ CEBALLOS

DATOS DE CONVENIO

Buscar por: * Convenio*

CODIGO	NOMBRE DE CONVENIO	ESTADO		
NIT	NOMBRE DE LA EMPRESA	TIPO		
NATURALEZA	DIRECCION	DEPARTAMENTO	CIUDAD	
PERSONA JURIDICA	TELEFONO	CORREO ELECTRONICO	FECHA DE INICIO	FECHA FINALIZACION

Ilustración 9

Iteración Universitaria

Dirección de Sistemas y Tecnología - Área Desarrollo de software
usuario MIGUEL OCTAVIO ORTIZ CEBALLOS

DATOS DE PASANTIAS

ACTA DE COMITE	NOMBRE DE LA PASANTIA	ESTADO	FACULTAD	PROGRAMA	SEDE/SECCIONAL/EXTENSION	PASANTES	FECHA DE INICIO	FECHA FINALIZACION
----------------	-----------------------	--------	----------	----------	--------------------------	----------	-----------------	--------------------

SOPORTE

TIPO DE ARCHIVO *
Seleccione * :

Seleccionar archivo No se eligió archivo

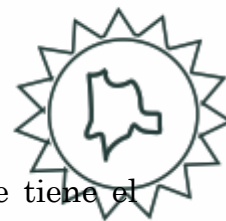
El peso del documento soporte no debe superar 2048 Kb

DESCRIPCIÓN * :

Ilustración 10

Descripción:

1. Se establece una opción para la búsqueda de un convenio por medio de tres filtros que podrá seleccionar el usuario.
2. Se muestran los datos pertinentes al convenio buscado



3. Se muestran los datos apropiados a la empresa con el que se tiene el convenio.
4. Se muestran las pasantías y su respectiva información, pertenecientes al convenio investigado
5. El usuario una vez escogida la pasantía requerida, podrá seleccionar el tipo de documento que desea cargar (Acta inicial, seguimiento, final)
6. Selecciona el documento (PDF, que no sea mayor de 2MB), desde la carpeta donde lo tiene alojado
7. El usuario podrá realizar una descripción del acta que cargo
8. El botón cargar información cumplirá con la función de ingresar la o las actas deseadas por el usuario a la pasantía solicitada por este



3.8. CONSULTAR CONVENIO

Iteración Universitaria

Dirección de Sistemas y Tecnología - Área Desarrollo de software

usuario MIGUEL OCTAVIO ORTIZ CEBALLOS

Inicio Consultar Convenio

los campos marcados con * son obligatorios

Buscar por: * Numero *

Codigo de convenio

Buscar

INFORMACION DE CONVENIO

NIT	EMPRESA	CONVENIO	ESTADO	FECHA DE INICIO	FECHA DE FINALIZACION
-----	---------	----------	--------	-----------------	-----------------------

Ilustración 11

Descripción:

1. Se establece una opción para la búsqueda de un convenio por medio de tres filtros que podrá seleccionar el usuario.
2. Se muestran los datos pertinentes al convenio buscado



3.9. CONSULTAR EMPRESA

The screenshot shows the 'Consultar Empresa' page. At the top, there is a dark green header with the text 'Iteración Universitaria' and navigation icons. Below the header, the user's profile is visible: 'MIGUEL OCTAVIO ORTIZ CEBALLOS'. The main content area has a breadcrumb trail: 'Inicio > Consultar Empresa'. A yellow warning box states 'los campos marcados con * son obligatorios'. Below this, there are two search filters: 'Buscar por:' with a dropdown menu set to 'Nombre', and 'Empresa a buscar:' with a text input field. A green 'Buscar' button is positioned below the filters. The section 'Información de la empresa' contains a table with the following columns: NIT, Nombre, Direccion, Departamento, Ciudad, Tipo, Telefono, and Correo. Two blue callout boxes with numbers '1' and '2' are present. Box '1' points to the search filters, and box '2' points to the table header.

Ilustración 12

Descripción:

1. Se establece una opción para la búsqueda de una empresa por medio de dos filtros que podrá seleccionar el usuario.
2. Se muestran los datos pertinentes de la empresa solicitada



3.10. INFORME PARA EVENTOS DE INTERACCION UNIVERSITARIA

Iteracion Universitaria

Dirección de Sistemas y Tecnología - Área Desarrollo de software

usuario MIGUEL OCTAVIO ORTIZ CEBALLOS

Inicio Informe Eventos

INFORME PARA EVENTOS DE EXTENSION UNIVERSITARIA

1. INFORMACION GENERAL

NOMBRE DEL EVENTO

ESCRIBIR EL NOMBRE DEL EVENTO

EDUCACION CONTINUA

ESCRIBIR

PROYECCION SOCIAL

ESCRIBIR

PROGRAMA ACADEMICO

ESCRIBIR

FACULTAD

Ilustración 13

Iteracion Universitaria

Dirección de Sistemas y Tecnología - Área Desarrollo de software

usuario MIGUEL OCTAVIO ORTIZ CEBALLOS

Inicio Informe Eventos

- Nombres completos de cada persona a certificar
- Tipo de documento
- Numero de documne

iueeducacioncontinua@ucundinamarca.edu.co

iuwebmaster@ucundinamarca.edu.co

Este documnto debe ser entregado en medio fisico y/o magnetico con las firmas correspondientes.

El registro fotografico debe de ser entregado unicamente en medio magnetico

FIRMA DEL REMITENTE

NOMBRES Y APELLIDOS COMPLETOS

CARGO

Exportar a pdf

Ilustración 14

Descripción:

1. Se establece el formato con sus respectivos campos para ser digitalizado por el usuario.
2. Se añade un botón para que se pueda exportar el informe en formato PDF y así poder imprimir dicho archivo.



3.11. EVALUACION PROYECCION SOCIAL

The screenshot shows a survey form on a grey background. At the top left is the logo 'e-encuesta.com'. The title 'PROCESO GESTIÓN INTERACCIÓN UNIVERSITARIA' is centered. Below it is the section '1. EVALUACION DE EVENTOS Y/O ACTIVIDADES ACADEMICAS'. The form contains three main sections: 'NOMBRE DEL EVENTO: (*)' with a text input field; 'NOMBRE DEL CONFERENCISTA: (*)' with a text input field; and 'TIPO DE EVENTO: (*)' with four radio button options: 'DIPLOMADO', 'CURSO', 'SEMINARIO', and 'CONFERENCIA'.

Ilustración 15

Este será el formato asignado por la oficina de interacción universitaria de la evaluación de proyección social.



4. CONTROL DE CAMBIOS DEL MANUAL

Actualización Nro.	Descripción del cambio	Versión del aplicativo	Fecha de cambio



www.unicundi.edu.co
unicundi@mail.unicundi.edu.co
Línea gratuita 018000 976000



GP-CER36541



CO-SC-CER395037



SC-CER365037

Dirección de Sistemas y Tecnología
sistemasytecnologia@mail.unicundi.edu.co
PBX: 828 14 83 Ext. 110
Sede Fusagasugá