



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA DE SOFTWARE

**PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO DE SOFTWARE**

**TEMA: DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL
SISTEMA DE CONTROL DE PASANTÍAS PARA LA
UNIVERSIDAD DE LAS FUERZAS ARMADAS- ESPE,
ENMARCADO EN EL “MODELO – VISTA – CONTROLADOR
(MVC)” Y UTILIZANDO LA HERRAMIENTA “YII” COMO
FRAMEWORK DEL LENGUAJE PHP**

AUTORES: RICARDO DAVID VÁSCONEZ GARCÉS

FAUSTO FABRICIO ROBAYO ZURITA

DIRECTOR: ING. EDGAR MONTALUISA

CODIRECTOR: ING. XIMENA LÓPEZ

LATACUNGA

2015

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA DE SOFTWARE

CERTIFICADO

ING. FABIÁN MONTALUISA (DIRECTOR DE TESIS)

ING. XIMENA LÓPEZ (CODIRECTOR DE TESIS)

CERTIFICAN

Que el presente trabajo de investigación, previo a la obtención del título de Ingeniero de Software, realizado por los estudiantes Ricardo David Vásconez Garcés y Fausto Fabricio Robayo Zurita; siendo parte del proyecto “Diseño de la estructura organizativa e implementación de una metodología para el desarrollo de software en la Fábrica de Software de la ESPE Extensión Latacunga.”, a fin de evaluar la factibilidad y viabilidad de la implementación de la metodología, razón por la cual se utiliza el marco teórico conceptual conjuntamente con Adela Sofía Gallardo Cueva y su aplicación ha sido cuidadosamente revisada por los suscritos y hemos podido constatar que cumple con todos los requisitos de fondo y de forma establecidos por la Universidad de las Fuerzas Armadas – ESPE por lo que autorizamos su presentación.

Latacunga, Mayo 20 de 2015

Ing. Fabián Montaluisa
DIRECTOR

Ing. Ximena López
CODIRECTOR

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA DE SOFTWARE

DECLARACIÓN DE RESPONSABILIDAD

NOSOTROS,

RICARDO DAVID VÁSCONEZ GARCÉS

FAUSTO FABRICIO ROBAYO ZURITA

DECLARAMOS QUE:

El contenido e información que se encuentra en esta Tesis denominada “DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE CONTROL DE PASANTÍAS PARA LA UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE, ENMARCADO EN EL “MODELO – VISTA – CONTROLADOR (MVC)” Y UTILIZANDO LA HERRAMIENTA “YII” COMO FRAMEWORK DEL LENGUAJE PHP” es responsabilidad exclusiva de los autores y han respetado derechos intelectuales de terceros, conforme a las fuentes que se incorporan en la bibliografía.

Latacunga, mayo 20 de 2015

Ricardo David Vásconez G.
C.I.:1804465944

Fausto Fabricio Robayo Z.
C.I. 1803282258

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA DE SOFTWARE

AUTORIZACIÓN

NOSOTROS,

RICARDO DAVID VÁSCONEZ GARCÉS

FAUSTO FABRICIO ROBAYO ZURITA

AUTORIZAMOS

A la Universidad de las Fuerzas Armadas ESPE, la publicación, en la biblioteca virtual de la Institución del trabajo de grado denominado “DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE CONTROL DE PASANTÍAS PARA LA UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE, ENMARCADO EN EL “MODELO – VISTA – CONTROLADOR (MVC)” Y UTILIZANDO LA HERRAMIENTA “YII” COMO FRAMEWORK DEL LENGUAJE PHP”, cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y autoría.

Latacunga, mayo 20 de 2015

Ricardo David Vásquez G.
C.I.:1804465944

Fausto Fabricio Robayo Z.
C.I. 1803282258

DEDICATORIA

El presente trabajo de tesis está dedicado a nuestras familias, porque ellos nos han enseñado con su ejemplo a luchar para conseguir nuestros más anhelados sueños.

David y Fabricio.

AGRADECIMIENTO

A todas las personas que en este arduo camino nos han apoyado y siempre han estado junto a nosotros y nos han ayudado a alcanzar este logro.

A la Universidad de las Fuerzas Armadas – ESPE por la excelente labor educativa que realiza y por habernos dado la oportunidad de realizar el presente trabajo de investigación.

David y Fabricio.

ÍNDICE DE CONTENIDOS

PORTADA	i
CERTIFICADO	ii
DECLARACIÓN DE RESPONSABILIDAD	ii
AUTORIZACIÓN	ii
DEDICATORIA	ii
AGRADECIMIENTO	iii
ÍNDICE DE CONTENIDOS	iv
ÍNDICE DE TABLAS	vii
ÍNDICE DE FIGURAS	viii
RESUMEN	ix
ABSTRACT	x
CAPÍTULO I	1
1. MARCO TEÓRICO	1
1.1. Patrones De Diseño.....	1
1.1.1 Patrones De Creación	3
1.1.2 Patrones Estructurales	4
1.1.3 Patrones De Comportamiento.....	5
1.2. Patrón De Diseño “Modelo – Vista – Controlador”	7
1.3. Arquitectura De Software.....	11
1.4. Frameworks De Desarrollo	12
1.4.1. Tipos De Framework Web.....	13
1.4.2. Características De Los Frameworks	13
1.5. Procesos De Desarrollo	14
1.5.1. Codificar Y Corregir.....	14
1.5.2. Modelo De Cascada.....	14

1.5.3. Desarrollo Ágil.....	15
1.5.4. Desarrollo Iterativo O Incremental.....	15
1.5.5. Modelo En Espiral	15
1.6. Metodología De Las Tic's De La Universidad De Las Fuerzas Armadas Espe Extensión Latacunga	16
1.6.1. Fases De La Metodología	16
a) Formalización De La Necesidad.....	17
b) Requerimientos	18
c) Diseño	19
d) Implementación	20
e) Pruebas	21
f) Puesta En Marcha	22
g) Artefactos Y Manuales	22
1.6.2. Formalización De La Necesidad.....	23
1.6.3. Requerimientos	27
1.6.4. Diseño	30
1.6.5. Implementación.....	35
1.6.6. Pruebas.....	37
1.6.7. Puesta En Marcha.....	39
1.6.8. Manuales.....	42
CAPÍTULO II.....	43
2. DESARROLLO DEL PRODUCTO SOFTWARE.....	43
2.1. Formalización De La Necesidad	43
2.2. Requerimientos.....	43
2.3. Diseño.....	43
2.4. Implementación	43

2.5. Pruebas	44
2.6. Liberación	44
CAPÍTULO III.....	45
3. MANUAL DE USUARIO Y CAPACITACIÓN AL CLIENTE	45
CAPÍTULO IV	46
4. CONCLUSIONES Y RECOMENDACIONES	46
4.1. Conclusiones	46
4.2. Recomendaciones	46
BIBLIOGRAFÍA	48
ANEXOS.....	49

ÍNDICE DE TABLAS

Tabla 1.1. Patrones de diseño	2
Tabla 1.2. Patrones de diseño - Ventajas y Desventajas.....	6
Tabla 1.3. Características de los Frameworks	133
Tabla 1.4. Formalización de Desarrollo de Software	244
Tabla 1.5. Planificación de Entregas.....	267
Tabla 1.6. Requisitos Específicos	29
Tabla 1.7. Diseño de Interfaces	312
Tabla 1.8. Diseño de base de datos	334
Tabla 1.9. Versión de Software	356
Tabla 1.10. Pruebas de Sistema.....	38
Tabla 1.11. Entrega del Sistema.....	401

ÍNDICE DE FIGURAS

Figura 1.1. MVC con PHP5.....	9
Figura 1.2. Proceso de desarrollo de software	144
Figura 1.3. Metodología para desarrollo de la Fábrica de Software.....	18

RESUMEN

En el presente proyecto se realizó el diseño, desarrollo e implementación del sistema de control de pasantías para la Universidad de las Fuerzas Armadas ESPE Extensión Latacunga, partiendo de la necesidad de tener un software para el seguimiento y control de los pasantes para lo cual en la actualidad se cuenta con formatos con los cuales se está trabajando, teniendo la necesidad de sistematizar todo el proceso para generar automáticamente los mismos y así poder administrar la información, mantener a mano un historial ordenado de todos los alumnos y docentes que intervienen en el proceso de pasantías, se trabajó con la “METODOLOGÍA DE LAS TIC’s DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE EXTENSIÓN LATACUNGA” la cual nos permite agilizar la documentación de software, esta se apoya el patrón “Modelo-Vista-Controlador (MVC)” con el framework de desarrollo “Yii” el mismo que nos ayudó a agilizar el desarrollo del sistema y trabajar a la par con la metodología. El desarrollo en tres capas “Modelo-Vista-Controlador” y el uso del framework Yii nos permitió desarrollar en lenguaje “PHP” proporcionando la seguridad requerida por considerarse información como reservada. El sistema desarrollado nos permite generar los formatos específicos para cada estudiante, nos proporciona un conjunto de reportes de caso, estudiante y reportes de cada una de las Carreras. El sistema maneja perfiles de usuarios para cada uno de los roles involucrados en todo el proceso de pasantías.

PALABRAS CLAVE

- **TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN**
- **DESARROLLO DE SOFTWARE**
- **APLICACIONES WEB**
- **PROGRAMACIÓN WEB**

ABSTRACT

This project deals with the design, development and implementation of an internship control system for the Universidad de las Fuerzas Armadas ESPE, at its Latacunga campus, starting from the need of having software for the tracking and control of interns. Currently, paper forms are being used, which creates a need for the systematization of all the process so that those forms can be generated automatically in order to manage the information and keep an orderly register of all the students and teachers who intervene in the internships process. Work will be developed through as “ICT Methodology of the Universidad de las Fuerzas Armadas ESPE – Latacunga Campus” which has already been implemented. This methodology will facilitate documentation of the software. The methodology is based on the pattern “Model-View-Controller (MVC)” and on “Yii” development framework which will help to speed up the development of the system and to work along with the methodology. Development in three layers “Model-View-Controller” and use of the Yii framework will allow establishment of the PHP language which supplies required safety since the information is deemed classified. The developed system will allow generation of the specific forms for each student; it will supply a set of case reports, student reports and reports for each of the degrees. The system will manage user profiles for each of the roles involved in all the internships process.

KEY WORDS

- **INFORMATION AND COMMUNICATION TECHNOLOGIES**
- **SOFTWARE DEVELOPMENT**
- **WEB APPLICATIONS**
- **WEB PROGRAMMING**

CAPÍTULO I

1. MARCO TEÓRICO

1.1. PATRONES DE DISEÑO

En la actividad productiva del desarrollo del Software, cada una de las tareas debe presentar su planificación para asegurar la concepción de metas, manteniendo latente dificultades y brechas que han sido solucionadas; analizando un entorno análogo podemos observar que en un edificio, un auto o un sistema informático, se debe recordar que los problemas que se presenten en el proceso alguien ya los resolvió, es ahí donde surgen los patrones de diseño los cuales ayudan a utilizar las soluciones ya probadas y documentadas por otras personas a problemas que son comunes en una situación en particular como es el caso del desarrollo del producto software.

Para los desarrolladores de sistemas, lo ideal sería contar con una colección de estos patrones con los problemas que resuelven, pero se debe considerar que los mismos tienen su grado de complejidad, partiendo del hecho que son creados por otras personas y aplicados por quienes tienen la necesidad de dar solución a un problema, sin embargo, una vez que se comprende su funcionamiento y aplicación los diseños serán mucho más flexibles, modulares y especialmente reutilizables dentro de un macro entorno que presta su facilidad de aceptación.

Gamma en su obra manifiesta que los patrones de diseño son “descriptores de clases y objetos relacionados que están particularizados para resolver un problema de diseño general en un determinado contexto” (Erich, Helm, Johnson, & Vlissides, 2003).

Existen dos criterios base para la clasificación de los patrones de Diseño como se puede observar en la Tabla 1.1.

Tabla 1.1.
Patrones de diseño

		Propósito		
		De creación	Estructurales	De comportamiento
Ámbito	Clase	Factory Method	Adapter (de clases)	Interpreter Template Method
	Objeto	Abstract Factory Builder Prototype Singleton	Adapter (de objetos) Bridge Composite Decorator Facade Flyweight Proxy	Chain of Responsibility Command Iterator Mediator Memento Observer State Strategy Visitor

Fuente: (Erich, Helm, Johnson, & Vlissides, 2003)

En base al propósito; es decir, en función de lo que cada patrón realiza, tenemos la siguiente clasificación: creacionales, estructurales y de comportamiento.

- **Patrones Creacionales:** Están relacionados con los procesos de creación de objetos. Sin descuidar la instancia de creación como su configuración, es decir, ayudan a encapsular y abstraer dicha creación.
- **Patrones Estructurales:** Tratan sobre la composición de clases u objetos, separan la interfaz de la implementación, se ocupan de como las clases y objetos se agrupan para formar estructuras más grandes.
- **Patrones de Comportamiento:** Describen el modo en que las clases y objetos interactúan y distribuyen sus responsabilidades, así como los algoritmos que encapsulan: es decir, muestran la comunicación entre los objetos.

Existen patrones que aplican a las clases, los mismos que se ocupan en las relaciones de clases y subclases que se crean gracias a la herencia establecida en los tiempos de compilación; así como patrones que se aplican a los objetos y a su vez tratan de las relaciones entre ellos que pueden cambiar en tiempo de ejecución.

Las diversas formas de organizar patrones nos permiten utilizarlos juntos o en forma alterna para interactuar con sus componentes sin desvirtuar el efecto del negocio en la aplicación. “Tener muchas formas de pensar en los patrones le hará comprender mejor qué es lo que hacen, como compararlos y cuándo aplicarlos.” (Erich, Helm, Johnson, & Vlissides, 2003).

A continuación una lista resumen con los patrones de diseño orientado a objetos más utilizados según (Erich, Helm, Johnson, & Vlissides, 2003)

1.1.1 Patrones de creación

- **Abstract Factory.-** Proporciona una interfaz para crear familias de objetos o que dependen entre sí, sin especificar sus clases concretas.
- **Builder.-** Separa la construcción de un objeto complejo de su representación, de forma que el mismo proceso de construcción pueda crear diferentes representaciones.
- **Factory Method.-** Define una interfaz para crear un objeto, pero deja que sean las subclases quienes decidan qué clase instanciar.

Este patrón se utiliza cuando se tiene múltiples variantes de una sola entidad. Por ejemplo, se tiene una clase `button` la cual tiene diferentes variaciones como: `ImageButton`, `InputButton` y `FlashButton`; dependiendo del lugar, se necesita crear diferentes botones, es aquí donde se puede utilizar `Factory Method` para crear los botones.

- **Prototype.-** Especifica los tipos de objetos a crear por medio de una instancia prototípica, y crear nuevos objetos copiando este prototipo.
- **Singleton.-** Garantiza que una clase sólo tenga una instancia y proporciona un punto de acceso global a ella.

Si se requiere pasar una instancia específica de una clase a otra, se puede usar `Singleton` para evitar tener que pasar la instancia vía el constructor o un argumento, con lo cual se logra acceder desde distintos lugares de la programación e incluso de diferentes clases.

1.1.2 Patrones estructurales

- **Adapter.-** Convierte la interfaz de una clase en otra distinta que es la que esperan los clientes con la finalidad de que pueda ser utilizada por una clase que de otra forma no lo haría.

Un buen ejemplo, es cuando se crea una clase dominio para las clases de tabla y llamar sus funciones una a una, se puede encapsular todos estos métodos en un método que utilice una clase adaptador. Esto no sólo permite reutilizar cualquier acción que se desee, sino también evitará que se tenga que reescribir el código si se requiere usar la misma acción en un sitio distinto.

- **Bridge.-** Desvincula una abstracción de su implementación, de manera que ambas puedan variar de forma independiente.
- **Composite.-** Combina objetos en estructuras de árbol para representar jerarquías de parte-todo, esto permite que los clientes traten de manera uniforme a los objetos individuales y a los compuestos.
- **Decorator.-** Añade dinámicamente nuevas responsabilidades a un objeto, proporcionando una alternativa flexible a la herencia para extender la funcionalidad.

El momento más adecuado para usar Decorator es cuando se tiene una entidad que necesita tener un nuevo comportamiento sólo si la situación así lo requiere, por ejemplo, se tiene un enlace HTML que realiza cosas diferentes dependiendo de la página en la que se encuentre, como puede ser mostrar el link subrayado si se encuentra en una página y resaltado si está en otra.

- **Facade.-** Proporciona una interfaz unificada para un conjunto de interfaces de un subsistema
- **Flyweight.-** Usa el compartimiento para permitir un gran número de objetos de grano fino de forma eficiente.
- **Proxy.-** Proporciona un sustituto o representante de otro objeto para controlar el acceso a éste.

1.1.3 Patrones de comportamiento

- **Chain of Responsibility.-** Evita acoplar el emisor de una petición a su receptor, al dar a más de un objeto la posibilidad de responder a la petición, para esto crea una cadena con los objetos receptores y pasa la petición a través de la cadena hasta que esta sea tratada por algún objeto.
- **Command.-** Encapsula una petición en un objeto, permitiendo así paramétrica a los clientes con distintas peticiones, encolar o llevar un registro de las peticiones y poder deshacer la operaciones.
- **Interpreter.-** Dado un lenguaje, define una representación de su gramática junto con un intérprete que usa dicha representación para interpretar las sentencias del lenguaje.
- **Iterator.-** Proporciona un modo de acceder secuencialmente a los elementos de un objeto agregado sin exponer su representación interna.
- **Mediator.-** Define un objeto que encapsula cómo interactúan un conjunto de objetos. Promueve un bajo acoplamiento al evitar que los objetos se refieran unos a otros explícitamente, y permite variar la interacción entre ellos de forma independiente.
- **Memento.-** Representa y externaliza el estado interno de un objeto sin violar la encapsulación, de forma que éste puede volver a dicho estado más tarde.
- **Observer.-** Define una dependencia de uno-a-muchos entre objetos, de forma que cuando un objeto cambia de estado se notifica y actualizan automáticamente todos los objetos.
- **State.-** Permite que un objeto modifique su comportamiento cada vez que cambia su estado interno. Parecerá que cambia la clase del objeto.
- **Strategy.-** Define una familia de algoritmos, encapsula uno de ellos y los hace intercambiables. Permite que un algoritmo varíe independientemente de los clientes que lo usan.
- **Template Method.-** Define en una operación el esqueleto de un algoritmo, delegando en las subclases algunos de sus pasos. Permite

que las subclases redefinan ciertos pasos del algoritmo sin cambiar su estructura.

- **Visitor.-** Representa una operación sobre los elementos de una estructura de objetos. Permite definir una nueva operación sin cambiar las clases de los elementos sobre los que opera.

El uso de patrones de diseño no es solo un estilo arquitectural es también una forma de afrontar los proyectos a nivel de trabajo del equipo de desarrollo, ciclo de vida del proyecto, lenguaje a utilizar con los expertos en el negocio, etc.; sin embargo también identifica una serie de patrones de diseño y estilo de arquitectura concreta, es también, una aproximación concreta para diseñar software basándose sobre todo en la importancia del dominio del negocio, sus elementos y comportamientos y las relaciones entre ellos; tiene sus ventajas y desventajas las cuales se resumen en la Tabla 1.2.

Tabla 1.2.
Patrones de diseño - Ventajas y Desventajas

Ventajas	Desventajas
Permite reducir operaciones complejas presentando son soluciones concretas: Los patrones son recetas de diseño, se pueden clasificar, manteniendo independencia para exponerlos como simple.	No posee ingeniería inversa
Las soluciones técnicas dada una determinada situación presenta un escenario interesante de desacoplamiento entre los objetos utilizando un lenguaje específico y otros	Es difícil interpretar el patrón: los patrones no se reflejan en el código empleado.
Se aplican en soluciones muy comunes: Proceden de la experiencia y tiene utilidad demostrada en problemas del diseño orientado a objetos.	El patrón de diseño es sumamente costoso y difícil en lenguajes que no se alinean al paradigma de desarrollo.

CONTINÚA



Presenta soluciones simples: Resuelven un problema particular en la mayoría de aplicaciones utilizando un pequeño número de clases con flexibilidad y finamente con calidad de software.	Su utilización es complicada por el hecho que las tecnologías actuales incentivan a los desarrolladores para particionar la aplicación en la fase de diseño
La utilidad se vuelve cada vez más importante: Nos permite reducir tiempo de desarrollo por lo tanto costo incrementando calidad	Tiempo: El tiempo dedicación en la fase inicial es elevado siendo relativo ya que su ventaja de mantenimiento es productiva
Estructura general: Su estructura es conocida por todos los programadores, de tal manera que su producto al ser independiente no resulte distinta de su forma de trabajar.	
Permiten tener una estructura de código común a todos los proyectos que implemente una funcionalidad genérica.	
Una aplicación con MVC es mucho más mantenible, extensible y modificable	

En la Tabla 1.2. se puede evidenciar que las ventajas da una diferencia competitiva en el uso de patrones de diseño, considerando que existe trabajo ya realizado, probado y aceptado entorno de desarrollo de producto software aportando con una solución satisfactoria en tiempo y costo aceptable sobre todo con una calidad marcada.

En diseño de aplicaciones es fundamental anticiparse a nuevos requisitos y a cambios en los ya existentes de manera que el diseño soporte estas variaciones y evolucionen sin mayores complicaciones, sin correr el riesgo de rediseñar por completo. Los cambios no previstos pueden implicar el rediseño de la aplicación lo que involucra: redefiniciones y nuevas implementaciones de clases, modificar los clientes y repetir el ciclo de pruebas, lo cual siempre resulta ser muy costoso. Los patrones de diseño ayudan a evitar este tipo de problemas al dejar que algún aspecto de la estructura del sistema varíe de manera independiente, lo que da como resultado sistemas más robustos frente a cambios concretos.

1.2. PATRÓN DE DISEÑO “MODELO – VISTA – CONTROLADOR”

Para el diseño de aplicaciones Web con interfaces enriquecidas y que ofrezcan una fuerte interactividad con el usuario se utiliza el patrón de diseño

Modelo-Vista-Controlador. Aquí se conjuga la interfaz lógica y la lógica de negocio esto se debe a la abstracción del objeto en un orden de relación jerárquica entre ellos, es decir la herencia guardan un relación de orden jerárquico entre si donde se pueden establecer claras diferencias prácticas, esto trae consigo mayor trabajo y una alta probabilidad de error.

Con este patrón se trata de realizar un diseño que separe la vista del modelo, con la finalidad de mejorar la reusabilidad, de esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos.

En el patrón de diseño de software Modelo-Vista-Controlador (MVC) todo el proceso está dividido en 3 capas:

- **Modelo:** En esta capa se hace el levantamiento de todos los objetos que debe utilizar el sistema, es decir, es el proveedor de recursos. Es en donde se encapsulan los datos y la lógica del negocio, esto hace que sea independiente de la base de datos que se utilice así como de los medios de representación de los datos.
- **Vista:** Es la capa de presentación la cual muestra la información del modelo al usuario, se agrupan todas las clases y archivos que tengan relación de la interfaz de usuario. Cada vista tiene asociado un componente controlador que se indica a continuación.
- **Controlador:** El controlador viene a ser el orquestador de los diferentes eventos generados en la interfaz de usuario, se encarga de llamar en el modelo al experto del negocio que sabe que es lo que hay que hacer con la petición del usuario. Una vez que el modelo ha realizado su tarea se lo comunica al controlador. El controlador invoca a la vista o interfaz para que se actualice con los cambios hechos en el modelo. En resumen, el Controlador es el que escucha los cambios en la vista y se los envía al modelo, el cual le regresa los datos a la vista, es un ciclo donde cada acción del usuario causa que se inicie un nuevo ciclo.

En la Figura 1.1. se muestran las responsabilidades de cada capa del patrón de diseño MVC y cómo interactúan entre sí.

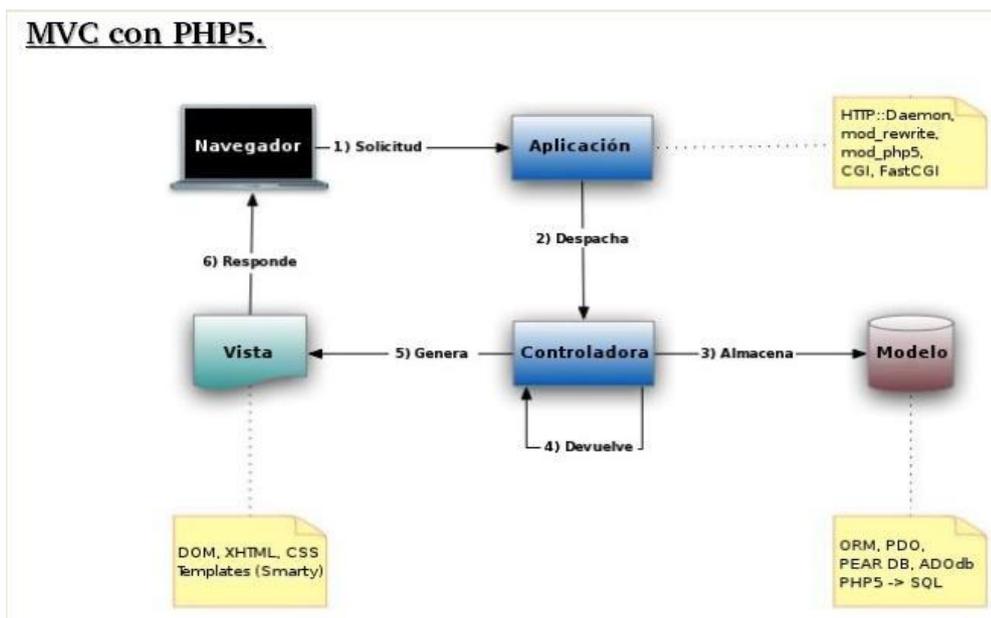


Figura 1.1. MVC con PHP5
Fuente: (Caballero & Lara, 2006)

La interacción que se muestra en la Figura 1 es la siguiente:

1. Solicitud: El usuario ejecuta un navegador, pide un URL solicitando a la Aplicación.
2. Despacha: La Aplicación delega a la capa **Controladora** la petición del usuario.
3. Almacena: Consulta en la capa **Modelo** por medio de la capa de datos del manejador de base de datos existente.
4. Devuelve: La capa de Datos devuelve los resultados en datos puros a la capa **Controladora**.
5. Genera: La capa **Controladora** genera la capa de **Vista** en base a los tipos de datos generados.
6. Responde: La **Vista** generada se envía como respuesta a la solicitud del navegador.

Con lo expuesto se resume las responsabilidades básicas de cada componente del patrón MVC de la siguiente manera:

- El **modelo** es responsable de:
 - Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.

- Define las reglas de negocio.
- El **controlador** es responsable de:
 - Recibe los eventos de entrada.
 - En base a reglas de gestión de eventos se realizan peticiones al modelo o a las vistas.
- Las **vistas** son responsables de:
 - Recibir datos del modelo y los muestran al usuario.
 - Tienen un registro de su controlador asociado (normalmente porque además lo instancia).

MVC implementa varios patrones a la vez, así tenemos que la relación entre la Vista y el Controlador es una implementación del patrón Strategy. MVC utiliza otros patrones de diseño, tales como el Factory Method y el Decorator. Factory Method se utiliza para especificar la clase controlador predeterminada de una vista y Decorator para añadir la capacidad de desplazamiento de una vista, sin embargo, las principales relaciones se dan entre los patrones Observer, Composite y Strategy.

“Si una aplicación no está diseñada siguiendo el patrón de diseño MVC, inevitablemente se tendrá problemas de escalabilidad, mantenimiento y de extensión” (Caballero & Lara, 2006). Se puede afirmar que en aplicaciones Web la programación utilizando el patrón de diseño MVC no es una alternativa sino una necesidad que evitará en el futuro muchos dolores de cabeza gracias a sus características que en resumen son las siguientes:

- Patrón de diseño orientado a objetos.
- Adecuado para aplicaciones Web con alta interacción humana. Separa clara y consistentemente las preocupaciones en las capas indicadas anteriormente.
- Permite múltiples representaciones (vistas) de la misma información
- (modelo).
- Facilidad para agregar, eliminar o modificar interfaces de usuario. Facilita los desarrollos simultáneos con actualizaciones de interfaces, lógica del negocio o incluso introducir una nueva aplicación sin la necesidad de afectar a otro código fuente.

- Garantiza la reducción del código fundamentalmente porque los modelos pueden ser utilizados en varias vistas.
- Los desarrolladores centran su atención en un solo aspecto de la aplicación al mismo tiempo.

Estas características conducen a que un proyecto de software Web se implemente de una forma ágil con un producto sostenible y mantenible en el tiempo y que en el futuro puede migrar en cualquier dirección. Estas son las razones fundamentales por las que se utilizará el patrón MVC en el presente proyecto.

1.3. ARQUITECTURA DE SOFTWARE

El diseño global del sistema usando un patrón definido se denomina arquitectura.

La arquitectura de software, tiene que ver con el diseño y la implementación de estructuras de software de alto nivel. Es el resultado de ensamblar un cierto número de elementos arquitectónicos de forma adecuada para satisfacer la mayor funcionalidad y requerimientos de desempeño de un sistema, así como requerimientos no funcionales, como la confiabilidad, escalabilidad, portabilidad y disponibilidad. (Kruchten, 1995)

La arquitectura de software de un programa o de un sistema computacional está definida por la estructura, comprendida por los elementos de software, las propiedades visibles de esos elementos y las relaciones entre ellos. (Gomaa, 2011)

Incluyendo:

- La descripción de los componentes con los cuales se construyen los sistemas
- Las interacciones entre esos componentes
- Patrones para guiar la composición
- Restricciones sobre dichos patrones
- Componentes: servidores, clientes, bases de datos, filtros, capas en un sistema jerárquico, etc.
- Interacciones: llamadas a procedimientos, protocolos C/S, protocolos de acceso a BD, etc.

La arquitectura de software se encarga de:

- Diseño preliminar o de alto nivel.
- Organización a alto nivel del sistema, incluyendo aspectos como la descripción y análisis de propiedades relativas a su estructura y control global, los protocolos de comunicación y sincronización utilizados, la distribución física del sistema y sus componentes, etc.
- Otros aspectos relacionados con el desarrollo del sistema y su evolución y adaptación al cambio:

Composición, reconfiguración, reutilización, escalabilidad, mantenibilidad, etc.

Los aspectos que la arquitectura de software no se ocupa:

- Diseño detallado.
- Diseño de algoritmos.
- Diseño de estructuras de datos.

1.4. FRAMEWORKS DE DESARROLLO

La palabra Framework por su conceptualización es utilizada en diversas áreas de negocio, académica, investigación, etc. Por ejemplo encontramos frameworks para el fortalecimiento en estimulación temprana, en el área legislativa en contratación pública, así como también en el área de software no tiene una utilización específica, en este trabajo teórico práctico lo orientamos al desarrollo de software y en concreto en el ámbito de aplicaciones web; ya que como objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

Un framework Web, por tanto, podemos definirlo como un conjunto de partes específicas que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web sin llegar a obtener los límites tope del mismo, permitiendo así su crecimiento y complejidad a lineado a un esquema y misión empresarial.

1.4.1. Tipos de Framework WEB

“Existen varios tipos de frameworks Web: orientados a la interfaz de usuario, como Java Server Faces, orientados a aplicaciones de publicación de documentos, como Coocoon, orientados a la parte de control de eventos, como Struts y algunos que incluyen varios elementos como Tapestry.

La mayoría de frameworks Web se encargan de ofrecer una capa de controladores de acuerdo con el patrón MVC o con el modelo 2 de Servlets y JSP, ofreciendo mecanismos para facilitar la integración con otras herramientas para la implementación de las capas de negocio y presentación.” (Gutiérrez., 2006)

1.4.2. Características de los frameworks

A continuación enunciamos una serie de características que podemos encontrar en prácticamente todos los frameworks existentes.

Tabla 1.3.
Características de los Frameworks

Abstracción de URLs y sesiones.	No es necesario manipular directamente las URLs ni las sesiones, el framework ya se encarga de hacerlo.
Acceso a datos.	Incluyen las herramientas e interfaces necesarias para integrarse con herramientas de acceso a datos, en Base de Datos, XML, etc.
Controladores.	La mayoría de frameworks implementa una serie de controladores para gestionar eventos, como una introducción de datos mediante un formulario o el acceso a una página. Estos controladores suelen ser fácilmente adaptables a las necesidades de un proyecto concreto.
Autenticación y control Incluyen mecanismos para la identificación de usuarios de acceso.	Incluyen mecanismos para la identificación de usuarios mediante login y password y permiten restringir el acceso a determinadas páginas a determinados usuarios.
Internacionalización.	
Separación entre diseño y contenido.	

Fuente: (Gutiérrez., 2006)

1.5. PROCESOS DE DESARROLLO

Un proceso de desarrollo de software o llamado ciclo de vida es la estructura básica para el desarrollo del producto software, existes varios modelos a seguir para establecer el proceso de desarrollo de software de manera que éste cumpla los requisitos del cliente tanto en tiempo, costo, alcance y calidad. Este proceso es afectado a las habilidades e ingenio del grupo de trabajo.

Lo indicado en términos globales se muestra en la Figura 1.2.



Figura 1.2. Proceso de desarrollo de software
Fuente: (Ivar, 2004).

El proceso de desarrollo de software no es único. Existen varios modelos para el para el desarrollo muchos cuentan con pros y contras, en el proyecto se debería escoger el que más se ajuste a la necesidad o también se puede trabajar con modelos híbridos, es decir combinar varios modelos. Existen muchos modelos genéricos de proceso de software que vamos a revisar brevemente a continuación.

1.5.1. Codificar Y Corregir

El ciclo de vida codificación y corrección es más una estrategia a la falta de experiencia o presión que se ejerce sobre el grupo de desarrollo para cumplir con el tiempo de entrega, sin dedicar tiempo al diseño, el grupo de trabajo empieza a trabajar directamente en el código y realiza sus pruebas, obteniendo inevitables errores que se solucionan antes de la entrega del producto Software.

1.5.2. Modelo de Cascada

El modelo de cascada es un proceso donde el grupo de trabajo tiene que seguir las siguientes fases sucesivamente:

- 1) Especificación de requisitos
- 2) Diseño de software
- 3) Integración
- 4) Pruebas(o validación)
- 5) Despliegue(o instalación)
- 6) Mantenimiento

Para pasar de fase se debe finalizar la fase anterior, se puede hacer una revisión al terminar la fase para estar seguro o poder corregir y seguir a la siguiente fase.

1.5.3. Desarrollo Ágil

“El desarrollo ágil de software utiliza un desarrollo iterativo como base para abogar por un punto de vista más ligero y más centrado en las personas que en el caso de las soluciones tradicionales. Los procesos ágiles utilizan retroalimentación en lugar de planificación, como principal mecanismo de control. La retroalimentación se canaliza por medio de pruebas periódicas y frecuentes versiones del software.” (Mann, 1997)

1.5.4. Desarrollo Iterativo o Incremental

El desarrollo iterativo recomienda la construcción de secciones reducidas del software que poco a poco se irán incrementando para facilitar la detección de errores antes de que sea demasiado tarde, el desarrollo interactivo es ideal para clientes que no tienen una idea clara de lo que desean.

1.5.5. Modelo En Espiral

La principal característica del modelo en espiral es la gestión de riesgos de forma periódica en el ciclo de desarrollo. La ventaja de este modelo es que tiene un análisis interactivo y concienzudo de los riesgos que nos sirve especialmente en sistemas de gran escala. El espiral se visualiza como un proceso que pasa a través de algunas interacciones con el diagrama de los cuatro cuadrantes:

- 1) Crea planes para identificar el objetivo
- 2) Analiza los riesgos es la fase principal del modelo
- 3) Desarrollo y Verificación

4) Planificación de Software

Cada proyecto de software requiere de una forma particular para abordar el problema. Las propuestas comerciales y académicas actuales promueven procesos iterativos.

1.6.METODOLOGÍA DE LAS TIC's DE LA UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE EXTENSIÓN LATACUNGA

La metodología de desarrollo de software constituye la guía para el proceso de desarrollo de software, debiendo acoplarse a los procesos internos de las empresas de desarrollo. La metodología deberá proporcionar las guías necesarias que orienten el desarrollo, proporcionando los suministros necesarios para documentar el trabajo realizado sin llegar a constituirse en una receta que conlleve la creación de documentos que no aportan significativamente a la obtención del producto software.

En la ESPE Extensión Latacunga se vive una realidad particular, producto del régimen militar que guía los procesos administrativos que están orientados hacia los ejes de Educación, Investigación y Extensión. En este ámbito se propone una metodología que sea capaz de ofrecer un camino seguro que guíe el desarrollo de software permitiendo lograr un producto software que cumpla los requerimientos internos de la Universidad así como proyectos externos. Estos proyectos deberán adaptarse para poder ser desarrollados en un entorno regido por las políticas institucionales.

La Metodología proporcionará un conjunto de fases a ser seguidas y dentro de las cuales se desarrollarán un grupo específico de entregables que sustentarán el trabajo realizado, propiciando la entrega de productos de software y la consecución de un buen nivel de calidad reflejado en la operatividad de la aplicación.

1.6.1. Fases de la Metodología

La metodología de desarrollo software que es objeto de esta propuesta consta de varias fases que orientarán el desarrollo del producto software en un proceso que se ejecuta al iniciar el proyecto que no deberá exceder de 3 días, un proceso iterativo el cual dará como resultado el desarrollo de versiones de software con operatividad autónoma que se ejecutarán en un

tiempo de 2 a 4 semanas, y un proceso de entrega donde el usuario final podrá usar el sistema para ejecutar sus procesos administrativos y se proporcionará la capacitación en un tiempo entre 3 a 5 días.

Las fases de la Metodología de Desarrollo de Software propuesta que plantea, se esquematizan en la Figura 1.3. y son las siguientes:

- Formalización de la Necesidad
- Requerimientos
- Diseño
- Implementación
- Pruebas
- Puesta en Marcha

La metodología propicia el desarrollo en un entorno en el cual los requerimientos se presentan de forma urgente y en la cual el contacto con el dueño del procesos suele ser dificultoso por cuestiones de la organización y los procesos internos, así como la urgencia de contar con la herramienta de software que ayude a acelerar el trabajo de forma automatizada.



Figura 1.3. Metodología para desarrollo de la Fábrica de Software
Fuente: (López & Montaluisa, 2015)

a) Formalización de la Necesidad

La fase de Formalización de la Necesidad constituye el primer proceso de la metodología que se ejecutará por única vez al inicio del proceso metodológico.

Esta fase determina el inicio del proyecto, propiciada por la iniciativa de desarrollo de una aplicación de software como respuesta a una necesidad. Considerando las circunstancias que rodean el entorno en el cual se originan estas necesidades, en esta fase se establece el inicio del proyecto de desarrollo mediante una disposición que es propiciada por la Unidad Operativa Requirente, esta necesidad se modelará mediante un pedido formal que se denominará “Formalización de la Necesidad”. La Formalización de la Necesidad establecerá los responsables y una primera indagación de lo requerido, citando las leyes, reglamentos y procesos establecidos que respalden dicha necesidad.

La Formalización de la Necesidad tendrá como finalidad dar inicio formalmente al proceso de desarrollo de software, comprometer a los actores a aportar en todas las fases requeridas para el desarrollo, e identificar el entorno en el cual se desarrollará y se ejecutará la aplicación de software desarrollada.

En esta fase se establecerá un cronograma de entregas parciales que determinará el alcance del proyecto y de las versiones de software a desarrollarse, estas versiones tendrán autonomía y que en su conjunto conformarán la aplicación de software resultante del proyecto.

Se propicia garantizar los compromisos adquiridos tanto por parte del desarrollador como del requirente.

b) Requerimientos

La fase de Requerimientos es la segunda fase del proceso metodológico, y se constituye como un componente que es parte del proceso repetitivo para el desarrollo de las versiones de software. Esta fase se ejecutará cada vez que se repita el proceso de desarrollo de una nueva versión, en esta fase se detallará la descripción formal de los requerimientos logrando plasmar el alcance del sistema.

Se considera en la metodología que el desarrollo del proyecto se ejecuta en un entorno complejo que dificulta la obtención de los requerimientos y debido al corto tiempo que se establece para el desarrollo de los proyectos se propone la esquematización de historias de usuarios con la variante que

el técnico responsable de la elaboración del sistema será el encargado de recopilar y plasmar en el formato la información necesaria para detallar el requerimiento y luego deberá confrontarlo con el usuario responsable del proceso a fin de legalizarlo. Para esta fase deberá aprovechar al máximo el tiempo disponible del personal operativo responsable de los procesos y recopilar la información necesaria aplicando técnicas como:

- Entrevistas al personal involucrado con los procesos a automatizar
- Recopilación de documentos de apoyo y formatos que hayan sido establecidos para el trabajo diario
- Observación de reglamentos e instructivos desarrollados para la ejecución de proyectos que están relacionados con el requerimiento

Los requerimientos describirán de la mejor forma posible la información necesaria para obtener versiones de software de rápida entrega y que contemplen funcionalidades que el usuario final pueda iniciar a ejecutar, los requerimientos serán agrupados y levantados desde dos puntos de vista:

- La necesidad del usuario
- El criterio del técnico que permitirá estructurar paquetes de software de uso autónomo.

En la definición de los requerimientos se establecerá una sección que permita establecer los aspectos necesarios que serán validados en las siguientes fases del desarrollo a fin de establecer si el software desarrollado cumple las expectativas del usuario que lo requiere.

c) Diseño

La fase de diseño constituye la tercera fase y es parte del proceso iterativo de desarrollo de la aplicación de software, su ejecución se refleja en los modelos producto de los requerimientos establecidos para la versión de software en ejecución.

El proceso de diseño contemplará la esquematización de los requerimientos agrupados por funcionalidades, por consiguiente generará modelos parciales que provean funcionalidad independientes.

La fase de diseño considerará los siguientes aspectos que deberán ser modelados:

- **Diseño de Datos.-** El diseño establecerá el modelo de datos que dará soporte al sistema en desarrollo, se realizará mediante la aplicación de software relacionado directamente con el manejador de base de datos que se haya definido para el proyecto. Por ejemplo para MySQL se usará MySQL Workbench.
- **Diseño de Interfaces.-** El Diseño de interfaces se establecerá de acuerdo al Framework a utilizarse, se realizará un único diseño base para todas las interfaces de mantenimiento de las tablas en la cual se establecerá el posicionamiento para:
 - Texto en el formulario
 - Cajas de entrada de datos (texto, combos, casillas de selección, etc.)
 - Botones de comando y enlaces de funcionalidad
 - Gráficos y logos

Se personalizará aquellas interfaces que requieran de complejidad en la implementación y que difieren del esquema general de entradas y presentación de datos. El diseño de interfaces determina en forma gráfica mediante esquemas la distribución y operatividad deseada de la interface la que concordará con los requerimientos establecidos.

d) Implementación

La fase de Implementación es la cuarta fase del proceso iterativo de desarrollo y se enfocará en obtener código de forma rápida, con la mayor fiabilidad posible y con un alto grado de calidad.

Para lograr un estilo homogéneo y eficiente se deberá observar los diseños establecidos de interfaces y se utilizará técnicas de programación que permitan acelerar la generación de código, tal como Frameworks que posibiliten generar código fuente con una arquitectura de software MVC (Modelo Vista Controlador).

El objetivo al integrar el Framework y la arquitectura MVC es permitir un desarrollo rápido de cada módulo y las versiones establecidas en el

cronograma de entregas, pasando de la necesidad al producto software de la forma más rápida posible y la entrega al usuario final luego de las pruebas y validaciones necesarias.

La Implementación generará versiones de software que serán autónomos y fácilmente acoplables en un producto final.

El proceso para la generación de código establece seguir varios pasos de forma metódica, los que tomando como insumo los diseños de datos e interfaces generarán el código:

- Creación de Tablas de Datos en el manejador de base de datos
- Uso del generador de código
- Personalización del Modelo
- Personalización de la Vista
- Personalización del Controlador
- Pruebas Unitarias (Funcionalidad del código)

Para la implementación, así como para el resto de fases se recomienda el uso de herramientas de software dentro del marco que regula a las instituciones públicas. El decreto 1014 recomienda el uso del Software Libre, y en este marco podemos usar herramientas de desarrollo como NetBeans, así como herramientas de gestión de versiones como SubVersion (SVN).

e) Pruebas

La fase Pruebas es la quinta fase y es parte del proceso iterativo de desarrollo. Una vez finalizada la generación del código se realizarán pruebas del software a fin de garantizar la calidad mediante la concordancia entre los requerimientos, el diseño y el producto resultante.

Las pruebas se establecerán en 2 puntos a fin de minimizar los fallos residuales en el software:

- **Pruebas de Desarrollo.-** Las realizará el técnico a cargo del desarrollo y se basa en examinar de forma detallada los objetos creados y su funcionalidad. Propicia la eliminación de errores a nivel de código la funcionalidad individual del módulo.

- **Pruebas de Función.-** Las realizará un técnico a fin encontrar residuos de fallos que no han sido detectados y también establecerá la correcta relación entre los módulos desde el punto de vista funcional del software, de esta manera se asegura que la versión de software sea un elemento entregable al usuario final. En esta evaluación se contrapondrá los requerimientos establecidos para la versión de software contra el desempeño de la aplicación desarrollada.

Conseguiremos entonces una versión que pasará a estado de producción y podrá ser implantada.

f) Puesta en Marcha

La Fase de puesta en marcha se ejecutará al finalizar cada una de las versiones de software planificadas. La ejecución de esta fase requiere la puesta en marcha de la versión de software, realizando paralelamente la capacitación sobre las funcionalidades incluidas en la determinada versión.

Se orientará al usuario final a utilizar las funcionalidades implementadas y se determinará el grado de satisfacción sobre los requerimientos.

Se realizará un Acta de Conformidad en la cual se indicará las funcionalidades implementadas en la versión y se registrarán las novedades suscitadas en la capacitación.

Mientras se avanza en el proceso de Puesta en Marcha por parte de un técnico, el grupo de trabajo iniciará inmediatamente a trabajar en la nueva versión.

Los fallos residuales detectados en la Puesta en Marcha serán incluidos como parte de la nueva versión a fin de viabilizar de forma fluida el desarrollo del software.

g) Artefactos y Manuales

Para lograr la operatividad de la metodología se plantea un conjunto de artefactos de apoyo, que permitirán documentar el proceso de desarrollo e integrar el esfuerzo de las personas involucradas en el proyecto. Estos artefactos se establecerán en cada una de las fases de la metodología.

Se requiere además de los formatos de base para los manuales que estarán a disposición del cliente o usuario final para poder implantar el sistema o capacitar a nuevos funcionarios que no fueron parte del grupo de personal capacitado durante la ejecución del proyecto.

Para poder organizar los entregables que se definen en la metodología se ha establecido un árbol de directorios que contendrán cada uno de los entregables de forma organizada y nos permitirán que todos los miembros del equipo de desarrollo sean capaces de encontrar fácilmente los artefactos que han desarrollado otros colaboradores.

- Raíz [Repositorio de Proyectos de Desarrollo]
 - Proyecto [Nombre Corto del Proyecto]
 - Entregables [Documentos de Texto y Modelos]
 - Código [Versiones de Código Fuente]
 - Sistema Versión 1.0 [Primera Versión]
 - Sistema Versión 2.0 [Segunda Versión]
 - Manuales [Versiones de Manuales]

Una vez establecido el repositorio para los artefactos que serán desarrollados como producto de la metodología procederemos a detallar cada uno de ellos de acuerdo a la fase en la cual se elaborarán.

1.6.2. Formalización de la Necesidad

En esta primera fase se establecen los documentos que permiten dar legalidad al pedido y proporcionan el marco de responsabilidad de las áreas involucradas en el futuro desarrollo.

Se plantea la utilización de los siguientes documentos:

- **Formalización de Desarrollo de Software.-** Este documento recopila la necesidad del desarrollo expresado en una disposición, memorando o pedido formal para el desarrollo de la aplicación. Estará compuesto por una descripción general del proyecto de desarrollo, la especificación de los responsables para la ejecución del proyecto, legalizado por las firmas del requirente y del responsable del grupo de desarrollo.

Tabla 1.4. Formalización de Desarrollo de Software
Formalización de Desarrollo de Software

Plantilla: “FORMALIZACIÓN DE DESARROLLO DE SOFTWARE”		
Función de la Plantilla: <p>Formalizar el pedido de desarrollo de software, logrando la identificación de las entidades participantes en el proceso de desarrollo, y realizando acuerdos que permitirán una adecuada planificación y garanticen la participación activa tanto del grupo de desarrollo como del requirente.</p>		
Objetivos de la Plantilla: <ul style="list-style-type: none"> • Formalizar la necesidad presentada mediante un pedido verbal o mediante una disposición documentada. • Identificar las personas que participarán en el proceso de desarrollo tanto de técnicos como de funcionarios requirentes. • Comprometer la participación del grupo de desarrollo y del requirente mediante el suministro de información y la entrega en los plazos establecidos del producto de software. 		
Alcance de la Plantilla: <p>Esta plantilla se usará al inicio del proyecto para formalizar el pedido.</p>		
Notas: <p>Esta plantilla se guardará como un archivo en Formato de Texto (MS Word 2007-2010) y su nombre seguirá la nomenclatura siguiente: Formalizacion.[nombre_de_proyecto].v[número_version(0.0)].docx</p>		
Versiónes de la Plantilla:		
Versión:	Fecha:	Responsable:
1.0	10-noviembre-2014	Ing. Edgar Montaluisa Ing. Ximena López

CONTINÚA



FORMALIZACIÓN DE DESARROLLO DE SOFTWARE	
Proyecto: [Nombre del Proyecto]	
Fecha de Reunión: [Fecha de Reunión entre las partes]	Documento de Requerimiento: [No. de documento de requerimiento]
Unidad Requirente: [Nombre de la Unidad Requirente]	Integrantes Unidad Requirente: [Personal que participará directamente en el proyecto]
Unidad Desarrollo: [Nombre de la Unidad Desarrollo]	Integrante Grupo de Desarrollo: [Personal del Grupo de Desarrollo que participará directamente en el proyecto]
Descripción: [Relato detallado del requerimiento, que permita establecer el objetivo principal que se persigue con su desarrollo; No incluye requisito solo una descripción global del proyecto a desarrollarse; Debe ser lo suficientemente claro para establecer el alcance global y las iteraciones que serán necesarias para la implementación]	
Nombre del Responsable RESPONSABLE UNIDAD REQUIRENTE	Nombre del Líder del Proyecto RESPONSABLE DEL GRUPO DE DESARROLLO

- **Planificación de Entregas.-** Este documento se establecerá como resultado del análisis del requerimiento general, refleja el cronograma de entregas parciales de las versiones resultantes del proceso de desarrollo, las versiones deberán ajustarse en el tiempo establecido

por la metodología, entre 2 y 4 semanas. Es necesario considerar que se ejecutará un proceso completo de desarrollo por cada versión. Este cronograma tendrá la aceptación del grupo de desarrollo y de la unidad requirente, siendo compromiso que a la entrega de cada versión se proceda inmediatamente a trabajar con las funcionalidades desarrolladas. En este documento se detallará de forma resumida que contendrá cada versión a entregar a fin de que sirva de guía para las fases posteriores.

Tabla 1.5.
Planificación de Entregas

<p>Plantilla:</p> <p>“PLANIFICACIÓN DE ENTREGAS”</p>
<p>Función de la Plantilla:</p> <p>Esta plantilla establecerá los tiempos en los cuales se realizará la entrega de las versiones de software planificadas en el desarrollo.</p>
<p>Objetivos de la Plantilla:</p> <ul style="list-style-type: none"> • Especificar los tiempos estimados en los cuales se entregarán las versiones de software. • Preparar a la Unidad Requirente para el proceso de inducción y organizar su trabajo para dar tiempo a la capacitación del nuevo sistema. • Organizar el trabajo del personal técnico para el cumplimiento de los tiempos estimados en el desarrollo de software.
<p>Alcance de la Plantilla:</p> <p>Esta plantilla se usará luego de haber formalizado el desarrollo de software y previo al proceso iterativo de desarrollo de la aplicación.</p>
<p>Notas:</p> <p>Esta plantilla se guardará como un archivo en Formato de Texto MS Word 2007-2010) y su nombre seguirá la nomenclatura siguiente: Planificacion.[nombre_de_proyecto].v[número_version(0.0)].docx</p>

CONTINÚA



Versiones de la Plantilla:			
Versión:	Fecha:	Responsable:	
1.0	10-noviembre-2014	Ing. Edgar Montaluisa Ing. Ximena López	
PLANIFICACIÓN DE ENTREGAS			
Proyecto: [Nombre del Proyecto]			
Fecha Elaboración: [Fecha de Elaboración del Documento]		Tiempo Total Entrega: [Tiempo Total de Desarrollo en días]	
Cronograma de Entrega de Versiones:			
Versión:	Tiempo Desarrollo	Fecha Entrega:	Resumen de la Entrega:
[#Versión]	[En días]	[Fecha]	[Breve descripción de la Funcionalidad a entregar en la versión]
Nombre del Responsable RESPONSABLE UNIDAD REQUIRENTE		Nombre del Líder del Proyecto RESPONSABLE DEL GRUPO DE DESARROLLO	

1.6.3. Requerimientos

En esta fase, que constituye parte del proceso iterativo de desarrollo, se inicia con la especificación detallada del problema en forma de Historia de Usuario, identificando los requerimientos y las relaciones que permitan llegar

a la comprensión del problema. En esta fase se utilizará el siguiente documento para cada una de las historias de usuarios identificadas:

- **Requisitos Específicos.-** Este documento basado en el formato de Historia de Usuarios sintetiza la información necesaria y suficiente para avanzar en el proceso de desarrollo. En ningún momento se pretenderá que su elaboración tome tanto tiempo que retrase el desarrollo de software, su objetivo será guiar y proporcionar los suministros para el diseño, implementación y pruebas del sistema. Para la especificación del requerimiento se utilizará de un documento que recopile esta necesidad de forma detallada.

Tabla 1.6.
Requisitos Específicos

Plantilla: “REQUISITOS ESPECÍFICOS”
Función de la Plantilla: Esta plantilla permitirá realizar el levantamiento de los requisitos funcionales detallando de la mejor manera cada uno de los requerimientos del usuario.
Objetivos de la Plantilla: <ul style="list-style-type: none"> • Especificar los requisitos funcionales de cada una de las versiones a desarrollar de la aplicación. • Definir las pruebas a realizarse como comprobación del software realizado, a fin de determinar el grado de implementación del requisito en el sistema.

CONTINÚA



Alcance de la Plantilla:		
Esta plantilla se usará de forma iterativa por cada ciclo que se realice en el desarrollo de una versión específica y será el primero en ser diseñado a fin de proporcionar los parámetros para el desarrollo de la aplicación en las siguientes fases.		
Notas:		
Esta plantilla se guardará como un archivo en Formato de Texto (MS Word 2007-2010) y su nombre seguirá la nomenclatura: Requisitos.[nombre_de_proyecto].v[número_version(0.0)].docx		
Versiones de la Plantilla:		
Versión:	Fecha:	Responsable:
1.0	10-noviembre-2014	Ing. Edgar Montaluisa Ing. Ximena López
REQUISITOS ESPECÍFICOS		
Proyecto:		
[Nombre del Proyecto]		
Fecha Elaboración:	Versión de Software:	
[Fecha de Elaboración del Documento]	[Número de la Versión de desarrollo de software a la que pertenece el requisito]	
ID de Requisito:	Título:	
[Número de Requisito en "R"+3 dígitos]	[Nombre que resumen la funcionalidad del requisito]	

CONTINÚA



Descripción: [Relato que detalla el proceso a automatizarse, identificando los actores, documentos de fuente de la información, documentos de salida de información, las validaciones requeridas, los atributos de información que deberán ser considerados. Incluirá un gráfico explicativo de requerirse]	
Estimación: [Número de Horas que se estima tomará su automatización]	Usuario: [Usuario quien requiere la funcionalidad del sistema]
Prioridad: [Escala en números enteros de 1 a 10 que indica la urgencia de realizar el requerimiento]	Dependencia: [ID de los requisitos de los cuales depende este nuevo requisito para poder ser implementado.]
Fecha Revisiones: [Fecha en formato dd-mm-aaaa de las revisiones realizadas, y breve descripción de la motivación de la revisión]	
Pruebas: [Se establecerá las pruebas que se realizarán en la aplicación desarrollada para poder verificar la correcta implementación del requisito. Se establecerán los datos de entrada y los resultados esperados]	
Usuario Responsable USUARIO DE UNIDAD REQUIRENTE	Nombre del Técnico TÉCNICO RESPONSABLE DEL DESARROLLO

1.6.4. Diseño

En el diseño se establecerán dos artefactos, los que detallarán aspectos como la interface del sistema y al almacenamiento de información. El

modelamiento de las interfaces estándar se realizará en la primera versión, y en las siguientes versiones solamente aquellas que difieran del estándar. El documento de interfaces será único ya que es un repositorio incremental de los diseños que se podrán usar en el desarrollo de la aplicación. El modelamiento de datos será incremental de acuerdo a los requerimientos establecidos en cada versión y a la Planificación de Entregas, en el documento se especificará la instancia correspondiente a cada una de las versiones implementadas. Se establecen dos artefactos para esta fase:

- **Diseño de Interfaces.-** Este documento detallará las interfaces estándar a utilizarse en todas las vistas de mantenimiento de la información, estas serán usadas durante el desarrollo de todas las versiones de software. Además proporcionará la plantilla para la documentación de nuevas interfaces que se vayan presentando en el desarrollo del proyecto y sus versiones. El diseño de la interface guardará concordancia con la utilización de las herramientas de desarrollo a fin de minimizar el paso del diseño a la implantación.

Tabla 1.7.
Diseño de Interfaces

Plantilla: “DISEÑO DE INTERFACES”
Función de la Plantilla: Esta plantilla detallará la ubicación básica de los componentes en las interfaces a fin de mantener una armonía en el diseño. Los procesos que son recurrentes se modelarán una sola vez, dando la opción de modelar de forma detallada solo las interfaces que requieren mayor complejidad.
Objetivos de la Plantilla: <ul style="list-style-type: none"> • Especificar el diseño de las interfaces a utilizar, usando diagramas de bloques indicando el posicionamiento de elementos como texto, cajas de entrada de datos, cajas de selección, imágenes, etc. • Detallar aquellas interfaces que requieran mayor complejidad de interpretación. • Guardar armonía en el diseño de las interfaces

CONTINÚA 

Alcance de la Plantilla:		
Esta plantilla se usará de forma iterativa en cada ciclo que se realice en el desarrollo de todas las versiones. Se diseñará luego de identificar si el requisito requiere un nuevo diseño de los anteriores ya existentes. El diseño es general y no específico a cada requerimiento.		
Notas:		
Esta plantilla se guardará como un archivo en Formato de Texto (MS Word 2007-2010) y su nombre seguirá la nomenclatura: Interfaces. [nombre_de_proyecto]. v[número_version(0.0)].docx		
Versiones de la Plantilla		
Versión:	Fecha:	Responsable:
1.0	10-noviembre-2014	Ing. Edgar Montaluisa Ing. Ximena López
DISEÑO DE INTERFACES		
Proyecto:		
[Nombre del Proyecto]		
Fecha Elaboración:	Técnico:	
[Fecha de Elaboración del Documento]	[Nombre del técnico que propuso el uso de la interface]	
Título:		
[Nombre que resumen la funcionalidad de la interface]		

CONTINÚA



<p>Descripción:</p> <p>[Relato que detalla la concepción de la interface, especificando en qué casos se debe utilizar este diseño. Se acompañará con una imagen que represente en Bloques el diseño]</p>
<p style="text-align: center;">Nombre del Técnico Responsable</p> <p style="text-align: center;">TÉCNICO RESPONSABLE DEL DESARROLLO</p>

- **Diseño de Base de Datos.-** El documento establece cada una de las versiones de incremento a la base de datos, capturando de forma gráfica y proporcionando los lineamientos de documentación de las innovaciones implementadas. El documento establecerá la operatividad que proporciona la versión de base de datos en base a los requisitos establecidos, especifica los almacenamientos creados en la versión.

Tabla 1.8.
Diseño de base de datos

<p>Plantilla:</p> <p>“DISEÑO DE BASE DE DATOS”</p>
<p>Función de la Plantilla:</p> <p>Detallar la funcionalidad de la base de datos que soportará la versión correspondiente de software desarrollado.</p>
<p>Objetivos de la Plantilla:</p> <ul style="list-style-type: none"> • Especificar el modelo de datos que soportará la implementación de una versión completa desarrollada. • Especificar los requerimientos que se han considerado en la versión de la Base de Datos. • Permitir la trazabilidad en la evolución de la Base de Datos

CONTINÚA



Alcance de la Plantilla:		
Esta plantilla se elaborará como resumen del modelamiento de datos, indicando los requisitos implementados y la versión final de la base de datos a ser entregada en la versión del Software.		
Notas:		
Esta plantilla se guardará como un archivo en Formato de Texto (MS Word 2007-2010) y su nombre seguirá la nomenclatura siguiente: Base.[nombre_de_proyecto].v[número_version(0.0)].docx		
Versiones de la Plantilla:		
Versión:	Fecha:	Responsable:
1.0	10-noviembre-2014	Ing. Edgar Montaluisa Ing. Ximena López
DISEÑO DE BASE DE DATOS		
Proyecto:		
[Nombre del Proyecto]		
Fecha Elaboración:	Versión del Software:	
[Fecha de Elaboración del Documento]	[Número de versión del software a la cual da soporte la Base de Datos]	
Requisitos Implementados:	Técnico:	
[ID de los Requisitos implementados en la base de datos]	[Nombre del técnico que realizó el modelo de la Base de Datos]	

CONTINÚA



Descripción:

[Relato que detalla las funcionalidades a las cuales da soporte la Base de Datos, especificando las implementaciones de la nueva versión. Será necesario anexar la referencia del archivo de modelo de ser necesario, usando el mismo nombre que este documento]

Nombre del Técnico Responsable

TÉCNICO RESPONSABLE DEL DESARROLLO

1.6.5. Implementación

La implementación al ser un proceso de desarrollo de la aplicación requiere la documentación de la versión creada, para lo cual se hará uso del documento:

- **Versión de Software.-** Este documento especificará en número de versión, los requisitos implementados, los parámetros necesarios para la implantación y las ubicaciones de los códigos fuentes. Cada versión incrementará la funcionalidad de la versión anterior y detallará si se han incluido correcciones a versiones anteriores como parte del proceso de desarrollo.

Tabla 1.9.**Versión de Software****Plantilla:**

“VERSIÓN DE SOFTWARE”

Función de la Plantilla:

Detallar la funcionalidad que se ha implementado en una versión específica de software, especificando la operatividad y las correcciones realizadas sobre la versión anterior.

CONTINÚA



Objetivos de la Plantilla. <ul style="list-style-type: none"> • Detallar las nuevas funcionalidades de la versión desarrollada. • Detallar las correcciones realizadas sobre una versión anterior. • Definir el entorno necesario para la puesta en marcha de la versión de software desarrollada. 		
Alcance de la Plantilla: <p>Esta plantilla se elaborará como resumen de la implementación de cada versión de software desarrollada, recopilando las funcionalidades y requerimientos implementados.</p>		
Notas: <p>Esta plantilla se guardará como un archivo en Formato de Texto (MS Word 2007-2010) y su nombre seguirá la nomenclatura siguiente: Software.[nombre_de_proyecto].v[número_version(0.0)].docx</p>		
Versiones de la Plantilla:		
Versión:	Fecha:	Responsable:
1.0	10-noviembre-2014	Ing. Edgar Montaluisa Ing. Ximena López
VERSIÓN DE SOFTWARE		
Proyecto: <p>[Nombre del Proyecto]</p>		
Fecha Elaboración: <p>[Fecha de Elaboración del Documento]</p>	Versión del Software: <p>[Número de versión del software]</p>	
Requisitos Implementados: <p>[ID de los requisitos que componen esta nueva versión de software]</p>	Requisitos Corregidos: <p>[ID de los requisitos revisados en esta nueva versión de software]</p>	

CONTINÚA



<p>Descripción:</p> <p>[Relato que detalla las funcionalidades implementadas en la versión de software desarrollado. Descripción de los requerimientos técnicos y operativos para la implantación de la versión. Detallará el repositorio en el cual se encontrará la versión del código fuente de la aplicación]</p>
<p style="text-align: center;">Nombre del Técnico Responsable</p> <p style="text-align: center;">TÉCNICO RESPONSABLE DEL DESARROLLO</p>

1.6.6. Pruebas

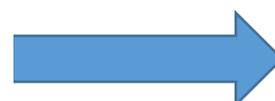
La ejecución de la fase de pruebas de función concluirá con la elaboración del documento que se denominará:

- **Pruebas del Sistema.-** Este documento se elaborará luego de haber retroalimentado hacia la fase de desarrollo los fallos detectados y se considere resueltos y superados todos estos. El documento se elaborará al finalizar y haber superado las pruebas, resumiendo las falencias encontradas y los fallos reportados y solucionados. Registrará el tiempo tomado para el proceso de revisión y el tiempo que fue requerido para la solución de fallos.

Tabla 1.10.
Pruebas de Sistema

<p>Plantilla:</p> <p>“PRUEBAS DE SISTEMA”</p>
<p>Función de la Plantilla:</p> <p>La función de esta plantilla es detallar los fallos detectados durante la revisión de funcionalidad del software y los resultados de la retroalimentación hacia el grupo de desarrollo, así como las soluciones establecidas.</p>

CONTINÚA



Objetivos de la Plantilla:		
<ul style="list-style-type: none"> • Detallar los fallos detectados sobre la versión final desarrollada. • Retroalimentar las soluciones aplicadas para solventar los fallos detectados. • Establecer el nivel de cumplimiento del proceso de desarrollo de software y la obtención de una versión de software de calidad. 		
Alcance de la Plantilla:		
Esta plantilla se elaborará al final de haber realizado las pruebas de funcionalidad del sistema y deberá mostrar un balance positivo sobre los fallos detectados y solucionados.		
Notas:		
Esta plantilla se guardará como un archivo en Formato de Texto (MS Word 2007-2010) y su nombre seguirá la nomenclatura siguiente: Pruebas.[nombre_de_proyecto].v[número_version(0.0)].docx		
Versiones de la Plantilla:		
Versión:	Fecha:	Responsable:
1.0	10-noviembre-2014	Ing. Edgar Montaluisa Ing. Ximena López
PRUEBAS DE SISTEMA		
Proyecto:		
[Nombre del Proyecto]		
Fecha Elaboración:	Versión del Software:	
[Fecha de Elaboración del Documento]	[Número de versión del software]	

CONTINÚA



Requisitos Validados: [ID de los requisitos que componen esta nueva versión de software y que han sido validados]		Técnico Revisor: [Nombre y Firma del Técnico encargado de la Revisión del software]
Tiempo de Revisión: [Tiempo en número de horas que se utilizaron para la revisión]		Tiempo de Soluciones: [Tiempo reportado por el Grupo de Desarrollo sobre los fallos reportados]
Descripción de Fallos: [Relato que detalla los fallos encontrados y como este afecta al desempeño del sistema, las acciones tomadas para solucionarlas y el estado actual del sistema revisado]		
No.	Fallo:	Solución:
[Secuencial de Fallos]	[Descripción del Fallo Reportado]	[Descripción de la Solución Aplicada por el grupo de desarrollo]
<p>Nombre del Técnico Responsable</p> <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>		

1.6.7. Puesta en Marcha

La puesta en marcha requerirá la formalización de la entrega de la versión a fin de iniciar con el trabajo por parte del requirente con la versión desarrollada. Al finalizar el proceso de puesta en marcha y capacitación se elaborará el documento:

- **Entrega del Sistema.-** En este documento se detallará la versión, las personas capacitadas, las novedades suscitadas en la capacitación y la implementación del Sistema. Se requiere de la aceptación mediante

la firma de los usuarios capacitados y la autoridad requirente. Se compromete al usuario a usar el sistema de forma que permita acoplar las nuevas versiones de forma continua y se establecen los compromisos a cumplirse en lo referente al registro de información usando el sistema.

Tabla 1.11.
Entrega del Sistema

Plantilla: “ENTREGA DEL SISTEMA”
Función de la Plantilla: Detallar los aspectos más importantes del proceso de entrega de la versión, indicando al requirente la operatividad implementada en esta versión del software y contando con la aceptación del mismo.
Objetivos de la Plantilla: <ul style="list-style-type: none"> • Detallar las funcionalidades entregadas en la versión del software. • Detallar las novedades de capacitación, tales como: personal capacitado y novedades de la implantación. • Propiciar la continuidad del desarrollo del proyecto, evidenciado la aceptación por parte del responsable del proceso y del técnico responsable del proyecto.
Alcance de la Plantilla: Esta plantilla se elaborará al final del ciclo de desarrollo de la versión de software, y en forma posterior al proceso de puesta en marcha y capacitación. De esta manera se finaliza el proceso de desarrollo de la versión y propicia la nueva iteración.
Notas: Esta plantilla se guardará como un archivo en Formato de Texto (MS Word 2007-2010) su nombre seguirá la nomenclatura siguiente: Entrega.[nombre_de_proyecto].v[número_version(0.0)].docx

CONTINÚA



Versiones de la Plantilla:		
Versión:	Fecha:	Responsable:
1.0	10-noviembre-2014	Ing. Edgar Montaluisa Ing. Ximena López
ENTREGA DEL SISTEMA		
Proyecto: [Nombre del Proyecto]		
Fecha Elaboración: [Fecha de Elaboración del Documento]		Versión del Software: [Número de versión del software]
Personal Capacitado: [Nombre de las personas y número de horas usadas en la inducción de la nueva versión del software, sumilla del Personal Capacitado]		Técnico Capacitador: [Nombre del técnico encargado de la implantación y Capacitación]
Novedades de Puesta en Marcha y la Capacitación: [Detalle de las novedades suscitadas en la capacitación y el estado de ejecución del sistema por parte de los capacitados]		
Detalle de la Versión: [Funcionalidades implantadas en la versión de Software]		
Nombre del Responsable RESPONSABLE UNIDAD REQUIRENTE		Nombre del Líder del Proyecto RESPONSABLE DEL GRUPO DE DESARROLLO

1.6.8. Manuales

El desarrollo de manuales se convierte en un trabajo que no es planificado en el tiempo de desarrollo, razón por la cual se propone que los manuales a ser incluidos en cada proyecto sean los siguientes:

- **Manual de Administrador.-** El manual de administrador deberá contar con las instrucciones necesarias para poner en funcionamiento el sistema, será desarrollado por el técnico encargado de la implantación del sistema, quien debe conocer los requerimientos técnicos de funcionamiento de la plataforma en la que se ejecuta el sistema. Deberá incluir lo siguiente:
 - Proceso de instalación y configuración de la Plataforma Base (Por ejemplo: Apache, PHP, MySQL)
 - Creación de la Base de Datos del Sistema a partir del Script y configuraciones de seguridad.
 - Instalación y configuración de la aplicación.
 - Registro de Usuarios.
 - Inicializaciones importantes para arrancar el sistema.
- **Manual de Usuario.-** El manual de usuario será desarrollado en versiones de acuerdo a las versiones de software que serán liberadas y puestas en marcha. Su desarrollo estará a cargo del técnico encargado de la puesta en marcha aprovechando la retroalimentación por parte del usuario final. En cada versión del manual de usuario deberá contener los siguientes aspectos:
 - Ingreso y salida al sistema
 - Manejo de los menús de opciones
 - Manejo de las interfaces desarrolladas por procesos
 - Manejo de reportes y exportación de datos a otros formatos
 - Índices de búsqueda temática
 - Gráficos ilustrativos

Los manuales estarán disponibles en repositorios en formato PDF que permita un fácil acceso vía internet

CAPÍTULO II

2. DESARROLLO DEL PRODUCTO SOFTWARE

2.1. FORMALIZACIÓN DE LA NECESIDAD

Mediante reuniones de trabajo y observación directa, se establece el compromiso para el desarrollo de este producto software en la metodología detallada en el capítulo anterior y en cumplimiento de la legislación, procedimiento y artefactos de desarrollo se obtiene el artefacto de formulación de software para el sistema de pasantías que podemos evidenciar en Anexo 1.

Nota:

Esta planificación se realizó con los requerimientos del Departamento de Eléctrica y Electrónica sin ajustarse a un estándar de la Universidad de las Fuerzas Armadas ESPE; ya que, no se contemplaba el mismo. Desde el mes junio del año 2014, como se evidencia en el memorándum número 2014-303-espe-ugvc, se genera formatos y procedimientos estándares que desvirtúan los formatos y procedimientos con los que se estaban trabajando en ese momento debiendo cambiar el cronograma de trabajo con la finalidad de ajustarse a una vigencia del tema real.

2.2. REQUERIMIENTOS

Mediante reuniones de trabajo y observación directa, se establece los requisitos para el desarrollo de este producto software y cumpliendo con la metodología se aplica los artefactos que se encuentran en Anexo 2.

2.3. DISEÑO

Mediante reuniones de trabajo y observación directa, se establece los requisitos para el desarrollo de este producto software y cumpliendo con la metodología se aplica los artefactos que se encuentran en Anexo 3.

2.4. IMPLEMENTACIÓN

Mediante reuniones de trabajo y observación directa, se establece los requisitos para el desarrollo de este producto software y cumpliendo con la metodología se aplica los artefactos que se encuentran en Anexo 4.

2.5. PRUEBAS

Mediante reuniones de trabajo y observación directa, se establece los requisitos para el desarrollo de este producto software y cumpliendo con la metodología se aplica los artefactos que se encuentran en Anexo 5.

2.6. LIBERACIÓN

Mediante reuniones de trabajo y observación directa, se establece los requisitos para el desarrollo de este producto software y cumpliendo con la metodología se aplica los artefactos que se encuentran en Anexo 6.

CAPÍTULO III

3. MANUAL DE USUARIO Y CAPACITACIÓN AL CLIENTE

El manual del usuario se ha realizado en un video, el mismo que requiere de un reproductor básico de formato MP4, el mismo que se encuentra al servicio de los involucrados del proceso.

La capacitación puntual al cliente queda a la disposición de la Unidad de Tecnología de la Información de ESPE Latacunga, ya que se realiza la entrega formal del sistema, su documentación y su manual de usuario.

CAPÍTULO IV

4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- En el presente tema investigativo el uso del patrón MVC mediante el framework Yii permitió reducir los tiempos de desarrollo en un 25% del previsto inicialmente, principalmente por la reutilización de los modelos de la base de datos en muchas opciones del sistema y la validación de formularios propia del framework.
- Al desarrollar el presente proyecto, se pudo determinar que la metodología implementada para las TIC's de la Universidad de las Fuerzas Armadas ESPE extensión Latacunga es de tipo ágil y minimiza los tiempos establecidos de desarrollo, salvo por algunos inconvenientes que suscitaron en el proceso de administración de pasantías de la Universidad, en donde se involucran las formas, formatos, actores, etc. que de cierta forma debemos comprometer en nuestras interfaces.
- En el desarrollo del presente proyecto se logró ejecutar cada una de las fases de desarrollo sin descuidar la generación de la documentación garantizando así la aplicación de la normativa, estandarización y buenas prácticas de desarrollo.
- La aplicación de los artefactos correspondientes a cada una de las fases de la metodología utilizada, permitió la documentación necesaria para liberar el software obtenido, a que otro equipo de trabajo tome la posta para su optimización.

4.2. RECOMENDACIONES

- Se recomienda el uso del software libre para desarrollo y ejecución de aplicaciones principalmente por la reducción de costos y los buenos resultados de su aplicación.
- Para proyectos similares, es recomendable la utilización de frameworks de desarrollo que combinados con la metodología de las TIC's de la Universidad de las Fuerzas Armadas ESPE extensión Latacunga, impulsan a dar soluciones rápidas y de alta calidad.

- Se recomienda la utilización de bases de datos de uso libre como MySQL o Postgresql, por cuanto no tienen costo y poseen excelentes características de rendimiento principalmente MySQL que es el estándar de facto para aplicaciones Web.
- Se recomienda contratar un certificado de seguridad SSL (Secure Socket Layer) con la finalidad de complementar la seguridad de la aplicación en la transferencia de datos entre el navegador y el servidor Web.
- Se recomienda actualizar la malla de formación de tal forma que los estudiantes perciban y vivan una experiencia real del desarrollo de software en ingeniería y con buenas prácticas.

BIBLIOGRAFÍA

- Caballero, L., & Lara, J. (2006). Tercer Congreso Nacional Software Libre Venezuela. Merida.
- Erich, G., Helm, R., Johnson, R., & Vlissides, J. (2003). *Patrones de Diseño*. Madrid: Addison Wesley.
- Gomaa, H. (2011). *Software modeling and design uml use cases patterns and software architectures*. Cambridge.
- Gutiérrez., J. J. (07 de 05 de 2006). *Lenguajes y Sistemas Informático*. Recuperado el 21 de 04 de 2015, de Lenguajes y Sistemas Informático:
http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf
- Ivar, J. (2004). *Aspect-Oriented Software Development With Use Cases*. Massachusetts: Addison-Wesley Professional .
- Kruchten, P. (1995). Modelo de arquitectura. *IEEE Software*, 45-50.
- López, X., & Montaluisa, F. (2015). *Diseño de la estructura organizativa e implementación de una metodología para el desarrollo de software en la Fábrica de Software de la ESPE Extensión Latacunga*. Latacunga: Universidad de las Fuerzas Armadas - ESPE.
- Mann, M. (1997). *Ingeniería de Software*. massachusetts: Mikel Angoar.

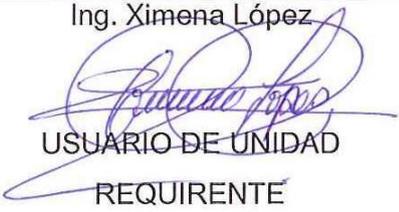
ANEXOS

Anexo 1 Formalización de la necesidad

FORMALIZACIÓN DE DESARROLLO DE SOFTWARE	
Proyecto: Sistema de Control de Pasantías	
Fecha de Reunión: 2014-02-19	Documento de Requerimiento: Memorándum No. 2014-045-ESPE-EL-c-1 del 21 de marzo de 2014 de la Unidad de Admisión y Registro.
Unidad Requirente: Universidad de las Fuerzas Armadas ESPE – Extensión Latacunga	Integrantes Unidad Requirente: Ing. Nancy Guerrón Ing. Ximena López Ing. Cesar Naranjo Ing. Misael Pazmiño Ing. José Quiroz
Unidad Desarrollo: Grupo de desarrollo de tesis	Integrantes Unidad de Desarrollo: Ricardo David Vásconez Garcés Fausto Fabricio Robayo Zurita
Descripción: El proyecto tiene como finalidad el conocer mediante técnicas apropiadas el proceso de seguimiento de pasantías de esta forma generar un producto software (análisis, diseño e implementación) que permitirá al administrador del proceso hacer un seguimiento minucioso efectivo y oportuno logrando así agilizar el proceso que actualmente es muy lento, debido a que se minimiza el trasapeló de documentos y la diversidad de los mismos en referencia a un solo estudiante.	
<p>Ing. Nancy Guerrón</p>  <p>RESPONSABLE UNIDAD REQUIRENTE</p>	<p>Fabricio Robayo</p>  <p>RESPONSABLE DEL GRUPO DE DESARROLLO</p>

PLANIFICACIÓN DE ENTREGAS			
Proyecto: Sistema de Control de Pasantías			
Fecha Elaboración: 2014-05-19		Tiempo Total Entrega: 60 días	
Cronograma de Entrega de Versiones:			
Versión:	Tiempo Desarrollo	Fecha Entrega:	Resumen de la Entrega:
1.0	10 días	2014-05-30	Definición de menús; operatividad de departamentos, carreras y empresas.
2.0	10 días	2014-06-13	Operatividad de convenios, indicadores de resultados, instrumentos de evaluación y tipos de pasantías.
3.0	5 días	2014-05-01	Operatividad coordinadores de pasantías, estudiantes y tutores académicos.
4.0	20 días	2014-05-30	Operatividad de las pasantías, tutores empresariales y programas de trabajo
5.0	10 días	2014-08-08	Operatividad del registro de calificaciones y desarrollo de los formatos de seguimiento de pasantías.
6.0	5 días	2014-08-15	Elaboración y definición de reportes.
Ing. Nancy Guerrón  RESPONSABLE UNIDAD REQUIRENTE		Fabricio Robayo  RESPONSABLE DEL GRUPO DE DESARROLLO	

Anexo 2 Requerimientos

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-05-19	Versión de Software: 1.0
ID de Requisito: R001-V1.0	Título: Definición de menús del sistema
Descripción: Se debe realizar el diseño de los menús que va a utilizar el sistema, siendo de fácil localización todos los procesos dentro del sistema y según el perfil de usuario registrado.	
Estimación: 14 horas	Usuario: Ing. Ximena López
Prioridad: 10	Dependencia: N/A
Fecha Revisiones: 2014-05-22 Se realiza la revisión de la estructura de los menús. 2014-05-30 Se realiza la revisión de la estructura de los menús debido a la actualización de la misma.	
Pruebas: El menú deberá ser de fácil uso para el usuario, pudiendo identificar de manera clara todos los módulos de la aplicación.	
Ing. Ximena López  USUARIO DE UNIDAD REQUIRENTE	Fabricio Robayo  TÉCNICO RESPONSABLE DEL DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-05-19	Versión de Software: 1.0
ID de Requisito: R002-V1.0	Título: Operatividad de los departamentos
Descripción: Se debe realizar el ingreso de los departamentos existentes, contando con ID que deberá ser numérico y auto incremental, ID de banner deberá ser alfanumérico, nombre del departamento deberá ser alfanumérico y nombre del director de este departamento que de igual forma deberá ser alfanumérico. Todos estos datos, exceptuando el ID de banner, son obligatorios.	
Estimación: 20 horas	Usuario: Ing. Ximena López
Prioridad: 9	Dependencia: N/A
Fecha Revisiones: 2014-05-23 Se realiza la revisión del funcionamiento de los departamentos.	
Pruebas: Se realizara las pruebas registrando los departamentos en el sistema con cada uno de los datos existentes.	
Ing. Ximena López  USUARIO DE UNIDAD REQUIRENTE	Fabricio Robayo  TÉCNICO RESPONSABLE DEL DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-05-19	Versión de Software: 1.0
ID de Requisito: R003-V1.0	Título: Operatividad de las carreras
Descripción: Se debe realizar el ingreso de las carrera existentes, contando con ID que deberá ser numérico y auto incremental, ID de Banner deberá ser alfanumérico, tipo de carrera que podrá ser tecnología o ingeniería; por ende; deberá ser alfanumérico, nombre de la carrera deberá ser alfanumérico, nombre del director de la carrera que deberá ser alfanumérico, numero de cedula del director de la carrera que deberá ser alfanumérico, siglas de la carrera que será alfanumérico, mínimo de créditos para poder cumplir las pasantías será numérico y el departamento al que pertenece este campo deberá ser numérico y será la clave foránea para relacionar con los departamentos. Todos los campos, exceptuando ID de banner, son obligatorios.	
Estimación: 26 horas	Usuario: Ing. Ximena López
Prioridad: 8	Dependencia: R002-V1.0
Fecha Revisiones: 2014-05-28 Se realiza la revisión del funcionamiento de las carreras.	

Pruebas:

Se realizara las pruebas registrando las carreras en el sistema con cada uno de los datos existentes.

Ing. Ximena López



USUARIO DE UNIDAD
REQUIRENTE

Fabricio Robayo



TÉCNICO RESPONSABLE DEL
DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-05-19	Versión de Software: 1.0
ID de Requisito: R004-V1.0	Título: Operatividad de las empresas
Descripción: Se debe realizar el ingreso de las empresas, contando con ID que deberá ser numérico y auto incremental, ID de banner que deberá ser alfanumérico, nombre de la empresa que deberá ser alfanumérico, actividad de la empresa que deberá ser alfanumérico, dirección de la empresa que deberá ser alfanumérico, representante legal de la empresa que deberá ser alfanumérico, cargo del representante legal de la empresa que deberá ser alfanumérico, título del representante legal de la empresa que deberá ser alfanumérico, teléfono de la empresa que deberá ser alfanumérico y correo electrónico de la empresa que de igual forma deberá ser alfanumérico. Todos los campos, exceptuando ID de banner, son obligatorios.	
Estimación: 20 horas	Usuario: Ing. Ximena López
Prioridad: 7	Dependencia: N/A
Fecha Revisiones: 2014-05-30 Se realiza la revisión del funcionamiento de las empresas.	

Pruebas:

Se realizaran las pruebas del funcionamiento de las empresas registrando cada una e identificando los campos obligatorios de cada una tanto en ingreso como en modificaciones.

Ing. Ximena López



USUARIO DE UNIDAD
REQUIRENTE

Fabricio Robayo



TÉCNICO RESPONSABLE DEL
DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-02	Versión de Software: 2.0
ID de Requisito: R001-V2.0	Título: Operatividad de los convenios
Descripción: Se debe realizar el ingreso de los convenios existentes con cada una de las empresas, contando con ID que deberá ser numérico y auto incremental, ID de banner que deberá ser alfanumérico, tipo de relación que será Marco Convenio o Carta de compromiso por ende alfanumérico, fecha de inicio y fecha de fin que deberán ser campos de fecha, numero de documento que será alfanumérico, estudiante que será numérico y campo foráneo para relacionar al estudiante en el caso de un convenio de tipo Carta de Compromiso y empresa que será numérico y campo foráneo para relacionar a la empresa. Todos los campos son obligatorios, exceptuando ID de banner y numero de documento.	
Estimación: 26 horas	Usuario: Ing. Ximena López
Prioridad: 10	Dependencia: R004-V1.0
Fecha Revisiones: 2014-06-05 Se realiza la revisión del funcionamiento de convenios ingresando los datos existentes.	

Pruebas:

Se realizaran las pruebas del funcionamiento de convenios ingresando los datos de los convenios existentes y de algunos ficticios para probar la relación de los convenios de tipo Carta de Compromiso.

Ing. Ximena López



USUARIO DE UNIDAD
REQUIRENTE

Fabricio Robayo



TÉCNICO RESPONSABLE DEL
DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-02	Versión de Software: 2.0
ID de Requisito: R002-V2.0	Título: Operatividad de los indicadores de resultados
Descripción: Se debe ingresar los indicadores de resultados existentes por cada carrera contando con ID que deberá ser numérico y auto incremental, ID de banner que deberá ser alfanumérico, descripción del indicador que deberá ser alfanumérico y carrera que será numérico y campo foráneo que relacionara a las carreras existentes. Todos los campos, exceptuando ID de banner, son obligatorios.	
Estimación: 18 horas	Usuario: Ing. Ximena López
Prioridad: 9	Dependencia: R003-V1.0
Fecha Revisiones: 2014-06-09 Se realiza la revisión del funcionamiento de los indicadores de resultados ingresando los existentes por cada carrera.	
Pruebas: Se realizaran las pruebas del funcionamiento del ingreso y modificación de los indicadores de resultados ingresando los existentes por cada carrera.	

Ing. Ximena López

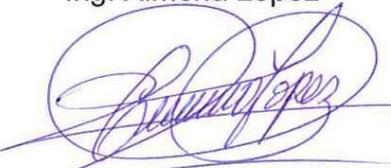


USUARIO DE UNIDAD
REQUIRENTE

Fabricio Robayo



TÉCNICO RESPONSABLE DEL
DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-02	Versión de Software: 2.0
ID de Requisito: R003-V2.0	Título: Operatividad de los instrumentos de evaluación
Descripción: Se deben ingresar los instrumentos de evaluación contando con ID que deberá ser numérico y auto incremental, descripción del instrumento que será alfanumérico y el criterio del instrumento de evaluación que deberá ser alfanumérico. Todos los campos son obligatorios.	
Estimación: 18 horas	Usuario: Ing. Ximena López
Prioridad: 8	Dependencia: N/A
Fecha Revisiones: 2014-06-11 Se realiza la revisión del funcionamiento de los instrumentos de evaluación.	
Pruebas: Se realizaran las pruebas del funcionamiento de ingreso y modificación de los instrumentos de evaluación registrando cada uno de ellos.	
<p>Ing. Ximena López</p>  <p>USUARIO DE UNIDAD REQUIRENTE</p>	<p>Fabricio Robayo</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-02	Versión de Software: 2.0
ID de Requisito: R004-V2.0	Título: Operatividad de los tipos de prácticas pre profesionales
Descripción: Se debe realizar el ingreso de los tipos de prácticas pre profesionales contando con ID que será numérico y auto incremental y el nombre del tipo de pasantía que será alfanumérico. Todos los campos son obligatorios	
Estimación: 18 horas	Usuario: Ing. Ximena López
Prioridad: 7	Dependencia: N/A
Fecha Revisiones: 2014-06-13 Se realiza la revisión del funcionamiento de los tipos de prácticas pres profesionales.	
Pruebas: Se realizaran las pruebas del funcionamiento ingresando los tipos de prácticas pre profesionales que hasta el momento son: Practicas pre profesionales no remuneradas y Pasantías	

Ing. Ximena López



USUARIO DE UNIDAD

REQUIRENTE

Fabricio Robayo



TÉCNICO RESPONSABLE DEL

DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-16	Versión de Software: 3.0
ID de Requisito: R001-V3.0	Título: Operatividad de los coordinadores de pasantías
Descripción: Se debe realizar el ingreso de los coordinadores de pasantías contando con ID que será numérico y auto incremental, ID de banner que deberá ser alfanumérico, cedula que será alfanumérico y validando que sea una cedula correcta, nombre del coordinador que será alfanumérico, apellido del coordinador que será alfanumérico, correo del coordinador validando que el mismo sea correcto este será alfanumérico, teléfono del coordinador que será alfanumérico y carrera que será numérico y campo foráneo que relacionara con cada una de las carreras existentes.	
Estimación: 13 horas	Usuario: Ing. Ximena López
Prioridad: 10	Dependencia: R003-V1.0
Fecha Revisiones: 2014-06-17 Se realiza la revisión del funcionamiento de los coordinadores de pasantías.	

Pruebas:

Se realizaran las pruebas ingresando todos los coordinadores existentes y relacionándolos con cada una de las carreras.

Ing. Ximena López



USUARIO DE UNIDAD
REQUIRENTE

Fabrizio Robayo



TÉCNICO RESPONSABLE DEL
DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-16	Versión de Software: 3.0
ID de Requisito: R002-V3.0	Título: Operatividad de los estudiantes
Descripción: Se debe realizar el ingreso de los estudiantes contando con ID que será numérico y auto incremental, ID de banner que será alfanumérico, cedula que será alfanumérico y validando que sea una cedula correcta, nombre del estudiante que será alfanumérico, apellido del estudiante que será alfanumérico, correo del estudiante validando que el mismo sea correcto este será alfanumérico, teléfono del estudiante que será alfanumérico y carrera a la que pertenece el estudiante que deberá ser numérico y campo foráneo para relacionarlo con las carreras. Todos los campos son obligatorios.	
Estimación: 14 horas	Usuario: Ing. Ximena López
Prioridad: 9	Dependencia: R003-V1.0
Fecha Revisiones: 2014-06-19 Se realiza la revisión del funcionamiento de los estudiantes.	
Pruebas: Se realizara las pruebas registrando cada estudiante y relacionándolo con su carrera correspondiente.	

Ing. Ximena López



USUARIO DE UNIDAD

REQUIRENTE

Fabrizio Robayo



TÉCNICO RESPONSABLE DEL

DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-16	Versión de Software: 3.0
ID de Requisito: R003-V3.0	Título: Operatividad de los tutores académicos
Descripción: Se debe realizar el ingreso de los tutores académicos contando con un ID que será numérico y auto incremental, ID de banner que será alfanumérico, cedula del tutor académico que será alfanumérico y validando que sea una cedula correcta, nombre del tutor académico que deberá ser alfanumérico, apellido del tutor académico que será alfanumérico, correo del tutor académico que será alfanumérico y validando que el mismo sea correcto, teléfono del tutor académico que será alfanumérico y el departamento al que pertenece el tutor académico este será numérico y campo foráneo para relacionarlo con los departamentos. Todos los campos son obligatorios.	
Estimación: 13 horas	Usuario: Ing. Ximena López
Prioridad: 8	Dependencia: R002-V1.0
Fecha Revisiones: 2014-06-20 Se realiza la revisión del funcionamiento de los tutores académicos.	

Pruebas:

Se realiza las pruebas registrando los tutores académicos.

Ing. Ximena López



USUARIO DE UNIDAD
REQUIRENTE

Fabricio Robayo



TÉCNICO RESPONSABLE DEL
DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-23	Versión de Software: 4.0
ID de Requisito: R001-V4.0	Título: Operatividad de las pasantías.
Descripción: Se debe realizar el ingreso de las pasantías contando con ID numérico y auto incremental, numero de solicitud que será alfanumérico, numero de solicitud que será alfanumérico, número de créditos aprobados que será numérico, fecha de inicio y fin de la pasantías que serán campos de fecha, número total de horas de la pasantía que será numérico, periodo académico deberá ser alfanumérico, observaciones deberá ser alfanumérico, hora de entrada y salida serán formato hora, nombre del proyecto a realizar deberá ser alfanumérico, área del conocimiento aplicada al proyecto deberá ser alfanumérico, URL del cronograma de trabajo en donde se almacenara el archivo adjunto este deberá ser alfanumérico, fecha de evaluación que será de tipo fecha, debilidades encontradas durante la pasantía que será de formato alfanumérico, recomendaciones mostradas luego de realizar la pasantía esta será alfanumérico, el estado de la pasantía siendo: negada, en progreso cumplida, rechazada y reprobada siendo alfanumérico, el tipo de pasantía que será numérico y campo foráneo para la relación con los tipos de pasantías existentes, el estudiante asociado a la pasantía que será numérico y campo foráneo para la relación con el estudiante, el convenio que será numérico y campo foráneo para la relación con los convenios	

existentes, el tutor empresarial que será numérico y campo foráneo para la relación con los tutores empresariales existentes y el tutor académico que será numérico y campo foráneo para la relación con los tutores académicos existentes.

La pasantía será ingresada en varias etapas siendo la primera ingresada por el rol de coordinador de pasantías quien ingresará: el estudiante, periodo académico, número de solicitud, numero de trámite, número de créditos aprobados, tipo de pasantía, convenio y el tutor académico.

La segunda etapa será ingresada por el tutor académico que ingresará: al tutor empresarial con todos sus datos (R002-V4.0) la fecha de inicio y fin de la pasantía, el horario de entrada y salida, el total de horas de la pasantía, el nombre del proyecto a realizarse, el área de conocimiento aplicada, la ruta en donde se almacenara el archivo del cronograma de trabajo y todos los programas de trabajo existentes (R003-V4.0)

La tercera etapa será ingresada con los datos entregados por el tutor empresarial y estos son: los instrumentos de evaluación (R003-V2.0) las debilidades encontradas en la pasantía, las recomendaciones existentes y la fecha en la que el tutor empresarial realizo la evaluación.

La cuarta etapa nuevamente será ingresada por el tutor académico una vez el tutor empresarial haya entregado todos los datos expuestos en el párrafo anterior deberá ingresar la calificación del formato número dos y la calificación del formato número cuatro además de una observación de estas calificaciones.

Con esto se da por terminado el proceso de creación de pasantías.

Estimación: 72 horas	Usuario: Ing. Ximena López
------------------------------------	--

<p>Prioridad:</p> <p>10</p>	<p>Dependencia:</p> <p>R004-V2.0 R002-V3.0</p> <p>R001-V2.0 R002-V4.0*</p> <p>R003-V3.0 R002-V2.0</p> <p>R003-V2.0 R003-V4.0*</p>
<p>Fecha Revisiones:</p> <p>2014-07-04 Se realiza la revisión del funcionamiento de las pasantías</p>	
<p>Pruebas:</p> <p>Se probara el funcionamiento de las pasantías realizando el ingreso de una prueba real práctica de una de ellas pasando por todos los estados existentes.</p>	
<p>Ing. Ximena López</p>  <p>USUARIO DE UNIDAD REQUIRENTE</p>	<p>Fabricio Robayo</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-23	Versión de Software: 4.0
ID de Requisito: R002-V4.0	Título: Operatividad de los tutores empresariales
Descripción: Se debe realizar el ingreso de los tutores empresariales contando con ID que será numérico y auto incremental, ID de banner que será alfanumérico, numero de cedula que será alfanumérico, nombre del tutor empresarial que será alfanumérico, apellido del tutor empresarial que será alfanumérico, correo del tutor empresarial que deberá ser alfanumérico y validando que sea un correo aceptado, número telefónico del tutor empresarial que será alfanumérico y función en la empresa que desempeña el tutor empresarial que será alfanumérica. Todos los datos del tutor empresarial serán ingresados desde la creación de la pasantía.	
Estimación: 44 horas	Usuario: Ing. Ximena López
Prioridad: 9	Dependencia: R001-V4.0
Fecha Revisiones: 2014-07-10 Se realiza la revisión del funcionamiento de los tutores empresariales.	

Pruebas:

Se probará el funcionamiento de los tutores empresariales realizando el ingreso de una práctica real perteneciente a una pasantía existente.

Ing. Ximena López



USUARIO DE UNIDAD

REQUIRENTE

Fabricio Robayo



TÉCNICO RESPONSABLE DEL

DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-23	Versión de Software: 4.0
ID de Requisito: R003-V4.0	Título: Operatividad de los programas de trabajo
Descripción: Se debe realizar el ingreso de los programas de trabajo contando con ID que será numérico y auto incremental, departamento al que pertenece en el programa que será alfanumérico, unidad operativa asignada en la empresa deberá ser alfanumérica, función asignada en la empresa será alfanumérica, actividades a realizar en la pasantía deberá ser alfanumérica y pasantía que será el campo con el cual relacionar al programa de trabajo y las pasantías que deberá ser numérico. Todos los datos de los programas de trabajo serán ingresados desde la creación de la pasantía.	
Estimación: 44 horas	Usuario: Ing. Ximena López
Prioridad: 8	Dependencia: R001-V4.0
Fecha Revisiones: 2014-07-18 Se realiza la revisión del funcionamiento de los programas de trabajo.	

Pruebas:

Se probará el funcionamiento de los programas de trabajo realizando el ingreso de una práctica real perteneciente a una pasantía existente.

Ing. Ximena López



USUARIO DE UNIDAD
REQUIRENTE

Fabricio Robayo



TÉCNICO RESPONSABLE DEL
DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-07-21	Versión de Software: 5.0
ID de Requisito: R001-V5.0	Título: Operatividad de las calificaciones.
Descripción: Se debe realizar el ingreso de las calificaciones contando con ID que será numérico y auto incremental, nota del formato numero dos que será numérica, nota del formato número cuatro será numérica (estas dos notas serán ingresadas por el tutor académico), fecha de evaluación que tendrá el formato de fecha y será ingresada automáticamente el momento en el que el tutor académico realice el ingreso de las notas antes mencionadas, nota del tutor empresarial que será numérica y será auto calculada según los criterios establecidos al momento del ingreso de los instrumentos de evaluación, observación de la calificación que será alfanumérico y pasantía que será numérico y el campo con el que se relacione la pasantía.	
Estimación: 32 horas	Usuario: Ing. Ximena López
Prioridad: 10	Dependencia: R001-V4.0
Fecha Revisiones: 2014-07-24 Se realiza la revisión del funcionamiento de las calificaciones.	

Pruebas: Se probará el funcionamiento de las calificaciones realizando el ingreso de una práctica real perteneciente a una pasantía existente.	
Ing. Ximena López  USUARIO DE UNIDAD REQUIRENTE	Fabricio Robayo  TÉCNICO RESPONSABLE DEL DESARROLLO

REQUISITOS ESPECÍFICOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-07-21	Versión de Software: 5.0
ID de Requisito: R002-V5.0	Título: Realización de los formatos para impresión.
Descripción: Se debe realizar el diseño de los formatos pre establecidos para la impresión establecida en los diferentes tiempos establecidos en los procesos de la pasantía. Siendo estos: <p style="text-align: center;"><i>Carta de compromiso</i></p>	

Estimación: 48 horas	Usuario: Ing. Ximena López
Prioridad: 9	Dependencia: R001-V1.0 R002-V1.0 R003-V1.0 R004-V1.0 R001-V2.0 R002-V2.0 R003-V2.0 R004-V2.0 R001-V3.0 R002-V3.0 R003-V3.0 R001-V4.0 R002-V4.0 R003-V4.0 R001-V5.0
Fecha Revisiones: 2014-08-08 Se realiza la revisión del funcionamiento de cada uno de os formatos.	
Pruebas: Se probaran las impresiones de dichos formatos revisando si los datos contenidos en los mismos corresponden a la información de la pasantía asociada.	

Ing. Ximena López



USUARIO DE UNIDAD

REQUIRENTE

Fabricio Robayo



TÉCNICO RESPONSABLE DEL

DESARROLLO

REQUISITOS ESPECÍFICOS

Proyecto:

Sistema de Control de Pasantías

Fecha Elaboración:

2014-08-11

Versión de Software:

6.0

ID de Requisito:

R001-V6.0

Título:

Diseño de los reportes de las prácticas pre profesionales.

Descripción:

Se debe realizar los repostes según lo correspondiente a:

Reporte – Resumen de Empresas

ORD.	EMPRESA	REGISTRADOS
1	ICONO SISTEMAS	2
2	COMANDANCIA GENERAL DEL EJERCITO - DIC	2
3	CORPORACION NACIONAL DE TELECOMUNICACIONES CNT	2
4	BAEEL SOFTWARE ECUADOR	1
TOTAL		7

© 2015 | Todos los Derechos Reservados.
Diseñado y Desarrollado por: David Vasconez y Fausto Robayo

Reporte – Detalle de Estudiantes por Empresa

EMPRESA: ICONO SISTEMAS

ORD.	NOMBRE	CEDULA	TELEFONO	CORREO
1	RICARDO DAVID VASCONEZ GASCES	1804455944	100298513	0955681942
2	FAUSTO FAELICIO ROBAYO ZURITA	1803282258	1002981287	0992693325
3	RICARDO DAVID VASCONEZ GALARZA	1804455944	100298513	0955681942

© 2015 | Todos los Derechos Reservados.
Diseñado y Desarrollado por: David Vasconez y Fausto Robayo

Reporte – Detalle de Practicas Pre Profesionales por Tutor Académico

Detalle de Pasantias por Tutor Académico

REPORTE DETALLADO DE PASANTIAS POR TUTOR ACADEMICO

TUTOR ACADEMICO: CESAR ARMANDO CASTRO NARVAEZ

ORD.	ESTUDIANTE	EMPRESA	FECHA INICIO	FECHA FIN	ESTADO
1	NIVAR CAMILO ACOSTA RODRIGUEZ	COFORACIÓN NACIONAL DE TELECOMUNICACIONES CNT	2015-05-01	2015-05-31	CUMPLIDA

© 2015 | Todos los Derechos Reservados.
Diseñado y Desarrollado por: David Vasconez y Fausto Robayo

Reporte – Detalle de Practicas Pre Profesionales por Estudiante

Detalle de Pasantias por Estudiante

REPORTE DETALLADO DE PASANTIAS POR ESTUDIANTE

ESTUDIANTE: RICARDO DAVID VASCONEZ GARCES

ORD.	EMPRESA	FECHA INICIO	FECHA FIN	N HORAS	ESTADO
1					NEGADA

© 2015 | Todos los Derechos Reservados.
Diseñado y Desarrollado por: David Vasconez y Fausto Robayo

Reporte – Detalle de Practicas Pre Profesionales por Estado

Detalle de Pasantias por Estado

REPORTE DETALLADO DE PASANTIAS POR ESTADO

ESTADIOS: En Progreso, Incompleta, En Progreso, Cumplidas, Rechazadas, Negadas, Reprobadas

ESTADO: EN PROGRESO

ORD.	ESTUDIANTE	TIPO	CREDITOS APROBADOS
1	RICARDO DAVID VASCONEZ GALARZA	PRÁCTICA PSE PROFESIONAL NO FEAMUNERADA	10

© 2015 | Todos los Derechos Reservados.
Diseñado y Desarrollado por: David Vasconez y Fausto Robayo

Estimación:

40 horas

Usuario:

Ing. Ximena López

Prioridad:

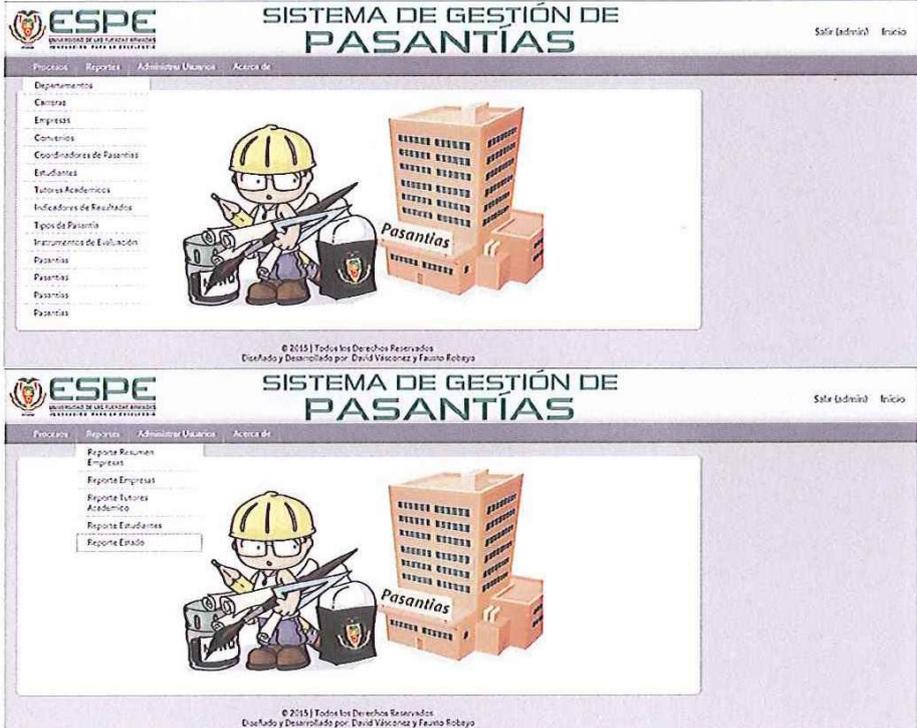
10

Dependencia:

R001-V1.0

	<p>R002-V1.0</p> <p>R003-V1.0</p> <p>R004-V1.0</p> <p>R001-V2.0</p> <p>R002-V2.0</p> <p>R003-V2.0</p> <p>R004-V2.0</p> <p>R001-V3.0</p> <p>R002-V3.0</p> <p>R003-V3.0</p> <p>R001-V4.0</p> <p>R002-V4.0</p> <p>R003-V4.0</p> <p>R001-V5.0</p> <p>R002-V5.0</p>
<p>Fecha Revisiones: 2014-08-15 Se realiza la revisión del funcionamiento de cada uno de los reportes.</p>	
<p>Pruebas: Se probaran los reportes revisando si los datos contenidos en los mismos corresponden a la información de las pasantías.</p>	
<p>Ing. Ximena López</p>  <p>USUARIO DE UNIDAD REQUIRENTE</p>	<p>Fabrizio Robayo</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>

Anexo 3 Diseño

DISEÑO DE INTERFACES	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-05-19	Técnico: David Vásconez Fabricio Robayo
Título: Diseño de los menús, operatividad de departamentos, carreras y empresas.	
Descripción:  <p>© 2015 Todos los Derechos Reservados Diseñado y Desarrollado por: David Vásconez y Fabricio Robayo</p>	

Procesos | Reportes | Administrar Usuarios | Acerca de | [Inicio > Departamentos](#)

Administrar Departamentos

Búsqueda Avanzada

Desplegando 1-3 de 3 resultados

ID de un Departamento	Nombre del Departamento	Siglas del Departamento	Director de Departamento	
	DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA	EDE	ARIANDO FERRAN ALVAREZ SALAZAR	i f n
	DEPARTAMENTO DE ENERGÍA Y MECÁNICA	DEM	SINAGA ANAQUELA MARCO ADOLEO	i f n
	DEPARTAMENTO DE CIENCIAS ADMINISTRATIVAS Y DE COMERCIO	CEAC	GALO RAMIRO CASTRO CLAVO	i f n

© 2015 | Todos los Derechos Reservados
 Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

Procesos | Reportes | Administrar Usuarios | Acerca de | [Inicio > Administrar](#)

Administrar Carreras

Búsqueda Avanzada

Desplegando 1-10 de 13 resultados

ID de una Carrera	Nombre de la Carrera	Siglas de la Carrera	Tipo de Carrera	Número de Carreras para la Carrera	Director de la Carrera	Código de la Carrera de la Carrera	Nombre del Departamento	
	INGENIERIA EN SOFTWARE	ISOW	INGENIERIA	100	LUCAS ROGERIO GARCÉS GUAYTA	10010615000	DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA	i f n
	INGENIERIA ELECTRONICA E INSTRUMENTACION	IELI	INGENIERIA	100	FRANLIN MANUEL SILVA MONTEFOS	1001062105	DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA	i f n
	INGENIERIA ELECTROMECANICA	IEME	INGENIERIA	100	TORRES VASCOZ PATA PEREZES	0501045005	DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA	i f n
	INGENIERIA AUTOMOTRIZ	IMAU	INGENIERIA	100	JUAN TRAZANO CASTRO CLAVO	1001050402	DEPARTAMENTO DE ENERGÍA Y MECÁNICA	i f n
	INGENIERIA COMERCIAL	ICOF	INGENIERIA	100	XAVIER HERRIGONES FABIAN BARRERA	1001060204	DEPARTAMENTO DE CIENCIAS ADMINISTRATIVAS Y DE COMERCIO	i f n
	INGENIERIA EN FINANZAS Y AUDITORIA	ICFP	INGENIERIA	100	TAPIA LEONILDO CESAR	0501020004	DEPARTAMENTO DE CIENCIAS ADMINISTRATIVAS Y DE COMERCIO	i f n
	INGENIERIA EN ADMINISTRACION TURISTICA Y HOTELERIA	ITUR	INGENIERIA	100	ALEDRY GEOVANNY ACOSTA FERNANDEZ	1001060403	DEPARTAMENTO DE CIENCIAS ADMINISTRATIVAS Y DE COMERCIO	i f n
	INGENIERIA EN MECATRONICA	IMET	INGENIERIA	100	VICENTE DAVID HILLO CARRASCO	1001050200	DEPARTAMENTO DE ENERGÍA Y MECÁNICA	i f n
	INGENIERIA EN PETROLOGIA	IPED	INGENIERIA	100	ROMAN NICOLAY RODRIGUEZ MARQUEL	1001020200	DEPARTAMENTO DE ENERGÍA Y MECÁNICA	i f n
	TECNOLOGIA EN COMPUTACION		TECNOLOGIA	100	LUCAS ROGERIO GARCÉS GUAYTA	1001001003	DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA	i f n

1 de 1 página < Anterior | 1 | 2 | Siguiente >

© 2015 | Todos los Derechos Reservados
 Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

Procesos | Reportes | Administrar Usuarios | Acerca de | [Inicio > Administrar](#)

Administrar Empresas

Búsqueda Avanzada

Desplegando 1-4 de 4 resultados

ID de una Empresa	Nombre de la Empresa	Actividad de la Empresa	Dirección de la Empresa	Representante Legal de la Empresa	Teléfono de la Empresa	
	ICONO SISTEMAS	INFORMATICA	MONTAÑO Y JUAN B VELA	VICTOR LUZURAGA	03242700	i f n
	COMANDAMON GENERAL DEL EJERCITO - DIC	DEFENSA DEL GOBIERNO	LA FRECOLETA	MARCO DUMTANA	022479045	i f n
	CORPORACION NACIONAL DE TELECOMUNICACIONES DNT	TELECOMUNICACIONES DEL PAIS	PIEDRA SALCEDO Y GUAYAZUL	DAVID VASCOZ	032254532	i f n
	BASEL SOFTWARE ECUADOR	DESARROLLO DE SISTEMAS	SAN MARTIN	MARIEL SILVA	S/N	i f n

© 2015 | Todos los Derechos Reservados
 Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

David Vásconez



TÉCNICO RESPONSABLE DEL DESARROLLO

DISEÑO DE INTERFACES

Proyecto:

Sistema de Control de Pasantías

Fecha Elaboración:

2014-06-02

Técnico:

David Vásquez

Fabrizio Robayo

Título:

Operatividad de convenios, indicadores de resultados, instrumentos de evaluación y tipos de pasantías.

Descripción:

ESPE SISTEMA DE GESTIÓN DE PASANTÍAS

Procesos Reportes Administración Usuarios Ayuda de Inicio Administrar

Administrar Convenios

Operaciones

Crear Convenio

Mostrando 1-12 de 12 resultados

ID Interno del Convenio	Tipo de Redacción del Convenio	Fecha Inicio del Convenio	Numero Documento del Convenio	Estudiante	Nombre de la Empresa	Acciones
100295518	CARTA DE COMPROMISO	2015-04-13	121243	RICARDO DAVID VÁSQUEZ GARCÉS	ICOND SISTEMAS	[Icono]
100295518	CARTA DE COMPROMISO	2015-04-15	121243	FABRIZIO FABIANO ROBAYO ZURITA	ICOND SISTEMAS	[Icono]
100000009	CARTA DE COMPROMISO	2015-04-01	8103	FABRIZIO FABIANO ROBAYO ZURITA	ICOND SISTEMAS	[Icono]
100222222	CARTA DE COMPROMISO	2015-04-01	03075	FABRIZIO FABIANO ROBAYO ZURITA	ICOND SISTEMAS	[Icono]
	CARTA DE COMPROMISO	2015-05-04	2015-04-001	PARCO ERICK PASTOR RIOS	COMANDANCIA GENERAL DEL EJERCITO-DIC	[Icono]
	CARTA DE COMPROMISO	2015-05-01	2015-04-001	PAULA VIVIANO CHICALZA CHALEA	COMANDANCIA GENERAL DEL EJERCITO-DIC	[Icono]
	CARTA DE COMPROMISO	2015-05-01	022356	YVIRA GABRIEL ACOSTA RODRIGUEZ	CORPORACION NACIONAL DE TELECOMUNICACIONES CNT	[Icono]
	CARTA DE COMPROMISO	2015-05-01	0012	RICARDO DAVID VÁSQUEZ GALARRAZA	BAEEL SOFTWARE ECUADOR	[Icono]
	CARTA DE COMPROMISO	2015-05-01	022356	PARCO ERICK PASTOR RIOS	CORPORACION NACIONAL DE TELECOMUNICACIONES CNT	[Icono]
	CARTA DE COMPROMISO	2015-04-30	022356	JUAN FRANCISCO FAREDES ANDINO	CORPORACION NACIONAL DE TELECOMUNICACIONES CNT	[Icono]

1 de 1 página < Anterior 1 2 Siguiente >

© 2014 | Todos los Derechos Reservados. Diseñado y Desarrollado por: David Vásquez y Fabrizio Robayo

Administrar Indicadores de Resultados

[Búsqueda Avanzada](#)

Mostrando 1-7 de 7 resultados

ID Nuevo Indicador de Resultados	Descripción Indicador de Resultados	Nombre de la Carrera	
	APLICAR POLÍTICAS Y PROCEDIMIENTOS A FIN DE GARANTIZAR SEGURIDAD, CONTROL Y EVALUACIÓN DE LA INFORMACIÓN, CUMPLIENDO CON EL MARCO LEGAL VIGENTE NACIONAL E INTERNACIONAL.	INGENIERÍA EN SOFTWARE	D F X
	APLICAR POLÍTICAS Y PROCEDIMIENTOS A FIN DE GARANTIZAR SEGURIDAD, CONTROL Y EVALUACIÓN DE LA INFORMACIÓN, CUMPLIENDO CON EL MARCO LEGAL VIGENTE NACIONAL E INTERNACIONAL.	INGENIERÍA ELECTRÓNICA E INSTRUMENTACIÓN	D F X
	APLICAR POLÍTICAS Y PROCEDIMIENTOS A FIN DE GARANTIZAR SEGURIDAD, CONTROL Y EVALUACIÓN DE LA INFORMACIÓN, CUMPLIENDO CON EL MARCO LEGAL VIGENTE NACIONAL E INTERNACIONAL.	INGENIERÍA ELECTROMECÁNICA	D F X
	APLICAR POLÍTICAS Y PROCEDIMIENTOS A FIN DE GARANTIZAR SEGURIDAD, CONTROL Y EVALUACIÓN DE LA INFORMACIÓN, CUMPLIENDO CON EL MARCO LEGAL VIGENTE NACIONAL E INTERNACIONAL.	INGENIERÍA AUTOMOTRIZ	D F X
	APLICAR POLÍTICAS Y PROCEDIMIENTOS A FIN DE GARANTIZAR SEGURIDAD, CONTROL Y EVALUACIÓN DE LA INFORMACIÓN, CUMPLIENDO CON EL MARCO LEGAL VIGENTE NACIONAL E INTERNACIONAL.	INGENIERÍA COMERCIAL	D F X
	APLICAR POLÍTICAS Y PROCEDIMIENTOS A FIN DE GARANTIZAR SEGURIDAD, CONTROL Y EVALUACIÓN DE LA INFORMACIÓN, CUMPLIENDO CON EL MARCO LEGAL VIGENTE NACIONAL E INTERNACIONAL.	TECNOLOGÍA EN COMPUTACIÓN	D F X
	SEGUNDO INDICADOR DE PRUEBA	INGENIERÍA EN SOFTWARE	D F X

 © 2015 | Todos los Derechos Reservados.
 Diseñado y Desarrollado por: David Váscquez y Fausto Robayo

Operaciones

[← Crear Indicador de Resultados](#)

Administrar Instrumentos de Evaluación

[Búsqueda Avanzada](#)

Mostrando 1-2 de 2 resultados

Criterio de Evaluación	Descripción de Evaluación	
CONTROL DE AVANCE DE ACTIVIDADES	SE RESPALDARÁ CON UN CRONOGRAMA PROGRAMADO VS REAL	D F X
RESULTADOS ALCANZADOS	PRESENTA INFORME INDICANDO LOS RESULTADOS QUE SE LOGRARON CON LA PRÁCTICA FRE PROFESIONAL NO REMUNERADA O PASANTÍA EN RAZÓN DEL CUMPLIMIENTO DE METAS Y OBJETIVOS	D F X

 © 2015 | Todos los Derechos Reservados.
 Diseñado y Desarrollado por: David Váscquez y Fausto Robayo

Operaciones

[← Crear Instrumento de Evaluación](#)

Administrar Tipos de Pasantías

[Búsqueda Avanzada](#)

Mostrando 1-2 de 2 resultados

Nombre del Tipo de Pasantía	
PASANTÍA	D F X
PRÁCTICA FRE PROFESIONAL NO REMUNERADA	D F X

 © 2015 | Todos los Derechos Reservados.
 Diseñado y Desarrollado por: David Váscquez y Fausto Robayo

Operaciones

[← Crear Tipo de Pasantía](#)

David Váscquez



TÉCNICO RESPONSABLE DEL DESARROLLO

DISEÑO DE INTERFACES

Proyecto:

Sistema de Control de Pasantías

Fecha Elaboración:

2014-06-16

Técnico:

David Vásconez

Fabricio Robayo

Título:

Operatividad de coordinadores de pasantías, estudiantes y tutores académicos.

Descripción:

ID Bases del Coordinador	Código del Coordinador	Nombre del Coordinador	Apellido del Coordinador	Correo del Coordinador	Nombre de la Carrera	
100047543	1011194724	XIMENA DEL ROCIO	LOPEZ CHICO	xlope@hotmail.com	INGENIERÍA EN SOFTWARE	p / /
100010010	0511493125	CÉSAR ALFREDO	NARANJO HUILLO	ca.naranjo@espe.edu.ec	INGENIERÍA ELECTRÓNICA E INSTRUMENTACIÓN	p / /
100047543	1011194724	XIMENA DEL ROCIO	LOPEZ CHICO	ximena_lopez@espe.edu.ec	TECNOLOGÍA EN COMPUTACIÓN	p / /
100010619	0502692483	HECTOR COCHES	TERAN HERRERA	hcteran@espe.edu.ec	INGENIERÍA EN MECATRÓNICA	p / /
100010619	0502692483	JOSE LIZANZO	OLIVERA ERAZO	jolivera@espe.edu.ec	INGENIERÍA AUTOMOTRIZ	p / /
100007413	0501375411	WILLIAM MOISES	BONILLA DÍAZ	wbonilla@espe.edu.ec	INGENIERÍA EN PETROQUÍMICA	p / /
100040329	0502433128	CARLOS GEOVANNY	ALFAN YANZE	cgalfan@espe.edu.ec	INGENIERÍA EN ADMINISTRACIÓN TURÍSTICA Y HOTELERA	p / /
100007920	0502283126	MARTHA DECILIA	LODADA OREJUELA	mlorejuela@espe.edu.ec	INGENIERÍA EN FINANZAS Y AUDITORÍA	p / /
100007920	0502283126	MARTHA DECILIA	LODADA OREJUELA	mlorejuela@espe.edu.ec	INGENIERÍA COMERCIAL	p / /
100012443	0500814179	EUTIMIO MISHEL	RAZMIGO DÍAZ	emrazmigo@espe.edu.ec	INGENIERÍA ELECTRÓNICA E INSTRUMENTACIÓN	p / /

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fabricio Robayo

Procesos | Reportes | Administrar Usuarios | Acerca de | Inicio | Administrar

Administrar Estudiantes

Búsqueda Avanzada

Mostrando 1-7 de 7 resultados

ID Estudiante	Cédula del Estudiante	Nombres del Estudiante	Apellidos del Estudiante	Correo del Estudiante	Nombre de la Carrera		
10029583	1804469344	RICARDO DAVID	VÁSQUEZ GARCÉS	ricdavid@gmail.com	INGENIERÍA EN SOFTWARE	p	x
10031287	1803282458	FAUSTO FABRICO	ROBAYO ZURITA	fabrico_fausto@hotmail.com	INGENIERÍA EN SOFTWARE	p	x
10029514	1807120253	MARCO ERICK	PASTUÑA FIOS	e1ckg@uniba.gov.ec	TECNOLOGÍA EN COMPUTACIÓN	p	x
10031356	660248261	PAUL VINICIO	CHICAIZA CHAULA	paull31@hotmail.com	INGENIERÍA EN SOFTWARE	p	x
10029597	172849794	YMER CAMILO	ACOSTA RODRIGUEZ	ymerim@gmail.com	INGENIERÍA EN SOFTWARE	p	x
10029513	1804469344	RICARDO DAVID	VÁSQUEZ CALARZA	rvasconez@espe.edu.ec	INGENIERÍA EN SOFTWARE	p	x
10029537	1800120187	JUAN FRANCISCO	FAJARES ANGIÑO	jfajares@espe.edu.ec	INGENIERÍA AUTOMOTRIZ	p	x

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

Operaciones

[← Crear Estudiante](#)

Procesos | Reportes | Administrar Usuarios | Acerca de | Inicio | Administrar

Administrar Tutores Académicos

Búsqueda Avanzada

Mostrando 1-2 de 2 resultados

ID Tutor Académico	Cédula del Tutor Académico	Nombres del Tutor Académico	Apellidos del Tutor Académico	Correo del Tutor Académico	Nombre del Departamento		
10000001	1800200183	CESAR ARMANDO	CASTRO NÚÑEZ	ccarcastro@espe.edu.ec	DEPARTAMENTO DE ELECTRICAL Y ELECTRÓNICA	p	x
10000771	1800200183	JUAN CARLOS	SORIA RAMÍREZ	juan_soria@espe.edu.ec	DEPARTAMENTO DE ELECTRICAL Y ELECTRÓNICA	p	x

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

Operaciones

[← Crear Tutor Académico](#)

David Vásconez



TÉCNICO RESPONSABLE DEL DESARROLLO

DISEÑO DE INTERFACES

Proyecto:

Sistema de Control de Pasantías

Fecha Elaboración:

2014-06-23

Técnico:

David Vásconez

Fabricio Robayo

Título:

Operatividad de pasantías, tutores empresariales y programas de trabajo.

Descripción:

The screenshot displays the 'SISTEMA DE GESTIÓN DE PASANTÍAS' web application. At the top, there is a header with the ESPE logo and the system title. Below the header, there are navigation tabs for 'Procesos', 'Reportes', 'Administrar Usuarios', and 'Ayuda de'. The main content area is titled 'Administrar Pasantías' and shows a table of data. The table has columns for 'Estudiante', 'Período Académico', 'Número de Solicitud de la Pasantía', 'Nombre del Tipo de Pasantía', 'Tutor Académico', and 'Acciones'. The table lists 10 rows of student data. To the right of the table, there are 'Operaciones' buttons for 'Crear Pasantía' and 'Formular NDS'. At the bottom of the page, there is a copyright notice: '© 2015 | Todos los Derechos Reservados. Diseñado y Desarrollado por David Vásconez y Fabricio Robayo'.

Estudiante	Período Académico	Número de Solicitud de la Pasantía	Nombre del Tipo de Pasantía	Tutor Académico	Acciones
RICARDO DAVID VASCONEZ GARCÉS	AGOSTO 2015 - SEPTIEMBRE 2015	012788	PRÁCTICA PRE PROFESIONAL NO SEMANAL		P W 0
VINICIO GILBERTO ACOSTA RODRIGUEZ	AGOSTO 2015 - SEPTIEMBRE 2015	012789	PRÁCTICA PRE PROFESIONAL NO SEMANAL	CESAR ARMANDO CASTRO NARANJE	P W 0
RICARDO DAVID VASCONEZ GALARRA	AGOSTO 2015 - SEPTIEMBRE 2015	012790	PRÁCTICA PRE PROFESIONAL NO SEMANAL	JUAN CARLOS SORIA RAMIREZ	P W 0
RICARDO DAVID VASCONEZ GALARRA	AGOSTO 2015 - SEPTIEMBRE 2015	012791	PRÁCTICA PRE PROFESIONAL NO SEMANAL		P W 0
RICARDO DAVID VASCONEZ GALARRA	AGOSTO 2015 - SEPTIEMBRE 2015	012792	PRÁCTICA PRE PROFESIONAL NO SEMANAL		P W 0
RICARDO DAVID VASCONEZ GALARRA	AGOSTO 2015 - SEPTIEMBRE 2015	012793	PRÁCTICA PRE PROFESIONAL NO SEMANAL	JUAN CARLOS SORIA RAMIREZ	P W 0
JUAN FRANCISCO PAREDES ANDINO	AGOSTO 2015 - SEPTIEMBRE 2015	012794	PRÁCTICA PRE PROFESIONAL NO SEMANAL		P W 0
JUAN FRANCISCO PAREDES ANDINO	AGOSTO 2015 - SEPTIEMBRE 2015	012795	PRÁCTICA PRE PROFESIONAL NO SEMANAL	JUAN CARLOS SORIA RAMIREZ	P W 0
JUAN FRANCISCO PAREDES ANDINO	AGOSTO 2015 - SEPTIEMBRE 2015	012796	PRÁCTICA PRE PROFESIONAL NO SEMANAL		P W 0

SS Pasantías - Create Tutor... x +

localhost:8080/bsic?AdministrarUsuarios

ESPE
UNIVERSIDAD DE LOS RÍOS
UNIVERSIDAD PARA LA SOSTENIBILIDAD

SISTEMA DE GESTIÓN DE PASANTÍAS

Salir (Admin) Inicio

Procesos Reportes Administrar Usuarios Acerca de

Tipo de Tutor Empresarial > Crear

Crear Tutor Empresarial

Los campos con * son requeridos

Tutor Empresarial Banner

Tutor Empresarial Cédula*

Tutor Empresarial Nombre*

Tutor Empresarial Apellido*

Tutor Empresarial Correo*

Tutor Empresarial Teléfono*

Tutor Empresarial Función*

Crear

Operaciones

- < Listar Tipo Empresarial
- < Administrar Tutor Empresarial

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

ESPE
UNIVERSIDAD DE LOS RÍOS
UNIVERSIDAD PARA LA SOSTENIBILIDAD

SISTEMA DE GESTIÓN DE PASANTÍAS

Salir (Admin) Inicio

Procesos Reportes Administrar Usuarios Acerca de

Tipo de Programa Trabajo > Crear

Crear Programa Trabajo

Los campos con * son requeridos

Programa Trabajo Departamento*

Programa Trabajo Unidad Operativa*

Programa Trabajo Función Asignada*

Programa Trabajo Actividades*

Crear y Nuevo Solo Crear

Operaciones

- < Listar Programa Trabajo
- < Administrar Programa Trabajo

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

David Vásconez



TÉCNICO RESPONSABLE DEL DESARROLLO

DISEÑO DE INTERFACES

Proyecto:

Sistema de Control de Pasantías

Fecha Elaboración:

2014-07-21

Técnico:

David Vásconez

Fabricio Robayo

Título:

Operatividad de calificaciones y formatos

Descripción:

The screenshot displays the web interface for the 'SISTEMA DE GESTIÓN DE PASANTÍAS'. The header includes the ESPE logo and navigation links like 'Inicio', 'Salir', 'Inicio', 'Procesos', 'Reportes', 'Administrar Usuarios', and 'Ayuda de'. The main content area is titled 'Actualizar Calificación 2' and contains a form with the following fields: 'Calificación Nota 2' (value: 18), 'Calificación Nota 1' (value: 20), and 'Observación' (value: NINGUNA). A 'Guardar' button is located below the form. On the right side, there is a sidebar with the heading 'Operaciones' and a list of actions: 'Lista Calificación', 'Crear Calificación', 'Ver Calificación', and 'Administrar Calificación'. The footer of the interface contains the text: '© 2015 | Todos los Derechos Reservados. Diseñado y Desarrollado por: David Vásconez y Fausto Robayo'.

Ver Pasantía

Pasantía: 1
Numero de Solicitud de la Pasantía: 012753
Numero de Trámite de la Pasantía:
Créditos Aprobados: 150
Fecha de Inicio: 2015-05-01
Fecha de Fin: 2015-05-31
Total de Horas de la Pasantía: 72
Periodo Académico: AGOSTO 2015 - SEPTIEMBRE 2015
Observaciones:
Estado de la Pasantía: CUMPLIDA
Horario de Entrada: 8:00
Horario de Salida: 17:00
Nota:
Etiquetas de Horas:
Nombre del Proyecto: NOMBRE DEL PROYECTO
Area de Conocimiento: REGIMACION
URL Cronograma de Trabajo: el nombre del archivo
Nombre del Tipo de Pasantía: PRÁCTICA PRE PROFESIONAL NO REMUNERADA
Estudiante: YMER CAMILO ACOSTA RODRIGUEZ
Tutor Empresarial: SOFIA ACELA GALLARDO CUEVA
Tutor Académico: CESAR ARMANDO CASTRO NARVAEZ
Pasantías: El estudiante ha cumplido 72 horas de practicas pre profesionales

Pasantías cumplidas:
El estudiante ha cumplido 72 horas de practicas pre profesionales

Operaciones

- ← Crear Pasantía
- ← Actualizar Pasantía
- ← Eliminar Pasantía
- ← Administrar Pasantía
- ← Formato 01 Proyecto Práctica Pre Profesional
- ← Formato 02 Control de Avance de Actividades Semanal
- ← Formato 03 Evaluación de Práctica Pre Profesional Tutor Empresarial
- ← Formato 05 Evaluación de Práctica Pre Profesional Tutor Académico
- ← Certificado

David Vásconez



TÉCNICO RESPONSABLE DEL DESARROLLO

DISEÑO DE INTERFACES

Proyecto:

Sistema de Control de Pasantías

Fecha Elaboración:

2014-08-11

Técnico:

David Vásconez

Fabricio Robayo

Título:

Diseño de los reportes

Descripción:

SISTEMA DE GESTIÓN DE PASANTÍAS

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fabricio Robayo

Resumen de Empresas
REPORTE GENERAL DE EMPRESAS

ORD.	EMPRESA	REGISTRADOS
1	ICONO SISTEMAS	2
2	COMANDANCIA GENERAL DEL EJERCITO - DIC	2
3	CORPORACION NACIONAL DE TELECOMUNICACIONES CNT	2
4	BABEL SOFTWARE ECUADOR	1
TOTAL		7

SISTEMA DE GESTIÓN DE PASANTÍAS

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fabricio Robayo

Detalle de Estudiantes por Empresa
REPORTE DETALLADO DE ESTUDIANTES POR EMPRESA

EMPRESA: ICONO SISTEMAS

ORD.	NOMBRE	CEDULA	TELEFONO	CORREO
1	RICARDO DAVID VASCONEZ GARCES	1804485944	100298513	0595681942
2	FAUSTO FABRICIO ROBAYO ZURITA	1803282558	100381287	0592693325
3	RICARDO DAVID VASCONEZ GALARZA	1804485944	100298513	0595681942

TUTOR ACADÉMICO: CESAR ARMANDO CASTRO NARVAEZ ▾

Detalle de Pasantías por Tutor Académico

REPORTE DETALLADO DE PASANTÍAS POR TUTOR ACADÉMICO

TUTOR ACADÉMICO: CESAR ARMANDO CASTRO NARVAEZ

ORD	ESTUDIANTE	EMPRESA	FECHA INICIO	FECHA FIN	ESTADO
1	YIMIR CAMILO ACOSTA RODRIGUEZ	COOPERACION NACIONAL DE TELECOMUNICACIONES CNT	2015-05-01	2015-05-31	CUMPLIDA

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

Operaciones

- ◀ Imprimir
- ◀ Exportar a Hoja de Cálculo

ESTUDIANTES: RICARDO DAVID VASCONEZ GARCES ▾

Detalle de Pasantías por Estudiante

REPORTE DETALLADO DE PASANTÍAS POR ESTUDIANTE

ESTUDIANTE: RICARDO DAVID VASCONEZ GARCES

ORD.	EMPRESA	FECHA INICIO	FECHA FIN	N HORAS	ESTADO
1					NEGADA

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

Operaciones

- ◀ Imprimir
- ◀ Exportar a Hoja de Cálculo

ESTUDIANTES: En Progreso ▾

- Selección
- En Progreso
- Cumplidas
- Rechazadas
- Negadas
- Reprobadas

Detalle de Pasantías por Estado

REPORTE DETALLADO DE PASANTÍAS POR ESTADO

ESTADO: EN PROGRESO

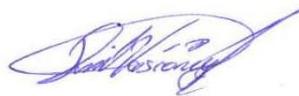
ORD.	ESTUDIANTE	TIPO	CREDITOS APROBADOS
1	RICARDO DAVID VASCONEZ GALARZA	PRÁCTICA PFE PROFESIONAL NO REMUNERADA	150

© 2015 | Todos los Derechos Reservados
Diseñado y Desarrollado por: David Vásconez y Fausto Robayo

Operaciones

- ◀ Imprimir
- ◀ Exportar a Hoja de Cálculo

David Vásconez



TÉCNICO RESPONSABLE DEL DESARROLLO

DISEÑO DE BASE DE DATOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-05-19	Versión del Software: 1.0
Requisitos Implementados: R001-V1.0 R002-V1.0 R003-V1.0 R004-V1.0	Técnico: David Vásconez Fabricio Robayo
Descripción:	
<pre> -- sgp_departamento CREATE TABLE sgp_departamento (departamento_id INT(11) PRIMARY KEY, departamento_id_banner VARCHAR(10), departamento_nombre VARCHAR(100), departamento_siglas VARCHAR(5), departamento_director VARCHAR(100)) -- sgp_carrera CREATE TABLE sgp_carrera (carrera_id INT(11) PRIMARY KEY, carrera_id_banner VARCHAR(10), carrera_nombre VARCHAR(50), carrera_siglas VARCHAR(5), carrera_tipo VARCHAR(20), carrera_minimo_creditos VARCHAR(10), carrera_director VARCHAR(100), carrera_director_cedula VARCHAR(15), carrera_departamento INT(11) FOREIGN KEY REFERENCES sgp_departamento(departamento_id)) -- sgp_empresa CREATE TABLE sgp_empresa (empresa_id INT(11) PRIMARY KEY, empresa_id_banner VARCHAR(10), empresa_nombre VARCHAR(50), empresa_actividad VARCHAR(50), empresa_direccion VARCHAR(100), empresa_representante_legal VARCHAR(50), empresa_titulo VARCHAR(100), empresa_cargo VARCHAR(200), empresa_telefono VARCHAR(10), empresa_correo VARCHAR(50)) </pre>	
<p>Fabricio Robayo</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>	

DISEÑO DE BASE DE DATOS	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-02	Versión del Software: 2.0
Requisitos Implementados: R001-V2.0 R002-V2.0 R003-V2.0 R004-V2.0	Técnico: David Vásquez Fabricio Robayo
Descripción:	
<pre> graph TD app_tipo_pasantia --> app_departamento app_tipo_pasantia --> app_carrera app_instrumento_evaluacion --> app_carrera app_convencio --> app_carrera app_convencio --> app_empresa app_indicador_resultados --> app_carrera </pre> <p>app_tipo_pasantia</p> <ul style="list-style-type: none"> app_tipo_pasantia_id : int(11) app_tipo_pasantia_nombre : varchar(50) <p>app_departamento</p> <ul style="list-style-type: none"> departamento_id : int(11) departamento_id_banner : varchar(10) departamento_nombre : varchar(100) departamento_siglas : varchar(5) departamento_director : varchar(100) <p>app_carrera</p> <ul style="list-style-type: none"> carrera_id : int(11) carrera_id_banner : varchar(10) carrera_nombre : varchar(50) carrera_siglas : varchar(5) carrera_tipo : varchar(20) carrera_minimo_creditos : varchar(10) carrera_director : varchar(100) carrera_director_cedula : varchar(15) carrera_departamento : int(11) <p>app_instrumento_evaluacion</p> <ul style="list-style-type: none"> evaluacion_id : int(11) evaluacion_criterio : varchar(100) evaluacion_descripcion : varchar(500) <p>app_convencio</p> <ul style="list-style-type: none"> convencio_id : int(11) convencio_id_banner : varchar(10) convencio_tipo_de_relacion : varchar(50) convencio_fecha_inicio : date convencio_fecha_fin : date convencio_numero_documento : varchar(15) convencio_empresa : int(11) convencio_estudiante : int(11) <p>app_indicador_resultados</p> <ul style="list-style-type: none"> indicador_resultados_id : int(11) indicador_resultados_id_banner : varchar(10) indicador_resultados_descripcion : varchar(500) indicador_resultados_carrera : int(11) <p>app_empresa</p> <ul style="list-style-type: none"> empresa_id : int(11) empresa_id_banner : varchar(10) empresa_nombre : varchar(50) empresa_actividad : varchar(50) empresa_direccion : varchar(100) empresa_representante_legal : varchar(50) empresa_titulo : varchar(100) empresa_cargo : varchar(200) empresa_telefono : varchar(10) empresa_correo : varchar(50) 	
<p>Fabricio Robayo</p> <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>	

DISEÑO DE BASE DE DATOS

Proyecto:

Sistema de Control de Pasantías

Fecha Elaboración:

2014-06-23

Versión del Software:

4.0

Requisitos Implementados:

R001-V4.0

R002-V4.0

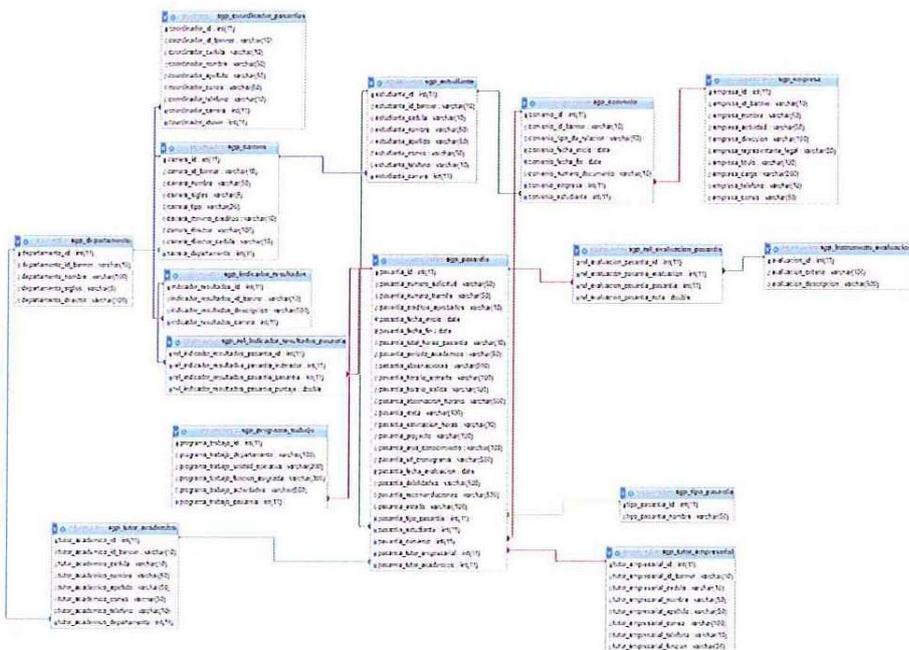
R003-V4.0

Técnico:

David Vásquez

Fabrizio Robayo

Descripción:



Fabrizio Robayo

TÉCNICO RESPONSABLE DEL DESARROLLO

Anexo 4 Implementación

VERSIÓN DE SOFTWARE	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-05-19	Versión del Software: 1.0
Requisitos Implementados: R001-V1.0 R002-V1.0 R003-V1.0 R004-V1.0	Requisitos Corregidos: N/A
Descripción: <p>Se realiza el diseño e implementación de los menús que contendrá el sistema; el registro, modificación y eliminación de los departamentos, carreras y empresas.</p> <p>Para realizar el ingreso, modificación o eliminación de carreras se debe tomar en cuenta que es dependiente de los departamentos; es decir, para eliminar un departamento, este no debe tener carreras asignadas a ese departamento.</p> <p>El código perteneciente a la versión 1.0 se encontrará en el repositorio de las TIC's sede Latacunga en la carpeta asignada para el efecto.</p>	
David Vásquez  TÉCNICO RESPONSABLE DEL DESARROLLO	

VERSIÓN DE SOFTWARE	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-02	Versión del Software: 2.0
Requisitos Implementados: R001-V2.0 R002-V2.0 R003-V2.0 R004-V2.0	Requisitos Corregidos: N/A
Descripción: Se realiza el registro, modificación y eliminación de los convenios, indicadores de resultados, instrumentos de evaluación y tipos de pasantías. Para realizar el ingreso, modificación o eliminación de convenios se debe tomar en cuenta que es dependiente de las empresas; es decir, para eliminar una empresa, esta no debe tener convenios asignados a esa empresa. Para realizar el ingreso, modificación o eliminación de indicadores de resultados se debe tomar en cuenta que es dependiente de las carreras; es decir, para eliminar una carrera, esta no debe tener indicadores de resultados asignados a esa carrera. El código perteneciente a la versión 2.0 se encontrará en el repositorio de las TIC's sede Latacunga en la carpeta asignada para el efecto.	
David Vásquez  TÉCNICO RESPONSABLE DEL DESARROLLO	

VERSIÓN DE SOFTWARE	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-16	Versión del Software: 3.0
Requisitos Implementados: R001-V3.0 R002-V3.0 R003-V3.0	Requisitos Corregidos: N/A
<p>Descripción: Se realiza el registro, modificación y eliminación de los coordinadores de pasantías, estudiantes y tutores académicos.</p> <p>Para realizar el ingreso, modificación o eliminación de coordinadores de pasantías se debe tomar en cuenta que es dependiente de las carreras; es decir, para eliminar una carrera, esta no debe tener coordinadores de pasantías asignados a esa carrera.</p> <p>Para realizar el ingreso, modificación o eliminación de estudiantes se debe tomar en cuenta que es dependiente de las carreras; es decir, para eliminar una carrera, esta no debe tener estudiantes asignados a esa carrera.</p> <p>Para realizar el ingreso, modificación o eliminación de tutores académicos se debe tomar en cuenta que es dependiente de los departamentos; es decir, para eliminar un departamento, este no debe tener tutores académicos asignados a ese departamento.</p> <p>El código perteneciente a la versión 3.0 se encontrará en el repositorio de las TIC's sede Latacunga en la carpeta asignada para el efecto.</p>	
<p>David Vásquez</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>	

VERSIÓN DE SOFTWARE	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-23	Versión del Software: 4.0
Requisitos Implementados: R001-V4.0 R002-V4.0 R003-V4.0	Requisitos Corregidos: N/A
<p>Descripción:</p> <p>Se realiza el registro, modificación y eliminación de las pasantías, tutores empresariales y programas de trabajo.</p> <p>Para realizar el ingreso, modificación o eliminación de pasantías se debe tomar en cuenta que es dependiente de los estudiantes, convenios, tutores empresariales y tutores académicos; es decir, para eliminar un estudiante, convenio, tutor empresarial o tutor académico, estos no deben tener pasantías asignados a esos estudiantes, convenios, tutores empresariales o tutores académicos.</p> <p>Para realizar el ingreso, modificación o eliminación de programas de trabajo se debe tomar en cuenta que es dependiente de las pasantías; es decir, para eliminar una pasantía, esta no debe tener programas de trabajo asignados a esa pasantía.</p> <p>El código perteneciente a la versión 4.0 se encontrará en el repositorio de las TIC's sede Latacunga en la carpeta asignada para el efecto.</p>	
<p>David Vásconez</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>	

VERSIÓN DE SOFTWARE	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-07-21	Versión del Software: 5.0
Requisitos Implementados: R001-V5.0 R002-V5.0	Requisitos Corregidos: N/A
Descripción: Se realiza el registro, modificación y eliminación de las calificaciones y el diseño e implementación de los formatos pre establecidos para la impresión en los diferentes procesos de la pasantía. Para realizar el ingreso, modificación o eliminación de calificaciones se debe tomar en cuenta que es dependiente de las pasantías; es decir, para eliminar una pasantía, esta no debe tener calificaciones asignadas a esa pasantía. El código perteneciente a la versión 5.0 se encontrará en el repositorio de las TIC's sede Latacunga en la carpeta asignada para el efecto.	
David Vásquez  TÉCNICO RESPONSABLE DEL DESARROLLO	

VERSIÓN DE SOFTWARE	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-08-11	Versión del Software: 6.0
Requisitos Implementados: R001-V6.0	Requisitos Corregidos: N/A
Descripción: <p>Se realiza el diseño e implementación de los reportes asignados con su respectivo botón para exportar a PDF y a Excel.</p> <p>El código perteneciente a la versión 6.0 se encontrará en el repositorio de las TIC's sede Latacunga en la carpeta asignada para el efecto.</p>	
<p>David Vásquez.</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>	

Anexo 5 Pruebas

PRUEBAS DE SISTEMA	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-05-30	Versión del Software: 1.0
Requisitos Validados: R001-V1.0 R002-V1.0 R003-V1.0 R004-V1.0	Técnico Revisor: Fabricio Robayo
Tiempo de Revisión: 4h	Tiempo de Soluciones: 5h
Descripción de Fallos: Se ingresó una carrera pero al seleccionar el departamento no se mostraron todos los departamentos que esto afecta al ingreso de la carrera porque no se puede seleccionar la carrera específica que necesitamos, se cambiara la consulta en la base de datos para que se visualicen todos los departamentos el sistema se encuentra en desarrollo	

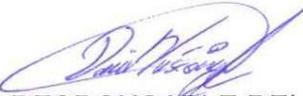
No.	Fallo:	Solución:
001-V1.0	No se visualizan todas las carreras	Modificar el barrido de la consulta a la base de datos para que se visualicen todas las carreras

David Vásconez



TÉCNICO RESPONSABLE DEL DESARROLLO

PRUEBAS DE SISTEMA	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-13	Versión del Software: 2.0
Requisitos Validados: R001-V2.0 R002-V2.0 R003-V2.0 R004-V2.0	Técnico Revisor: Fabricio Robayo
Tiempo de Revisión: 4h	Tiempo de Soluciones: 5h
Descripción de Fallos: Los indicadores de resultados se visualizan con el mismo formato para todas las carreras, afecta el desempeño en la aplicación ya que el formato es específico de cada carrera y no es un general, la solución será generar un formulario donde se pueda modificar los formatos para cada carrera	

No.	Fallo:	Solución:
001-V2.0	Fallo en los Indicadores de resultados por carrera	General una nueva tabla en la base de datos y crear un formulario para el ingreso para que los indicadores sean específicos
<p>David Vásquez</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>		

PRUEBAS DE SISTEMA	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-06-20	Versión del Software: 3.0
Requisitos Validados: R001-V3.0 R002-V3.0 R003-V3.0	Técnico Revisor: Fabricio Robayo
Tiempo de Revisión: 2h	Tiempo de Soluciones: 3h
Descripción de Fallos: En el formulario de estudiante no me valida el número de cedula y el campo de correo electrónico, afecta al desempeño de la aplicación que se puede mandar datos basura a la base de datos la solución es generar funciones para que se validen los campos antes de que se envíen a la base de datos el sistema está en desarrollo	

No.	Fallo:	Solución:
001-V3.0	Validación de campos de Estudiante	Realizar función para validar los campos
<p>David Vásquez</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>		

PRUEBAS DE SISTEMA	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-07-18	Versión del Software: 4.0
Requisitos Validados: R001-V4.0 R002-V4.0 R003-V4.0	Técnico Revisor: Fabricio Robayo
Tiempo de Revisión: 8h	Tiempo de Soluciones: 10h
Descripción de Fallos: En el formato de pasantía se listan en un combo todos los nombre de los estudiantes ingresados en toda la carrera el combo se genera dependiendo el número de estudiantes y con aumento de estudiantes se va a generar un combo que va a dificultar la búsqueda del estudiante en específico, se debe solucionar para que la búsqueda sea más fácil y no por un combo si no por una consulta directa a la base de datos	

No.	Fallo:	Solución:
001-V4.0	Búsqueda de estudiante	Crear una función para que por medio de autocomplete haga la búsqueda del estudiante específico
<p>David Vásquez</p>  <p>TÉCNICO RESPONSABLE DEL DESARROLLO</p>		

PRUEBAS DE SISTEMA		
Proyecto: Sistema de Control de Pasantías		
Fecha Elaboración: 2014-03-11	Versión del Software: 5.0	
Requisitos Validados: R001-V5.0 R002-V5.0	Técnico Revisor: Fabricio Robayo	
Tiempo de Revisión: 4h	Tiempo de Soluciones: 5h	
Descripción de Fallos: El formulario de calificación no se genera en los formatos, solución revisar porque en el formato no se visualiza las calificaciones		
No.	Fallo:	Solución:
001-V5.0	Formatos Incompletos	Revisar el diseño de los formatos
David Vásconez		
		
TÉCNICO RESPONSABLE DEL DESARROLLO		

PRUEBAS DE SISTEMA	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2014-08-15	Versión del Software: 6.0
Requisitos Validados: R001-V6.0	Técnico Revisor: Fabricio Robayo
Tiempo de Revisión: 2h	Tiempo de Soluciones: 3h
Descripción de Fallos: Los datos generados en los reportes de necesita generar en formato XML, la solución es generar los reportes en XML con alguna herramienta que lo permita	

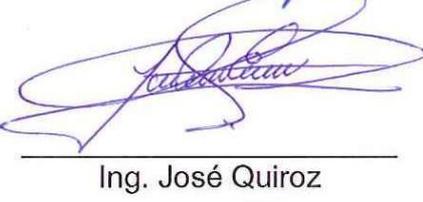
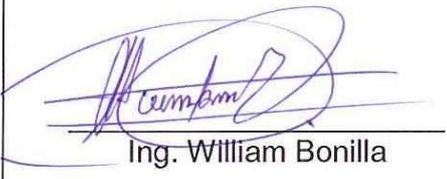
No.	Fallo:	Solución:
001-V6.0	Generar reportes en XML	Buscar una herramienta que genere los reportes en XML.

David Vásconez



TÉCNICO RESPONSABLE DEL DESARROLLO

Anexo 6 Liberación

ENTREGA DEL SISTEMA	
Proyecto: Sistema de Control de Pasantías	
Fecha Elaboración: 2015-05-20	Versión del Software: 6.3
Personal Capacitado:  Ing. Ximena López  Ing. Cesar Naranjo  Ing. Misael Pazmiño  Ing. José Quiroz  Ing. William Bonilla  Ing. Héctor Terán	Técnico Capacitador: Ricardo David Vásquez Garcés Fausto Fabricio Robayo Zurita

Novedades de Puesta en Marcha y la Capacitación:

Dentro de las novedades expresadas de los asistentes se encuentra: la no satisfacción en el formato número 06 (resumen de las pasantías realizadas para la Universidad de las Fuerzas Armadas Matriz) no es el solicitado, ya que se trabajó con el formato 2015; sin embargo, el nuevo es evaluado pero los requerimientos no se encuentran considerados en la base de datos. No se contemplaron porque no estaban en los requerimientos iniciales.

Detalle de la Versión:

La versión del software gestiona todo el proceso de prácticas pre profesionales; iniciando desde el ingreso de departamentos, carreras, empresas, convenios, indicadores de resultados, instrumentos de evaluación, tipos de pasantías, coordinador de pasantías, estudiantes, tutores académicos, pasantías, tutores empresariales, programas de trabajo, calificaciones; además administra los convenios relacionados con cada uno de los estudiantes y empresas relacionadas; como también las practicas pre profesionales durante todo su proceso; la asignación del estudiante a la pasantía, la validación del número de créditos para la aceptación de la pasantía, la asignación del convenio y el tutor académico, la asignación del tutor empresarial, las fechas y horarios de cada pasantía, los proyectos que se realizaran en cada una, los programas de trabajo asociados a la pasantía, las calificaciones y evaluaciones de los tutores académicos y empresariales. Adicionalmente el sistema genera todos los formatos requeridos por el proceso que son:

Carta de compromiso

Formato 1: Proyecto Práctica Pre Profesional

Formato 2: Control de Avance de Actividades

Formato 3: Evaluación Tutor Empresarial

Formato 4: Informe

Formato 5: Evaluación Tutor Académico

Formato 6: Resumen de Prácticas Pre Profesionales 2015

Formato 7: Modelo de Certificado

Y los reportes requeridos que son:

Reporte – Resumen de Empresas

Reporte – Detalle de Estudiantes por Empresa

Reporte – Detalle de Practicas Pre Profesionales por Tutor Académico

Reporte – Detalle de Practicas Pre Profesionales por Estudiante

Reporte – Detalle de Practicas Pre Profesionales por Estado

Ing. Armando Álvarez



RESPONSABLE UNIDAD
REQUIRENTE

Fabricio Robayo



RESPONSABLE DEL GRUPO DE
DESARROLLO

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE

CARRERA DE INGENIERÍA DE SOFTWARE

CERTIFICACIÓN

Se certifica que el presente trabajo fue desarrollado por los señores Ricardo David Vásconez Garcés y Fausto Fabricio Robayo Zurita bajo nuestra supervisión

Ing. Fabián Montaluisa
DIRECTOR DE PROYECTO

Ing. Ximena López
COORDIRECTOR DE PROYECTO

Ing. Lucas Garcés
DIRECTOR DE CARRERA

Dr. Rodrigo Vaca
SECRETARIO ACADÉMICO