

Universidad de Sancti Spíritus “José Martí Pérez”

Facultad de Ingeniería

Carrera Ingeniería Informática



**Aplicación web para mejorar la gestión de la información
de los recursos humanos de la UEB de Investigación y
Proyectos Hidráulicos de Sancti Spíritus.**

**Trabajo de Diploma para optar por el Título de Ingeniería
Informática.**

Autora: Dunia Alonso León.

Tutores: MsC. Yaikiel Hernández Díaz.

Ing. Héctor Pérez Hernández.

Sancti Spíritus, junio, 2013



*...Querido me es el sueño, y más el de piedra;
Mientras que perdure el daño y la vergüenza:
No ver, no sentir, es mi gran ventura;
Pero no me despierte, ¡oh! hable suavemente...*

Ergo Proxy

*A las personas de mi vida,
Especialmente
A mi madre y abuela.*

Agradecimientos

Quiero agradecer...

A *Alejandro* : ¿Y esa luz? ¿Es tu sombra?

Por dar, por estar, por ser...

Por tu ayuda,

Por tus clases



A mi mamá , a mi papá , a mi amigo *Orelvis*



Porque son parte de mi,

Porque son esenciales,

Porque los quiero.

A *Belkís* , mi suegra:

Por enseñarme y ayudarme,

Porque haz sido extraordinaria,

Por tu apoyo,

Y porque *Ale* es lo mejor que me ha pasado

A mi tutor *Yaikiel* ,

Por tu apoyo,

Por enseñarme y ayudarme.



A los amigos y las personas de mi vida, de mi historia:

Por creer...

Por ser.



A cada detalle que hizo posible que llegase hasta aquí...

Gracias!

Resumen

Una empresa cuenta con diversos tipos de recursos que le permiten funcionar y alcanzar sus metas. Los empleados, trabajadores y colaboradores son quienes conforman lo que se conoce como recursos humanos de una entidad. Por lo general, la política de recursos humanos (RR.HH.) se centra en lograr que los empleados estén en la misma sintonía que la empresa, haciendo coincidir los anhelos y las aspiraciones de los trabajadores con la estrategia de la compañía.

La importancia de los recursos humanos no ha pasado inadvertida por la industria de software. Existen ahora diversas aplicaciones disponibles para ayudar al departamento de recursos humanos en sus tareas.

Los mismos por sus características muy genéricas no se corresponden con el problema actual que presenta la UEB de Investigaciones y Proyectos Hidráulicos de Sancti Spíritus por lo que se hace necesario otra solución. El sistema requiere de nuevas funcionalidades por lo cual se implementará una nueva aplicación.

La UEB de Investigaciones y Proyectos Hidráulicos de Sancti Spíritus reconoce la importancia y necesidad de la plena satisfacción del capital humano con que cuenta, a partir de la adopción de un Sistema de Gestión de Recursos Humanos para el mejoramiento continuo y alcanzar un desempeño laboral superior.

En la implementación de la aplicación se utilizó como lenguaje PHP 5, que permitió un código eficiente y rápido a la hora de procesar la información, para la persistencia de los datos se utilizó PostgreSQL 9.0 por considerarse un sistema gestor de bases de datos muy potente y como servidor web se escogió Apache por la seguridad y estabilidad que presenta. El sistema se implementó haciendo uso del framework Symfony 2.2 el cual incluye facilidades de desarrollo para aplicaciones web con arquitectura tres capas y utiliza el patrón Modelo Vista Controlador.

Abstract

A company has diverse types of resources that allow it to work and to reach its goals. The employees, workers and collaborators are who conform what is known as human resources of an entity.

In general, the human resources politics (which can be abbreviated as RR.HH.) it is centered in achieving the employees to be in the same wavelength of the company, making coincide the yearnings and the aspirations of the workers with the strategy of the company.

The importance of the human resources has not happened inadvertent for the software industry. There are up too date diverse available applications to help the department of human resources in their tasks.

The same ones for their very generic characteristics do not correspond with the current problem which presents the Company for what becomes necessary another solution. The system requires reason why of new functionalities a new application it was implemented.

The UEB of Investigations and Hydraulic Projects of Sancti Spíritus recognizes the importance and necessity of the complete satisfaction of the human capital with which it counts, starting from the adoption of a System of Administration of human resources for the continuous improvement and to reach an acting labor superior.

In the implementation of the application it was used as language PHP which allowed an efficient and quick code when processing the information, for the persistence of the data PostgreSQL it was used to be considered a very potent SGBD and It was set as servant web Apache it was chosen by the security and stability that it presents.

Índice

Introducción	1
Capítulo I: Fundamentación Teórica	5
1.1 Sistema de Gestión.	5
1.1.1 Gestión.	5
1.1.2 Tipos de Gestión.....	¡Error! Marcador no definido.
1.1.3 Sistema de Gestión.	7
1.1.4 Tipos de Sistemas de Gestión.	¡Error! Marcador no definido.
1.2 Recursos Humanos.....	6
1.3 Sistemas de Gestión de Recursos Humanos.	10
1.4 Metodologías de desarrollo de software.	12
1.4.1 Características de las metodologías de desarrollo.¡Error! Marcador no	
definido.	
1.4.2 Comparación Ágil - No-Ágil.....	12
1.5 Tecnologías y herramientas.	16
Introducción.....	¡Error! Marcador no definido.
2.1 Propuesta de solución.....	26
2.2 Fase Planificación.	26

2.3 Fase de Diseño.....	34
Diagrama de clases persistente.....	¡Error! Marcador no definido.
2.4 Fase de Desarrollo.....	39
2.5 Fase Prueba.....	¡Error! Marcador no definido.
Conclusiones	¡Error! Marcador no definido.
Capítulo III: Respuesta a la Solución Propuesta	¡Error! Marcador no definido.
3.1 Caso de estudio.....	¡Error! Marcador no definido.
3.2 Planificación.....	¡Error! Marcador no definido.
3.3 Diseño.....	¡Error! Marcador no definido.
3.4 Desarrollo.....	¡Error! Marcador no definido.
3.5 Prueba	¡Error! Marcador no definido.
Conclusiones	48
Conclusiones generales.....	49
Recomendaciones	50
Anexos	51

Introducción

Las Tecnologías de la Información y las Comunicaciones (TIC) han adquirido un valor extraordinario en el mundo moderno. Prácticamente están presentes en todos los niveles de actividad y ramas de la economía, la política y la sociedad.

La aparición de las TIC ha impulsado la expansión del conocimiento en todas las direcciones de la sociedad, tanto en los países desarrollados como en los subdesarrollados.

En Cuba, el campo de la Gestión de la Información (GI) ha experimentado cambios ventajosos tras la utilización de las TIC, para el desarrollo de grandes sistemas de gestión y control, los cuales han facilitado el viejo proceso de recogida y análisis de información, permitiendo una mejor centralización, accesibilidad y manipulación de la misma. Además, se han proporcionado nuevos resultados estadísticos debido al descubrimiento de nuevas interrelaciones entre los datos manejados, permitiendo lograr mejoras en la toma de decisiones en el proceso de gestión de la información.

La gestión de recursos humanos se ha definido como “la ciencia y la práctica que se ocupan de la naturaleza de las relaciones de empleo y del conjunto de decisiones, acciones y cuestiones vinculadas a dichas relaciones” (Ferris, Rosen y Barnum 1995).

Aborda las políticas y las prácticas empresariales que consideran la utilización y la gestión de los trabajadores como un recurso de la actividad en el contexto de la estrategia general de la empresa encaminada a mejorar la productividad y la competitividad. Se trata de un término que suele describir el enfoque empresarial de la administración de personal basado en la prioridad concedida a la participación de los trabajadores, normalmente, aunque no siempre, en centros de trabajo sin presencia sindical, con el fin de motivarles para que aumenten su productividad.

La UEB de Investigaciones y Proyectos Hidráulicos de Sancti Spíritus reconoce desde su estrategia corporativa la importancia y necesidad de la plena satisfacción del capital humano con que cuenta, a partir de la adopción de un Sistema de Gestión Integrada de Capital Humano como decisión estratégica de la alta dirección que incluye la búsqueda de soluciones a los problemas y la toma de decisiones que tiene como objetivo la integración de la Gestión de Recursos Humanos para el mejoramiento continuo y alcanzar un desempeño laboral superior, donde participan activamente los trabajadores y la organización sindical.

Con su implementación, evaluación y mejora pretendemos alcanzar mejores resultados en la gestión del capital humano y por ende contribuir a elevar significativamente la motivación y el compromiso del capital humano de la organización. Esta aspiración está en correspondencia con las aspiraciones del país y sector de contar con un recurso humano más comprometido y competente.

Dicha empresa cuenta con un sistema automatizado implementado en Access mediante el cual se gestionan los recursos. El uso de la misma ha presentado algunas desventajas funcionales como: problemas de actualización de la base de datos, no logra explotar al máximo las condiciones que en medios informáticos cuenta la entidad, interfaz poco intuitiva para el trabajo de los directivos, así como conflictos de almacenamiento de indicadores que exceden los valores registrados en la base de datos. La empresa tiene dentro de sus lineamientos lograr un desarrollo en el área informática que le permita tener un mejor aprovechamiento de sus potencialidades.

A partir de la situación problemática se propone el siguiente **problema a resolver**:

¿Cómo mejorar la gestión de los Recursos Humanos en La UEB de Investigaciones y Proyectos Hidráulicos de Sancti Spíritus?

Debido a esto se ha propuesto como **objetivo general** desarrollar el módulo para la gestión de los recursos humanos del IPHSG.

Dicho objetivo ha sido desglosado en los siguientes **objetivos específicos**:

1. ¿Cuáles son los fundamentos teóricos y metodológicos que sustentan el desarrollo de un software que mejore la gestión de la información de los recursos humanos?
2. ¿Cómo planificar un software que mejore la gestión de la información de los recursos humanos de la UEB de Investigación y Proyectos Hidráulicos de Sancti Spíritus?
3. ¿Cómo diseñar un software que mejore la gestión de la información de los recursos humanos de la UEB de Investigación y Proyectos Hidráulicos de Sancti Spíritus?
4. ¿Cómo implementar un software que mejore la gestión de la información de los recursos humanos de la UEB de Investigación y Proyectos Hidráulicos de Sancti Spíritus?

Para darle cumplimiento al objetivo general se plantean las siguientes **tareas de investigación**:

1. Determinar los fundamentos teóricos y metodológicos que sustentan el desarrollo de un software que mejore la gestión de la información de los recursos humanos.
2. Planificar un software que mejore la gestión de la información de los recursos humanos de la UEB de Investigación y Proyectos Hidráulicos de Sancti Spíritus.
3. Diseñar un software que mejore la gestión de la información de los recursos humanos de la UEB de Investigación y Proyectos Hidráulicos de Sancti Spíritus.
4. Implementar un software que mejore la gestión de la información de los recursos humanos de la UEB de Investigación y Proyectos Hidráulicos de Sancti Spíritus.

Una vez culminada las tareas de la investigación se pretende obtener como resultado: la implementación de un sistema informático que gestione los recursos humanos en la UEB de Investigación y Proyectos Hidráulicos, contando con una base de datos propia que almacene toda la información requerida.

El presente documento está estructurado en tres capítulos:

Capítulo I: Fundamentación teórico y metodológica que sustenta el desarrollo de un software para mejorar la gestión de la información de los recursos humanos.

En este capítulo se abordan la fundamentación teórico y metodológica que sustenta el desarrollo de un software para mejorar la gestión de la información de los recursos humanos y los conceptos asociados al dominio del problema. Se explica en detalles el problema a resolver y se describe el campo de acción donde se desarrolla. Así como las tecnologías y metodologías utilizadas para su desarrollo teniendo en cuenta las tendencias actuales en el desarrollo de las TIC.

Capítulo II: Planificación y diseño del software para mejorar la gestión de la información de los recursos humanos en la UEB Investigación y Proyectos Hidráulicos de Sancti Spíritus.

En este capítulo se explican a partir de la metodología *Extreme Programming* (XP) las historias de usuarios generadas por los requerimientos funcionales del sistema, el plan de entrega, iteraciones, reuniones diarias y las tarjetas CRC pertenecientes a la fase de planificación y diseño brindando así una visión más clara del producto y una mejor comprensión del negocio.

Capítulo III: Implementación y prueba del software para mejorar la gestión de la información de los recursos humanos en la UEB Investigación y Proyectos Hidráulicos de Sancti Spíritus.

En este capítulo se presentan las tareas de programación por iteraciones en las cuales se analiza el tiempo real de programación por Bundle y las tareas de programación por historias de usuarios.

Capítulo I: Fundamentación teórico y metodológica que sustenta el desarrollo de un software para mejorar la gestión de la información de los recursos humanos.

En el presente capítulo se ofrece la fundamentación teórica y metodológica sobre la gestión de usuarios además de analizar a fondo las causas que dan lugar a la situación problemática y describir los conceptos asociados. También se realiza un análisis sobre las tecnologías, lenguajes y herramientas que van a ser las utilizadas en el proceso de desarrollo del sistema.

1.1 Gestión.

Actividades coordinadas para dirigir y controlar una entidad (Sherman, 2006). Son guías para orientar la acción, previsión, visualización y empleo de los recursos y esfuerzos a los fines que se desean alcanzar, la secuencia de actividades que habrán de realizarse para lograr objetivos y el tiempo requerido para efectuar cada una de sus partes y todos aquellos eventos involucrados en su consecución.

Una gestión, entonces, podrá estar orientada a resolver un problema específico, a concretar un proyecto, un deseo, pero también puede referir a la dirección y administración que se realiza en una empresa, una organización, un negocio, e incluso a nivel de gobierno, es común que la tarea que lleva a cabo el gobierno de un determinado país sea también denominada como gestión (María Isabel Día).

En tanto, nos podemos encontrar con diversos tipos de gestión, dependiendo del ámbito en el cual se desarrolle la misma, así aparecerán la gestión social, la gestión de proyectos, la gestión del conocimiento y la gestión ambiental.

La Gestión Social se ocupa de construir diversos espacios destinados a la interacción social y a la superación de aquellos problemas u obstáculos que se presentan en las comunidades y que impiden el normal funcionamiento y existencia de algunos grupos.

La Gestión de Proyectos se encargará de administrar y organizar los recursos con el claro objetivo que se pueda concretar todo el trabajo que requiere un proyecto dentro del tiempo pautado y con los recursos que se disponen, ni más ni menos.

La Gestión del Conocimiento una cuestión ampliamente difundida a nivel organizacional, ya que se ocupará de la transferencia de precisamente el conocimiento y la experiencia entre los miembros que componen la organización y de esta manera el conocimiento resultará siendo un recurso disponible para todos los componentes de la misma.

Y finalmente aparece la Gestión Ambiental la que es en realidad un concepto de gestión no tan difundido, pero decisivo a la hora de la vida, ya que la misma refiere al conjunto de diligencias dedicadas al manejo del sistema ambiental en base al desarrollo sostenible. A través de esta se organizarán todas aquellas actividades orientadas a darle a una comunidad la mejor calidad de vida posible.

1.2 Recursos Humanos.

En la administración de empresas, se denomina recursos humanos (RRHH) al trabajo que aporta el conjunto de los empleados o colaboradores de una organización. Pero lo más frecuente es llamar así a la función o gestión que se ocupa de seleccionar, contratar, formar, emplear y retener a los colaboradores de la organización. Estas tareas las puede desempeñar una persona o departamento en concreto —los profesionales en Recursos Humanos— junto a los directivos de la organización.

El objetivo básico que persigue la función de Recursos Humanos con estas tareas es alinear el área o profesionales de RRHH con la estrategia de la organización, lo que permitirá implantar la estrategia organizacional a través de las personas, quienes son consideradas como los únicos recursos vivos e inteligentes capaces de llevar al éxito organizacional y enfrentar los desafíos que hoy en día se percibe en la fuerte competencia mundial. Es imprescindible resaltar que no se administran personas ni recursos humanos, sino que se administra con las personas viéndolas como agentes activos y proactivos dotados de inteligencia, creatividad y habilidades intelectuales.

Generalmente la función de Recursos Humanos está compuesta por áreas tales como reclutamiento y selección, contratación, capacitación, administración o gestión del personal durante la permanencia en la empresa (Sherman, 2001).

Recursos humanos es el conjunto de capacidades, conocimientos, habilidades, motivaciones, actitudes y valores de los trabajadores, los colectivos laborales y las entidades, y es el recurso principal y decisivo dentro del proceso de trabajo que actúa sobre el resto de los elementos que en él intervienen (George, 2001).

1.3 Sistema de Gestión.

Un Sistema de Gestión es un conjunto de etapas unidas en un proceso continuo, que permite trabajar ordenadamente una idea hasta lograr mejoras y su continuidad.

Se establecen cuatro etapas en este proceso, que hacen de este sistema, un proceso circular virtuoso, pues en la medida que el ciclo se repita recurrente y recursivamente, se logrará en cada ciclo, obtener una mejora.

Las cuatro etapas del sistema de gestión son:

1. Etapa de Ideación
2. Etapa de Planeación
3. Etapa de Implementación
4. Etapa de Control

Etapa de Ideación: el objetivo es trabajar en la idea que guiará los primeros pasos del proceso de creación que se logra con el sistema de gestión propuesto.

Etapa de Planeación (Planificación): Dentro del proceso, la planificación constituye una etapa fundamental y el punto de partida de la acción directiva, ya que supone el establecimiento de sub-objetivos y los cursos de acción para alcanzarlos.

En esta etapa, se definen las estrategias que se utilizarán, la estructura organizacional que se requiere, el personal que se asigna, el tipo de tecnología que se necesita, el tipo de recursos que se utilizan y la clase de controles que se aplican en todo el proceso.

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

Etapa de Control: el control es una función administrativa, esencialmente reguladora, que permite verificar (o también constatar, palpar, medir o evaluar), si el elemento seleccionado (es decir, la actividad, proceso, unidad, sistema, etc.), está cumpliendo sus objetivos o alcanzando los resultados que se esperan.

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

Los Sistemas de Gestión permiten a cualquier empresa que los implante, la obtención de innumerables beneficios y la mejora de la gestión de todos y cada uno de sus procesos.

Sistemas de Gestión de la Seguridad y Salud en el Trabajo (SST).

Diariamente más organizaciones están mostrando interés en lograr demostrar el firme desempeño de la Seguridad y Salud en el Trabajo (SST), mediante el control de los riesgos laborales y amparándose en lo indicado en la legislación.

Beneficios de la implantación de un Sistema de Gestión de la Seguridad y Salud en el Trabajo (SST):

- Mayor control de los riesgos para la Seguridad y Salud de los Trabajadores (SST).
- Disminuir la inestabilidad laboral y aumentar la productividad.
- Cumplir con la legislación vigente en materia de Prevención de Riesgos Laborales (PRL).
- Reducir costes y sanciones administrativas derivadas del incumplimiento de la legislación en materia de PRL.
- Fomentar la cultura preventiva mediante la integración de la Prevención de Riesgos Laborales en los procesos generales desarrollados por la organización.

🚦 Sistemas de Gestión de la I+D+i (Investigación + Desarrollo + innovación).

Los Sistemas de Gestión de la I+D+i permiten a las empresas y organismos mejorar las actividades de I+D+i que desarrollan y proporcionan pautas para organizar y gestionar todo lo relacionado con este tipo de actividades.

Beneficios de la implantación de un Sistema de Gestión de la I+D+i:

- Integración de la actividades de I+D+i en los procesos generales de la organización.
- Evidenciar la transparencia de las actividades de I+D+i.
- Aportar valor añadido en las actividades de I+D+i.
- Mejorar la imagen de la organización y hacerla más competitiva.

🚦 Sistemas de Gestión de la Responsabilidad Social

Los Sistemas de Gestión de la Responsabilidad Social permiten a cualquier organización la integración de la responsabilidad social empresarial dentro de su estrategia de negocio.

Beneficios de la implantación de un Sistema de Gestión de la Responsabilidad Social:

- Integración dentro de la gestión de la empresa de los aspectos económicos-financieros y de buena dirección junto con los ambientales y sociales.
- Lograr la satisfacción de todas las partes interesadas, no solo de clientes.
- Mejorar la imagen de la empresa y diferenciarse de la competencia.

Sistemas de Gestión de Recursos Humanos.

Los Sistemas de Gestión de Recursos Humanos (SGRH) establecen una relación entre la gestión de recursos humanos y tecnología de la información.

Los HRMS permiten a las empresas automatizar muchos aspectos de la gestión de los recursos humanos, con la doble ventaja de reducir la carga de trabajo del departamento de recursos humanos, y aumentar la eficiencia del departamento de normalización de procesos de RRHH.

El departamento de RRHH juega un papel vital en garantizar el buen funcionamiento de una empresa por lo que es más importante el seguimiento y el análisis de las pautas de trabajo y horario de la mano de obra, lo que permite una mejor gestión de la información que forma las estrategias.

La importancia de los recursos humanos no ha pasado inadvertida por la industria de software. Existe ahora una amplia gama de aplicaciones disponibles para ayudar al departamento de recursos humanos en sus tareas, haciendo posible la automatización de ciertas tareas y ayudar en la organización de muchas otras.

Existen diferentes programas o software destinados a la gestión de recursos humanos.

1. **Autotask:** Contiene todo lo que se necesita para organizar, automatizar y optimizar una empresa desde una única plataforma basada en nube y accesible desde cualquier sitio.

2. **GestPeople:** Podrás conocer de forma rápida información relacionada con tu organización, estructura organizativa, perfiles profesionales, competencias, objetivos, evaluación del desempeño y empleados. Permite al departamento de RRHH acceder a la información de cada empleado desde diferentes sitios, conociendo de forma ágil y sencilla su mapa de ocupaciones, conociendo sus objetivos y sus competencias entre otras aplicaciones.
3. **WorkMeter:** Te ayuda a mejorar la productividad e incrementar el rendimiento de tu empresa fomentando el compromiso, la motivación laboral y la autogestión de los empleados. Ofrece los datos necesarios para implementar políticas de mejores prácticas que permiten aumentar y mejorar la productividad. Incrementa la implicación de los recursos humanos.
4. **E-volution:** Simplifica y automatiza procesos de administración de personal y nóminas. Gestión de posiciones, gestión de competencias, administración del tiempo, administración del personal.
5. **SimpleHRM:** Es una solución Open Source o de código abierto con la cual podrás gestionar de una manera muy completa y eficiente los procesos relacionados con los recursos humanos de tu empresa.
6. **Perfil GX:** Es un software creado en Perú para la administración integral del área de recursos humanos de una empresa y comprende el planeamiento organizacional, selección de personal, administración de contratos. Adicionalmente contiene módulos de capacitación, bienestar social, remuneraciones, planillas, generación de archivos PDT Planilla Electrónica PLAME para Sunat, AfpNet, Eps, para contabilidad, bancos.

Los mismos por sus características muy genéricas no se corresponden con el problema actual que presenta la UEB por lo que se hace necesario otra solución. El sistema requiere de nuevas funcionalidades por lo cual se implementará una nueva aplicación que resuelva todos los problemas ganando así en seguridad, portabilidad y accesibilidad.

1.4 Metodologías de desarrollo de software.

En cuanto al concepto de metodologías de desarrollo varios autores han tratado el tema:

Se define metodología como un conjunto de filosofías, etapas, procedimientos, reglas, técnicas, herramientas, documentación y aspectos de formación para los desarrolladores de sistemas de información (MADDISON 1983).

Según (PIATTINI 1996) una metodología de desarrollo es “un conjunto de procedimientos, técnicas, herramientas, y un soporte documental que ayuda a los desarrolladores a realizar nuevo software”.

En un proyecto de desarrollo de software la metodología define quién debe hacer, qué, cuándo y cómo debe hacerlo. La metodología constituye la columna vertebral del proceso de desarrollo de software. No existe una metodología universal. Las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea configurable (MARQUÈS, 1995).

Las metodologías son el camino a seguir para desarrollar software de una manera sistemática, las metodologías persiguen tres necesidades principales:

- Mejores aplicaciones, conducentes a una mejor calidad.
- Un proceso de desarrollo controlado.
- Un proceso normalizado en una organización, no dependiente del personal.

1.4.1 ¿Metodología ágil o tradicional?

Por la experiencia adquirida en el desarrollo de aplicaciones de software podemos analizar y comparar las diferencias entre las metodologías ágiles y las tradicionales más utilizadas para el desarrollo de software.

Desde hace un tiempo el desarrollo de aplicaciones de software ha aumentado la atención e inclinación de la gente a su adquisición, sin embargo su alto costo, demora en su

implementación y sobre todo la fiabilidad que le pueda brindar el software son los aspectos más observados por el cliente.

Es por eso que al paso del tiempo se han desarrollado nuevas metodologías de desarrollo de software, pensando cada día más en disminuir tiempo y costo y aumentar la calidad y por ende su fiabilidad, para satisfacer las expectativas del cliente y darles un plus que cada empresa le da para mantener la competitividad dentro del mercado de desarrollo de software.

Las metodologías ágiles se han convertido en la solución a todos estos problemas, debido a su simplicidad en sus reglas y prácticas las cuales están orientadas al trabajo en grupos más pequeños, los que implican un menor costo en el desarrollo del software. Además de un menor costo, la simplicidad que se aplica en metodologías ágiles reduce el tiempo de análisis y diseño, además de disminuir la documentación que se realiza dentro del proceso de desarrollo de software.

Las metodologías ágiles se han convertido en las herramientas más utilizadas por las empresas desarrolladoras de software debido al bajo costo en la producción de software y a la reducción en el tiempo de entrega del producto, sin descuidar la calidad y fiabilidad que debe mantener un producto de software que fue desarrollado por un método tradicional.

Las Metodologías Ágiles o “ligeras” constituyen un nuevo enfoque en el desarrollo de software, mejor aceptado por los desarrolladores de e-projects que las metodologías convencionales (ISO-9000, CMM, etc.) debido a la simplicidad de sus reglas y prácticas, su orientación a equipos de desarrollo de pequeño tamaño, su flexibilidad ante los cambios y su ideología de colaboración.

Entre las metodologías ágiles más destacadas hasta el momento podemos nombrar:

- XP
- Scrum

- Crystal Methodologies.
- Dynamic Systems Development Method (DSDM)
- Feature Driven Development (FDD)
- Adaptive Software Development (ASD)
- Lean Development (LD)

Se puede decir que XP le dio conciencia al movimiento actual de metodologías ágiles. Fue creada por Kent Beck en 1996 y define una manera de reunir a clientes y programadores en un equipo firmemente integrado, con condiciones de trabajo que promueven la comunicación y solución de un problema. Debido a la interacción intensa, funciona mejor para equipos pequeños y medianos requiriendo de una gran colaboración y disciplina.

XP se encuentra entre las llamadas Metodologías Ágiles, que surgieron como una alternativa a los procesos de desarrollo de software tradicionales. Sus principios fundamentales son: realimentación entre el cliente y el equipo de desarrollo; comunicación fluida entre todos los participantes; simplicidad en las soluciones implementadas.

Entre las principales características de XP que la diferencian de otras metodologías de desarrollo se encuentran: es una metodología adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. Es una de las metodologías de desarrollo de software con más éxito en la actualidad. Se utiliza en proyectos con equipo de desarrollos pequeños y con plazo de entrega corto. La metodología consiste en una programación rápida o extrema. Una particularidad es que tiene como miembro del equipo al usuario final. (CASTILLO, 2009)

Esta metodología de desarrollo del software está dividida en 4 fases:

1ª Fase: planificación del proyecto. En esta fase se describen las historias de usuarios las cuales son realizadas con la misma finalidad que los casos de usos pero con algunas

diferencias. También se hace necesario la creación de un plan de publicaciones para indicar las historias de usuarios correspondientes que serán publicadas en las versiones del programa. El proyecto se divide en iteraciones de corta duración y al comenzar cada iteración es donde se seleccionan las historias de usuario definidas anteriormente en el plan.

El objetivo principal es mostrar una versión nueva a cada instante para mostrarle al cliente, saber lo que opina y luego seguir programando de forma que se mantenga la comunicación entre el cliente y los desarrolladores. Para lograr una planificación eficiente es necesario desarrollar reuniones diarias para que los desarrolladores expongan sus problemas, soluciones e ideas de forma conjunta.

2ª Fase: diseño del proyecto. Sugiere diseños simples, sencillos y libres de complejidad. Usar glosarios de términos y una correcta especificación de los nombres de métodos y clases que ayude a comprender el diseño y facilite sus posteriores ampliaciones y la reusabilidad del código. Propone además usar la técnica de refactorizar lo que significa mejorar y modificar la estructura y codificación de códigos ya creados sin alterar su funcionalidad.

3ª Fase: implementación del proyecto. Para la codificación se utilizan patrones de codificación ya creados y estándares. El modelo de trabajo se realiza usando repositorios de códigos y modelos de desarrollo colectivo.

4ª Fase: pruebas al software. Hay que asegurarse de que todo lo que se hace funcione correctamente. Para ello, lo mejor es desarrollar la prueba desde el momento que se conocen los casos de uso (o, según XP, las historias del usuario). Por ello, lo mejor es desarrollar las pruebas antes de generar el código para tener una prueba más objetiva del correcto funcionamiento de éste.

Tabla 1. Comparación entre metodologías ágiles y tradicionales.

Metodología Ágil	Metodología Tradicional
Pocos artefactos y roles	Más artefactos y más roles
No existe un contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo (además in-situ)	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (< 10 integrantes) y trabajando en el mismo sitio	Grupos grandes
Menos énfasis en la arquitectura	La arquitectura es esencial

Con lo anterior planteado, es conveniente para el equipo de desarrollo la selección de una metodología ágil, debido a que el equipo es pequeño, presenta poco tiempo de desarrollo y alto riesgo. La metodología ágil que se utilizará será XP ya que la misma se ajusta las necesidades de la UEB y al desarrollo del software en cuestión.

1.5 Tecnologías y herramientas.

1.5.1 Lenguajes del lado del servidor.

PHP

Es un lenguaje de programación utilizado para la creación de aplicaciones web. PHP es un acrónimo que significa "*PHP HyperText Pre-processor*", (inicialmente se llamó *Personal Home Page*). PHP es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. Surgió en 1995, desarrollado por PHP Group. PHP no necesita ser compilado para ejecutarse. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas (Heredia, 2005).

1.5.2 Lenguajes del lado del cliente.

HTML

Desde el surgimiento de internet se han publicado sitios web gracias al lenguaje HTML. Es un lenguaje estático para el desarrollo de sitios web (acrónimo en inglés de *HyperText Markup Language*, en español Lenguaje de Marcas de Hipertexto). Desarrollado por el *World Wide Web Consortium (W3C)*. Este lenguaje se caracteriza por ser un lenguaje de fácil aprendizaje, estático, lo admiten todos los exploradores, pero la interpretación de cada navegador puede ser diferente (Webmaster, 2009).

Hojas de Estilo en Cascada (CSS)

CSS (*Cascade Style Sheets*), proporcionan una manera muy eficiente de aplicar estilos y formato a las páginas web y a prácticamente cualquiera de sus elementos. Además, permiten definir nuevas clases de estilos y subclases de etiquetas HTML o de otras clases. Estas opciones la convierten en una herramienta muy poderosa y flexible (Ruiz, 2006).

CSS, es una tecnología que permite crear páginas web de una manera más exacta, ya que trata de dar la separación definitiva de la lógica (estructura) y el físico (presentación) del documento. El estilo lógico se refiere a la lógica del documento: cabeceras, párrafos, no se preocupa de la apariencia final, sino de la estructura del documento. Por el contrario, el estilo físico no se preocupa de la estructura del documento, sino por la apariencia final: párrafos con un cierto tipo de letra, tablas con un determinado color de fondo, etc.

JavaScript.

Este es un lenguaje interpretado, no requiere compilación. Fue creado por Brendan Eich en la empresa *Netscape Communications*. Utilizado principalmente en páginas web. Es similar a Java, aunque no es un lenguaje orientado a objetos, el mismo no dispone de herencias. La mayoría de los navegadores en sus últimas versiones interpretan código Java Script. El código JavaScript puede ser integrado dentro de las páginas web. Los script tienen capacidades limitadas, por razones de seguridad, el código visible por cualquier usuario y el código deben descargarse completamente (Álvarez, 2001).

1.5.3 Framework.

El concepto framework se emplea en muchos ámbitos del desarrollo de sistemas software, no sólo en el ámbito de aplicaciones Web. En general, con el término framework, se está refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta (Gutiérrez, 2006)

Algunas de las características presentes en casi todos son: abstracción de URL y sesiones (no es necesario manipular directamente las URLs ni las sesiones, el framework se encarga de esto); acceso datos (incluyen las herramientas e interfaces necesarias para integrarse con herramientas de acceso a datos, en bases de datos, etc.); controladores (la mayoría de los frameworks que existen implementa una serie de controladores para gestionar eventos) (Gutiérrez, 2006).

Actualmente existen en el mundo una gran variedad de frameworks para PHP y JavaScript. Algunos de estos son Zend Framework, CodeIgniter, Symfony, etc. El programador no necesita plantearse una estructura global de la aplicación, sino que el framework le proporciona un esqueleto que tiene que adecuar a sus necesidades. Facilita la colaboración. Cualquiera que haya tenido que interpretar el código fuente de otro programador sabrá lo difícil que es entenderlo y modificarlo; por tanto, todo lo que sea definir y estandarizar va a ahorrar tiempo y trabajo a los desarrollos colaborativos. Es más fácil encontrar herramientas (utilidades, bibliotecas) adaptadas al framework concreto para facilitar el desarrollo.

Ejemplos de Framework.

CodeIgniter

Framework increíblemente ligero que usa el diseño MVC y contiene una muy buena estructuración de sus librerías. Usarlo es tan fácil como descomprimir su contenido en una carpeta en el servidor, sólo con eso ya funcionará. Es uno de los frameworks más

indicados para las personas que quieran iniciarse en el uso de los mismos (Álvarez, 2009).

Zend Framework.

Zend Framework es un framework PHP orientado a objetos 100%, con una fuerte implementación del patrón MVC, una abstracción de datos muy simple de usar y un rendimiento muy estable, siendo uno de los frameworks más robustos de la actualidad. Además, al ser un framework de código abierto, evitaremos posibles problemas con licencias como nos podría ocurrir con algún framework restrictivo o de pago (Gutiérrez, 2006).

Symfony.

Symfony2 es la versión más reciente de Symfony, el popular framework para desarrollar aplicaciones PHP. Se anunció por primera vez a principios de 2009 y supone un cambio radical tanto en arquitectura interna como en filosofía de trabajo respecto a sus versiones anteriores. Symfony2 ha sido ideado para exprimir al límite todas las nuevas características de PHP 5.3 y por eso es uno de los frameworks PHP con mejor rendimiento. Su arquitectura interna está completamente desacoplada, lo que permite reemplazar o eliminar fácilmente aquellas partes que no encajan en tu proyecto. Symfony2 también es el framework que más ideas incorpora del resto de frameworks, incluso de aquellos que no están programados con PHP. (Eguiluz, 2011)

Selección del Framework.

En Symfony, el modelo contiene la información necesaria para poder generar de forma automática el código de las operaciones de insertar, obtener, modificar y borrar (CRUD), de forma que se simplifica el desarrollo inicial de la parte de administración de las aplicaciones. Es independiente del gestor de bases de datos, gracias a su capa de abstracción y es muy adecuado para metodologías ágiles de desarrollo como XP. Por todas las funcionalidades que brinda se seleccionó el framework Symfony2 para el desarrollo de la aplicación web en cuestión.

1.5.4 Sistemas gestores de Bases de Datos

Los sistemas de gestión de base de datos (SGBD); (en inglés: *Database management system*, abreviado *DBMS*) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Su propósito general es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante, para un buen manejo de datos (Sandoval Méndez, 2009).

El objetivo fundamental de un SGBD consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, o sea, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado.

En el mundo existen varios tipos de SGBD clasificados en licenciamiento comercial como Oracle, SQL SERVER, etc. y los de licenciamiento libre entre los que se encuentran MYSQL, PostgreSQL, FireBird, SQLite, etc.

MySQL

MySQL es un gestor de base de datos sencillo de usar e increíblemente rápido. También es uno de los motores de base de datos más usados en Internet (Riveros, 2008)

A continuación se muestran las características de MySQL por las que fue seleccionado como sistema gestor de bases de datos para la implementación de la aplicación web:

- Probado con un amplio rango de compiladores diferentes.
- Funciona en diferentes plataformas.
- Un sistema de reserva de memoria muy rápido basado en threads.
- Joins muy rápidos usando un multi-join de un paso optimizado.

- Tablas hash en memoria, que son usadas como tablas temporales.
- Las funciones SQL están implementadas usando una librería altamente optimizada y deben ser tan rápidas como sea posible. Normalmente no hay reserva de memoria tras toda la inicialización para consultas.
- El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado (linkado) en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible.
- Soporte completo para las cláusulas SQL GROUP BY y ORDER BY. Soporte de funciones de agrupación (COUNT(), COUNT(DISTINCT ...), AVG(), STD(), SUM(), MAX(), MIN(), y GROUP_CONCAT()).
- Soporte para LEFT OUTER JOIN y RIGHT OUTER JOIN cumpliendo estándares de sintaxis SQL y ODBC.
- Soporte para alias en tablas y columnas como lo requiere el estándar SQL.
- DELETE, INSERT, REPLACE, y UPDATE devuelven el número de filas que han cambiado (han sido afectadas). Es posible devolver el número de filas que serían afectadas usando un flag al conectar con el servidor.

PostgreSQL

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (*ORDBMS*) que ha sido desarrollado de varias formas desde 1977. Comenzó como un proyecto denominado Ingres en la Universidad Berkeley de California (Riveros, 2008)

PostgreSQL es uno de los gestores de bases de datos de código abierto más avanzado hoy en día; soporta casi toda la sintaxis SQL y ofrece muchas características modernas (Group, T.P.G.D., 2009):

- Consultas complejas.
- Integridad transaccional.
- Control de concurrencia multi-versión (MVCC, *Multi Version Concurrency Control*).

Además, PostgreSQL puede ser ampliado por el usuario de muchas formas a diferencia de MySQL y otros gestores de licenciamiento libre, por ejemplo, mediante la adición de nuevos (Johnson, 2007b): tipos de datos; funciones; operadores.

El lenguaje PL/pgSQL es uno de los más utilizados ya que en él es posible realizar cálculos, manejo de cadenas y consultas dentro del servidor de la base de datos. Además cuenta con las siguientes ventajas (Riveros, 2008):

- Estabilidad y confiabilidad legendaria: puesto que nunca ha presentado caídas en varios años de operación de alta actividad, es decir que funciona correctamente.
- Extensible: el código fuente está disponible para todos sin costo.
- Multiplataforma: puede ser utilizado en cualquier plataforma.
- Diseñado para ambientes de alto volumen: usa una estrategia de almacenamiento de filas llamada MVCC para conseguir una mejor respuesta en ambientes de grandes volúmenes.
- Herramientas gráficas de diseño y administración de bases de datos: existen varias herramientas gráficas de alta calidad para administrar las bases de datos y para hacer diseño de bases de datos.

Selección del gestor de base datos.

Los gestores de base de datos antes mencionados, presentan características que lo hacen muy buenos para el desarrollo de la solución propuesta, pero entre los

requerimientos del cliente se propone usar PostgreSQL ya que es libre, es uno de los gestores de base de datos más robusto y potente que existe, el mismo funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema. También en nuestro país existe una comunidad Postgres y una gran bibliografía del mismo.

1.5.5 Entorno de Desarrollo Integrado

NetBeans IDE

El NetBeans IDE es un producto libre y gratuito sin restricciones de uso y de amplias comodidades para el programador. Para la realización de este proyecto se utilizó el IDE 7.2.1, una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero funciona perfectamente para la escritura de PHP ya que presenta un buen completamiento de código y reconoce todas las funciones y estructuras de este lenguaje.

Zend Studio 8.0

Se trata de un programa orientada a desarrollar aplicaciones web utilizando PHP. El programa, sirve de editor de texto para páginas PHP, también proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código, aunque para estos habrá que disponer de la parte del servidor. El programa está escrito en Java por lo que se puede usar en Windows, Linux y MacOS (Álvarez, 2005).

Selección del Entorno de Desarrollo Integrado

Tanto el NetBeans 7.2.1 como Zend Studio 8.0 son dos entornos de desarrollo integrados muy buenos para lograr una productividad en proyectos que se realicen con PHP, debido a que el equipo de desarrollo tiene experiencia del uso del IDE NetBeans y Zend Studio no es libre, se toma a NetBeans 7.2.1 para el desarrollo de la solución.

1.5.6 Herramientas Case

Visual Paradigm.

Es un software privativo gratuito para modelado en UML. Esta herramienta tiene unas características gráficas muy cómodas, que facilitan la realización de los diagramas de modelado que sigue el estándar de UML, los mismo son: Diagramas de clase, Casos de Uso, Comunicación, Secuencia, Estado, Actividad, Componentes, etc.

Visual Paradigm ofrece un entorno de creación de diagramas para UML 2.0. Diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad. Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación. Capacidades de ingeniería directa e inversa. Modelo y código que permanece sincronizado en todo el ciclo de desarrollo. Disponibilidad de múltiples versiones, para cada necesidad. Disponibilidad en múltiples plataformas.

Características:

- Producto de calidad.
- Soporta aplicaciones web.
- Las imágenes y reportes generados, no son de muy buena calidad.
- Varios idiomas.
- Generación de código para Java y exportación como HTML.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.

Rational Rose Enterprise

Existen herramientas CASE de trabajo visuales como el Analise, el Designe, el Rational Rose, que permiten realizar el modelado del desarrollo de los proyectos, en la actualidad la mejor y más utilizada en el mercado mundial es Rational Rose y es la que se utiliza en la modelación de este proyecto. "Rational Rose cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los

usuarios y certificación de las distintas fases. Es la herramienta CASE que comercializan los desarrolladores de UML y que soporta de forma completa la especificación del UML.

Rose es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros de equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Una de las grandes ventajas de Rose es que utiliza la notación estándar en la arquitectura de software(UML), la cual permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común, además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto.

Selección de la herramienta case

Visual Paradigm es la herramienta case que se utilizará para el desarrollo de la solución, ya que el equipo de trabajo dispone de gran experiencia con la herramienta, además de ser una herramienta libre.

Conclusiones.

Los conceptos expuestos a lo largo de este capítulo son la fundamentación teórica que garantiza el cumplimiento del objetivo general de este trabajo: Desarrollar el módulo para la gestión de los recursos humanos del IPHSG.

Se definió la para la solución del sistema quedando conformada de la siguiente manera: (XP) como metodología de desarrollo. Como lenguajes: PHP 5 del lado del servidor, HTML5, CCS y JavaScript por parte del cliente. Symfony2 como framework de desarrollo. PostgreSQL como gestor de base de datos. Apache como servidor web. NetBeans IDE 7.2 como entorno de desarrollo. Visual Paradigm para el modelado de prototipos y de la base de datos.

Capítulo II: Planificación y diseño del software para mejorar la gestión de la información de los recursos humanos en la UEB Investigación y Proyectos Hidráulicos de Sancti Spiritus.

Después de haber analizado el estado del arte y elegidas las herramientas y metodología a utilizar, estamos en condiciones de plantear una solución a nuestro problema utilizando la metodología de desarrollo XP. En este capítulo se ejecutan las fases de Planificación y Diseño propias de la metodología propuesta para el desarrollo del sistema así como los artefactos generados en dicha fase.

2.1 Propuesta de solución.

En el presente trabajo se propone la implementación de una aplicación web para la Gestión de los Recursos Humanos con su implementación, evaluación y mejora pretendemos alcanzar mejores resultados en la gestión del capital humano y por ende contribuir a elevar significativamente la motivación y el compromiso del capital humano de la organización. Esta aspiración está en correspondencia con las aspiraciones del país y sector de contar con un recurso humano más comprometido y competente.

Es un sistema que integra el conjunto de políticas, normativas, objetivos, metas, responsabilidades, funciones, procedimientos, herramientas y técnicas que permiten llevar a cabo los procesos de gestión de recursos humanos, centrado en la estrategia y en las competencias laborales, que tiene como objetivo la integración de la Gestión de Recursos Humanos para el mejoramiento continuo y alcanzar un desempeño laboral superior.

2.2 Fase Planificación.

XP plantea la planificación como un permanente diálogo entre la parte empresarial y técnica del proyecto, en la que los primeros decidirán:

Alcance: ¿Qué es lo que el software debe de resolver para que genere valor?

Prioridad: ¿Qué debe ser hecho en primer lugar?

Composición de las versiones: ¿Cuánto es necesario hacer para saber si el negocio va mejor con software que sin él? En cuanto el software aporte algo al negocio se deben tener lista las primeras versiones.

Fechas de versiones: ¿Cuáles son las fechas en la presencia del software en cuanto a sus primeras versiones?

Consecuencias: Informar sobre las consecuencias de la toma de decisiones por parte del negocio.

Procesos: ¿Cómo se organiza el trabajo y el equipo?

Programación detallada: Dentro de una versión ¿Qué problemas se resolverán primero?

2.2.1 Historias de usuario.

En XP la gestión de requisitos es extremadamente simple, el cliente escribe y prioriza las historias de usuario que expresan las necesidades del sistema. Los programadores estiman el esfuerzo asociado y las dependencias entre ellas. Para planificar el trabajo desde el punto de vista técnico las historias de usuario son divididas en tareas para las cuales también se realiza una estimación. Teniendo en cuenta el esfuerzo asociado a las historias de usuario y las prioridades del cliente se define una versión que sea de valor para el cliente y que tenga una duración de unos pocos meses.

Las historias de usuario tienen el mismo propósito que los casos de uso, pero no son lo mismo. Las escriben los propios clientes, tal y como ven ellos las necesidades del sistema. Por tanto serán descripciones cortas y escritas en el lenguaje del usuario, sin terminología técnica.

Conducen el proceso de creación de los test de aceptación. Uno o más de uno de estos test se utilizarán para verificar que las historias de usuario han sido implementadas correctamente.

La principal diferencia es el nivel de detalle. Las historias de usuario solamente proporcionaran los detalles sobre la estimación del riesgo y cuánto tiempo conllevará la implementación de dicha historia de usuario.

Las historias de usuario deben tener el detalle mínimo como para que los programadores puedan realizar una estimación poco riesgosa del tiempo que llevará su desarrollo. Cuando comience la implementación, los desarrolladores dialogarán directamente con el cliente para obtener todos los detalles necesarios. También conducirán el proceso de creación de los test de aceptación.

Durante este proceso se identificaron 33 historias de usuarios. A continuación se muestran algunas de las historias de usuarios. Para definir las historias de usuario utilizamos la siguiente planilla, que contiene todos los datos necesarios para desarrollar la funcionalidad descrita.

La misma cuenta con el número de la historia de usuario, los usuarios que tienen acceso a la misma, el nombre, la prioridad que tiene esta historia de usuario en el negocio y su riesgo, los puntos estimados que son los días que demorará la realización de dicha historia de usuario, cuenta con la iteración a la cual ha sido asignada, aparecen los programadores responsables del desarrollo de la historia de usuario, además de una descripción la cual ejemplifica las funcionalidades de cada historia de usuario, las mismas también cuentan con una observación donde aparecen todos los datos que contiene una historia de usuario.

Tabla 1: Representación de la Historia de Usuario Nro. 1.

Historia de Usuario	
Número: 1	
Nombre Historia de Usuario: Gestionar trabajador	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 4
Puntos Estimados: 4	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de un trabajador.	
Observaciones: Trabajador posee atributos como: CI, nombre, primer apellido, segundo apellido, número tarjeta, fecha de nacimiento, nombre de la madre, nombre del padre, sexo, dirección, color de piel, estatura, peso, si tiene salida al exterior, fecha baja.	

Tabla 2: Representación de la Historia de Usuario Nro. 8.

Historia de Usuario	
Número: 8	
Nombre Historia de Usuario: Gestionar entidad	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de una entidad.	
Observaciones: Entidad posee atributos como: nombre.	

Tabla 3: Representación de la Historia de Usuario Nro. 2.

Historia de Usuario	
Número: 2	
Nombre Historia de Usuario: Gestionar área de trabajo	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de un área de trabajo.	
Observaciones: Área de trabajo posee atributos como: nombre e identificador.	

Tabla 4: Representación de la Historia de Usuario Nro. 14.

Historia de Usuario	
Número: 14	
Nombre Historia de Usuario: Gestionar organización	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de una organización.	
Observaciones: La organización posee atributos como: el nombre.	

Una vez definidas las historias de usuarios, con los grados de complejidad de las mismas daremos continuación al plan de entrega de las mismas.

2.2.2 Plan de entregas

Las historias de usuario servirán para crear el plan estimado de entregas. Se convocará una reunión para crear el plan de entregas. El plan de entregas se usará para crear los planes de iteración para cada iteración. Con cada historia de usuario previamente evaluada en tiempo de desarrollo ideal, el cliente las agrupará en orden de importancia.

Una semana ideal es cuánto tiempo costaría implementar dicha historia si no tenemos nada más que hacer, incluyendo la parte de test correspondiente.

De esta forma se puede trazar el plan de entregas en función de estos dos parámetros: tiempo de desarrollo ideal y grado de importancia para el cliente. Las iteraciones individuales son planificadas en detalle justo antes de que comience cada iteración.

Tabla: 5 Estimación de esfuerzos por Historia de Usuario.

Bundle	Historias de Usuario
Recursos Humanos	<ol style="list-style-type: none"> 1. Gestionar trabajadores. 2. Gestionar áreas de trabajo. 3. Gestionar tipos de misión. 4. Gestionar causas de baja de trabajo. 5. Gestionar idiomas. 6. Gestionar grupos salariales. 7. Gestionar grupos sanguíneos. 8. Gestionar entidades. 9. Gestionar ausencias. 10. Gestionar provincias. 11. Gestionar ubicaciones de defensa. 12. Gestionar especialidades. 13. Gestionar tipo de contratos. 14. Gestionar organizaciones. 15. Gestionar cargos.

2.2.3 Plan de iteraciones.

El plan de iteración consiste en seleccionar las historias de usuario que, según el plan de entregas, corresponderían a esta iteración. También se eligen qué pruebas de aceptación fallidas se corregirán.

Cada historia de usuario se transformará en tareas de desarrollo. Cada tarea de desarrollo corresponderá a un periodo ideal de uno a tres días de desarrollo.

Es necesario mantener vigiladas la velocidad del proyecto y el movimiento de historias de usuario. Puede ser necesario volver a calcular las historias de usuario y negociar el plan de entrega cada de tres a cinco iteraciones. Como estaremos siempre implementando las historias de usuario más importantes para el cliente, estaremos haciendo lo máximo posible por nuestro cliente y la dirección.

1ra Iteración.

En esta iteración se implementarán todas las historias de usuarios que tienen prioridad alta es decir las de riesgo 4 y 5, de esta forma se van creando las funcionalidades principales del sistema que dan soporte a la implementación de las demás funcionalidades. Además se tendrá la primera versión de prueba, la cual será mostrada al cliente con el objetivo de obtener una retroalimentación para el grupo de trabajo.

2da Iteración.

El objetivo de esta iteración es la implementación de las restantes funcionalidades con prioridad media las cuales presentan riesgo 3. La versión de pruebas de esta iteración junto con la pasada será mostrada al cliente con el objetivo de obtener su valoración y los cambios en el caso de que se encuentren.

3ra Iteración.

En esta iteración serán implementadas las funcionalidades de prioridad baja con riesgo 2. Como resultado de esta iteración se tendrá la versión 1.0 del producto final, adicionando lo concerniente a la interfaz Web. A partir de este momento el sistema será puesto a prueba por un período de tiempo para evaluar el desempeño del mismo.

2.2.4 Plan de duración de las iteraciones

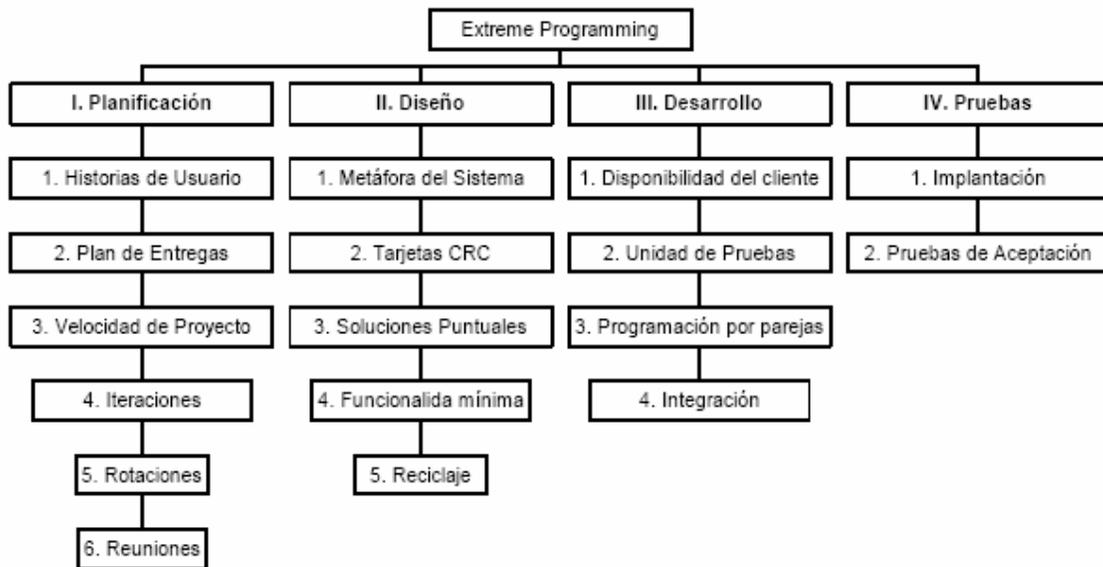
Siguiendo el desarrollo de la metodología XP se crea el plan de duración de las iteraciones. En este plan se especifica más detalladamente el orden de desarrollo de las historias de usuarios dentro de cada iteración así como la estimación completa de dicha iteración.

Tabla 6: Plan de duración de las Iteraciones.

Iteraciones.	Orden de las Historias de Usuarios a implementar.	Duración total de las iteraciones.
1ra Iteración.	<ol style="list-style-type: none">1. Gestionar trabajadores.2. Gestionar áreas de trabajo.3. Gestionar idiomas.4. Gestionar grupos salariales.5. Gestionar entidades.6. Gestionar ausencias.7. Gestionar especialidades8. Gestionar organizaciones.9. Gestionar cargos.	11 Semanas
2da Iteración.	<ol style="list-style-type: none">1. Gestionar tipos de misión.2. Gestionar ubicaciones de defensa.3. Gestionar tipo de contratos.	2 Semanas
3ra Iteración.	<ol style="list-style-type: none">4. Gestionar causas de baja de trabajo.5. Gestionar grupos sanguíneos.6. Gestionar provincias.	2 Semanas

Una vez realizado los artefactos pertenecientes a la fase de planificación a partir de estos se le dará comienzo a la fase de diseño la cual es de gran importancia para el desarrollo del producto.

Trabajando con Extreme Programming



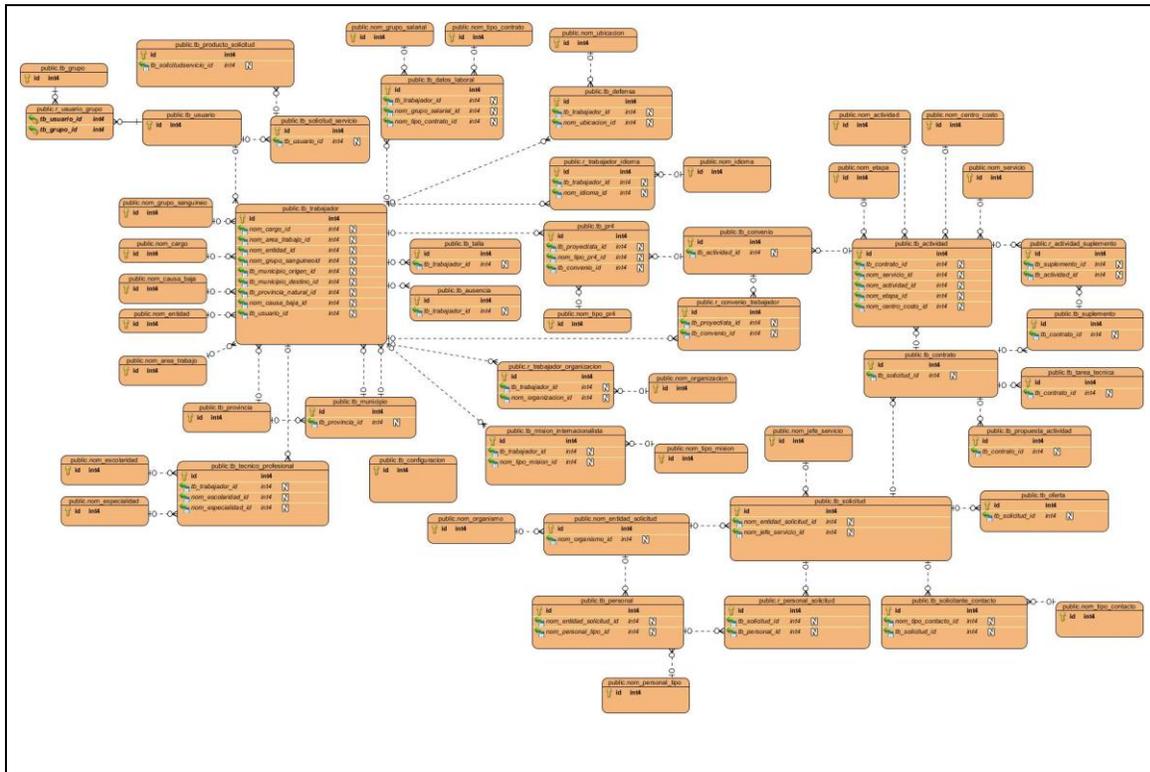
2.3 Fase de Diseño.

El diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar. Los diseños deben de ser sencillos. En esta fase se realiza el diagrama de clases persistentes y las tarjetas CRC (Cargo o Clase, Responsabilidad y Colaboración) para una mayor concentración y apariencia del desarrollo al programador (Fases-Programacion Extrema, 2012).

2.3.1 Modelo Entidad Relación

Estas son las tablas de la base de datos con sus relaciones. La más importante es la tabla trabajador pues es en ella donde se realizan la mayoría de las funcionalidades y la que más peso tiene en el modelo relacional.

Fig. 1. Modelo Entidad Relación



Tarjetas CRC (Cargo o clase, Responsabilidad y Colaboración)

Para poder diseñar el sistema como un equipo se debe cumplir con tres principios: Cargo o Clase, Responsabilidad y Colaboración (CRC). Las tarjetas CRC permiten desprenderse del método de trabajo basado en procedimientos y trabajar con una metodología basada en objetos. Las tarjetas CRC permiten que el equipo completo contribuya en la tarea del diseño.

Una tarjeta CRC representa un objeto. El nombre de la clase se coloca a modo de título en la tarjeta, las responsabilidades se colocan a la izquierda, y las clases que se implican en cada responsabilidad a la derecha, en la misma línea que su requerimiento correspondiente.

Tabla 7: Tarjeta CRC Clase tb_trabajador

Clase: tb_trabajador	
Responsabilidad	Colaboración
Insertar nuevos trabajadores.	nom_area_trabajo nom_cargo r_trabajador_idioma tb_datos_laboral tb_talla tb_defensa tb_tecnico_profesional nom_causa_baja tb_ausencia r_trabajador_organizacion tb_mision_internacionalista tb_municipio tb_provincia nom_grupo_sanguineo nom_entidad tb_usuario tb_pr4 r_convenio_trabajador

Tabla 8: Tarjeta CRC Clase nom_entidad

Clase: nom_entidad	
Responsabilidad	Colaboración
Insertar nueva entidad.	tb_trabajador

Tabla 9: Tarjeta CRC Clase nom_organizacion

Clase: nom_organizacion	
Responsabilidad	Colaboración
Insertar nueva organización.	r_trabajador_organizacion

2.4 Conclusiones

Durante el desarrollo de este capítulo se arribaron a las siguientes conclusiones parciales:

1. Se realizó una descripción de los procesos vinculados al campo de acción donde se refleja la complejidad del negocio.
2. En la fase de planificación se generaron los siguientes artefactos:
 - Las historias de usuario.

- El plan de entrega.
 - Plan de iteraciones.
 - Plan de duración de las iteraciones.
3. En la fase de diseño se obtuvieron los artefactos siguientes:
- El diagrama de clases persistentes.
 - Las Tarjetas CRC.

Capítulo III: Implementación y Prueba del software para mejorar la gestión de la información de los recursos humanos en la UEB Investigación y Proyectos Hidráulicos de Sancti Spíritus.

La implementación dentro de la metodología XP debe realizarse de forma iterativa, esta característica trae consigo que después del desarrollo de cada iteración surja un producto funcional que debe ser mostrado al cliente y previamente probado para incrementar la visión de los desarrolladores y clientes de posibles cambios y soluciones. En este capítulo se detallan las tres iteraciones llevadas a cabo durante la construcción del sistema exponiéndose las tareas generadas por las historias de usuarios, así como las pruebas de aceptación efectuadas sobre el sistema.

3.1 Fase de Desarrollo.

En esta fase XP plantea la implementación de cada una de las historias de usuarios. En un comienzo de la fase se hace un chequeo de cada una de las HU junto con el plan de iteraciones y se modifica en caso de ser necesario. Como parte de este plan se crean tareas para ayudar a organizar la implementación exitosa de las HU. Estas tareas pueden escribirse en un lenguaje técnico, al contrario de las historias de usuario que son escritas en el lenguaje del cliente (Fases-Programación Extrema, 2012).

Teniendo en cuenta la planificación realizada anteriormente, se llevaron a cabo tres iteraciones de desarrollo sobre el sistema, obteniéndose como finalidad un producto con todas las restricciones y características deseadas para ser utilizado. A continuación se detallan las tareas de programación por iteraciones.

3.1.1 Tareas de programación por iteraciones.

Cada HU como funcionalidad de la aplicación está compuesta por una o varias tareas de programación, éstas no son más que pasos lógicos a seguir por el programador para realizar la implementación de una HU.

En la **iteración 1** se implementaron las historias de usuario de mayor prioridad, que son sin dudas el esqueleto del negocio anteriormente descrito, con el fin de obtener una

versión del producto con algunas de las funcionalidades críticas para ser mostrado al cliente y tomar nuevas iniciativas de forma rápida.

Tabla 10: Tiempo real de implementación por Bundle (Iteración 1).

Bundle	Historias de Usuario	Tiempo de Implementación (Semana)	
		Estimación	Real
Recursos Humanos	1. Gestionar trabajadores.	1	0.9
	2. Gestionar áreas de trabajo	0.7	0.7
	3. Gestionar idiomas.	0.6	0.6
	4. Gestionar grupos salariales.	0.6	0.6
	5. Gestionar entidades.	0.4	0.4
	6. Gestionar ausencias.	0.4	0.4
	7. Gestionar organizaciones.	0.4	0.4
	8. Gestionar cargo	0.4	0.4

Tabla 11: Tarea de programación por Historias de Usuario. (Iteración 1)

Historia de Usuario.	Tarea de programación por historias de usuario.
Gestionar trabajadores.	1 - Mostrar interfaz "Gestionar trabajadores". 2 - Adicionar trabajadores. 3 - Actualizar trabajadores. 4 - Eliminar trabajadores. 5 - Listar trabajadores.
Gestionar áreas de trabajo	1 - Mostrar interfaz "Gestionar áreas de trabajo". 2 - Adicionar áreas de trabajo. 3 - Actualizar áreas de trabajo. 4 - Eliminar áreas de trabajo. 5 - Listar áreas de trabajo.
Gestionar idiomas.	1 - Mostrar interfaz "Gestionar idiomas". 2 - Adicionar idiomas. 3 - Actualizar idiomas. 4 - Eliminar idiomas. 5 - Listar idiomas.
Gestionar grupos salariales.	1 - Mostrar interfaz "Gestionar grupos salariales". 2 - Adicionar grupos salariales. 3 - Actualizar grupos salariales. 4 - Eliminar grupos salariales. 5 - Listar grupos salariales.
Gestionar entidades.	1 - Mostrar interfaz "Gestionar entidades". 2 - Adicionar entidades. 3 - Actualizar entidades. 4 - Eliminar entidades. 5 - Listar entidades.
Gestionar ausencias.	1 - Mostrar interfaz "Gestionar ausencias". 2 - Adicionar ausencias. 3 - Actualizar ausencias. 4 - Eliminar ausencias. 5 - Listar ausencias.
Gestionar organizaciones.	1 - Mostrar interfaz "Gestionar organizaciones". 2 - Adicionar organizaciones. 3 - Actualizar organizaciones. 4 - Eliminar organizaciones. 5 - Listar organizaciones.
Gestionar cargo	1 - Mostrar interfaz "Gestionar cargo". 2 - Adicionar cargo. 3 - Actualizar cargo. 4 - Eliminar cargo. 5 - Listar cargo.

En la **iteración 2** Las historias de usuarios que se implementan son las de prioridad media.

Tabla 12: Tiempo real de implementación por Bundle (Iteración 2).

Bundle	Historias de Usuario	Tiempo de Implementación (Semana)	
		Estimación	Real
Recursos Humanos	1. Gestionar tipos de misión.	0.4	0.4
	2. Gestionar ubicaciones de defensa.	0.4	0.4
	3. Gestionar tipo de contratos.	0.4	0.4

Tabla 13: Tarea de programación por historias de usuario. (Iteración 2)

Historia de Usuario.	Tarea de programación por historias de usuario.
Gestionar tipos de misión.	1 - Mostrar interfaz "Gestionar tipos de misión". 2 - Adicionar tipos de misión. 3 - Actualizar tipos de misión. 4 - Eliminar tipos de misión. 5 - Listar tipos de misión.
Gestionar ubicaciones de defensa.	1 - Mostrar interfaz "Gestionar ubicaciones de defensa". 2 - Adicionar ubicaciones de defensa. 3 - Actualizar ubicaciones de defensa. 4 - Eliminar ubicaciones de defensa. 5 - Listar ubicaciones de defensa.
Gestionar tipo de contratos.	1 - Mostrar interfaz "Gestionar tipo de contratos". 2 - Adicionar tipo de contratos. 3 - Actualizar tipo de contratos. 4 - Eliminar tipo de contratos 5 - Listar tipo de contratos.

En la **iteración 3** se implementan las historias de usuario con baja prioridad. Estas funcionalidades reflejan en la aplicación un ambiente más amigable y cómodo para el cliente.

Tabla 14: Tiempo real de implementación por Bundle (Iteración 3).

Bundle	Historias de Usuario	Tiempo de Implementación (Semana)	
		Estimación	Real
Recursos Humanos	1. Gestionar causas de baja de trabajo.	0.1	0.1
	2. Gestionar grupos sanguíneos.	0.4	0.1
	3. Gestionar provincias.	0.3	0.3

Tabla 15: Tarea de programación por historias de usuario. (Iteración 3)

Historia de Usuario.	Tarea de programación por historias de usuario.
Gestionar causas de baja de trabajo.	<ul style="list-style-type: none"> 1 - Mostrar interfaz "Gestionar causas de baja de trabajo". 2 - Adicionar causas de baja de trabajo. 3 - Actualizar causas de baja de trabajo. 4 - Eliminar causas de baja de trabajo. 5 - Listar causas de baja de trabajo.
Gestionar grupos sanguíneos.	<ul style="list-style-type: none"> 1 - Mostrar interfaz "Gestionar grupos sanguíneos". 2 - Adicionar grupos sanguíneos. 3 - Actualizar grupos sanguíneos. 4 - Eliminar grupos sanguíneos. 5 - Listar grupos sanguíneos.
Gestionar provincias.	<ul style="list-style-type: none"> 1 - Mostrar interfaz "Gestionar provincias". 2 - Adicionar provincias. 3 - Actualizar provincias. 4 - Eliminar provincias. 5 - Listar provincias.

Una vez concluido el estudio del primer capítulo y realizada las fases de planificación y diseño correspondientes al siguiente capítulo, así como la implementación del sistema estamos en condiciones de realizar las pruebas de aceptación del mismo.

3.2 Fase Prueba

Uno de los pilares fundamentales de XP es el proceso de pruebas, el cual anima a los desarrolladores a probar constantemente tanto como sea posible. Mediante esta filosofía se reduce el número de errores no detectados así como el tiempo entre la introducción de éste en el sistema y su detección. Todo esto contribuye a elevar la calidad de los productos desarrollados y a la seguridad de los programadores a la hora de introducir cambios o modificaciones (Allende, 2006; Crispin & House, 2002).

La metodología XP divide las pruebas en dos grupos: pruebas unitarias, desarrolladas por los programadores, encargadas de verificar el código de forma automática y las pruebas de aceptación, destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además de comprobar que dicha funcionalidad sea la esperada por el cliente (Allende, 2006).

3.2.1 Las pruebas de aceptación

El objetivo de estas pruebas es verificar los requisitos, por este motivo, los propios requisitos del sistema son la principal fuente de información a la hora de construir las pruebas de aceptación.

Las pruebas de aceptación son creadas a partir de las historias de usuario. Durante una iteración la historia de usuario seleccionada en la planificación de iteraciones se convertirá en una prueba de aceptación. El cliente o usuario especifica los aspectos a testear cuando una historia de usuario ha sido correctamente implementada.

Una historia de usuario puede tener más de una prueba de aceptación, tantas como sean necesarias para garantizar su correcto funcionamiento y no se considera completa hasta que no supera sus pruebas de aceptación. Esto significa que debe desarrollarse un nuevo test de aceptación para cada iteración o se considerará que el equipo de desarrollo no realiza ningún progreso.

Una prueba de aceptación es como una caja negra. Cada una de ellas representa una salida esperada del sistema. Es responsabilidad del cliente verificar la corrección de las pruebas de aceptación y tomar decisiones a cerca de las mismas.

La garantía de calidad es una parte esencial en el proceso de XP. La realización de este tipo de pruebas y la publicación de los resultados debe ser lo más rápido posibles, para que los desarrolladores puedan realizar con la mayor rapidez los cambios que sean necesarios.

Tabla: 16 Pruebas De Aceptación del escenario Adicionar Provincia.

Escenario	Descripción	Variable 1	Respuesta del Sistema
Adicionar Provincia.	Funcionalidad mediante la cual se adicionan provincias al sistema.	V	Adiciona una nueva provincia en el sistema.
		Nombre	
		Dato	
		Cienfuegos	Muestra un mensaje de error. "Este valor no es válido"
		I	
		Nombre	
Dato	23sd		
23sd			

		I	Muestra un mensaje.
		Nombre	“Por favor, rellene este campo”
		Dato	
		Null	

Tabla: 17 Pruebas De Aceptación del escenario Adicionar Provincia.

Escenario	Descripción	Variable 1	Variable 2	Variable 3	Respuesta del Sistema
Adicionar Ausencia.	Funcionalidad mediante la cual se adicionan ausencias al sistema.	V	V	V	Adiciona una nueva ausencia en el sistema.
		causa	días laborables	promedio trabajadores	
		Dato	Dato	Dato	
		certificado	20	60	
		I	V	V	Muestra un mensaje de error.
		causa	días laborables	promedio trabajadores	“Este valor no es válido”
		Dato	Dato	Dato	
		5	veinte	cuarenta	

		V causa	I días laborables	V promedio trabajadores	Muestra un mensaje. "Por favor, rellene este campo"
		Dato certificado	Dato Null	Dato 40	

3.3 Conclusiones

Durante la elaboración de este capítulo se arribaron a las siguientes conclusiones parciales:

1. En la fase de implementación se presentaron las Tareas de programación por iteraciones, donde se encuentra las tareas de programación por usuarios y el tiempo real de implementación por Bundle.
2. En la fase de prueba se conformaron los diseños de caso de prueba, realizando las pruebas de aceptación y quedando de esta manera la validación del sistema.

Conclusiones generales

1. Se realizó una revisión bibliográfica sobre las distintas herramientas y tecnologías necesarias para el desarrollo de la aplicación.
2. Se elaboraron las historias de usuario, plan de entrega y el plan de iteraciones correspondientes a la etapa de planificación.
3. Se realizaron las tarjetas CRC y el diagrama de las clases persistentes correspondientes a la etapa de Diseño.
4. Se implementó un sistema web el cual gestiona los recursos humanos de la UEB de Investigación y Proyectos Hidráulicos de Sancti Spíritus.
5. Se realizaron los diseños de casos de pruebas correspondientes a las pruebas de aceptación.

Recomendaciones

Continuar con la implementación de más funcionalidades de interés para la Empresa siguiendo la arquitectura propuesta.

Bibliografías

1. Alonso, L. C. (2006). *Modulo Admisión del Sistema Automatizado para la Gestión de Información de la Misión Milagro*. Ciudad de la Habana.
2. Álvarez, M. A. (2005). *Evaluando Zend Studio*. Extraído el 15 Febrero, 2010, de <http://www.maestrosdelweb.com/editorial/zendstudio/>
3. Álvarez, M. A. (2009). *CodeIgniter es un framework PHP para la creación rápida de aplicaciones web*. . Extraído el 15 Febrero, 2010, de Presentación general del framework y primeras notas para empezar a usarlo.: <http://www.desarrolloweb.com/articulos/codeigniter.html>
4. *Comparación de herramientas CASE*.

http://html.rincondelvago.com/herramientas-case_4.html (5/11/2007)
5. Cuenca, C. L. (2006). *Arquitectura del servidor Apache*. Extraído el 15 Febrero, 2010, de <http://www.desarrolloweb.com/articulos/1112.php>
6. Eguiluz, Javier. (2011). Desarrollo web ágil con Symfony2.
7. Group, T.P.G.D. (2009). *PostgreSQL 8.3.7 Documentation [PostgreSQL 8.3.7 Documentación]*. Extraído el 14 Enero, 2010, de <http://www.postgresql.org/docs/8.3/interactive/intro-whatism.html>
8. Gutiérrez, J. J. (2006). *¿Qué es un framework web?* Extraído el 14 Enero, 2010, de <http://jordisan.net/blog/2006/que-es-un-framework/>¿Qué es un framework?
9. Heredia, H. (2005). *¿Qué es el PHP?* Extraído el 26 Enero, 2010, de <http://www.maestrosdelweb.com/editorial/phpintro/>
10. Johnson, A. (2007). *MySQL*. Extraído el 8 Abril, 2010, de Software Libre: http://softwarelibre.net63.net/?page_id=69

11. Johnson, A. (2007). *PostgreSQL*. Extraído el 8 Abril, 2010, de Software Libre: http://softwarelibre.net63.net/?page_id=69
12. *NetBeans*. (2009). Recuperado el 10 de Noviembre de 2010, de <http://es.wikipedia.org/wiki/Netbeans>
13. Pérez Valdés, D. (2007). *Los diferentes lenguajes de programación para la web*. Extraído el 14 Enero, 2010, de <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>
14. Pressman, R. S. (2007). *Ingeniería del Software Un enfoque práctico*. La Habana: Editorial Félix Varela.
15. Riveros, F. (2008). *Gestor de Base de Datos: MySQL, PostgreSQL, SQLite*. Extraído el 15 Febrero, 2010, de http://www.eaprende.com/base_de_datos_SQL_Server_con_PHP_y_ADODB.html
16. Ruiz, M. H. (2006). *Programación WEB avanzada Soluciones rápidas y efectivas para desarrolladores de sitios*. La Habana: Félix Varela.
17. Sandoval Méndez, S. (2009). *Gestores de Bases de Datos*. Extraído el 25 Marzo, 2010, de <http://www.chacharaselnido.com/ITVG/GBD%20EXPO.pptx>
18. *Tecnología CASE*. <http://www.iscmolina.com/Herramientas%20CASE.html> (5/12/2007)
19. Una introducción a Apache. (2006). *Una introducción a Apache*. Extraído el 15 Febrero, 2010, de http://linux.ciberaula.com/articulo/linux_apache_intro/
20. Webmaster. (2009). *HTML, lenguaje de Hipertexto*. Extraído el 10 Febrero, 2010, de Mundo Webmaster: <http://mundowebmaster.wordpress.com/2009/02/21/html-lenguaje-de-hipertexto/>

Anexos

Anexo 1 Representación de la Historia de usuario: Gestionar tipo de contrato

Historia de Usuario	
Número: 13	
Nombre Historia de Usuario: Gestionar tipo de contrato	
Prioridad en Riesgo: Baja	Riesgo en Desarrollo: 3
Puntos Estimados: 3	Iteración Asignada: 3ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de un contrato.	
Observaciones: Tipo de contrato posee atributos como: nombre.	

Anexo 2 Representación de la Historia de usuario: Gestionar cargo

Historia de Usuario	
Número: 15	
Nombre Historia de Usuario: Gestionar cargo	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de un cargo.	
Observaciones: El cargo posee el atributo nombre.	

Anexo 3 Representación de la Historia de usuario: Gestionar idioma

Historia de Usuario	
Número: 5	
Nombre Historia de Usuario: Gestionar idioma	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de un idioma.	
Observaciones: Idioma posee atributos como: nombre.	

Anexo 4 Representación de la Historia de usuario: Gestionar provincia

Historia de Usuario	
Número: 10	
Nombre Historia de Usuario: Gestionar provincia	
Prioridad en Riesgo: Media	Riesgo en Desarrollo: 3
Puntos Estimados: 3	Iteración Asignada: 2da
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de una provincia.	
Observaciones: Provincia posee atributos como: el nombre.	

Anexo 5 Representación de la Historia de usuario: Gestionar especialidades

Historia de Usuario	
Número: 12	
Nombre Historia de Usuario: Gestionar especialidades	
Prioridad en Riesgo: Media	Riesgo en Desarrollo: 4
Puntos Estimados: 4	Iteración Asignada: 2da
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de una especialidad.	
Observaciones: Especialidad posee el atributo nombre.	

Anexo 6 Representación de la Historia de usuario: Gestionar organización

Historia de Usuario	
Número: 14	
Nombre Historia de Usuario: Gestionar organización	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de una organización.	
Observaciones: La organización posee atributos como: el nombre.	

Anexo 7 Representación de la Historia de usuario: Gestionar grupo sanguíneo

Historia de Usuario	
Número: 7	
Nombre Historia de Usuario: Gestionar grupo sanguíneo	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de un grupo sanguíneo.	
Observaciones: Grupo sanguíneo posee el atributo grupo donde se especifica el grupo al que pertenece.	

Anexo 8 Representación de la Historia de usuario: Gestionar área de trabajo

Historia de Usuario	
Número: 2	
Nombre Historia de Usuario: Gestionar área de trabajo	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de un área de trabajo.	
Observaciones: Área de trabajo posee atributos como: nombre e identificador.	

Anexo 9 Representación de la Historia de usuario: Gestionar causa de baja de trabajo

Historia de Usuario	
Número: 4	
Nombre Historia de Usuario: Gestionar causa de baja de trabajo	
Prioridad en Riesgo: Media	Riesgo en Desarrollo: 4
Puntos Estimados: 4	Iteración Asignada: 2da
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de una causa de baja de trabajo.	
Observaciones: Causa de baja de trabajo posee atributos como: nombre	

Anexo 10 Representación de la Historia de usuario: Gestionar grupo salarial

Historia de Usuario	
Número: 6	
Nombre Historia de Usuario: Gestionar grupo salarial	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de un grupo salarial.	
Observaciones: Grupo salarial posee el atributo nombre.	

Anexo 11 Representación de la Historia de usuario: Gestionar ausencia

Historia de Usuario	
Número: 9	
Nombre Historia de Usuario: Gestionar ausencia	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de una ausencia.	
Observaciones: Ausencia posee atributos como: causa, fecha, días laborables, promedio de trabajadores.	

Anexo 12 Representación de la Historia de usuario: Gestionar trabajador

Historia de Usuario	
Número: 1	
Nombre Historia de Usuario: Gestionar trabajador	
Prioridad en Riesgo: Media	Riesgo en Desarrollo: 4
Puntos Estimados: 4	Iteración Asignada: 2da
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de un trabajador.	
Observaciones: Trabajador posee atributos como: CI, nombre, primer apellido, segundo apellido, número tarjeta, fecha de nacimiento, nombre de la madre, nombre del padre, sexo, dirección, color de piel, estatura, peso, si tiene salida al exterior, fecha baja.	

Anexo 13 Representación de la Historia de usuario: Gestionar entidad

Historia de Usuario	
Número: 8	
Nombre Historia de Usuario: Gestionar entidad	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de una entidad.	
Observaciones: Entidad posee atributos como: nombre.	

Anexo 14 Representación de la Historia de usuario: Gestionar tipo misión

Historia de Usuario	
Número: 3	
Nombre Historia de Usuario: Gestionar tipo misión	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de una misión internacionalista.	
Observaciones: Misión internacionalista posee atributos como: fecha de inicio, fecha de fin y país.	

Anexo 15 Representación de la Historia de usuario: Gestionar ubicaciones de defensa

Historia de Usuario	
Número: 11	
Nombre Historia de Usuario: Gestionar ubicaciones de defensa	
Prioridad en Riesgo: Alta	Riesgo en Desarrollo: 5
Puntos Estimados: 5	Iteración Asignada: 1ra
Programador: Dunia Alonso León	
Descripción: Permite la inserción, modificación y la eliminación de la defensa.	
Observaciones: Defensa posee atributos como: número unidad regular, cumplimiento SMG, y si es combatiente o no.	

Anexo 16 Representación de la Tarjeta CRC de la Clase: **tb_ausencia**

Clase: tb_ausencia	
Responsabilidad	Colaboración
Insertar ausencia.	tb_trabajador

Anexo 17 Representación de la Tarjeta CRC de la Clase: **tb_datos_laboral**

Clase: tb_datos_laboral	
Responsabilidad	Colaboración
Insertar datos laborales.	tb_trabajador nom_tipo_contrato nom_grupo_salarial

Anexo 18 Representación de la Tarjeta CRC de la Clase: tb_talla

Clase: tb_talla	
Responsabilidad	Colaboración
Insertar talla.	tb_trabajador

Anexo 19 Representación de la Tarjeta CRC de la Clase: tb_defensa

Clase: tb_defensa	
Responsabilidad	Colaboración
Gestionar defensa.	tb_trabajador nom_ubicacion

Anexo 20 Representación de la Tarjeta CRC de la Clase: **tb_provincia**

Clase: tb_provincia	
Responsabilidad	Colaboración
Insertar provincia.	tb_trabajador tb_municipio

Anexo 21 Representación de la Tarjeta CRC de la Clase: **tb_municipio**

Clase: tb_municipio	
Responsabilidad	Colaboración
Insertar municipio.	tb_trabajador tb_provincia

Anexo 22 Representación de la Tarjeta CRC de la Clase: tb_mision_internacionalista

Clase: <u>tb_mision_internacionalista</u>	
Responsabilidad	Colaboración
Insertar nueva misión internacionalista.	tb_trabajador nom_tipo_mision

Anexo 23 Representación de la Tarjeta CRC de la Clase: tb_trabajador

Clase: <u>tb_trabajador</u>	
Responsabilidad	Colaboración
Insertar nuevos trabajadores.	nom_area_trabajo nom_cargo r_trabajador_idioma tb_datos_laboral tb_talla tb_defensa

	tb_tecnico_profesional
	nom_causa_baja
	tb_ausencia
	r_trabajador_organizacion
	tb_mision_internacionalista
	tb_municipio
	tb_provincia
	nom_grupo_sanguineo
	nom_entidad
	tb_usuario
	tb_pr4
	r_convenio_trabajador

Anexo 24 Representación de la Tarjeta CRC de la Clase: tb_tecnico_profesional

Clase: tb_tecnico_profesional	
Responsabilidad	Colaboración
Insertar Técnicos Profesionales	Tb_trabajador Nom_escolaridad Nom_especialidad

Anexo 25 Representación de la Tarjeta CRC de la Clase: nom_entidad

Clase: nom_entidad	
Responsabilidad	Colaboración
Insertar nueva entidad.	tb_trabajador

Anexo 26 Representación de la Tarjeta CRC de la Clase: nom_area_trabajo

Clase: nom_area_trabajo	
Responsabilidad	Colaboración
Insertar nueva área de trabajo.	tb_trabajador tb_solicitud_servicio

Anexo 27 Representación de la Tarjeta CRC de la Clase: nom_cargo

Clase: nom_cargo	
Responsabilidad	Colaboración
Insertar nuevo cargo.	tb_trabajador

Anexo 28 Representación de la Tarjeta CRC de la Clase: nom_idioma

Clase: nom_idioma	
Responsabilidad	Colaboración
Insertar nuevo idioma.	r_trabajador_idioma

Anexo 29 Representación de la Tarjeta CRC de la Clase: nom_tipo_contrato

Clase: nom_tipo_contrato	
Responsabilidad	Colaboración
Insertar nuevo tipo de contrato.	tb_datos_laboral

Anexo 30 Representación de la Tarjeta CRC de la Clase: nom_grupo_salarial

Clase: nom_grupo_salarial	
Responsabilidad	Colaboración
Insertar nuevo grupo salarial.	tb_datos_laboral

Anexo 31 Representación de la Tarjeta CRC de la Clase: nom_ubicacion

Clase: nom_ubicacion	
Responsabilidad	Colaboración
Insertar nueva ubicación.	tb_defensa

Anexo 32 Representación de la Tarjeta CRC de la Clase: nom_ especialidad

Clase: nom_ especialidad	
Responsabilidad	Colaboración
Insertar nuevo técnico profesional.	tb_tecnico_profesional

Anexo 33 Representación de la Tarjeta CRC de la Clase: nom_ escolaridad

Clase: nom_ escolaridad	
Responsabilidad	Colaboración
Insertar nueva escolaridad.	tb_tecnico_profesional

Anexo 34 Representación de la Tarjeta CRC de la Clase: nom_causa_baja

Clase: nom_causa_baja	
Responsabilidad	Colaboración
Insertar nueva causa de baja de trabajo	tb_trabajador

Anexo 35 Representación de la Tarjeta CRC de la Clase: nom_organizacion

Clase: nom_organizacion	
Responsabilidad	Colaboración
Insertar nueva organización.	r_trabajador_organizacion

Anexo 36 Representación de la Tarjeta CRC de la Clase: nom_grupo_sanguineo

Clase: nom_grupo_sanguineo	
Responsabilidad	Colaboración
Insertar nuevo grupo sanguíneo.	tb_trabajador

Anexo 37 Representación de la Tarjeta CRC de la Clase: nom_tipo_mision

Clase: nom_tipo_mision	
Responsabilidad	Colaboración
Insertar nueva misión.	tb_mision_internacionalista

Anexo 38 Portal



IPH
UEB de Investigaciones y Proyectos
Hidráulicos Sancti Spiritus

UN PROYECTO ÚNICO POR EL AGUA DE TODOS

[Misión](#) [Visión](#) [Valores Compartidos](#) [Competencias Organizacionales](#) [Acceso usuarios](#)

Bienvenidos a la sección del Portal!!!

En esta sección podrá encontrar todo sobre la UEB de Investigación y Proyectos Hidráulicos Sancti Spiritus.

Solo está funcional el Panel, para acceder a él debe primero estar autenticado, la autenticación puede realizarse a través del vínculo Acceso usuarios.

En tu móvil



iPhone iPad Android

Síguenos en ...



Puedes vernos en las redes sociales

IPH	Preguntas frecuentes	Noticias	Síguenos
Misión	Pregunta 1	Noticia 30	
Visión	Pregunta 2	Noticia 29	
Valores Compartidos	Pregunta 3	Noticia 28	
Competencias	Pregunta 4	Noticia 27	
Organizacionales	Pregunta 5	Noticia 26	
	Ver todas	Ver todas	

© Todos los derechos reservados 2013 | IPH

Anexo 39 Secciones de trabajo



IPH
UEB de Investigaciones y Proyectos
Hidráulicos Sancti Spiritus

UN PROYECTO ÚNICO POR EL AGUA DE TODOS

Misión Visión Valores Compartidos Competencias Organizacionales Panel Hola admin # | Cerrar sesión

Inicio > Panel

Secciones de trabajo

Dirección

- Director
- Jefe de servicio
- Especialistas de áreas

Contratación

- Contratador
- Especialista Contratador
- Control de la producción

Proyecto e Investigación

- Proyectista
- Especialista Proyectista
- Investigador
- Especialista Investigador

Recursos Humanos

- Recursos Humanos
- Especialista Recursos Humanos

Usuarios

- Informático

En tu móvil



iPhone iPad Android

Síguenos en ...



Puedes vernos en las redes sociales

IPH

Misión
Visión
Valores Compartidos
Competencias Organizacionales

Preguntas frecuentes

Pregunta 1
Pregunta 2
Pregunta 3
Pregunta 4
Pregunta 5

[Ver todas](#)

Noticias

Noticia 30
Noticia 29
Noticia 28
Noticia 27
Noticia 26

[Ver todas](#)

Síguenos

© Todos los derechos reservados 2013 | IPH

Anexo 40 Recursos Humanos



IPH
UEB de Investigaciones y Proyectos
Hidráulicos Sancti Spiritus

**UN PROYECTO ÚNICO POR EL AGUA
DE TODOS**

Misión Visión Valores Compartidos Competencias Organizacionales Panel

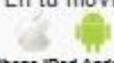
Hola admin 00 | [Cerrar sesión](#)

Inicio > Panel > **Recursos Humanos**

Recursos Humanos

 Gestionar trabajadores	 Gestionar ausencias
 Gestionar áreas de trabajo	 Gestionar provincias
 Gestionar tipos de misión	 Gestionar ubicaciones de defensa
 Gestionar causas de baja de trabajo	 Gestionar especialidades
 Gestionar idiomas	 Gestionar tipo de contrato
 Gestionar grupos salariales	 Gestionar organizaciones
 Gestionar grupos sanguíneos	 Gestionar cargos
 Gestionar entidades	

En tu móvil



iPhone iPad Android

Síguenos en ...



Puedes vernos en las redes sociales

IPH Misión Visión Valores Compartidos Competencias Organizacionales	Preguntas frecuentes Pregunta 1 Pregunta 2 Pregunta 3 Pregunta 4 Pregunta 5 Ver todas	Noticias Noticia 20 Noticia 29 Noticia 25 Noticia 27 Noticia 26 Ver todas	Síguenos
--	--	--	-----------------

© Todos los derechos reservados 2012 | IPH

Anexo 41 Crear especialidad



IPH
UEB de Investigaciones y Proyectos
Hidráulicos Sancti Spiritus

UN PROYECTO ÚNICO POR EL AGUA DE TODOS

[Misión](#) [Visión](#) [Valores Compartidos](#) [Competencias Organizacionales](#) [Panel](#) Hola admin !!! | [Cerrar sesión](#)

[Inicio](#) > [Panel](#) > [Recursos Humanos](#) > [Especialidades](#)

Crear especialidad

Nombre

[Regresar al listado](#)

En tu móvil



iPhone iPad Android

Síguenos en ...



Puedes vernos en las
redes sociales

IPH	Preguntas frecuentes	Noticias	Síguenos
Misión	Pregunta 1	Noticia 30	
Visión	Pregunta 2	Noticia 29	
Valores Compartidos	Pregunta 3	Noticia 28	
Competencias Organizacionales	Pregunta 4	Noticia 27	
	Pregunta 5	Noticia 26	
	Ver todas	Ver todas	

© Todos los derechos reservados 2013 | IPH

Anexo 42 Crear entidad



IPH
UEB de Investigaciones y Proyectos
Hidráulicos Sancti Spíritus

UN PROYECTO ÚNICO POR EL AGUA DE TODOS

[Misión](#) [Visión](#) [Valores Compartidos](#) [Competencias Organizacionales](#) [Panel](#) Hola admin !!! | [Cerrar sesión](#)

[Inicio](#) > [Panel](#) > [Recursos Humanos](#) > **Entidades**

Crear entidad

Nombre

[Regresar al listado](#)

En tu móvil



iPhone iPad Android

Síguenos en ...



Puedes vernos en las redes sociales

IPH

- Misión
- Visión
- Valores Compartidos
- Competencias Organizacionales

Preguntas frecuentes

- Pregunta 1
- Pregunta 2
- Pregunta 3
- Pregunta 4
- Pregunta 5

[Ver todas](#)

Noticias

- Noticia 30
- Noticia 29
- Noticia 28
- Noticia 27
- Noticia 26

[Ver todas](#)

Síguenos

© Todos los derechos reservados 2013 | IPH

Anexo 43 Crear área de trabajo



IPH
UEB de Investigaciones y Proyectos
Hidráulicos Sancti Spiritus

UN PROYECTO ÚNICO POR EL AGUA DE TODOS

[Misión](#) [Visión](#) [Valores Compartidos](#) [Competencias Organizacionales](#) [Panel](#) Hola admin !!! | [Cerrar sesión](#)

[Inicio](#) > [Panel](#) > [Recursos Humanos](#) > **Áreas de trabajo**

Crear área de trabajo

Nombre

Identificador

[Regresar al listado](#)

En tu móvil



iPhone iPad Android

Síguenos en ...



Puedes vernos en las redes sociales

IPH

- Misión
- Visión
- Valores Compartidos
- Competencias Organizacionales

Preguntas frecuentes

- Pregunta 1
- Pregunta 2
- Pregunta 3
- Pregunta 4
- Pregunta 5

[Ver todas](#)

Noticias

- Noticia 30
- Noticia 29
- Noticia 28
- Noticia 27
- Noticia 26

[Ver todas](#)

Síguenos

© Todos los derechos reservados 2013 | IPH

Anexo 44 Crear trabajador



IPH
Unidad de Investigaciones y Proyectos
Hidráulicos Sancti Spiritus

**UN PROYECTO ÚNICO POR EL AGUA
DE TODOS**

Misión Visión Valores Compartidos Competencias Organizacionales Perfil Perfil admin (0) | Cambiar idioma

[Inicio](#) > [Perfil](#) > [Recursos Humanos](#) > [Trabajador](#)

Crear trabajador

En tu móvil

Phone iPad Android

Síguenos en
...

Pueden vernos en las
redes sociales

CI	<input type="text"/>
Nombre	<input type="text"/>
Primer apellido	<input type="text"/>
Segundo apellido	<input type="text"/>
Nro tarjeta	<input type="text"/>
Fecha nacimiento	<input type="text" value="01"/> / <input type="text" value="01"/> / <input type="text" value="2009"/>
Nombre madre	<input type="text"/>
Nombre padre	<input type="text"/>
Sexo	<input type="text"/>
Dirección	<input type="text"/>
Color de piel	<input type="text"/>
Estatura	<input type="text"/>
Peso	<input type="text"/>
Salida anterior	<input type="checkbox"/>
Fecha baja	<input type="text" value="01"/> / <input type="text" value="01"/> / <input type="text" value="2009"/>
Cargo	<input type="text"/>
Área de trabajo	<input type="text"/>
Entidad	<input type="text"/>
Grupo sanguíneo	<input type="text"/>
Municipio origen	<input type="text"/>
Municipio destino	<input type="text"/>
Provincia natural	<input type="text"/>
Causa baja	<input type="text"/>
Usuario	<input type="text"/>

[Regresar al listado](#)

IPH Misión Visión Valores Compartidos Competencias Organizacionales	Preguntas frecuentes Pregunta 1 Pregunta 2 Pregunta 3 Pregunta 4 Pregunta 5 Ver todas	Noticias Noticia 20 Noticia 29 Noticia 24 Noticia 27 Noticia 26 Ver todas	Siguientes
---	--	--	-------------------

© Todos los derechos reservados 2013 | IPH

Anexo 45 Crear organización



IPH
UEB de Investigaciones y Proyectos
Hidráulicos Sancti Spiritus

UN PROYECTO ÚNICO POR EL AGUA DE TODOS

[Misión](#) [Visión](#) [Valores Compartidos](#) [Competencias Organizacionales](#) [Panel](#) Hola admin !!! | [Cerrar sesión](#)

[Inicio](#) > [Panel](#) > [Recursos Humanos](#) > [Organizaciones](#)

Crear organización

Nombre

[Regresar al listado](#)

En tu móvil



iPhone iPad Android

Síguenos en ...



Puedes vernos en las
redes sociales

IPH

- Misión
- Visión
- Valores Compartidos
- Competencias Organizacionales

Preguntas frecuentes

- Pregunta 1
- Pregunta 2
- Pregunta 3
- Pregunta 4
- Pregunta 5

[Ver todas](#)

Noticias

- Noticia 30
- Noticia 29
- Noticia 28
- Noticia 27
- Noticia 26

[Ver todas](#)

Síguenos

© Todos los derechos reservados 2013 | IPH