|  |  |
| --- | --- |
|  | Sede Concepción Talcahuano  JetCat – Primera entrega – Anexos |
|  |  |
|  |  |
|  | Miguel Montero  Cristián De la Rivera |
|  |  |
|  | |  | | --- | |  | | Concepción, diciembre de 2012 | |

# Anexos

## Factibilidades

### Detalle de costos factibilidad técnica

#### Solución 1

#### Hardware

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Elemento** | **Cantidad** | **Valor unitario** | **Valor Bruto** | **Valor Total** |
| **Equipos personal** | 3 | $ 479.990 | $ 403.353 | $ 1.210.059 |
| **Router** | 1 | $ 26.790 | $ 22.513 | $ 22.513 |
| **Hosting** | 1 | $ 46.173 | $ 38.801 | $ 38.801 |
| **Impresora** | 1 | $ 39.990 | $ 33.605 | $ 33.605 |
| **Disco duro portátil** | 1 | . $ 59.990 |  |  |
| **Elemento** | **Durac. Proy** | **Duración** | **Cost. mens.** | **Costo por Hr** |
| **Equipos personal** | 4 meses | 3 años | $ 33.613 | $ 187 |
| **Router** | 4 meses | 5 años | $ 375 | $ 2 |
| **Hosting** | 4 meses | 4 años | $ 808 | $ 4 |
| **Impresora** | 4 meses | 2 años | $ 1.400 | $ 8 |

#### Software

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Elemento** | **Cantidad** | **Valor Unitario** | **Valor Bruto** | **Nº Act** | **Valor Act** |
| Web2py | 3 | $ - | $ - | 0 | $ - |
| Notepad++ | 3 | $ - | $ - | 0 | $ - |
| Netbeans | 2 | $ - | $ - | 0 | $ - |
| Dreamweaver | 1 | $ 193.116 | $ 162.282 | 0 | $ - |
| XAMPP | 3 | $ - | $ - | 0 | $ - |
| Microsoft Office 2010 | 2 | $ 169.400 | $ 142.353 | 0 | $ - |
| Adobe Reader | 3 | $ - | $ - | 0 | $ - |
| GIT | 3 | $ - | $ - | 0 | $ - |
| Visual Paradigm | 1 | $ 144.716 | $ 121.610 | 0 | $ - |
| **Elemento** | **Valor Total** | **Duración** | **Costo mensual** | **Cost por Hr** |  |
| Web2py | $ - | 6 años | $ - | $ - |  |
| Notepad++ | $ - | 6 años | $ - | $ - |  |
| Netbeans | $ - | 6 años | $ - | $ - |  |
| Dreamweaver | $ 162.282 | 6 años | $ 2.254 | $ 13 |  |
| XAMPP | $ - | 6 años | $ - | $ - |  |
| Microsoft Office 2010 | $ 284.706 | 4 años | $ 5.931 | $ 33 |  |
| Adobe Reader | $ - | 6 años | $ - | $ - |  |
| GIT | $ - | 6 años | $ - | $ - |  |
| Visual Paradigm | $ 121.610 | 6 años | $ 1.689 | $ 9 |  |

#### Remuneración del personal:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Cargo** | **Sueldo Base** | **Gratificación** | **Total Imp** | **Movilización** | **Colación** |
| **Jefe proyecto** | $878.565 | $72.042 | $950.607 | $40.000 | $40.000 |
| **Analista** | $453.726 | $72.042 | $525.768 | $40.000 | $40.000 |
| **Programador** | $674.071 | $72.042 | $746.113 | $40.000 | $40.000 |
| **Cargo** | **Total Haber** | **Mutual ACHS** | **Seg Cesantia** | **Feriado Prop** | **Finiquito** |
| **Jefe proyecto** | $1.030.607 | $9.031 | $28.518 | $60.084 | $85.850 |
| **Analista** | $605.768 | $4.995 | $15.773 | $35.316 | $50.460 |
| **Programador** | $826.113 | $7.088 | $22.383 | $48.162 | $68.815 |
| **Cargo** | **Costo Total** | **Valor H. Extra** | **Leyes Soc. H.Ex** | **Costo H.E** | **Cost H. Normal** |
| **Jefe proyecto** | $1.214.090 | $6.833 | $270 | $7.103 | $6.745 |
| **Analista** | $712.312 | $3.529 | $139 | $3.668 | $3.957 |
| **Programador** | $972.562 | $5.242 | $207 | $5.450 | $5.403 |

#### Gastos generales

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Elemento** | **Cantidad** | **Consumo (W/h)** | **Consumo (Kw/h)** | **Costo Hora** | **Total gastos** |
| **Impresora** | 1 | 10 | 0,01 | $ 1,13 | $ 2,51 |
| **Router** | 1 | 12,5 | 0,0125 | $ 1,41 | $ 2,79 |
| **Equipos personal** | 3 | 35 | 0,035 | $ 11,87 | $ 13,24 |
| **Elemento** | **Total costos gen. por hr** | **Total proyecto** |  |  |  |
| **Impresora** | $ 2,51 |  |  |  |  |
| **Router** | $ 2,79 |  |  |  |  |
| **Equipos personal** | $ 39,72 |  |  |  |  |
| **Total** | $ 79,72 | $ 57.399 |  |  |  |

|  |  |
| --- | --- |
| **Internet** | |
| Valor internet mensual | $    24.990 |
| Valor total por hora | $       34,71 |

|  |  |
| --- | --- |
| **Luz** | |
| Kwh | 113 |
| Costo fijo | 990 |
| Costo fijo por hora | 1,375 |

### Detalle de cotización de hardware y software utilizado

#### Hardware:

* Hosting Google App Engine <https://developers.google.com/appengine/>
  + 0 clp (Debajo de 500mb y 5 millones de visitas al mes)
* Hosting Interactive Online <http://interactiveonline.com/>
  + U$D 95,4 Anual
  + 46.173 clp (38.801 clp s/iva)
* Notebook: <http://www.pcfactory.cl/producto/10112-Notebook.DM4-3080LA.Core.i5-2450M.6GB.640GB.14.Windows.7.Home.Premium>
  + 499.990 clp (420.160 clp s/iva)
* Router: <http://www.pcfactory.cl/producto/5979-Router.Inalambrico-N.DIR-615>
  + 26.790 clp (22.513 clp s/iva)
* Impresora: <http://www.pcfactory.cl/producto/8553-Impresora.Laserjet.BN.P1102W>
  + 39.990 clp (33.605 clp s/iva)
* HDD portátil <https://www.pcfactory.cl/?categoria=422&papa=312&pagina=1>
  + 59.990 clp (50411 clp s/iva)

#### Software:

* web2py 1.99.7 <http://www.web2py.com/>
  + 0 clp [LGPLv3](http://www.gnu.org/licenses/lgpl.html) (libre)
* django 1.4 <https://www.djangoproject.com/>
  + 0clp BSD license (libre)
* Pyfdpf 1.54 <http://code.google.com/p/pyfpdf/>
  + 0 clp GNU Lesser GPL (libre)
* Python 2.7 <http://www.python.org/>
  + 0 clp PSF LICENSE (libre)
* Notepad++ 6.1.2 <http://notepad-plus-plus.org/>
  + 0 clp GPL (libre)
* NetBeans IDE 7.1.2 <http://netbeans.org/>
  + 0 clp GPL v2 (libre)
* Eclipse 3.7.2 <http://www.eclipse.org/>
  + 0 clp EPL (libre)
* Jre 6u32 <http://www.java.com/es/download/>
  + 0 clp Libre
* XAMPP 1.7.7 <http://www.apachefriends.org/es/xampp.html>
  + 0 clp GPL (libre)
* Microsoft Office 2010 <http://www7.buyoffice.microsoft.com/latam/product.aspx?sku=10234449&cache=395768723&action=buy&culture=es-MX>
  + U$D 350.00 Privativa
  + 169.400 clp (142.353 clp s/iva)
* Dreamweaver CS6 <http://www.adobe.com/products/dreamweaver/buying-guide.html>
  + U$D 399 Privativa
  + 193.116 clp (162282 clp s/iva)
* Adobe Reader 10.1.3 <http://get.adobe.com/es/reader/?promoid=DINRS>
  + 0 clp Libre
* Microsoft Expression Web 4 <http://www.microsoft.com/expression/products/Purchase.aspx>
  + U$D 149 Privativa
  + 72.116 clp (60.601 clp s/iva)
* Visual Paradigm for UML Standard
  + U$D 299.00
  + 144.716 clp (121.610 clp s/iva)
* Git 1.7.10.1 <http://git-scm.com/>
  + 0 clp (libre)

## Documentación de metodología usada

### Estándar de modelo entidad-relación y Base de datos

El modelo de entidad relación está construido en la herramienta CASE ‘*Visual Paradigm Comunity Edition*’ y, por lo tanto, está regido por las reglas de diagramación que posee este software. Estas limitaciones responden mayormente a estándares de presentación gráfica de la estructura lógica diseñada.

Además de lo anterior, cualquier nombre utilizado en la creación de Bases de datos, tablas o campos no puede tener espacios, números ni caracteres especiales; dentro de estos caracteres especiales se incluye la letra “ñ” y los signos de interrogación y exclamación. Se excluye el guion bajo “\_”, utilizado para representar espacios.

Los nombres de las tablas empiezan con mayúsculas (letra inicial) y siguen con minúsculas. En el caso de las tablas que se compongan lógicamente de otras, se puede poner un nombre representativo a la lógica de la relación o las dos tablas principales (con la primera letra de cada tabla en mayúscula), sin espacio entre ellas. (e.g. TablaunoTablados).

El nombre de los campos debe ir con minúsculas, evitando usar múltiples palabras. Se puede usar abreviaciones de mínimo 3 caracteres, aunque no es recomendado. Para el caso de las llaves foráneas, se debe colocar el nombre de la tabla de la que proviene la llave (Incluyendo su inicial en mayúscula), seguido de un guion bajo y el nombre del campo (e.g. Tablauno\_id)

### Modelo por Prototipos

Un cliente a menudo define un conjunto de objetivos generales para el software, pero no identifica los requisitos detallados de entrada, procesamiento, o salida. En otros casos, el responsable del desarrollo del software puede no estar seguro de la eficacia de un algoritmo, de la capacidad de adaptación de un sistema operativo, o de la forma en que debería tomarse la interacción hombre-máquina. En éstas y en otras muchas situaciones, un paradigma de construcción de prototipos puede ofrecer el mejor enfoque.

El paradigma de construcción de prototipos comienza con la recolección de requisitos. El desarrollador y el cliente encuentran y definen los objetivos globales para el software, identifican los requisitos conocidos, y las áreas del esquema en donde es obligatoria más definición. Entonces aparece un «diseño rápido». El diseño rápido se centra en una representación de esos aspectos del software que serán visibles para el usuario/cliente. (P ej. enfoques de entrada y formatos de salida). El diseño rápido lleva a la construcción de un prototipo. El prototipo lo evalúa el cliente/usuario y lo utiliza para refinar los requisitos del software a desarrollar. La interacción ocurre cuando el prototipo satisface las necesidades del cliente, a la vez que permite que el desarrollador comprenda mejor lo que se necesita hacer.

Este método contiene condiciones únicas de aplicación, en donde los encargados del desarrollo tienen poca experiencia o información, o donde los costos y riesgos de que se cometa un error pueden ser altos. Así mismo este método resulta útil para probar la facilidad del sistema e identificar los requerimientos del usuario, evaluar el diseño de un sistema o examinar el uso de una aplicación.

El método del prototipo de sistemas consta de 5 etapas:

1. Identificación de requerimientos conocidos: La determinación de los requerimientos de una aplicación es tan importante para el método de desarrollo de prototipos como lo es para el ciclo de desarrollo de sistemas o análisis estructurado. Por consiguiente, antes de crear un prototipo, los analistas y usuario deben de trabajar juntos para identificar los requerimientos conocidos que tienen que satisfacer.
2. Desarrollo de un modelo de trabajo: Es fácil comenzar el proceso de construcción del prototipo con el desarrollo de un plan general que permita a los usuarios conocer lo que se espera de ellas y del proceso de desarrollo. Un cronograma para el inicio y el fin de la primera interacción es de gran ayuda.

El profesional de sistema para construcción inicial del prototipo emplea cualquier herramienta, como Lenguajes de Cuarta Generación, Generadores de Reportes, Generadores de Pantallas en el desarrollo de un prototipo se preparan los siguientes componentes:

* El lenguaje para el diálogo o conversación entre el usuario y el sistema
* Pantallas y formatos para la entrada de datos
* Módulos esenciales de procesamientos.
* Salida del sistema la incorporación en la interfaz de entrada/salida de características representativas de las que serán incluidas en el sistema final permite una mayor exactitud en el proceso de evaluación.

1. Utilización del prototipo: Es responsabilidad del usuario trabajar con el prototipo y evaluar sus características y operación. La experiencia del sistema bajo condiciones reales permite obtener la familiaridad indispensable para determinar los cambios o mejoras que sean necesarios, así como las características inadecuadas.
2. Revisión del prototipo: Durante la evaluación los analistas de sistemas desean capturar información sobre los que les gusta y lo que les desagrada a los usuarios.

Los cambios al prototipo son planificados con los usuarios antes de llevarlos a cabo, sin embargo es el analista responsable de tales modificaciones.

1. Repetición del proceso las veces que sea necesarias: El proceso antes descrito se repite varias veces, el proceso finaliza cuando los usuarios y analistas están de acuerdo en que el sistema ha evolucionado lo suficiente como para incluir todas las características necesarias.

Problemas Candidatos: Para decidir si el prototipo debe incluirse o no Ciclo de Desarrollo de Sistema de Información, el profesional considera los siguientes factores:

* Problemas no estructurado, novedosos y complejos, de información personalizada del usuario ,ya que sus salidas no son predecibles y definidas
* Problemas de ambiente Inestable, el profesional también debe evaluar el contexto del sistema
* Experiencia en diseños similares
* No se conocen los requerimientos, la naturaleza del sistema es tal que existe poca información con respecto a las características que debe tener el nuevo sistema para satisfacer las necesidades del usuario.
* Los requerimientos deben evaluarse, se conocen los requerimientos aparentes de información pero es necesario verificarlos y evaluarlos
* Costos altos, donde la inversión involucra gran cantidad de recursos financieros y humanos.
* Altos riesgo, la evaluación inexacta de los requerimientos o el desarrollo incorrecto ponen en peligro a la organización.
* El usuario, donde no está dispuesta examinar modelos en papel, o no sabe lo que quiere pero lo reconocerá cuando lo vea.
* Tecnologías Nuevas, la falta de experiencia en el uso de dichas tecnologías, junto con el deseo de instalar nuevas tecnología hace que sea propicio el uso del prototipo.

Estrategias para el Desarrollo de Prototipos: Se puede desarrollar un prototipo para cada uno de los componentes de la aplicación Prototipos por Pantallas.

La interface entre el sistema y el usuario es la pantalla de visualización, esta es el vehículo para presentar la información tal como ésta es proporcionada al sistema o como es recuperada de éste.

Los prototipos de pantalla permiten evaluar la posición de información sobre la pantalla, los encabezados, los botones, mensajes.

También permite la reacción de los usuarios por la cantidad de información sobre la pantalla. La creación de un prototipo de pantalla conduce a:

* Que debe presentarse como información sobre la pantalla principal
* Cuál pertenece a una pantalla de detalle

Prototipos para Procedimientos de Procesamientos: Las funciones de procesamiento incluye entradas, cálculos, recuperar información y actividades de salidas.

Como los datos pocas veces son ingresados de la forma correcta o en la secuencia válida, es por ello que la aplicación se diseña para asegurar la detección de errores.

El objetivo es determinar si los procedimientos de aplicación fueron desarrollados adecuadamente.

La evaluación de los procedimientos y la observación de errores y equivocaciones cometidas por los individuos cuando emplean el prototipo, pueden sugerir la adición de características de manejo de errores que no se habían anticipado.

Prototipos de Funciones Básicas: Para determinar los requerimientos de una aplicación no es necesario desarrollar todos los módulos del sistema, sino los básicos, son aquellos que forman el núcleo de la aplicación.

Incluye las funciones primarias de la aplicación como edición y validación, y excluye las secundarias como el manejo de archivos que no forman parte del procesamiento esencial.

Ejemplo de modelo por prototipo:

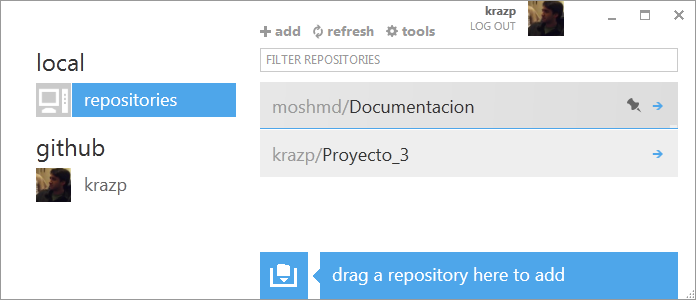
## Control de versiones a SW a entregar

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Descripción: C:\Users\cliente\Dropbox\TIPI\Jetcat tumb x64.png | | **Procedimiento para control de versiones** | | | | Código: 001 | |
| **Formato para control de versiones** | | | | Versión SW: 1.0 | |
| Control de versiones del documento: | | | | | | | |
| **VERSIÓN** | **FECHA** | | **MODIFICADO** | | **REVISADO** | | **APROBADO** |
| 1.0 | 28-09-2012 | | Miguel Montero | | Cristian De la Rivera | | Silvia R. |
| 1. **Nombre para referenciar el cambio o actividad:** | | | | | | | |
| **Tipo de cambio:** | | | | | | | |
| **Elemento Modificado:** | | | | | | | |
| Documentación | | | | Código | | | |
| * **Actualización** * **Modificación** * **Borrar** * **Agregar** * **Otro** | | | | * **Actualización** * **Modificación** * **Eliminar** * **Agregar** * **Reestructurar** * **Otro** | | | |
| 1. **Origen de la solicitud del cambio:** | | | | | | | |
| Documentación | | | | Código | | | |
| * Nueva documentación * Error en modificación * Cambio en política de documentación * Otro | | | | * Solicitud de mejora * Fallos * Otro | | | |
| 1. **Prioridad:** | | | | | | | |
| * Alta * Media * Baja | | | | | | | |
| 1. **Es necesario que este cambio se haya realizado antes del día:** | | | | | | | |
| Aviso requerido: \_\_\_\_\_\_\_\_\_ días antes. | | | | | | | |
| 1. **Fecha estimada para la realización del cambio y tiempo (horas/días) requeridos para su realización** | | | | | | | |
| 1. **Descripción del cambio (Documentación/Código)** | | | | | | | |
| 1. **Justificación de cambio / Objetivos del cambio** | | | | | | | |
| 1. **Comentarios:** | | | | | | | |
| Archivado por: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Firma:\_\_\_\_\_\_\_\_\_\_\_\_\_ | | | | | | | |

El mecanismo principal de control de versiones para el desarrollo del proyecto está basado en el protocolo de git, para el cual se utiliza principalmente el software *github for Windows* y complementariamente la consola *Shell* de git (En casos específicos donde se requieran comandos especializados).

Básicamente, git puede funcionar de manera local o remota, donde todos los cambios realizados por cualquier miembro del equipo se van guardando de manera individual. Por otra parte si se esta trabajando con el mismo archivo en git, se comparan las líneas de del archivo (Como líneas de código cuando corresponde) y se guarda de forma individual. En el caso de que esto causara un conflicto se muestra un mensaje con una advertencia.

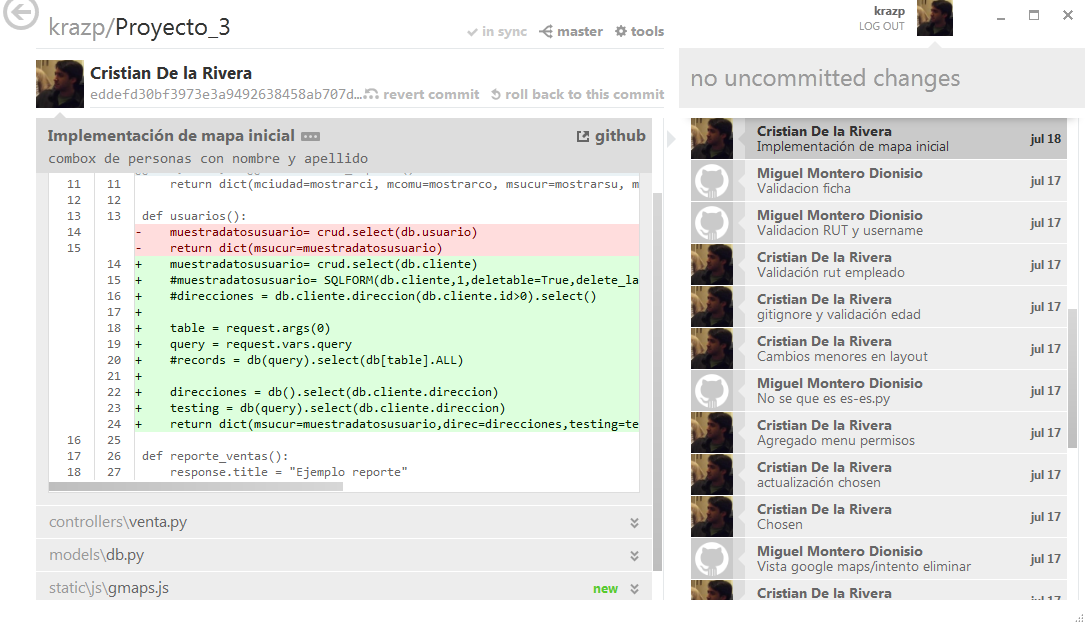
Capítulo 11, imagen 1



Menú principal de *github for Windows*

El control de software y documentación que se realiza a través de este medio permite la identificación de cambios de forma individual por cada miembro del equipo, además del detalle de los cambios realizado en cada *commit*. Además, existe la oportunidad de revertir el estado del elemento controlado a estados anteriores o revertir cambios específicos en caso de detectarse un error. En el desarrollo del proyecto, se maneja de forma separada el código fuente de la aplicación y la documentación.

Capítulo 11, imagen 2



Lista de cambios y detalle en *github for Windows*

Las posibilidades del software permiten analizar el progreso de cada miembro en particular y del proyecto en general, asistido por los nombres de cada ‘actualización’ o commit; en conjunto con las fechas. Una vez alcanzado hitos de programación avanzados, cada commit será identificado con un número de versión que se acerque a la versión 1.0 para la entrega del prototipo final, avanzando un número para cada adición de funcionalidades completas y decimales para la mejora de funcionalidades existentes.

## Documentación de seguimiento y controles

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Descripción: C:\Users\cliente\Dropbox\TIPI\Jetcat tumb x64.png | | **Procedimiento para seguimiento y controles** | | | | Código: 001 | |
| **Formato para seguimiento y controles** | | | | Versión: 1.0 | |
| Control de versiones del documento: | | | | | | | |
| **VERSIÓN** | **FECHA** | | **MODIFICADO** | | **REVISADO** | | **APROBADO** |
| 1.0 | 28-09-2012 | | Miguel Montero | | Cristian De la Rivera | | Silvia R. |
| 1. **Nombre para referenciar seguimiento y control:** | | | | | | | |
| **Elemento controlado:** | | | | | | | |
| Documentación | | | | Código | | | |
| Estado: | | | | Estado: | | | |
| 1. **Origen de la solicitud del cambio:** | | | | | | | |
| Documentación | | | | Código | | | |
| Comentario: | | | | Comentario: | | | |
| 1. **Es necesario que este cambio se haya realizado antes del día:** | | | | | | | |
| Aviso requerido: \_\_\_\_\_\_\_\_\_ días antes. | | | | | | | |
| 1. **Fecha estimada para la realización del cambio y tiempo (horas/días) requeridos para su realización** | | | | | | | |
| 1. **Descripción del cambio (Documentación/Código)** | | | | | | | |
| 1. **Justificación de cambio / Objetivos del cambio** | | | | | | | |
| 1. **Etapa en la que se realiza el seguimiento y control**  * Análisis * Diseño * Construcción * Implementación | | | | | | | |
| 1. **Comentarios:** | | | | | | | |
| Archivado por: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Fecha: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Firma:\_\_\_\_\_\_\_\_\_\_\_\_\_ | | | | | | | |

## Documentación de técnica de recolección de datos

A continuación se presentan tres técnicas para obtener información de la empresa que sea útil para el desarrollo del proyecto. Se utiliza una entrevista, un cuestionario y un proceso de observación *in situ.* En el caso de la entrevista y el cuestionario, se muestran primero las estructuras de éstos para después mostrar un ejemplo contestado de cada uno. Los objetos de estudio de estas técnicas corresponden respectivamente a:

**Administrador**: Buscando obtener información de la visión administrativa de la empresa.

**Personal**: Con intención de conocer la postura del lado operacional de la empresa.

**Observación *in situ***: Analizando la estructura de funcionamiento a escala general de la empresa.

### Entrevista

Entrevista aplicada

(Contestada por el administrador)

1. ¿Cuánto tiempo lleva en este negocio?

25 años

1. ¿Cuánto tiempo lleva trabajando con el sistema actual de manejo de información y porqué lo eligió?

25 años y porque era el único que existía en ese tiempo y el más accesible

1. ¿Cómo describiría el actual flujo de información de la empresa?

Es un mecanismo que manejo yo de forma manual

1. ¿En que siente que el sistema actual lo ayuda o lo perjudica?

Me permite mantener el control, pero siempre se pierden papeles o son difíciles de encontrar. También el sistema actual nos perjudica en que es lento y nos ayuda en que no se necesita ninguna capacitación especial para usarlo.

1. ¿Qué lo “impulsó” a cambiar el sistema?

Bueno, otras empresas ya tienen implementados estos sistemas y trabajan más rápido. También por que las nuevas tecnologías están de moda

1. ¿Cuál es su opinión sobre el sistema de registro de información actual de la empresa?

Que es muy lento para estos días en que todo es rápido.

1. Si pudiera cambiar un aspecto de como se maneja la información en la empresa, ¿cuál sería?

Que todo esté ordenado y sea más fácil de encontrar.

1. ¿Cuál es el volumen de información que registra la empresa?

Uff, creo que es bastante para el poco personal que maneja esta información

1. ¿Cuántas personas se dedican a registrar esta información?

Dos personas se reparten el trabajo, la secretaria y yo, el administrador

¿Cree usted que podría haber algún riesgo al implementar un nuevo sistema en la empresa?  
Mmm… todo cambio trae algún tipo de riesgo.

1. ¿Cómo desea utilizar la información que entregue la aplicación?

Me interesa poder ver cosas en la pantalla y tener archivos excel

1. ¿Qué tareas considera monótonas en su quehacer laboral diario?

El contar y registrar el flujo de caja.

1. ¿Cuáles son los principales problemas que se encuentra en el trabajo normalmente?

Encontrar tiempo para lograr manejar todas las cosas que se necesitan hacer.

1. ¿Cuánto conocimiento general tiene sobre computación?

Diría que termino medio.

1. ¿Qué programas usa actualmente en su trabajo?

En el trabajo sólo usamos el correo y Word.

1. ¿Cuál es el programa que usa más seguido?

Word

1. ¿Qué tarea es la que más le quita tiempo productivo?

El registrar el flujo de caja.

1. ¿Qué problema identificaría como el peor factor para la productividad de los empleados de la empresa?

La desorganización es lo que más nos afecta

1. ¿Cuál cree usted es la característica que menos le gusta a los empleados de la empresa sobre el uso del sistema de fluyo de información actual?

Que a veces no saben donde buscar las cosas o lo que hay que guardar

1. ¿Qué es lo que más le interesa que haga el sistema por usted?

Bueno, que pueda guardar las fichas y las ventas, y que haga lo del flujo de caja ya que nos toma mucho tiempo hacerlo a mano.

1. ¿Puede describir específicamente las tareas que necesita que realice el sistema?

Manejar los datos y el asunto del flujo de caja que ya dije.

1. ¿Cuántas personas deben interactuar con el sistema?

Dos trabajadores al principio

1. ¿Cuántos empleados tienen relación directa con el proyecto?

Ahora dos trabajadores.

1. ¿Cuáles son sus expectativas con respecto a la facilidad de uso del sistema?

Espero que sea fácil de usar para que no sea tan necesario las capacitaciones.

1. ¿Cuáles serían los objetivos específicos que le gustaría obtener del sistema?

Que me ayude con el negocio y haga el trabajo más fácil

1. ¿Además de las funcionalidades técnicas del software, desea lograr algo más con este proyecto?

Que el negocio siga creciendo, poder atraer más clientes, saber de donde vienen para saber donde hacer propaganda.

### Cuestionario 1

(Aplicado a la secretaria)

1. ¿Cuánto tiempo lleva trabajando en la empresa?

5 años

2. Según su sensación, ¿Qué tan importante es su gestión dentro del funcionamiento de la empresa?

Nada – poco – normal – bastante – mucho

3. ¿Qué tan satisfecho está con el sistema de registro de información actual de la empresa?

Nada – poco – normal – bastante – mucho

4. ¿Qué tan eficientemente logra trabajar con el sistema actual de la empresa?

Nada – poco – normal – bastante – mucho

5. ¿Qué es lo que más le perjudica a la hora de intentar realizar una tarea en el trabajo?

Que hay poco personal.

6. ¿Cuál es la herramienta más importante o útil a la hora de realizar su trabajo diario?

El lápiz y la calculadora.

7. ¿Cuál es el principal problema que le impide realizar alguna tarea que requiera hacer en el trabajo?

Que hay poco personal, entonces todo es más lento.

8. ¿Cómo logra solucionar o sortear el problema de la pregunta anterior?

Hay que apurar o simplificar algunas cosas, o dejarlas para después

9. Si pudiera cambiar un aspecto del funcionamiento actual de la empresa ¿Cuál sería?

Que las cosas serán más ordenadas para que no se pierdan

10. ¿Cree que la empresa necesita automatizar este sistema?

Si – No

11. ¿Cree que este nuevo sistema mejorará sus funciones en la empresa?

Si – No

12.-. ¿En forma cree que mejoraría la empresa con un nuevo sistema de información?

En que sería más rápido.

13. ¿Cree que la empresa está en condiciones para este cambio?

Si – No

14. De 1 a 10, ¿Qué tanto cree que cambiaría su esquema de trabajo con un nuevo sistema de información?

1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10

15. De 1 a 10, ¿Qué tan importante para su trabajo es algún tipo de software?

1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10

16. ¿Cómo le gustaría que fuese un software con el que deba trabajar?

Claro, rápido, eficaz y fácil

17. ¿Qué funciones desearía que tuviera el sistema?

Que recibiera información de los laboratorios

18. ¿Qué tanto cree usted que la empresa necesita modernizar su sistema de manejo de información?

Nada – poco – normal – bastante – mucho

19. De 1 a 10, ¿Qué nivel de conocimientos computacionales considera tener?

1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10

20. ¿Con qué frecuencia utiliza el computador?

Nada – poco – normal – bastante – mucho

21. ¿Ha trabajado con computadores en la empresa?

Si – No

22. ¿Debió aprender nuevas habilidades al comenzar a trabajar en esta empresa?

Si – No

23. ¿Está dispuesto a aprender nuevas habilidades computacionales si la empresa moderniza su actual sistema de manejo de información?

Si – No

24. Según su opinión, ¿Qué tanto afecta en el éxito de la empresa el sistema de información que posea?

Nada – poco – normal – bastante – mucho

25. Según su opinión, ¿Qué tanto influye en su desarrollo como profesional capacitarse en nuevas tecnologías?

Nada – poco – normal – bastante – mucho

### Observación *In Situ*

En un proceso de observación por el analista, se analiza el trabajo realizado en la empresa de forma presencial y éste genera una serie de observaciones relevantes para el desarrollo.

* Se logro observar que varias tareas que se efectúan dentro de la veterinaria se realizan de forma manual, tales como: cotizaciones, emisión de boletas, registro de datos de clientes con sus respectivas mascotas, compras que realiza la misma empresa para insumos o los mismos clientes dentro de la veterinaria.
* Todas las operaciones definitivas las realiza el administrador y éste desea mantener este poder inalterado
* Las hojas de información luego de utilizadas eran almacenadas en distintos lugares sin un orden especifico.
* Se determino que no existe un registro claro con respecto a los productos que se venden, es decir, no tienen una estadística clara de que es lo que falta o está demás.
* No se sabe de manera certera que productos son los más requeridos por los clientes.
* La empresa no cuenta con algún método de publicidad que le permita hacerse conocer dentro del sector en el que opera o sus alrededores.
* No se tiene conocimiento del origen de los clientes nuevos o antiguos.
* Existe una alta desorganización en la comunicación del personal.
* Se percibe un deterioro progresivo de parte de las hojas que contienen información.
* Existe una buena respuesta de los clientes con la cercanía del personal, especialmente cuando muestran cercanía conociendo datos como el nombre.
* Ocasionalmente el administrador debe consultar información al personal, entorpeciendo sus labores.
* El personal expresa conocimiento de las costumbres de consumo de ciertos clientes, pero en un número reducido.
* Ocasionalmente ocurren conflictos al intentar obtener la información de dueño de una mascota.
* Ocasionalmente se presentan problemas por falta de stock de algún medicamento, como vacunas.
* Se presentaron casos de pérdida de información de clientes, debiendo contactarlos de un modo informal, mostrando poco profesionalismo.
* El formato de los documentos para guardar información es irregular
* Se nota un vínculo de comunicación directa entre el personal, a pesar de su desorganización
* No existe una clasificación clara de los lugares donde se guarda información.
* No existe un conocimiento común de los lugares donde se guarda la información.
* Se presencia la toma de información por parte del personal, dejando a la empresa sin este material al no tener respaldo.
* Ocasionalmente el servicio al cliente se ve retrasado por reunir datos de él o documentos necesarios para brindarle el servicio.
* Cuando se presenta un problema, se evidencia que el personal no sabe pasos claros a seguir para solucionar el conflicto.
* Existe ocasionalmente colisión de autoridades al entregar órdenes. (Por ejemplo, veterinario discordante con administrador)
* Ocasionalmente varios empleados desean acceso a cierta información simultáneamente, no pudiendo acceder a ésta al mismo tiempo.
* Se observa información multiplicada pero inconsistente.
* No existe conocimiento claro ni registros de quien accede a los distintos materiales o datos de la empresa.

## Planificación de control de riesgos identificados

**Problemas con entrevistas**

• Antes

* Se establece la reunión con cliente, especificando el lugar y la hora por algún medio escrito, como correo electrónico.
* Se revisa la fecha de reunión, un día hábil antes de esta se contacta con el cliente para recordar y confirmar la reunión.

• Durante

* Se registran los datos relevantes de la reunión, así como apreciaciones que tenga el jefe de proyecto respecto a la naturaleza del cliente y la empresa; para ajustarse a futuros comportamientos.
* Se registran problemas que ocurren dentro de la reunión dentro de una pauta, para su posterior estudio y planificación de mejoras.

• Después

* Hay que considerar que esta nueva entrevista debe ocurrir lo más pronto posible. Se contacta al cliente para notificar la pérdida de una reunión y planificar la siguiente. Una vez acordado, se debe notificar al SQA, incluyendo la fecha de la siguiente reunión.

**Fallo de hardware o software**

• Antes

* Establecer plan de emergencia ante la contingencia, ya sea de hardware o software. Éstos se deben actualizar constantemente para adaptarse a los cambios constantes.
* Se revisa constantemente el hardware y software utilizado para asegurar que funciona correctamente

• Durante

* Se registra información sobre los problemas en un formulario de fallas de hardware o software.
* Se avisa al resto del personal relacionado con la falla y se toman medidas en el momento para mitigar el efecto de ésta.

• Después

* Evaluar datos registrados de fallos de hardware o software y tomar acciones correctivas correspondientes para evitar que se repita.
* Notificar al Jefe de Proyecto y SQA para ajustar la planificación de acuerdo al impacto de la falla.
* Se deciden soluciones para revertir el daño realizado; remplazando o ajustando prácticas de trabajo para minimizar el impacto

**Falta de hardware o herramienta de software**

• Antes

* Establecer el hardware o herramienta de software necesario para el desarrollo de alguna etapa de trabajo con antelación.
* Se debe hacer una revisión y verificación de hardware y herramientas de software que serán necesarias para el trabajo.

• Durante

* Se registra un formulario de hardware y herramienta de software existentes, como información de los componentes existentes.
* Se informa de la inexistencia del hardware o herramienta de software al personal relacionado con éste y se realiza una reunión de emergencia para diseñar estrategias que mitiguen el efecto de ésta carencia.
* Se aplican las estrategias anteriormente decididas y se documentan.

• Después

* Se evalúan los datos registrados de inexistencias y agenda una reunión para decidir estrategias para mitigar el daño al proyecto y la posible adquisición del hardware o software faltante.
* Se documentan las soluciones planteadas y los resultados de sus aplicaciones.

**Problemas base de datos**

• Antes

* Se establece el tipo de base de datos a implementar, contemplando su robustez y funcionalidad
* Revisión de la base de datos desarrollada, probando su estabilidad frente a varias consultas y documentando los resultados.

• Durante

* Se cierran las conexiones de la base de datos y se identifica el problema en caso de error con mensaje o código.
* Establecer formulario con errores producidos en la BDD.
* Revisión de formulario con la contingencia.
* Se aísla el problema por medio de consultas directamente al motor de base de datos.

• Después

* Se realiza una reunión para identificar el problema y decidir soluciones.
* Evaluar base de datos y aplicar acciones correctivas a la configuración o modificación de la aplicación o la BDD, para cada error

**Falta de equipo necesario**

• Antes

* Establecer el equipo necesario para el desarrollo del proyecto.
* Establecer pruebas parciales de los elementos involucrados en la implementación.
* Detectar posibles falencias o necesidades adicionales.
* Revisión de equipo, pruebas, falencias y/o necesidades adicionales.

• Durante

* Valida y verifica la existencia del equipo necesario para aguantar el personal,
* Registra formulario con personal existente necesario.
* Verifica formulario.

• Después

* Se busca las opciones que existen de equipo que cumpla los requerimientos y se selecciona a opción que cumpla mejor las necesidades del proyecto. Esto se debe notificar al SQA si contempla un retraso significativo dentro del desarrollo del proyecto.

**Caída de servicios**

• Antes

* Se debe mantener organizada la información de los distintos servicios que utiliza el sistema.
* Revisión constante de los servicios que utiliza el sistema para su correcto funcionamiento.
* Las consideraciones relevantes de cualquier aspecto relacionado con servicios debe ser agregada a la planificación temporal.

• Durante

* Detener todas las actividades que dependan del servicio.
* En caso de desperfecto parcial, bajar el servicio que esta funcionando mal.
* Revisar operaciones anexas al servicio para verificar su correcto funcionamiento.
* Establecer formulario para caída de servicios.
* Comunicarse con la empresa proveedora de servicios para adquirir información sobre la caída.
* Reunión de emergencia para mitigar efectos de caída.

• Después

* Mantención de los servicios que se utilizan dentro del Software.
* Reunión para decidir soluciones, aplicarlas y documentarlas.
* Establecer mecanismos para reducir el impacto en caso de repetición del problema.

**Requerimientos no viables**

• Antes

* Establecer objetivos necesarios para el proyecto.
* Realizar estudio o análisis de técnica a utilizar para recolección de datos.
* Realizar entrevista para toma de requerimientos.
* Realizar investigación de factibilidades relacionadas al área de negocio del cliente.

• Durante

* Registrar problemas que ocurran en el análisis.
* Verificar formulario con errores durante el análisis.
* Realizar acciones correctivas.

• Después

* En caso de que el conflicto sea detectado después, se debe contactar al cliente para informarle sobre conflicto, con posibles soluciones o alternativas disponibles que posee.

**Error en la planificación**

• Antes

* Establecer planificación temporal.
* Establecer tiempos de holgura para el desarrollo de la planificación.

• Durante

* Si se detecta una falla en la planificación, debe identificarse que ocasionó el problema y estimar el grado de daño que producirá esto sobre el desarrollo del proyecto.
* Registrar contingencias que ocurran dentro del desarrollo de la planificación.
* Verificar registro de contingencia.

• Después

* El jefe de proyecto toma medidas para neutralizar cualquier problema que pueda estar afectando la planificación y actualizar ésta para reflejar la nueva situación. Se debe incluir los cambios realizados en la planificación al cliente cuando esta ocasione cambios importantes.

**El sistema requiere más esfuerzo del presupuestado**

• Antes

* Establecer claramente las tareas y los tiempos.
* Evaluar el esfuerzo real requerido por tarea.
* Actualizar planificaciones.

• Durante

* Establecer formulario para incidencias dentro del desarrollo.
* Registrar problemas ocurridos dentro del tiempo en que se desarrolle el proyecto.
* Verificar formulario con problemas.

• Después

* Se deben buscar alternativas para aumentar la productividad del personal para mitigar los efectos de este problema sobre el desarrollo del proyecto.

**Cambio de requerimiento**

• Antes

* Establecer los requerimientos del usuario.
* Verificar que se cumplan los requerimientos dentro del proyecto.

• Durante

* Se debe analizar el requerimiento modificado por el cliente y evaluar el impacto que tiene sobre el diseño actual del software.
* Registrar requerimientos dentro de un formulario.
* Verificar datos del formulario

• Después

* Se debe identificar las partes del sistema comprometidas por el cambio y notificar al SQA. Este debe estimar el efecto que tendrá el cambio sobre el desarrollo y actualizar la planificación para reflejar estos cambios. Finalmente se rediseña las secciones involucradas para que cumplan con los nuevos requerimientos.

## Representación gráfica de planificación temporal

*Ver carta Gantt anexada*

## Otros

* Formulario para problema con entrevistas

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO ENTREVISTAS “001” | | | |
| **Encargado entrega** | : | Cristian De la Rivera |  |
| **Fecha** | : | 10/09/2012 |  |
| **Hora** | : | 14:00 PM |  |
| **Establece fecha reunión** | | | |
| **Confirmar fecha reunión** | | | |
| **Notas** | | | |
| **Registro de problemas en lo que concierne a la reunión** | | | |
| **Comentario**  : | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

* Formulario para fallas de hardware y/o software

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO HW/SW FALLAS “001” | | | |
| **Encargado entrega** | : | Cristian De la Rivera |  |
| **Fecha** | : | 11/11/2012 |  |
| **Hora** | : | 15:00 PM |  |
| **1**.**Tipo Hardware** | | | |
| * Notebook/PC * Impresora * Router * Internet * Disco duro * Otro (especificar) | | | |
| **2. Tipo Software** | | | |
| * Web2py * Notepad++ * Netbeans * Dreamweaver * XAMPP * Microsoft Office 2010 * Adobe Reader * GIT * Visual Paradigm * Otro (especificar) | | | |
| **Comentario:** | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

* Formulario para hardware y/o software faltantes dentro del proyecto

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO HW/SW Faltante “001” | | | |
| **Encargado entrega** | : | Cristian De la Rivera |  |
| **Fecha** | : | 11/11/2012 |  |
| **Hora** | : | 15:00 PM |  |
| **Tipo hardware faltante** | | | |
| Especificar: | | | |
| **Tipo software faltante** | | | |
| Especificar: | | | |
| **Comentario:** | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

* Formulario para problemas de BDD

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO PROBLEMAS BDD “001” | | | |
| **Encargado entrega** | : | Miguel Montero D. |  |
| **Fecha** | : | 11/10/2012 |  |
| **Hora** | : | 16:00 PM |  |
| **Establecer hora y fecha en el que se cierre de conexión** | | | |
| **Identificar y establecer los errores producidos** | | | |
| **Acción correctiva de contingencia** | | | |
| **Comentarios** | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

* Formulario para falta de equipo necesario

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO FALTA DE EQUIPO NECESARIO “001” | | | |
| **Encargado entrega** | : | Cristian De la Rivera |  |
| **Fecha** | : | 11/10/2012 |  |
| **Hora** | : | 16:00 PM |  |
| **Establecer equipo necesario para desarrollo de software** | | | |
| **Identificar posibles candidatos** | | | |
| **Verificar disponibilidad de los candidatos a cargos** | | | |
| **Contratación de personal** | | | |
| **Comentarios** | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

* Formulario para caída de servicios

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO CAÍDA SERVICIOS “001” | | | |
| **Encargado entrega** | : | Cristian De la Rivera |  |
| **Fecha** | : | 11/10/2012 |  |
| **Hora** | : | 16:00 PM |  |
| **Identificar servicio con malfuncionamiento** | | | |
| **Realizar operaciones correctivas** | | | |
| **Verificar servicios anexos que pudieron fallar** | | | |
| **Comentarios** | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

* Formulario requerimientos no viables

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO REQUERIMIENTOS NO VIABLES “001” | | | |
| **Encargado entrega** | : | Álvaro Salas |  |
| **Fecha** | : | 11/10/2012 |  |
| **Hora** | : | 14:00 PM |  |
| **Identificar requerimientos no viables** | | | |
| **Realizar operaciones correctivas del requerimiento** | | | |
| **Establecer requerimiento modificado fecha y hora**  Fecha: \_\_\_\_\_\_\_\_\_\_ Hora: \_\_\_\_\_\_\_\_\_\_ | | | |
| **Comentarios** | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

* Formulario error en la planificación

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO ERROR PLANIFICACIÓN “001” | | | |
| **Encargado entrega** | : | Roberto Andrade |  |
| **Fecha** | : | 11/10/2012 |  |
| **Hora** | : | 14:00 PM |  |
| **Identificar problemas dentro de la planificación** | | | |
| **Registrar contingencia** | | | |
| **Establecer operación correctiva** | | | |
| **Establecer acción correctiva que se lleva a cabo (fecha/hora)**  **Fecha: \_\_\_\_\_\_\_\_\_\_ Hora: \_\_\_\_\_\_\_\_\_\_** | | | |
| **Comentarios** | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

* El sistema requiere más esfuerzo del presupuestado

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO ESFUERZO MAYOR AL PRESUPUESTADO “001” | | | |
| **Encargado entrega** | : | Miguel Montero |  |
| **Fecha** | : | 11/10/2012 |  |
| **Hora** | : | 14:00 PM |  |
| **Evaluar tareas que se realizaran** | | | |
| **Delimitar las tareas que se llevaran a cabo** | | | |
| **Establecer cuales tareas no se realizaran** | | | |
| **Realizar acciones correctivas en la documentación y planificación temporal.** | | | |
| **Registrar fecha en la que se le dará a conocer la noticia al cliente**  **Fecha: \_\_\_\_\_\_\_\_\_** | | | |
| **Comentarios** | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

* Cambio de requerimiento

|  |  |  |  |
| --- | --- | --- | --- |
| FORMULARIO CAMBIO REQUERIMIENTO “001” | | | |
| **Encargado entrega** | : | Miguel Montero |  |
| **Fecha** | : | 11/10/2012 |  |
| **Hora** | : | 14:00 PM |  |
| **Evaluar requerimientos que se modificaran** | | | |
| **Establecer la razón del cambio** | | | |
| **Registrar datos a cambiar dentro del desarrollo** | | | |
| **Evaluar factibilidad de los cambios e impacto dentro del proyecto** | | | |
| **Registrar fecha de la decisión**  **Fecha: \_\_\_\_\_\_\_\_\_** | | | |
| **Comentarios** | | | |
| **FIRMA(s)** | | | |
|  |  |  |  |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Encargado entrega |  | Jefe proyecto | Timbre recepción |

**Detalle roles profesionales**

**Programador**: El programador se encargará del desarrollar el sistema. Además del desarrollo general, es responsable de tareas asociadas directamente a la operación de técnicas y tecnologías que se utilizarán en el proyecto, como los respaldos y configuraciones de software. Los programadores trabajan en conjunto y según objetivos dispuestos por las necesidades que identifique el jefe de proyecto a partir de la información que posee.

**Tester**: El rol de tester tiene la responsabilidad de probar el sistema en búsqueda de fallas y vulnerabilidades, poniendo a prueba los distintos elementos del sistema para comprobar su calidad en temas como comprobaciones y manejo de excepciones. En el caso de la propuesta presente, el tester realizará pruebas de tipo *white box*, pues este rol es realizado por personal que conoce los detalles programáticos del software.

**Jefe de proyecto:** Es quien dirige el proyecto, se encarga de llevar a cabo la planificación y organización de cada una de las etapas del proyecto. El jefe de proyecto es responsable del equipo de trabajo, considerando las capacidades de cada miembro dentro del marco de plazos y necesidades que tiene el desarrollo. Es su responsabilidad mantener al personal motivado y checkear el cumplimiento de todas las reglas de planificación que estén disponibles.

**SQA**: Es quien brinda a la administración la seguridad de que los procesos establecidos están siendo implementados correctamente. El rol de SQA debe velar que se cumplan los métodos correctos para el desarrollo, según los estándares definidos adecuadamente para cada caso. Esta labor apunta a dar valor al desarrollo del proyecto y a reducir los riesgos asociados a este, aspectos que el SQA debe tener en consideración al ejercer sus labores. Cabe mencionar que este individuo será externo al equipo de trabajo que desarrolle el Software.

**Analista**: El rol de analista es quien se encarga de tomar los requerimientos para desarrollar el sistema. Debe mantener una buena comunicación con el cliente y es su responsabilidad aplicar correctamente las herramientas y técnicas relacionadas a identificar de forma útil las necesidades del cliente. El analista debe entender el enfoque que se debe dar al proyecto y como se estructura cada necesidad en una arquitectura coherente, sobre la que se basará el desarrollo del proyecto.

En el proyecto a desarrollar, considerando su tamaño limitado y personal específico, varios de estos roles serán cumplidos por un miembro del equipo, en algunos casos compartiendo las tareas de un rol específico entre el equipo, como es el caso del Jefe de proyecto que también asumirá el cargo de analista programador.

**Estándares – Clasificación de requerimientos**

Los requerimientos se muestran con un nombre descriptivo, una clasificación por tipo y una clasificación importancia. El tipo del requerimiento puede ser Funcional o No Funcional.

Un requerimiento funcional responde a necesidades internas, que forman parte de la lógica con la que debe ser construida el software. Éstas responden a la lógica de negocios de la empresa a la que se le crea la solución. Por otro lado, los requerimientos No Funcionales responden a requerimientos no relacionados con la lógica interna, que pueden ser accesorios o cosméticos pero aun así son requeridos por el cliente. El software podría solucionar los mismos problemas sin ellos, pero no podrían adecuarse completamente a las necesidades de la empresa.

Adicionalmente, la clasificación por Importancia puede ser Alta, Medio o Baja. Esta clasificación se centra en la influencia que tiene el requerimiento para cumplir los objetivos planteados. Una importancia Alta representa una función necesaria para el funcionamiento integral del software para responder a todas o la mayoría de las necesidades de la empresa. Una importancia media significa un trastorno moderado a los objetivos específicos, pero que no significan una incapacidad total de responder al objetivo general. Mientras que una prioridad Baja no trastorna de forma significa el cumplimiento de los objetivos, aunque si afecta un objetivo específico de alguna manera.

**Estándares – Plan de contingencia**

Para el plan de contingencia, se ha considerado el estándar ISO-31000(2009) Manejo de riesgos – principios y guías, que define conceptos asociados y entrega guías sobre la identificación y orientación que debe tener el manejo de riegos en una empresa.

Los estándares de forma para el formato de documentación de contingencia están basados en el documento *Planificación y gestión de sistemas de información; de la escuela superior de informática – Universidad Castilla-La mancha*. Disponible en:

<http://alarcos.inf-cr.uclm.es/doc/pgsi/doc/teo/7/pgsi-trans7.pdf>

Además, se ha considerado en la confección y estructuración del plan aspectos expuestos en el *libro Desarrollo y gestión de proyectos informáticos de Steve McConnell, capítulo 5: Gestión de riesgos*

La identificación de riegos considera un nombre descriptivo del riesgo, junto a su descripción que debe incluir su responsable y lo que implica para el desarrollo. Esta identificación clasifica los riesgos según su naturaleza lógica y física; para después clasificar la fase del desarrollo en la que se centran. Esta identificación es la pauta en la que se basa la priorización de los riesgos y su posterior plan de tratamiento. Esta identificación se presenta en una tipografía Lucida Sans, tamaño 9, con formato negrita en el título del riesgo y un texto en formato normal justificado para la descripción de éste.

La priorización de riesgos está basada en Impacto (el trastorno que ejercen sobre la planificación del proyecto) y la probabilidad (Que tan posible es que el riesgo se cumpla). Ambos se clasifican entre baja-media-alta; en donde:

**Impacto Bajo:** Supone un trastorno menor del desarrollo del proyecto que es solucionado dentro del mismo día en que surge el problema.

**Impacto Medio:** Trastorno mediano del desarrollo, en donde el problema se soluciona en más de un día, pero en un plazo menor a una semana.

**Impacto alto:** Trastorno grave del desarrollo, en donde afecta en gran medida el progreso de éste y su solución considera un tiempo mayor a una semana.

**Probabilidad baja:** Es muy poco probable que se presente el problema, estimándose que no debería ocurrir dentro del periodo que dura el desarrollo.

**Probabilidad media:** Existe una posibilidad real de que se presente el problema, donde se estima que éste se presentará por lo menos una vez dentro del periodo de desarrollo.

**Probabilidad alta**: Es esperable que este problema se presente, estimándose que este ocurra varias veces durante el desarrollo del proyecto.

En ambos casos se asigna un puntaje de 1, 2 o 3 en las opciones baja, media y alta; respectivamente. La multiplicación del puntaje de estos dos factores da lugar a la prioridad, en donde un mayor puntaje señala la mayor importancia y atención que se le debe dar a ese aspecto.

Esto está organizado en una tabla que considera el nombre del riesgo, la probabilidad e impacto de ese riesgo específico y una columna final con la multiplicación del valor de los dos factores mencionados.

La planificación de control de riesgos debe incluir el riesgo, el responsable de manejarlo y estar dividido en “antes” y “durante” para cada riesgo específico. El antes se refiere a medidas y acciones para evitar que la contingencia ocurra y el “durante” señala medidas para manejar el riesgo una vez que ocurra. Cuando ocurre un riesgo, se debe rellenar un formulario de manejo de contingencias (Disponible en anexo)

Para el seguimiento, se creará una lista con los riesgos considerados, las veces que se presentó en la semana, las veces que se presentó la semana pasada, las semanas consecutivas que se ha presentado y el estado actual de tratamiento. Esta lista se presenta en formato tabla y es responsabilidad del Jefe de proyecto actualizar su contenido semanalmente.