**INACAP TEMUCO**

INGENIERÍA EN INFORMÁTICA

**Taller Integral de Proyecto Informático**

Sistema Web para la Gestión de mantenimientos y garantías para proyectos post-implementación para la empresa Lazos

inacap

**Nombre Alumno (a): Tracy Solange Padilla Sierra**

**Luis Alberto Lizama Bustos**

**Juan Carlos Vallejos** **Obreque**

Nombre Profesor: Marisel Angélica Prado González

Sección: 075 Vespertino

Fecha: 21 de Octubre del 2013

Índice de contenido

[CAPÍTULO I INTRODUCCIÓN 4](#_Toc369914350)

[1.1. Introducción 5](#_Toc369914351)

[1.2. Definición del Problema 5](#_Toc369914352)

[CAPÍTULO II OBJETIVOS 6](#_Toc369914353)

[2.1. Título 6](#_Toc369914354)

[2.2. Objetivos 6](#_Toc369914355)

[2.2.1 Objetivo General 6](#_Toc369914356)

[2.2.2 Objetivos Específicos 6](#_Toc369914357)

[2.3. Alcance y Limitaciones 7](#_Toc369914358)

[2.3.1 Alcance 7](#_Toc369914359)

[2.3.1 Limitaciones 8](#_Toc369914360)

[CAPÍTULO III MARCO TEÓRICO 9](#_Toc369914361)

[3.1. Fundamentos Teóricos 9](#_Toc369914362)

[3.2. Antecedentes del Proyecto de SW 12](#_Toc369914363)

[3.3. Desarrollo del Problema 16](#_Toc369914364)

[3.4. Arquitectura 19](#_Toc369914365)

[3.5. Lenguajes de Programación 23](#_Toc369914366)

[CAPÍTULO IV METODOLOGÍA DEL DESARROLLO 26](#_Toc369914367)

[4.1. Estudio de Factibilidad 27](#_Toc369914368)

[4.1.1 Alternativas de Solución 27](#_Toc369914369)

[4.1.2 Evaluación de Alternativas 28](#_Toc369914370)

[4.1.3 Selección de Alternativa de Solución 38](#_Toc369914371)

[4.2. Planificación de la solución (Carta Gantt) 38](#_Toc369914372)

[4.3. Desarrollo del Requerimiento 38](#_Toc369914373)

[4.3.1 Definición de Requerimientos 38](#_Toc369914374)

[4.3.2 Análisis del Requerimiento 38](#_Toc369914375)

[4.4. Diseño Lógico 38](#_Toc369914376)

[4.4.1 Modelo de Casos de Uso 38](#_Toc369914377)

[4.4.2 Definición de Actores 38](#_Toc369914378)

[4.4.3 Definición de Casos de Uso 39](#_Toc369914379)

[4.4.4 Descripción de Casos de Uso 39](#_Toc369914380)

[4.4.5 Grafico de Casos de Uso 39](#_Toc369914381)

[4.4.6 Diagrama de Secuencia 39](#_Toc369914382)

[4.5. Diseño Físico 39](#_Toc369914383)

[4.5.1 Diseño Arquitectónico 39](#_Toc369914384)

[4.5.2 Definición del Modelo de Datos 39](#_Toc369914385)

[4.5.3 Diseño de la Interfaz 39](#_Toc369914386)

[4.6. Implementación 39](#_Toc369914387)

[CAPÍTULO V CONCLUSIONES 40](#_Toc369914388)

[5.1. Bibliografía 40](#_Toc369914389)

Índice de Tablas

[Tabla 1 - Recurso Humano 30](#_Toc369914337)

[Tabla 2 - Recursos Consumibles 30](#_Toc369914338)

[Tabla 3 - Recursos Teconológicos 31](#_Toc369914339)

[Tabla 4 - Recursos Teconológicos Depreciación 32](#_Toc369914340)

[Tabla 5 - Recursos de Operación 33](#_Toc369914341)

[Tabla 6 - Resumen del presupuesto del proyecto 34](#_Toc369914342)

[Tabla 7 - Costos del proyecto 35](#_Toc369914343)

[Tabla 8 39](#_Toc369914344)

Índice de Figuras

[Ilustración 1 – Empresa Lazos 10](#_Toc369914345)

[Ilustración 2 - Arquitectura 18](#_Toc369914346)

[Ilustración 3 - Capas fisica 20](#_Toc369914347)

[Ilustración 4 - Arquitectura logica 21](#_Toc369914348)

[Ilustración 5 - Diseño en cascada 26](#_Toc369914349)

# CAPÍTULO I INTRODUCCIÓN

## Introducción

La empresa Lazos S.A. Solicita un sistema para la gestión de la post-entrega de sus sistemas desarrollados. Actualmente existe un sistema de ticket para la atención al cliente, pero no un sistema que cubra la necesidad tanto para la empresa ni para sus propios clientes.

## Definición del Problema

La idea es implementar sistema que cumpla con los estándares de escalabilidad de la empresa, ya que para ésta puede llegar a ser un gran problema la gestión de la post-implementación de sus proyectos, la información que se encuentra hoy en día se encuentra muy dispersa, porque cada jefe de proyecto maneja la post-entrega de los sistemas implementados, y el medio de comunicación para la gestión de la post-entrega es sólo es por correo electrónico o llamadas telefonicas, lo cual provoca una desorganización en la gestión de los proyectos entregados.

# CAPÍTULO II OBJETIVOS

## Título

Sistema Web para la Gestión de mantenimientos y garantías para proyectos post-implementación para la empresa Lazos.

## Objetivos

### 2.2.1 Objetivo General

Crear un sistema Web para efectuar la gestión de la post-entrega de sistemas desarrollados por la empresa Lazos.

### 2.2.2 Objetivos Específicos

* Establecer un modelo de datos escalable para las funcionalidades del sistema.
* Categorizar la centralización de las solicitudes del cliente estableciendo tipos y estados para el ciclo de vida de las solicitudes.
* Registrar la información de la gestión histórica de los proyectos guardando los distintos cambios que va sufriendo la solicitud en el tiempo.
* Indicar vía correo electrónico a los usuarios de los eventos que se producen en el sistema, utilizando el registro de los usuarios que se encuentren registrados en éste.
* Representar la información estadísticamente para ayudar a la toma de decisiones.

## Alcance y Limitaciones

### 2.3.1 Alcance

Inicialmente el proyecto comenzará con la toma de requerimientos, ésta se procesará previó al análisis y desarrollo de este sistema. Se realizarán reuniones con la contraparte (Lazos) y nuestro equipo de trabajo, las cuales quedarán formalizadas según se especifique en la carta Gantt.

A continuación se solicitará la información y/o documentación necesaria para la implementación y documentación de las maquetas, las cuales son necesarias para definir el alcance de los componentes visuales que conformarán el sistema. Durante la ejecución del proceso de análisis, el cliente podrá evaluar y comentar las maquetas para poder analizar y comprender qué es realmente lo que quiere y necesita.

Al momento de tener las maquetas aprobadas por el cliente, se pasará a la siguiente etapa, que será la implementación del documento maqueta. Este documento será el primer entregable a cliente. Cuando el cliente apruebe el documento se pasará a la siguiente etapa, la cual será comenzar con la confección del documento de análisis o diseño funcional del sistema (UML, flujos, etc.), el cual será el segundo entregable, y posterior a su confección se comenzará con el desarrollo de los componentes gráficos para la implementación de este sistema.

Posteriormente se construirá un modelo de datos para la implantación de este sistema correspondiente al tercer entregable al cliente. Dicho modelo debe cumplir con la nomenclatura estándar que utilice la contraparte para sus sistemas, la cual deberá entregar con anticipación, de lo contrario se definirá una interna.

Una vez finalizado el desarrollo de los tres primeros entregables, se generarán planes de pruebas para poder evaluar que lo implementado realmente cumpla con las funcionalidades solicitadas por el cliente. Estos serán el cuarto entregable al cliente.

Posteriormente al desarrollo del sistema se generarán manuales de usuarios (definidos por perfiles), y también se entregará el código fuente del sistema. Éstos serán el quinto y sexto entregable a cliente.

Finalmente al cierre del proyecto con la contraparte se capacitará a todo el personal que utilizará el sistema, para que los usuarios tengan los conocimientos necesarios al momento de manipular el sistema.

### 2.3.1 Limitaciones

* Cualquier requerimiento posterior a la toma de requerimientos no será considerado dentro del alcance de este sistema.
* Las funcionalidades en esta etapa del proyecto no se integrarán con otros sistemas o módulos.
* No se incluirá mantenimiento para las funcionalidades del sistema, sólo implementación.

# CAPÍTULO III MARCO TEÓRICO

## Fundamentos Teóricos

**Empresa Lazos**

La marca Lazos nace en 1997 como respuesta a los altos requerimientos tecnológicos en servicios de conectividad y comunicaciones, en particular en la implantación y ejecución del proyecto Enlaces. Lazos fue pionero (1997 - 1999) en ofrecer comercialmente en la IX Región el servicio de acceso a Internet conmutado (ISP). En el 2009 se decide constituir Lazos S.A. como empresa privada independiente manteniendo la participación accionaria principal. El 2011, como prueba de su madurez, se adjudica dos convenios marcos para vender sus servicios al Gobierno de Chile [[1]](#footnote-1):

* Convenio marco de desarrollo de software
* Convenio marco de webhosting

**Misión**

Mejorar la eficiencia y eficacia de los procesos de gestión y comunicación de las organizaciones, a través del desarrollo, operación y mantención de servicios de software Web, con fuerte orientación a la satisfacción del cliente.

**Objetivos**

* Diseñar y construir soluciones innovadoras para el mejoramiento de la gestión.
* Buscar altos niveles de satisfacción de sus clientes.
* Mejorar la rentabilidad de las organizaciones.
* Fortalecer las capacidades de gestión de sus clientes.

**Cargos**

* Gerente.
* Gerente de Proyectos.
* Jefe de Operaciones.
* Jefe de Proyecto.
* Jefe de Desarrollo.
* Ingeniera QA.
* Administrador de Bases de Datos.
* Webmaster.
* Ingeniero de Software.
* Administrador de Sistemas y Redes.
* Diseñador de Software.
* Ejecutiva Soporte Técnico.
* Ejecutivo Soporte Remoto.
* Asistente Administrativo.



Ilustración 1 – Empresa Lazos

## Antecedentes del Proyecto de SW

Debido a la importancia que se genera en la atención al cliente y la organización de la información, es necesario mejorar las condiciones en que la empresa en cuanto a gestión de sus proyectos.

Hoy en día en el mercado existen varios tipos de sistemas que realizan procesos y tareas similares a las que requiere la empresa, pero no las realizan en su totalidad como la empresa lo requiere.

A continuación se detallan algunos de los sistemas que existen en el mercado:

**SAAS barlotask**

Es un sistema que permite rápida y ágilmente gestionar las tareas de su personal, asignado prioridades, responsables, fechas límites, entre otros. Por Internet y desde cualquier lugar[[2]](#footnote-2).

Este sistema permite:

* Dar finalizada una nueva Tarea
* Mantenerla en estado borrador hasta “cerrarla”
* Adjuntar documentos
* Verificar Tareas anteriores
* Perfiles:
  + Colaborador
  + Gerente
  + Administrador/Control
* Avisos por mail de las nuevas tareas Asignadas
* Listado de Tareas Pendientes, ordenadas por prioridad o fecha

Ventajas del Sistema Workflow de Gestión de Tareas

* Mejora la asignación y Control de Tareas
* Mejora los tiempos de las presentaciones
* Centralización de la Información
* Descentralización de la actualización
* Rápida puesta en marcha

**OsTicket**

Es un sistema automatizado de soporte al cliente, que integra discretamente todos los tickets creados vía email o por formulario web dentro de una interface web simple[[3]](#footnote-3):

* Administra, organiza y archiva todas las solicitudes de soporte.
* Los clientes, al abrir una consulta recibirán un e-mail de auto-respuesta.
* Los clientes pueden ver el estado de los tickets que han abierto y su historial en línea, utilizando su número de consulta.
* OsTicket es una aplicación de código abierto escrita principalmente usando el lenguaje de programación PHP.
* Es gratis

**Mantis Bug Tracker**

Es un software que constituye una solución muy completa para gestionar tareas en un equipo de trabajo. Es una aplicación OpenSource hecha en php y mysql, destaca por su facilidad y flexibilidad de instalar y configurar. Esta aplicación se utiliza para testear soluciones, hacer un registro histórico de alteraciones y gestionar equipos remotamente[[4]](#footnote-4).

Una de las características de Mantis es que cuenta con una gran cantidad de posibilidades para su configuración:

* Permitir configurar la transición de estados (abierto, encaminado, testear, devuelto, cerrado, reabierto...).
* Puede especificar un número indeterminado de estados para cada tarea (abierta, encaminada, testear, devuelta, cerrada, reabierta…).
* Permite introducir diferentes perfiles (programador, tester, coordinador, visualizador…).

Mantis incluye filtros, un sistema de búsqueda, tiene soporte para varios idiomas y también informa por e-mail la resolución de los errores de los tickets que se han informado.

**Projects and teamwork**

Es un sistema que da solución a la gestión de proyectos, tanto en entornos distribuidos como centralizados, que requieren de escalabilidad y flexibilidad, también incluye una nube. Puede coordinar todo su equipo, planificar los proyectos, hacer seguimiento, establecer prioridades y aumentar su productividad, con una solución que se adapta a cada organización y no requiere largos periodos de formación o configuración[[5]](#footnote-5).

Algunas características incluidas:

* Comunicación
* Editor Gantt avanzado
* Importación desde MS Project
* Equipo de Proyecto
* Multi-idioma
* Multi-moneda
* Estimación de esfuerzo
* Seguimiento de las tareas
* Coste de los recursos
* Estructura Organizativa
* Procesos de Negocio
* Seguimiento de Proyectos

En nuestra investigación no se encontraron antecedentes de que exista un sistema que cumpla en su totalidad con los requisitos o funciones que la empresa necesita, ya que nuestro sistema estará acorde a los requerimientos solicitados, cumpliendo la función de dos sistemas en uno solo, ya que en el mercado se encuentran por separado: La gestión del cliente o soporte para el cliente y la gestión de tareas para el equipo de trabajo.

Los sistemas mencionados anteriormente son una opción que existe, pero sus valores son altamente elevados, populares que son utilizados por organizaciones grandes y están fuera del presupuesto de la empresa. A excepción de osticket ya que es completamente gratuito.

Pero la diferencia que existe entre los sistemas expuestos anteriormente en relación con nuestro sistema son las siguientes:

* + Está hecho a medida.
  + Las funciones están directamente relacionadas a las tareas que día a día realiza la empresa.
  + Los procesos son solamente los que la empresa solicita, a diferencia de los demás sistemas que se enfocan en un entorno general para organizaciones con funciones comunes, ya que para Lazos no son necesarias, como por ejemplo: gestiones para un gerente, o entidades externas.
  + Costos elevados por el prestigio y las funcionalidades que integra el sistema.

## Desarrollo del Problema

La empresa Lazos, dentro de su ámbito de negocio, su principal componente es el desarrollo de software, en el cual entregan sistemas a distintos clientes para su implementación. Una vez que el ciclo de desarrollo concluye comienza un periodo denominado “Garantía”, en el cual los clientes pueden reportar errores que presente el sistema, o pueden presentar nuevos requerimientos para agregarle a éste.

Dicho levantamiento de requerimientos o reporte de errores, actualmente para los sistemas que se encuentran en periodo de garantía, o que ya han sido implementados, se realiza por medios tales como conversaciones telefónicas o envío de correos electrónicos. Por lo cual, no queda respaldo centralizado de dicha información de manera que todo el personal interesado de la empresa esté informado, ni tampoco dicha información queda disponible para rápido acceso en el futuro.

Considerando lo anterior, el problema consiste en no tener un sistema que gestione el flujo de trabajo posterior a la implementación de los sistemas desarrollados por Lazos. Por lo tanto se propone crear un sistema acorde a la problemática detectada, en el cual se deben incluir las siguientes funcionalidades agrupadas por perfiles:

**Perfil Administrador:**

* Ingresar proyectos.
* Crear a los usuarios y clientes.
* Aprobar o rechazar las solicitudes de los clientes.
* Asignar una tarea (se puede asignar a otro administrador o desarrollador).
* Puede desempeñar las mismas acciones del perfil desarrollador.

**Perfil Encargado de las solicitudes:**

* Ingresar proyectos.
* Aprobar o rechazar las solicitudes de los clientes.
* Asignar una tarea (se puede asignar a otro encargado o desarrollador).
* Aprobar o rechazar la resolución de solicitudes.
* Puede desempeñar las mismas acciones del perfil desarrollador.

**Perfil Desarrollador:**

* Revisar la solicitud y su historico.
* Resolver y cambiar de estado.
* Adjuntar archivo y comentarios.

**Funcionalidades Generales:**

* Notificación vía correo electrónico de los cambios de estados a los usuarios asignados.
* Súper usuario puede crear usuarios internos como externos y puede crear proyectos.
* Todos los usuarios deben agregar un comentario al cambiar de estado.

Considerando la separación lógica agrupada por funcionalidades, las cuales responden a los actores involucrados internamente en el negocio de la empresa, es que se pretende tener un sistema que gestione el flujo de trabajo posterior a la implementación de los sistemas desarrollados por Lazos.

En este sistema existirá un registro histórico de la información de los proyectos, utilizará flujos de trabajo para poder saber en tiempo real y de manera centralizada el estado de las solicitudes y errores reportados para los proyectos que se encuentran en garantía.

## Arquitectura

Considerando las necesidades y la naturaleza Web de la solución propuesta, es que en términos de arquitectura, el desarrollo del software se basará en el paradigma cliente servidor, más específicamente en la lógica por capas.



Ilustración 2 - Arquitectura

La arquitectura cliente-servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un sólo programa. Los tipos específicos de servidores incluyen los servidores web, los servidores de archivo, los servidores del correo, etc. Mientras que sus propósitos varían de unos servicios a otros, la arquitectura básica seguirá siendo la misma.

Una disposición muy común son los sistemas multicapa en los que el servidor se descompone en diferentes programas que pueden ser ejecutados por diferentes computadoras aumentando así el grado de distribución del sistema.

En términos físicos la arquitectura se dividirá en tres capas, las cuales son: la capa del cliente o capa navegador, la cual representa todo lo que el usuario visualiza. La segunda capa corresponde al servidor Web, en el cual se ejecuta la lógica de negocio y se procesan las peticiones del cliente. Y la tercera capa corresponde al servidor de base de datos, donde se guardan los datos de manera persistente.



Ilustración 3 - Capas fisica

En términos lógicos la aplicación se separa en cuatro capas, las cuales son: la capa de presentación, que corresponde a todo lo que visualiza el usuario, es decir las vistas. La capa de negocio, en la cual se ejecuta toda la lógica especifica del sistema, como por ejemplo las validaciones y reglas del negocio internas de la empresa. La capa de acceso a datos, en la cual se encapsulan las llamadas y transacciones que se realizan con el servidor de base de datos. Y finalmente la capa de entidades de negocio, en la cual se definen los objetos comunes entre las tres primeras capas para transportar la información entre estas de manera coherente.



Ilustración 4 - Arquitectura logica

## Lenguajes de Programación

A continuación se describen los lenguajes de programación más populares o accesibles para desarrollar el software para la solución propuesta, de manera que se pueda elegir uno considerando sus características de acuerdo a las necesidades del proyecto.

**PHP**

PHP es un lenguaje que nació orientado a la programación web dinámica, se utiliza principalmente en aplicaciones de servidor, y el código puede ser también incrustado en contenido HTML. PHP es un lenguaje ampliamente utilizado en el desarrollo de aplicaciones Web debido a su naturaleza basada en software libre[[6]](#footnote-6).

Su sintaxis se asemeja a la de los lenguajes C, Java y Perl. La incrustación de código dentro del contenido HTML se realiza al utilizar las etiquetas <?php (código) ?>, es aquí donde se definen las clases en el caso de la orientación a objetos que implementa PHP desde las últimas versiones.

**.NET**

Microsoft.net es una plataforma de desarrollo y ejecución de aplicaciones que se incluye en Windows Server 2008, Windows Vista y Windows 7, o se puede instalar en Windows Xp, Windows Server 2003, y sus principales características son:

* Las aplicaciones de .NET se ejecutan en un entorno aislado del sistema operativo denominado runtime, característica que las hace ser portables.
* Es orientado a objetos y multi lenguaje.
* Esta orientado al desarrollo de aplicaciones empresariales complejas.
* Proporciona un modelo de desarrollo para distintos dispositivos de hadware (pc, tablets, etc.) y software (Windows, Web, consola, móviles, etc.).
* Se puede integrar con aplicaciones desarrolladas en COM y viceversa
* Integra compatibilidad con otros sistemas operativos al implementar estándares como xml, soap, wsdl, etc.

La plataforma de .NET está formada principalmente por:

* El Framework, que es el componente sobre el cual se desarrollan y ejecutan las aplicaciones.
* El runtime, que se denomina CLR (common language runtime) cuya función es administrar memoria, aislar el funcionamiento de las aplicaciones, brindar seguridad en la ejecución de estas y generar código nativo o compilado.
* El BLC (biblioteca de clase base) que proporciona todas las clases necesarias para implementar las funcionalidades mas comunes.
* Lenguajes de programación y compiladores, ya que .NET es multilenguaje, los lenguajes deben seguir un estándar llamado CLS (common language specification) y así son compatible entre si, ejemplos: visual basic.net, c-sharp, net.cobol, delphi, c++, python, etc.
* Herramientas y documentación, .NET proporciona utilidades para documentación y otras tareas de forma mas practica.

Actualmente .NET soporta desarrollo de aplicaciones Web montadas en servidores IIS utilizando la arquitectura ASP.NET, en la cual se puede desarrollar en los lenguajes C# o VB. Las aplicaciones .NET no están basadas en software libre, por lo que se deben costear las licencias de los servidores y otros componentes[[7]](#footnote-7).

**JAVA**

Java es un lenguaje orientado a objetos desarrollado por Sun Microsystems en los 90, nace principalmente del lenguaje C y C++, pero simplifica el modelo de objetos, tiene como filosofía la reutilización de código y le quita la responsabilidad al programador de manejar la memoria a través de punteros, ya que java discrimina cuando usar el paso por referencia o por valor, así como también el programador ya no debe escribir los destructores porque java libera memoria de forma automática con el garbage collector.

Las características de Java es que sus aplicaciones están compiladas en un bytecode el cual corre en una máquina virtual específica para una plataforma, por lo cual el código fuente será el mismo independiente del sistema en el que se ejecute lo que lo hace ser portable. Java también soporta programación estructurada al implementar el método main() como en C, y soporta de la orientación a objetos también las clases, polimorfismo, etc. pero a diferencia de C++ no existe la herencia múltiple, lo cual se trata de contrarrestar con la implementación de las interfaces. También en Java los tipos de datos primitivos existen al igual que en C y también existen sus equivalentes en objeto con métodos y propiedades asociadas que heredan de Object.

Java al tener estas características de automatización como el no manejar directamente la memoria, el garbage collector, más la máquina virtual, hace que sea un lenguaje menos eficiente en términos de velocidad y recursos, pero más potente en el desarrollo de aplicaciones empresariales complejas.

Actualmente Java es muy utilizado en aplicaciones Web, siendo estas desarrolladas bajo distintos Frameworks existentes como por ejemplo Servlets, JSP, Struts, o Spring. Java no está basado en código libre y se deben costear las licencias implicadas[[8]](#footnote-8).

# CAPÍTULO IV METODOLOGÍA DEL DESARROLLO

Considerando que el alcance del desarrollo del software para la solución es limitado, es que se optará por la metodología de desarrollo en cascada. Esta metodología de desarrollo es útil cuando los requerimientos son fijos, ya que el avance del desarrollo será de manera lineal.

El paradigma del modelo en cascada determina un enfoque sistemático y ordenado, secuencial, donde se inicia con la especificación de los requerimientos, luego la planificación y análisis, modelado, construcción y finalmente el despliegue del sistema con su periodo de pruebas y soporte.

Un inconveniente importante en este paradigma es que el cliente no puede visualizar o tener una idea de la funcionalidad real del sistema hasta que esté está terminado, desventaja la cual pretendemos suplir con un riguroso análisis inicial de la toma de requerimiento y confección de las maquetas del sistema.



Ilustración 5 - Diseño en cascada

## Estudio de Factibilidad

### 4.1.1 Alternativas de Solución

Considerando los requisitos a cubrir por el sistema, en base a lo solicitado por la empresa Lazos, es que las alternativas de solución a analizar las diferenciaremos en dos opciones. La primera, optar por una alternativa de software que se pueda adquirir en el mercado. La segunda, optar por desarrollar el software a medida.

**Software existente**

Para esta alternativa de solución se deberá tener en consideración los productos ya existentes en el mercado actual descritos en el marco teórico, tener en cuenta sus características, costos, y si cumplen o no con suplir los requerimientos que abarca el sistema.

**Software a medida**

Para esta alternativa de solución se deberá tener en consideración los lenguajes de programación descritos en el marco teórico, y la metodología de desarrollo de software descrito en la introducción de este capítulo.

### 4.1.2 Evaluación de Alternativas

A continuación se describe la evaluación y especificación de las dos alternativas descritas anteriormente para la solución, considerando los aspectos técnicos, económicos y operacionales de cada una, para finalmente realizar una comparación.

#### 4.1.2.1 Factibilidad Técnica

La factibilidad técnica tiene por objetivos evaluar si se cuenta con los recursos, conocimientos, herramientas, experiencia y el personal para llevar a cabo el diseño, implementación y operación de la soluciones propuestas.

Evaluando la factibilidad técnica de la primera alternativa (Software existente) encontramos que tecnicamente es más complejo para nuestro equipo de trabajo analizar software de terceros, lo que incrementaria los tiempos de desarrollo si se desea modificar por ejemplo el sistema OsTicket. En cambio si se opta por un sistema de código cerrado o de pago no existirian incovenientes tecnicos ya que el desarrollo sólo se limitaria a instalar y configurar dichas soluciones.

Evaluando la factibilidad técnica de la segunda alternativa (Software a medida) encontramos que tecnicamente es posible llevar a cabo el desarrollo del sistema, ya que nuestro equipo de trabajo tiene experiencia desarrollando sistemas en código libre y código licenciado.

También es importante destacar que cliente final, es decir, la empresa Lazos se dedica al desarrollo de software, por lo que ellos mismos también poseen el conocimiento técnico para posteriormente mantener el sistema.

Respecto al hardware, como se ha definido en el alcance del proyecto, ya estará definido por el cliente, por lo que no existirá impedimento de este tipo para la implementación de la primera o segunda solución.

Respecto a las tecnologías, actualmente los software de desarrollo o documentación asociada a las tecnologías son de disposición libre, y no existen impedimentos para manejarlos, ya que nuestro equipo de trabajo cuenta con los conocimientos en el manejo de herramientos de desarrollo.

#### 4.1.2.2 Factibilidad Económica

**4.1.2.2.1 Costos de Desarrollo del Proyecto**

Recurso Humano

En este estudio se incluye el costo del recurso humano para dar una aproximación más real del valor del proyecto.

La mano de obra es el recurso más importante y a la vez el más costoso que se ve involucrado en la realización de cualquier tipo de proyecto, esto no excluye a los proyectos de tipo informático, para determinar el salario mensual de cada uno de los desarrolladores se ha decidido tomar como referencia el equivalente al sueldo otorgado a los empleados de la empresa Lazos S.A. El cual se especificará a continuación:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Recurso | Cantidad | Meses | Salario Mensual ($) | Total ($) |
| Jefe de Proyecto | 1 |  |  |  |
| Programadores | 2 |  |  |  |
| Digitadores | 3 |  |  |  |
| Total | | | | $ |

Tabla 1 - Recurso Humano

Nota:

* 2 programadores que son los encargados de desarrollar la propuesta, lo que implica un costo total mensual por programador de $.
* 3 digitadores que serán los encargados de realizar la documentación y menuales del proyecto, cabe destacar que estos digitadores son el mismo personal encargado del desarrollo del proyecto, es decir, son los programadores y el jefe de proyecto. Lo que implica una remuneración por personal de $.

Recursos Consumibles

Los recursos consumibles que serán utilizados para la realización del proyecto los cuales se detallan a continuación, se estimaron en base a los requerimientos solicitados por parte de la empresa en base a proyectos realizados anteriormente y los precios se obtuvieron de cotizaciones realizadas.

|  |  |  |  |
| --- | --- | --- | --- |
| Detalle | Cantidad | Precio Unitario ($) | Total ($) |
| Fotocopias | 500 | 25 |  |
| Resmas de papel | 1 | 2.990 |  |
| Cartuchos de tinta de color | 1 | 12.990 |  |
| Cartucho de tinta negro | 2 | 9.990 |  |
| Gastos Varios |  |  |  |
| Total | | |  |

Tabla 2 - Recursos Consumibles

Nota:

* + Una cantidad aproximada de 500 fotocopias para lo que es la entrega de manuales informes que tiene un valor aproximado de $.
  + Una resmas de papel para la impresión del informe, documentos y anexos.
  + Cartuchos de tinta para la impresora de color y negro para realizar la impresión del informe, documentos y anexos.
  + Gastos varios incluye: anillado, carpetas para la entrega de manuales e informes, entre otros.

Recursos Teconológicos

Para el desarrollo del proyecto es necesario la adquisición de hardware que se detallará en el siguiente cuadro:

|  |  |  |  |
| --- | --- | --- | --- |
| Detalle | Cantidad | Precio Unitario ($) | Total ($) |
| Servidores (web y aplicaciones) | - | - | - |
| Notebook | 3 | $ 450.000 | $1.350.000 |
| Hosting | 1 | - | - |
| Dominio | 1 | $18.900 |  |
| Total | | |  |

Tabla 3 - Recursos Teconológicos

Nota:

* + Servidores (web y aplicaciones). La empresa cuenta con sus propios servidores web y aplicaciones con sus licencias respectivas, de esta forma se obtiene un ahorro significativo en este punto ya que son recursos altamente costosos para una implementacion propia los costos disminuyen al contratar servicios externos, pero no es necesario invertir en estos.
  + El hosting es otorgado por la misma empresa donde no se debe realizar inversión por parte de la implementación del proyecto.
  + El dominio tiene un valor aproximado de $18.900 por 2 años. Fue cotizado en ww.nic.cl

Los notebook que serán utilizadas para el desarrollo del proyecto se detalla en la factibilidad técnica, los valores correspondientes a la depreciación del equipo se pueden apreciar en el siguiente cuadro que muestra los costos de los recursos tecnológicos utilizados para el proyecto.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Detalle | Meses | Costos ($) | Total ($) | |
| Depreciación del Equipo | 4 | 45.000 | 135.000 | |
| Total | | | |  |

Tabla 4 - Recursos Teconológicos Depreciación

Nota:

* + Cabe destacar la depreciación del equipo a utilizar en el desarrollo de este proyecto es equivalente al 30% y nuestro proyecto se realizará en aproximadamente 4 meses, donde el total del valor de los equipos ($1.350.000) se debe multiplicar por el 30%, luego se debe dividir en 12 meses que es 1 año y el resultado se multiplica por los meses de uso del equipo que son 4 meses y nos da una depreciación total de los 3 equipos equivalente a $135.000.

Recursos de Operación

* + Energia Electrica

El Calculo de la energía eléctrica se realiza tomando en cuenta el consumo de energía utilizada por los equipos para el desarrollo del proyecto, cabe mencionar que la cantidad de energía consumida por cada uno de los equipos dependerá de las características propias de cada uno de ellos.

* + La siguiente tabla muestra un resumen de la energía aproximada que será utilizada en el desarrollo de este proyecto.

|  |  |  |  |
| --- | --- | --- | --- |
| Equipo | Cantidad | Consumo (watts) | Consumo Total (watts) |
| Notebook | 3 | 90 | 270 |
| Iluminación Ampolletas (Estación de trabajo) | 3 | 15 | 45 |
| Total |  |  | 315 |

Tabla 5 - Recursos de Operación

Convirtiendo a kilowatts: (315 w/h)/ 1000 = 0,315 kw/h

Consumo Mensual días hábiles: 0,315 x 4 Horas x 5 días x 4 semanas = 25,2 kw/h aproximados que se consumen al mes.

Consumo Mensual fin de semana: 0,315 x 8 Horas x 2 días x 4 semanas = 20,16 kw/h aproximados que se consumen al mes.

El total del consumo mensual aproximado es de = 45,36 kw/h al mes.

El costo del kilowatts por hora del servicio de energía eléctrica vigente hasta la fecha, para un consumo mensual menor a 220 kw/h es de $86,964.

Así: 45,36 kw/h \* $86,964 = $3944,68 de consumo mensual.

http://www.chilectra.cl/wps/wcm/connect/0a30ad0044100acc96999e65fe3686ef/Tarifas\_Suministro\_ClientesRegulados\_2013\_10\_01.pdf?MOD=AJPERES&Tipo=DOC

Finalmente en el estudio de factibilidad se presenta un cuadro resumen de los costos de desarrollo del proyecto en el que se consideraron los costos de Recurso Humano, Recursos Consumibles y Recursos Tecnológicos.

Tabla resumen del presupuesto del proyecto

|  |  |
| --- | --- |
| Recurso | Total ($) |
| Recurso Humano | $ |
| Recurso Consumible | $ |
| Recursos Tecnológicos | $ |
| Sub-total | $ |
|  |  |
| Total | $ |

Tabla 6 - Resumen del presupuesto del proyecto

Como se observa en la tabla anterior, el costo del desarrollo del proyecto asciende a la cantidad de $??????????????.

**4.1.2.2.2 Costos del Proyecto**

Se deben analizar los costos del proyecto con el fin de obtener un monto total de la inversión, de esta forma ver si el proyecto es rentable para la empresa.

La institución debe mantener una comunicación constante con sus clientes y en la actualidad sólo existe un sistema de ticket que genera principalmente una limitación en la comunicación y gestión entre la empresa y sus clientes. Este sistema pretende obtener principalmente los siguientes beneficios: Tener una mayor organización con la información de los proyectos, mantener registros históricos de las solicitudes del cliente estableciendo, tipos y estados en las solicitudes, representar la información estadísticamente para ayudar a la toma de decisiones.

El cuadro que se muestra a continuación son los recursos necesarios para llevar a cabo el desarrollo de este proyecto:

|  |  |  |  |
| --- | --- | --- | --- |
| Descripción | Cantidad | Precio Unitario ($) | Total ($) |
| Licencia de .NET |  |  |  |
| Licencia de SQL |  |  |  |
|  |  |  |  |
| Costo realización del proyecto |  |  |  |
|  |  |  |  |
|  |  |  |  |
| Total Costo del Proyecto | | | $ |

Tabla 7 - Costos del proyecto

La empresa Lazos S.A. Ya cuenta con varios de los bienes a utilizar para nuestro proyecto los cuales se detallaran a continuación:

* + A
  + A
  + A
  + A

Como hoy en día existen varios sistemas que realizan procesos similares, pero no en su totalidad, de igual forma los costos de estos son altamente elevados y no satisfacen las necesidades de la empresa. Nuestro proyecto es ampliamente favorable sobre las prioridades de la institución, por lo que consideramos que este proyecto es factiblemente económico y generara grandes beneficios.

#### 4.1.2.3 Factibilidad Operacional

/\*

Se refiere a todos aquellos recursos donde interviene algún tipo de actividad ( Procesos ), depende de los recursos humanos que participen durante la operación del proyecto. Durante esta etapa se identifican todas aquellas actividades que son necesarias para lograr el objetivo y se evalúa y determina todo lo necesario para llevarla a cabo.

- Operación garantizada.

- Uso garantizado.

Comprende una determinación de posibilidad que un nuevo sistema se use como se supone. Se deben considerar cuatro aspectos:

• La utilización de un nuevo sistema puede ser demasiado complejo para los usuarios de la organización o los operadores del sistema.

• Este nuevo sistema puede hacer que los usuarios se resistan a él como consecuencia de una técnica de trabajo, miedo a ser desplazado u otras razones.

• Un sistema nuevo puede introducir cambios demasiado rápidos que no permita al personal adaptarse a él y aceptarlo.

• La probabilidad de obsolescencia en el sistema. Cambios anticipados en la práctica o políticas administrativas pueden hacerse que un nuevo sistema sea obsoleto muy pronto.

\*/

Complejidad del sistema

Principalmente la idea principal es desarrollar un sistema con una interfaz amigable que sea facil y sencilla de usar.

Resistencia al cambio

Hoy en día los usuarios se resisten al cambio , no son tolerantes a usar sistemas nuevos prefieren seguir usando sistemas antiguos o seguir trabajando manualmente (planillas, excel, hojas, etc).

Adarpatación

Para que los usuarios se adapten facilmente al sistema se crearán manuales de usuarios y también se les capacitará a todo el personal lo cual tendremos total dispoción en resolver sus preguntas o dudas con respecto al sistema.

#### 4.1.2.4 Comparativa de las Alternativas

### 4.1.3 Selección de Alternativa de Solución

#### 4.1.3.1 Beneficios de la Solución

## Planificación de la solución (Carta Gantt)

## Desarrollo del Requerimiento

### 4.3.1 Definición de Requerimientos

### 4.3.2 Análisis del Requerimiento

#### 4.3.2.1 Requisitos Funcionales

#### 4.3.2.2 Requisitos No Funcionales

## Diseño Lógico

### 4.4.1 Modelo de Casos de Uso

### 4.4.2 Definición de Actores

### 4.4.3 Definición de Casos de Uso

### 4.4.4 Descripción de Casos de Uso

### 4.4.5 Grafico de Casos de Uso

### 4.4.6 Diagrama de Secuencia

## Diseño Físico

### 4.5.1 Diseño Arquitectónico

### 4.5.2 Definición del Modelo de Datos

#### 4.5.2.1 Modelo de Datos

#### 4.5.2.2 Diccionario de Datos

### 4.5.3 Diseño de la Interfaz

#### 4.5.3.1 Interfaz 1

#### 4.5.3.2 Interfaz 2

#### 4.5.3.n Interfaz n

## Implementación

# CAPÍTULO V CONCLUSIONES

|  |  |
| --- | --- |
|  |  |
|  |  |

Tabla 8

## Bibliografía

1. Recuperado en Octubre del 2013, de [www.lazos.cl](http://www.lazos.cl) [↑](#footnote-ref-1)
2. Recuperado en Octubre del 2013, de http://www.barloventotech.com.ar/detalle.php?a=sistema-de-gestion-de-tareas---workflow-saas&t=19&d=44 [↑](#footnote-ref-2)
3. Recuperado en Octubre del 2013, de http://osticket.com/ [↑](#footnote-ref-3)
4. Recuperado en Octubre del 2013, de http://www.mantisbt.org/ [↑](#footnote-ref-4)
5. Recuperado en Octtubre del 2013, de http://www.itmplatform.com/es/projects-and-teamwork/pt-presentacion [↑](#footnote-ref-5)
6. Recuperado en Octubre del 2013, de http://php.net/ [↑](#footnote-ref-6)
7. Recuperado en Octubre del 2013, de http://www.asp.net/ [↑](#footnote-ref-7)
8. Recuperado en Octubre del 2013, de http://www.oracle.com/es/products/index.html [↑](#footnote-ref-8)