r

Administración de Proyectos UCSE-DAR

Universidad Católica de Santiago del Estero

Alumno

Bruno Curti

Profesores

Marcela Vera

Juan Carlos Ramos

Fecha

Martes 4 de Julio, 2017

**Preámbulo**

El presente trabajo tiene como objetivo realizar las fases de iniciación y planificación para el proyecto CABL-Dev (Sistema de gestión de tareas de desarrollo) plasmando los conocimientos de la materia Administración de Proyectos.

Se ha elegido abordar una problemática real el cual será caso de estudio para el Proyecto Final de la carrera Ingeniería en Informática.

**Introducción**

CABL & Asociados se encuentra ubicado en la ciudad de Sunchales, provincia de Santa Fe, más específicamente en Avenida Belgrano 556.

La empresa se especializa en outsourcing, desempeñándose como una autentica fábrica de Software, cubriendo en sus trabajos desde la etapa de análisis hasta el servicio post-implementación, pasando por todas las etapas del ciclo del desarrollo de un sistema.

Las actividades que ofrece como parte del servicio de Outsourcing comprenden el desarrollo y mantenimiento de software, testing funcional e integral de aplicaciones y, especialmente, la Integración de aplicaciones con Servicios Web y Servicios de mensajería.

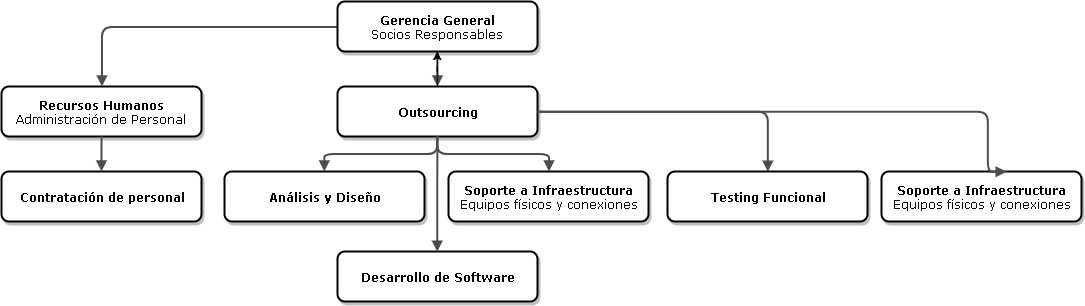
El principal potencial consiste en un equipo de Analistas Funcionales con experiencia en circuitos empresariales, junto a un grupo de desarrolladores capacitados en tecnologías Microsoft.

CABL & Asociados cuenta con Desarrolladores Seniors, Semi-Seniors y Juniors, Analistas de Procesos Seniors y Juniors y Líderes de Proyectos que pueden ser contratados para acoplarse al proyecto que el cliente demande.

El equipo de trabajo es versátil para trabajar con aplicaciones Web, Win-Forms y sobre todo se especializa en aplicaciones de integración (Web Services -WCF y asmx- y MSMQ).

CABL & Asociados SRL

**Organigrama**



**Business Case**

**Detalles del Proyecto**

**Organización:** CABL & Asociados SRL

**Proyecto:** CABL-Dev (Sistema de Gestión de Desarrollo)

****

**Patrocinador:** Carlos Blanche

**Director de proyecto:** Bruno Curti

**Fecha de Inicio:** 01/01/2017

**Fecha de finalización:** 01/06/2017

**Detalles del documento**

**Estado:** Aprobado **Versión:** 1.0

Rechazado

**Aprobado por: \_\_\_\_\_\_\_\_\_\_\_**

**Fecha: \_\_\_/\_\_\_/\_\_\_ Fecha actualización: \_\_\_/\_\_\_/\_\_\_**

**Comentarios:**

**Problemática**

En la actualidad, los miembros de la empresa gestionan las tareas de varias maneras. Algunos clientes para los cuales se hace outsourcing, ponen a disposición sus herramientas de gestión interna, por lo que los colaboradores que trabajan en dichos proyectos hacen uso de ellas, lo cual en la mayoría de los casos presenta bastantes limitaciones en cuanto a controles o exposición de trabajos. El relevamiento de cada actividad queda del lado del cliente, en éste caso.

Por su parte, hay otros desarrolladores abocados a diferentes proyectos quienes utilizan herramientas online como ser Trello ([www.trello.com](http://www.trello.com)) y Confluence [www.atlassian.com/software/confluence](http://www.atlassian.com/software/confluence)).

También algunos llevan un control personal bastante precarizado usando planillas de cálculo o apuntes en cuadernos hechos a mano por los mismos integrantes de la empresa, lo cual implica grandes posibilidades de error, o de extravió en el peor de los casos, haciendo de todo el proceso algo complejo y tedioso en muchas oportunidades.

Para cualquier caso, las historias de usuario o incidencias a resolver mediante tareas son siempre creadas por el mismo desarrollador. Esto es, por ejemplo, a partir de un dialogo con el cliente, ya sea cara a cara o por cualquier medio como ser Skype o el teléfono, el desarrollador persiste la necesidad planteada a resolver de alguna forma para dejar constancia formal de la misma.

La idea de este sistema también es corregir las formas de ese contacto, en el sentido de que el cliente tenga un formulario expuesto en la web en el cual solicite la revisión de algún error o plantee la necesidad de una nueva funcionalidad. De esta forma el desarrollador tiene una incidencia reportada por el cliente a partir de la cual puede hacer una estimación de esfuerzo y desglose de tareas, para responderle en primera instancia al cliente con un presupuesto y/o una solicitud de ampliar el detalle de la necesidad.

Hoy en día existen diversas herramientas para la gestión de tareas de desarrollo, algunas de ellas de acceso libre y otras pagas.

La página Trello, mencionada anteriormente permite gestionar el desarrollo por tareas, agrupadas en paneles de trabajo. Permite crear y ordenar tantos paneles como se quiera, las tareas disponen de un sistema de señalización para notificar de su prioridad e importancia. Se pueden adjuntar fotos, iconos, comentarios y además dividir la tarea en subtareas que según se completan van apareciendo en una línea porcentual hasta completarlas.

Como desventaja, no permite crear listas de trabajo repetitivas para poder hacer ciclos de tareas (Por ejemplo: X tareas cada lunes, o las tareas repetitivas de cada inicio de mes, que son de uso común en la empresa)

Otra herramienta disponible es Nozbe ([www.nozbe.com](http://www.nozbe.com)), una herramienta similar a Trello pero mucho más ágil y adecuada si se trabaja con un gran volumen de trabajo, es decir, con una enorme cantidad de elementos (tareas), ya que su motor de búsqueda/indexación maneja muy bien la gran cantidad de etiquetas que se pueden asignar a las tareas y subtareas que se definan. El motor de búsquedas de esta herramienta es su principal ventaja pensando a gran escala para cuando se cuente con un histórico de tareas muy grande por desarrollador, pero, como desventaja, el sistema no resulta muy intuitivo al principio y requiere conocerlo de manera completa para aprovecharlo al máximo, esto implica tiempos de capacitación y aprendizaje más prolongados que los necesarios en un sistema más dinámico.

Otra desventaja es su costo, actualmente unos 8€ aproximadamente por mes, y cabe mencionar que si bien existe, la versión gratuita es mucho más limitada en sus funcionalidades básicas y necesarias para la empresa como por ejemplo la generación de reportes.

Otra alternativa relevada es SeeNowDo (www.seenowdo.com), la cual ofrece una interfaz de usuario simple y sencilla para la creación y especificación de historias de usuario (a las cuales es posible asociarle tareas), así como también para la generación de sprints, y la correspondiente asignación de historias de usuario a sprints.

Si bien provee funcionalidad para estadísticas y generación de reportes, posee dos grandes limitaciones: por un lado, no existe el concepto de backlog. Todo debe desarrollarse dentro de sprints, lo cual hace difícil el manejo y gestión de cada historia, y la generación de reportes de rendimiento en un tiempo determinado por fechas. Por ejemplo, un reporte mensual de tareas.

Por otro lado, su performance no es buena. Aún con proyectos chicos la herramienta se vuelve lenta a medida que agregamos tareas a cada historia de usuario.

La idea de esta herramienta es más pensada a coordinar el día a día, y no la de establecer objetivos en el mediano-largo plazo, lo cual puede ser necesario en algunos proyectos.

**Marco teórico**

Es necesario conocer el vocabulario y los términos propios del tema a tratar en este proyecto. Como marco de trabajo, la empresa utiliza lo que conocemos como Scrum; cuya definición se asemeja a la definida en el contexto del rugby:

“Un scrum es un agrupamiento en Rugby, todos los integrantes del conjunto actúan para avanzar con la pelota”.

Dicha definición aplica al equipo de trabajo en la empresa.

En scrum se asume que:

* Los requerimientos cambiaran durante el proyecto.
* Se toma el cambio como una forma de entregar al final del desarrollo algo más cercano a la verdadera necesidad del Cliente.

Conceptos básicos:

**Product Backlog:** Es la acumulación de trabajos o son todos los trabajos, que han sido identificados, pero que no han sido completados, hace referencia al trabajo incompleto o hecho a medias y su unidad de medida son semanas.

Los trabajos están priorizados, y se los conoce también como historias de usuario.

**Sprint:** debe entenderse como un subconjunto de requerimientos, extraídos del Product Backlog, para ser ejecutadas durante un periodo definido por lo general entre 1 o 4 semanas.

Es la lista que descompone las funcionalidades del product backlog en las tareas necesarias para construir un incremento: una parte completa y operativa del producto.

**Incremento:** es la parte de producto producida en un sprint, y tiene como características que está completamente terminada y operativa: en condiciones de ser entregada al cliente final.

Cabe mencionar una de las grandes ventajas de la informática para las organizaciones, sobre todo para aquellas en constante crecimiento: una oficina sin papeles.

Si bien el papel es el soporte fundamental de múltiples actividades, la implementación de un sistema informático bajo el compromiso con el medio ambiente, contribuyen a la reducción de manera significativa del consumo de papel.

En contraparte, esto implica una serie de cambios y adaptaciones, no sólo tecnológicos, sino también culturales, administrativos y organizativos que se traducen en una mejora en la gestión y los procesos involucrados.

Así mismo, este sistema propone ser un centralizador de labor en el cual se implementa una base de datos común para registrar todas las tareas de cada desarrollador. Otra característica que hace que este registro de actividades sea centralizable, es que se trata de una aplicación web, expuesta en un dominio accesible desde cualquier lugar del planeta que tenga conexión a internet.

**Relevamiento**

A través de reuniones presenciales con los socios de la firma: Carlos Alberto Blanche, Pablo Angel Piovano y Maximiliano Astor, todos ellos ingenieros en informática, se fue obteniendo información sobre la forma de organización de las tareas del equipo de desarrollo, con el fin de armar una puesta en común y unificar la forma de organizar tareas mediante un proyecto con objetivos y alcances bien definidos.

Este proyecto buscara subsanar los problemas hoy presentes a la hora de priorizar, organizar, y exponer el esfuerzo dedicado en tareas de desarrollo, con el fin de que puedan integrarse otras funcionalidades como el contacto directo con los clientes a través de la plataforma expuesta en internet y un repositorio de código integrado, con la idea de asociar tareas de desarrollo a cambios en el código específicos para un control más exhaustivo.

**Objetivo general**

Siguiendo con los objetivos generales de la empresa en búsqueda de una mejora constante, es fundamental contar con una herramienta que administre el ciclo de vida de las tareas de cada colaborador, las cuales tendrán origen a partir de un solicitud directa del cliente.

El objetivo principal es diseñar un software que ayude al equipo de desarrollo a gestionar sus tareas diarias, para luego cumplir con los objetivos propuestos de manera eficaz y llevar un control del esfuerzo requerido por cada uno en cada proyecto particular.

Como objetivos secundarios, el poder gestionar un perfil de cada desarrollador para tener un detalle de los conocimientos de cada uno. A su vez integrar una agenda que permita guardar tareas y tener la opción de recordarlas mediante algún medio informático.

**Alternativas**

1. No realizar el nuevo sistema

Riesgos:

* Perder el control de clientes, sobre todo en su atención, lo que llevaría a una disconformidad y corte del servicio por parte de los mismos, en el peor de los casos.
* Los clientes no tienen en claro los costos en tiempo real de cada nuevo requerimiento, se pueden perder clientes por no responder consultas en tiempo y forma.
* La comunicación con los clientes es constante, lo cual genera desfasajes y pérdidas de tiempo.

1. Comprar un sistema enlatado

Riesgos:

* Que el sistema no cumpla con los requerimientos funcionales y no funcionales.
* El tiempo de adaptación será diferente, puesto que como se compra se usa, sin realizar modificaciones que ayudarían a mejorar la experiencia del empleado.
* Este tipo de softwares genéricos suele estar atado a actualizaciones “normalizadas” para todos sus clientes, con lo cual la empresa podría sufrir cambios en la aplicación que generan nuevos tiempos de adaptación o disconformidad.
* Imposibilidad de adaptar el software de acuerdo a las necesidades específicas del equipo.

**Solución propuesta**

Desarrollar un sistema Web utilizando el framework .NET, bajo la arquitectura MVC que permita la comunicación de incidencias y requerimientos del cliente hacia el equipo de desarrollo, y a su vez poder gestionar el ciclo de vida de cada tarea asociada a una solicitud de un cliente .

El enfoque Web da una versatilidad y una potencialidad muy amplia; permitiendo en un futuro próximo y según la administración de la empresa necesite, publicar el sitio Web de cara a internet, agregando una gran facilidad de comunicación para aquellos clientes que han adquirido un sistema de la empresa.

Por otro lado se buscará llevar a cabo muchas de las buenas prácticas que se encuentran en el rubro, permitiendo un sistema bien estructurado y buena mantenibilidad. Este último aspecto resulta muy útil, pensando en las potencialidades del sistema y el futuro crecimiento.

**Objetivos**

Particulares

* Desarrollar un sistema de gestión de tareas de desarrollo
* Contribuir a la resolución de una problemática real

Generales

* Facilitar la comunicación del cliente con el equipo de desarrollo
* Simplificar un control del esfuerzo requerido para cada tarea
* Dejar constancia escrita de las solicitudes del cliente

Beneficios globales

* Mejorar la imagen de la empresa
* Simplificar y organizar el trabajo diario
* Apoyar a la gerencia general en informes de desempeño

**Factibilidad**

Factibilidad técnica

Desde este punto de vista hay una curva de aprendizaje de algunas herramientas utilizadas en el desarrollo del sistema. No obstante, los involucrados ya poseen conocimientos en el desarrollo de aplicaciones de igual e incluso mayor magnitud.

Factibilidad financiera

Se ha optado por utilizar herramientas gratuitas para el desarrollo de software. Dentro del entorno .NET, Microsoft ofrece herramientas de desarrollo totalmente gratuitas: el Visual Studio ofrece una cuenta gratuita denominada “Visual Studio Community”, la cual es una versión básica para estudiantes y programación junior.

Estas ediciones utilizadas son iguales al entorno de desarrollo comercial, pero sin características avanzadas de integración, las cuales no son relevantes para este caso puntual.

Así mismo, utilizaremos una versión gratuita del reconocido motor de base de datos de Microsoft (SQL Server Express).

El proyecto no incluye costos de publicación y registros de nombres de dominio (NIC) y hosting. Ambos aspectos quedan como responsabilidad de la empresa y no están dentro de esta planificación o alcance.

CABL y asociados cuenta con dominios propios (e incluso un servidor) en los cuales puede alojar el sistema si así lo desea. Es una alternativa que en última instancia van a evaluar los socios.

Tomando como referencia los valores del Consejo de profesionales de ciencias informáticas de Córdoba (http://www.cpcipc.org.ar/content/honorarios), se utilizará como precio por hora: 200 $ para todos los casos.

Se estima durante la ejecución del mismo un total aproximado de 510 (3 meses) horas de trabajo

|  |  |  |
| --- | --- | --- |
|  | **Horas Requeridas** | **Precio Hora** |
| **Desarrollador** | 2 a 4 horas diarias |  |
| **Tester** | 4 horas semanales |  |
| **Analista** | 40 horas totales |  |

**Aproximación presupuesto:**

|  |  |
| --- | --- |
| 510 horas \* 200 (precio fijo por hora) | $102.000,00 |
| Suelo director de proyecto | 30.000,00 |
| **TOTAL** | **132.000,00** |

**Riesgos**

Como principal riesgo, se detecta la postergación de la fecha de finalización. Los involucrados no se dedicarán full-time a la realización de este proyecto, ya que en simultaneo continuarán con sus actividades diarias que surgen de la afectación a diferentes proyectos.

Debemos anticipar un posible incumplimiento de algunos plazos definidos en los entregables, ya que como se mencionó anteriormente, se debe tener en cuenta que cada uno de ellos posee actividades tanto laborales y personales fuera del proyecto. A raíz de esto, el tiempo disponible se ve acotado a la predisposición de cada colaborador e involucrado en sus momentos libres. De todas formas se estima que en condiciones normales o favorables el plazo de finalización será respetado.

**Project Charter**

**Información del proyecto:**

**Organización/Cliente:** CABL & Asociados SRL ****

**Fecha de inicio:** 01/01/2017

**Fecha de finalización:** 01/06/2017

**Justificación**

La realización de este proyecto tiene 2 orígenes atribuibles; en primera instancia, el trabajo final de grado obligatorio para la finalización de la carrera Ingeniería en informática de la Universidad Católica de Santiago del Estero Sede Rafaela.

Como caso de estudio, resulta significativo tomar una problemática real de la empresa CABL & Asociados SRL de la ciudad de Sunchales y reunir los esfuerzos necesarios para dar curso a una solución que permita plasmar los conocimientos y la formación adquirida a lo largo de la carrera.

Por el lado del cliente, resulta evidente la necesidad de una solución que automatice los registros de trabajo y actividades de cada desarrollador; que unifique y agilice la toma de datos de cada uno de ellos y que además proporcione reportes con cierta periodicidad, sin afectar ni entorpecer el trabajo de cada integrante de la empresa.

**Objetivo técnico del proyecto**

Desarrollar un sistema web usando el framework .NET, bajo una arquitectura MVC.

**Roles y responsabilidades**

|  |  |  |
| --- | --- | --- |
| **Rol** | **Persona asignada** | **Responsabilidades** |
| **Patrocinador** | Carlos Blanche | Proveer los medios necesarios para la realización del proyecto |
| **Director de proyecto** | Bruno Curti | Afrontar los desafíos del proyecto, resolver inconvenientes en su transcurso y entregar a tiempo un sistema de calidad que cumpla con las expectativas del cliente |
| **Analista** | Pablo Piovano | Realizar análisis de requerimientos, modelado, casos de uso, y especificaciones de los procesos del sistema |
| **Desarrolladores** | Emanuel Zanoni  Antonela Hernandez  Hernán Zanatta | Desarrollar la interfaz gráfica y codificar los módulos de procesos y acceso a datos del sistema.  Crear base de datos de la aplicación. |
| **Tester** | Lucas Mallía | Definir y llevar a cabo casos de prueba para detectar potenciales vulnerabilidades y bugs en el sistema. |
| **Usuarios claves** | Maximiliano Astor  Luisina Sambuelli | Colaborar con el análisis de los procesos de la organización. Participar en las reuniones propias del ciclo de vida del proyecto para analizar los entregables resultantes y especificar solicitudes de cambio y conformidad. |

**Interesados**

Los principales interesados del proyecto son:

* Directorio de CABL & Asociados SRL
* Líderes de equipos de desarrollo
* Desarrolladores
* Analistas
* Testers
* Potenciales clientes de la empresa
* Clientes de la empresa
* Estudiantes de la carrera de ingeniería en informática
* Sector RRHH

**Project Scope**

**Información del proyecto:**

**Organización/Cliente:** CABL & Asociados SRL ****

**Fecha de inicio:** 01/01/2017

**Fecha de finalización:** 01/06/2017

**Fecha de actualización:** 10/01/2017

**Preparado por:**  Bruno Curti (PM)

**Breve descripción del proyecto**

El proyecto se enfoca en el desarrollo de un sistema web que permita la comunicación de incidencias y requerimientos del cliente hacia el equipo de desarrollo, y a su vez poder gestionar el ciclo de vida de cada tarea asociada a una solicitud de un cliente, permitiendo registrar y administrar las tareas para un seguimiento real de las mismas y simplificar control sobre las mismas.

**Alcance del producto**

**Funcionalidades**

* Generar tareas:
  + Presupuestar un requerimiento del usuario
  + Aceptar requerimientos
  + Asignación de colaboradores
  + Asignación de esfuerzo (tiempo)
  + Adjuntar documentación
* Autogestión:
  + Consultas
  + Reportes
  + Estadísticas
* Gestionar perfil de desarrollador:
  + Datos personales
  + Grado de conocimientos
* Gestionar historias de usuario:
  + Solicitud de requerimiento
  + Seguimiento
  + Adjuntar documentación
* Funcionalidades implícitas:
  + Gestión desarrolladores
  + Gestión clientes
* El entregable final no incluye:
  + Servicio de hosting
  + Registro del dominio (NIC.AR)

Para la publicación de la aplicación en la web, el cliente cuenta con un dominio propio el cual pone a disposición para anexar un módulo con acceso restringido para aquellos interesados como ser los clientes y empleados de la empresa.

**Puntos críticos**

A lo largo del proyecto será muy importante contar con una buena predisposición de los usuarios y clientes para colaborar con el análisis y recopilación de los requerimientos. Es factible que se les pedirá asistir a las reuniones periódicas de control de avances del proyecto.

Esto resulta sumamente importante para que no haya desvíos críticos en el desarrollo del producto final, las cuales pueden hacer fracasar el proyecto si no son tenidas en cuenta en el tiempo adecuado.

El entregable final del proyecto implicará cambios significativos en la forma de trabajar de la organización, razón por la cual es importante capacitar y concientizar a los usuarios y al personal implicado sobre los beneficios y resultados de la implementación del software.

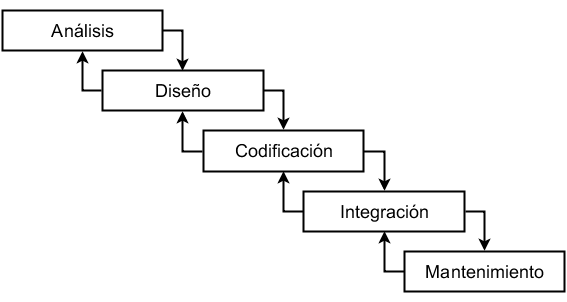
**Requisitos de aprobación**

Se preestablecen los puntos críticos a tener en cuenta en la aprobación final del producto:

* Software funcional
* Se requerirá que el producto cumpla con el documento OWASP Top 10 para prevenir los 10 riesgos más relevantes en relación a la seguridad en este tipo de aplicaciones. (<https://es.wikipedia.org/wiki/OWASP_Top_10>)
* Documentación del proyecto:
  + Manual de usuario para clientes
  + Manual de usuario para desarrolladores
* El código completo debe estar segurizado y alojado en el repositorio de código provisto por la empresa
* La capacitación y manual de usuario no deben excluir ninguna historia de usuario

**Ciclo de vida**

Al tener los objetivos del sistema bien claros y detallados desde un principio (debido a la naturaleza de la empresa), el producto se desarrollará mediante el ciclo de vida de cascada (o modelo lineal secuencial) con retroalimentación, ya que será un sistema cuyo modelo de negocios y circuito funcional no presentan dificultad para el análisis, teniendo en cuenta también la experiencia y capacidad de las personas involucradas en el desarrollo del sistema.



La elección del modelo con retroalimentación se debió a que en cada etapa del ciclo, contemplamos la posibilidad de que el cliente presente alguna disconformidad o pequeño cambio de alcance, por lo tanto este modelo nos da la posibilidad de volver a la etapa anterior y reorganizarla, como por ejemplo al realizar pruebas unitarias de Testing y detectar alguna inconsistencia, se vuelve a la etapa de codificación para solucionar el problema, o más común todavía, una vez que al cliente se le presenta un entregable funcional con los requerimientos que solicitó, suele ocurrírsele algún pequeño (o gran, en el peor de los casos) cambio funcional o de diseño.

**Estructura de descomposición del trabajo (EDT)**

1. **Planificación:**
   1. Business Case – Análisis de factibilidad
   2. Aprobación del proyecto
      1. Asignar al director de proyecto y miembros del equipo
      2. Creación del acta de constitución del proyecto
      3. Definición de objetivos y alcance
   3. Etapa de relevamiento
      1. Relevar información esencial asociada al sistema
      2. Definir y desarrollar mockups de los diferentes módulos
2. **Ejecución:**
   1. Etapa de iniciación
      1. Primer commit en el repositorio de código fuente (provisto por el patrocinador)
      2. Definir arquitectura de la aplicación
      3. Realización del modelo de datos – Modelado de clases
      4. Integración de librerías a usar dentro de la solución
   2. Codificación
      1. Perfiles de usuarios
         1. Modulo Clientes
            1. Importación desde planillas de calculo
            2. Alta/baja/Modificación/Consultas
         2. Modulo Desarrolladores
            1. Alta/Baja/Modificación/Consultas
         3. Administrativos
            1. Alta/Baja/Modificación/Consultas
      2. Módulo de seguridad
         1. Perfiles aislados
         2. Administración de cuentas de usuario
            1. Login/Logout/Reestablecer perfil
            2. Administración de perfiles
      3. Módulo de tareas
         1. Alta/Baja/Modificación/Consultas de:
            1. Esfuerzo
            2. Colaboradores
            3. Documentación asociada
            4. Cambios de estado
      4. Módulo de incidencias
         1. Alta/Baja/Modificación/Consultas de:
            1. Incidencias
            2. Requerimientos
            3. Documentación asociada
      5. Módulo de reportes
         1. Reporte por colaboradores de:
            1. Esfuerzo
            2. Fechas
            3. Contratiempos
         2. Reporte por clientes de
            1. Solicitudes
            2. Estado
            3. Presupuesto original
            4. Gasto final
   3. Testing:
      1. Seguridad (OWASP Top 10)
      2. Documentación de bugs encontrados
3. **Cierre del proyecto:**
   1. Commit final del proyecto en el repositorio
   2. Deploy de la solución
   3. Documentación del proyecto:
      1. Manual para clientes
      2. Manual para desarrolladores
4. **Evaluación del proyecto:**
   1. Lecciones aprendidas
   2. Conclusiones finales

**Cronograma:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Cronograma**  **CABL-Dev** | **Mes 1** | | | | **Mes 2** | | | | **Mes 3** | | | | **Mes 4** | | | | **Mes 5** | | | | **Mes 6** | | | |
| **1º** | **2º** | **3º** | **4º** | **1º** | **2º** | **3º** | **4º** | **1º** | **2º** | **3º** | **4º** | **1º** | **2º** | **3º** | **4º** | **1º** | **2º** | **3º** | **4º** | **1º** | **2º** | **3º** | **4º** |
| **1- Inicio del Proyecto** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2- Relevamiento inicial** |  | **x** | **x** | **x** | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.1 Reuniones con los clientes |  | **x** | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.2 Solicitud de información |  | **x** | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.3 Documentación |  |  |  |  | **x** | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.4 Clasificación y Priorización de Requerimientos |  |  |  |  |  | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.5 División de tareas |  |  |  |  |  |  | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3 - Análisis y Diseño** |  |  |  |  |  |  |  | **x** | **x** | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.1 Casos de Uso |  |  |  |  |  |  |  | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.2 Diagrama de clases |  |  |  |  |  |  |  | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.3 Diagrama de secuencia |  |  |  |  |  |  |  |  | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.4 Diagrama de actividades |  |  |  |  |  |  |  |  | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3.5 Modelo de Datos |  |  |  |  |  |  |  |  | **x** | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |  |
| **4 - Desarrollo** |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** |  |  |  |  |  |
| 4.1 Formularios ABM |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** | **x** | **x** |  |  |  |  |  |  |  |  |  |
| 4.2 Lógica interna |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** | **x** | **x** | **x** |  |  |  |  |  |  |
| 4.3 Reportes |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** | **x** | **x** | **x** | **x** |  |  |  |  |  |
| 4.4 Publicación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |  |  |  |
| **5 - Pruebas y Testing** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** | **x** |  |  |  |
| 5.1 Formularios ABM |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** |  |  |  |  |
| 5.2 Lógica interna |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** |  |  |  |  |
| 5.3 Reportes |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |  |  |
| 5.4 Prueba integral final |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** | **x** |  |  |  |
| **6 - Implementación** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** |  |
| 6.1 Instalación del sistema |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** |  |
| 6.2 Evaluación final |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** |  |
| **7 - Capacitación** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  | **x** |
| 7.1 Pre-implementación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |  |
| 7.2 Post-implementación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |
| **8 - Documentación** |  | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** |

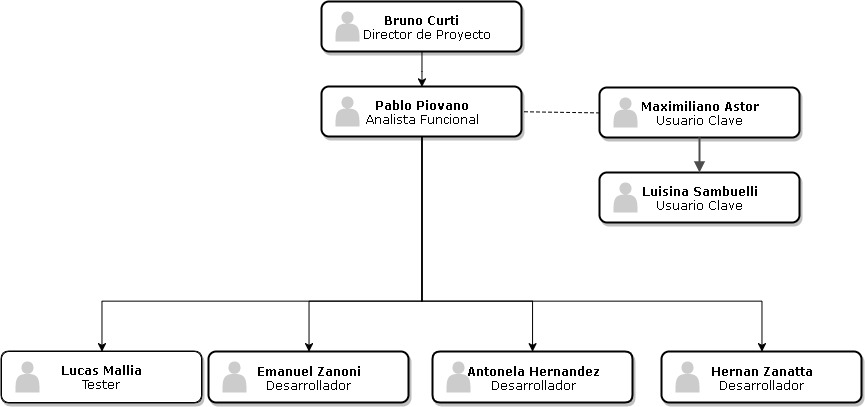
**Calendario del proyecto (Schedule):**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Etapa** | **Subetapa** | | **Fecha Inicio** | **Días** | **Fecha Fin** | **Actividad** |
| **Planificación** |  | | 02/01/2017 | 3 | 04/01/2017 | Business Case – Análisis de factibilidad |
| **Aprobación**  **De proyecto** | | 05/01/2017 | 1 | 05/01/2017 | Asignar al director del proyecto y miembros del equipo |
| 06/01/2017 | 1 | 06/01/2017 | Creación del acta de constitución del proyecto |
| 09/01/2017 | 1 | 09/01/2017 | Definición de objetivos y alcance |
| **Relevamiento** | | 10/01/2017 | 9 | 20/01/2017 | Relevar información esencial asociada al sistema |
| 23/01/2017 | 5 | 27/01/2017 | Definir y desarrollar mockups de los diferentes módulos |
| **Ejecución** | **Iniciación** | | 30/01/2017 | 1 | 30/01/2017 | Primer commit en el repositorio |
| 31/07/2017 | 4 | 03/02/2017 | Definir arquitectura de la aplicación |
| 06/02/2017 | 5 | 10/02/2017 | Realización del modelo de datos – Modelado de clases |
| 13/02/2017 | 5 | 17/02/2017 | Integración de librerías |
| **Codificación** | **Clientes** | 20/02/2017 | 3 | 22/02/2017 | Importación de planillas de cálculo |
| 23/02/2017 | 5 | 01/03/2017 | Alta/baja/modificación/consultas |
| **Desarrolladores** | 02/03/2017 | 7 | 10/03/2017 | Alta/baja/modificación/consultas |
| **Administrativos** | 13/03/2017 | 5 | 17/03/2017 | Alta/baja/modificación/consultas |
| **Seguridad** | 20/03/2017 | 2 | 21/03/2017 | Perfiles aislados |
| 22/03/2017 | 2 | 23/03/2017 | Administración de cuentas de usuario |
| 24/03/2017 | 2 | 27/03/2017 | Login/Logout/Reestablecer perfil |
| 28/03/2017 | 4 | 31/03/2017 | Administración de Perfiles |
| **Tareas** | 03/04/2017 | 5 | 07/04/2017 | Alta/baja/modificación/consultas |
| 10/04/2017 | 5 | 14/04/2017 | Administración de cuentas de usuario |
| 17/04/2017 | 2 | 18/04/2017 | Esfuerzo |
| 19/04/2017 | 3 | 21/04/2017 | Colaboradores |
| 24/04/2017 | 5 | 28/04/2017 | Documentación asociada |
| 01/05/2017 | 2 | 02/05/2017 | Cambios de estado |
| **Incidencias** | 03/05/2017 | 3 | 05/05/2017 | Alta/baja/modificación/consultas |
| 08/05/2017 | 2 | 09/05/2017 | Documentación asociada |
| **Reportes** | 10/05/2017 | 1 | 10/05/2017 | Reportes por colaboradores |
| 11/05/2017 | 1 | 11/05/2017 | Reportes por esfuerzo |
| 12/05/2017 | 1 | 12/05/2017 | Reportes por fecha |
| 15/05/2017 | 1 | 15/05/2017 | Reporte de contratiempos |
| 16/05/2017 | 1 | 16/05/2017 | Reporte por clientes |
| 17/05/2017 | 1 | 17/05/2017 | Reporte por solicitudes |
| 18/05/2017 | 1 | 18/05/2017 | Reporte por estado |
| 19/05/2017 | 1 | 19/05/2017 | Reporte por presupuesto original |
| 22/05/2017 | 1 | 22/05/2017 | Reporte gasto final |
| **Testing** |  | 23/05/2017 | 1 | 23/05/2017 | Seguridad (OWASP Top 10) |
|  |  | 24/05/2017 | 2 | 25/05/2017 | Documentación de bugs |
| **Cierre del proyecto** |  |  | 26/05/2017 | 1 | 26/05/2017 | Commit final del proyecto |
|  |  | 29/05/2017 | 1 | 29/05/2017 | Deploy de la solución |
| **Documentación** |  | 26/05/2017 | 4 | 29/05/2017 | Manual para clientes |
|  | 26/05/2017 | 4 | 29/05/2017 | Manual para desarrolladores |
| **Evaluación del proyecto** |  |  | 30/05/2017 | 1 | 30/05/2017 | Lecciones aprendidas |
|  |  | 31/05/2017 | 2 | 01/06/2017 | Conclusiones finales |

**Plan de Recursos Humanos:**

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Patrocinador:** |  | Carlos Blanche |
| **Director de proyecto:** |  | Bruno Curti |
| **Analista:** |  | Pablo Piovano |
| **Desarrolladores:** |  | Emanuel Zanoni  Antonela Hernandez  Hernán Zanatta |
| **Tester:** |  | Lucas Mallía |
| **Usuarios claves:** |  | Maximiliano Astor  Luisina Sambuelli |

**Organigrama de RRHH:**

****

Al ser un equipo relativamente pequeño, y ya tener experiencia en trabajos similares, es bastante democrático, pero sin dejar de ser ligeramente jerárquico.

**Sistema de recompensas:**

* Viernes con asado, si todo va al día (según cronograma)
* Gran cena de cierre

**Matriz de asignación de responsabilidades (RACI)**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Rol** | | **Descripción** |
| **R** | **Responsible** | **Responsable** | Refiere a quien efectivamente realiza la tarea. Lo habitual es que exista un único encargado (R) por tarea; si existe más de uno, el trabado se subdivide a un nivel más bajo usando RASCI. |
| **A** | **Accountable** | **Quien rinde cuentas** | Se responsabiliza de que la tarea se realice, y es quien rinde cuentas sobre su ejecución. Sólo puede existir una persona que deba rendir cuentas (A) de que el responsable (R) ejecute la tarea. |
| **C** | **Consulted** | **Consultado** | Posee algún conocimiento o capacidad requerida para realizar la tarea. (C) |
| **I** | **Informed** | **Informado** | Debe ser informado sobre el avance y resultados de la ejecución de la tarea. A diferencia de (C), la comunicación es unidireccional. |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Responsabilidad** | **Patrocinador** | **D.P.** | **Analista** | **Desarrolladores** | | | **Tester** | **Usuarios** | |
| **Carlos Blanche** | **Bruno**  **Curti** | **Pablo**  **Piovano** | **Emanuel**  **Zanoni** | **Antonela**  **Hernandez** | **Hernán**  **Zanatta** | **Lucas**  **Mallía** | **Maximiliano**  **Astor** | **Luisina**  **Sambuelli** |
| **Realización Business Case** | **C** | **R** |  |  |  |  |  | **C** | **C** |
| **Aprobación del proyecto** | **R** | **I** |  |  |  |  |  |  |  |
| **Relevamiento** |  | **A** | **R** |  |  |  |  | **C** | **C** |
| **Etapa de Iniciación** |  | **A** |  | **R** | **C** | **C** |  |  |  |
| **Clientes** |  | **A** |  |  | **R** |  |  |  |  |
| **Desarrolladores** |  | **A** |  |  |  | **R** |  |  |  |
| **Administrativos** |  | **A** |  | **R** | **C** | **C** |  |  |  |
| **Seguridad** |  | **A** |  |  | **R** |  |  |  |  |
| **Tareas** |  | **A** |  |  |  | **R** |  |  |  |
| **Incidencias** |  | **A** |  | **C** | **R** | **C** |  |  |  |
| **Reportes** |  | **A** |  | **C** | **R** | **C** |  |  |  |
| **Testing** |  | **A** |  |  |  |  | **R** |  |  |
| **Cierre del Proyecto** | **I** |  |  |  |  |  |  |  |  |
| **Evaluación** | **I** | **R** | **C** | **C** | **C** | **C** | **C** | **C** | **C** |

**Gestión de Costos**

Para la estimación de costos se utilizará como herramienta “Estimación ascendente”, tomando como base la partición estructurada de tareas.

Se elige ésta metodología ya que, si bien no se tiene en un alto nivel de detalle de cada actividad puntual, podemos obtener una estimación más precisa que si haríamos estimación análoga.

Se implementará en primera instancia estimación mediante PERT para aproximarnos a la cantidad de horas requerida de cada actividad/componente de nuestro WBS.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Actividad** | **Costo estimado expresado en horas** | | | |
| **Costo Optimista** | **Costo más probable** | **Costo**  **máximo** | **Costo**  **estimado** |
| **Relevar información esencial asociada al sistema** | **27** | **36** | **50** | **37** |
| **Definir y desarrollar mockups de los diferentes módulos** | **25** | **30** | **30** | **30** |
| **Primer commit en el repositorio** | **1** | **1** | **2** | **2** |
| **Definir arquitectura de la aplicación** | **4** | **6** | **8** | **6** |
| **Realización del modelo de datos – Modelado de clases** | **30** | **40** | **45** | **40** |
| **Integración de librerías** | **6** | **7** | **8** | **7** |
| **Importación de planillas de cálculo** | **16** | **20** | **25** | **21** |
| **Alta/baja/modificación/consultas (Clientes)** | **16** | **20** | **25** | **21** |
| **Alta/baja/modificación/consultas (Desarrolladores)** | **16** | **20** | **25** | **21** |
| **Alta/baja/modificación/consultas (Administrativos)** | **16** | **20** | **25** | **21** |
| **Perfiles aislados** | **4** | **4** | **6** | **5** |
| **Administración de cuentas de usuario** | **4** | **4** | **6** | **5** |
| **Login/Logout/Reestablecer perfil** | **2** | **2** | **3** | **3** |
| **Administración de Perfiles** | **4** | **4** | **8** | **5** |
| **Alta/baja/modificación/consultas (Tareas)** | **16** | **20** | **25** | **21** |
| **Administración de cuentas de usuario** | **4** | **4** | **8** | **5** |
| **Esfuerzo** | **2** | **2** | **4** | **3** |
| **Colaboradores** | **2** | **2** | **4** | **3** |
| **Documentación asociada** | **2** | **2** | **4** | **3** |
| **Cambios de estado** | **2** | **2** | **4** | **3** |
| **Alta/baja/modificación/consultas (Incidencias)** | **16** | **20** | **25** | **21** |
| **Documentación asociada** | **16** | **20** | **22** | **20** |
| **Reportes por colaboradores** | **8** | **10** | **12** | **10** |
| **Reportes por esfuerzo** | **8** | **10** | **12** | **10** |
| **Reportes por fecha** | **8** | **10** | **12** | **10** |
| **Reporte de contratiempos** | **8** | **10** | **12** | **10** |
| **Reporte por clientes** | **8** | **10** | **12** | **10** |
| **Reporte por solicitudes** | **8** | **10** | **12** | **10** |
| **Reporte por estado** | **8** | **10** | **12** | **10** |
| **Reporte por presupuesto original** | **8** | **10** | **12** | **10** |
| **Reporte gasto final** | **8** | **10** | **12** | **10** |
| **Seguridad (OWASP Top 10)** | **8** | **12** | **16** | **12** |
| **Documentación de bugs** | **8** | **10** | **12** | **10** |
| **Commit final del proyecto** | **2** | **2** | **4** | **3** |
| **Deploy de la solución** | **3** | **3** | **5** | **4** |
| **Manual para clientes** | **24** | **24** | **30** | **25** |
| **Manual para desarrolladores** | **24** | **24** | **30** | **25** |
| **Lecciones aprendidas** | **8** | **8** | **12** | **9** |
| **Conclusiones finales** | **8** | **8** | **12** | **9** |
| **Total** | | | | **490** |

Se estipulará un monto adicional para contingencias equivalente a 50hs de trabajo, lo cual representa un total de $10000 de reserva.

A su vez, desde 22/06/2017 hasta a la fecha 26/06/2017 se estiman 4 horas adicionales por semana destinadas al testing de los diferentes módulos de la aplicación. Esto equivale a un adicional de (4 x 5) 20 Hs.

**Aproximación final de costos:**

|  |  |  |
| --- | --- | --- |
| Horas desarrolladores | 490 Hs. \* $ 200 (precio hora) | $ 98.000,00 |
| Horas adicionales Testing | 20 Hs. \*$ 200 (precio hora) | $ 4.000,00 |
| Suelo director de proyecto | | $ 30.000,00 |
| Reserva para contingencias | | $ 50.000,00 |
| **TOTAL** | | **$ 182.000,00** |

**Plan de Riesgos**

Como se mencionó previamente en el Business Case, el principal riesgo identificado es la postergación de la fecha de finalización del proyecto. Los tiempos que poseen los involucrados se ven afectados por las responsabilidades que poseen diariamente.

Se requiere una actitud proactiva de cada uno de los miembros del equipo de proyecto para evitar demoras ante cualquier problema o eventualidad, una fluida comunicación entre los miembros y un alto grado de solidaridad.

El impacto de dicho riesgo se acrecienta a medida que el plazo de finalización del proyecto se posterga. Se compromete la implementación a tiempo del sistema para su utilización en la segunda mitad del año 2017.

Se utilizará como estrategia mitigar el riesgo, haciendo una monitorización y control exhaustivo del proyecto.

Como punto estratégico, en las reuniones de estado del proyecto se identificarán y analizarán los aspectos considerables a este riesgo puntual.

**Gestión de las comunicaciones**

El equipo está compuesto por 9 interesados directos:

* 1 patrocinador
* 1 DP
* 3 Desarrolladores
* 1 Tester
* 2 Usuarios claves

Por lo tanto se detectan 36 canales de comunicación [ ( 9 \* 8 ) / 2 = 36 ]

**Tecnología de las comunicaciones**

Se optó por el uso de 4 tecnologías de comunicación las cuales cada miembro del equipo conoce a la perfección dada su vasta experiencia en proyectos similares:

* Entrevistas: las mismas serán coordinadas entre el analista funcional con los usuarios claves para poder analizar y capturar los requerimientos necesarios para el sistema.
* Reuniones de avance cada 15 días: tendrán como objetivo evaluar el estado puntual de avance del proyecto. Se realizarán en la sala de reuniones de CABL & Asociados, previo acuerdo con los interesados. Se establece una duración máxima de 90 minutos.
* Correo electrónico: el objetivo es la constancia por escrito de breves comunicaciones; como ser consultas breves o la coordinación de alguna actividad.
* Videoconferencias/llamadas: para esto se preestablece el uso de la herramienta Skype, la cual todo el equipo hace un uso diario. Será para cuestiones que necesiten respuestas más urgentes, o aquellas consultas que precisen compartir pantallas o archivos relacionados al proyecto.

**Gestión de la calidad**

Con objetivo de prevenir errores y/o defectos en el entregable final del proyecto y con el fin de tener un cliente satisfecho, se trabajará con un enfoque proactivo para la gestión de calidad.

Se utilizará como metodología el proceso de desarrollo en cascada, la cual permite generar software de calidad buscando en forma permanente mejora continua y retroalimentación entre las diferentes etapas.

El área de testing destinada al proyecto participará para garantizar la calidad del entregable final.

Se busca mitigar cualquier costo de falla antes de que el producto llegue al cliente.

Como particularidades no se contará con auditorias puesto que no se tiene a involucrados para dicha actividad y la dimensión del proyecto no lo justifica. Del mismo modo, se descartara el uso de alguna norma de calidad preexistente.

El comité de control de cambios, en el cual participarán el patrocinador y el director de proyecto, será el responsable de aprobar y/o rechazar las solicitudes de cambio.

**Monitoreo y control**

El monitoreo y control del proyecto se llevara a cabo a lo largo de toda el ciclo de vida del proyecto, y estará a cargo del director de proyecto.

Para evaluar el desempeño del proyecto durante su ejecución se utilizara la Gestión del valor ganado, lo cual permitirá controlar la gestión integrada del alcance, la agenda y los costos.

La intención es detectar a tiempo aquellas partes del proyecto que requieran una atención especial e identificar acciones preventivas o correctivas para resolver cualquier problema de desempeño que tenga el equipo.

Cada 15 días, y posteriori a la reunión de avance, el director de proyecto entregará un informe a los interesados con la situación actual del proyecto. En este reporte se detallarán cambios, en caso de ser requeridos.

Ante la necesidad de adquirir algún cambio, el comité de cambios evaluará el impacto del mismo e informará la decisión final a todos los interesados.

El proyecto no requiere adquisiciones dado que la empresa ya cuenta cuenta tanto con las herramientas físicas como intangibles (Software).

**Conclusión**

A lo largo de este trabajo se ha trabajado con la “Guía de los fundamentos para la dirección de proyectos (Guía del PMBOK®)” aplicando conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto.

Si bien por el momento solo se han abordado las fases de iniciación y planificación, el presente trabajo dará continuidad al resto de las fases de la dirección de proyectos con el fin de cumplir con los requerimientos del Sistema de Gestión de Tareas CABL-Dev.

Es importante destacar que las diferentes estimaciones realizadas no ha resultado una tarea sencilla debido a la falta de experiencia en este tipo de labor.

Por otro lado es importante enfatizar la oportunidad de aplicar los conocimientos de la materia “Administración de Proyectos” a un caso de estudio real que dará soporte al Proyecto Final de la carrera “Ingeniería en informática”, y el cual se encuentra en curso actualmente por mi parte y los involucrados mencionados en este trabajo.

La administración de proyectos sin dudas asegura la materialización de los resultados esperados.