

Propuesta Técnica para el Sistema de Administración de convocatoria de auxiliares



CONSULTOR TIS

: Lic. Escalera Fernandez David.

RAZON SOCIAL DEL PROPONENTE

: "ActionSoft S.R.L."

E-MAIL DEL PROPONENTE

: actionsoft.corp@gmail.com

REPRESENTANTE LEGAL DE LA EMPRESA

: Lic. Orellana Claros Sergio Alvaro.

TELEFONO

: 68444975



1 Contenido

1	PROPUESTA DE SERVICIOS.....	4
1.1	Antecedentes.....	4
1.2	Identificación de los requerimientos del Sistema.....	4
1.2.1	Requisitos funcionales	4
1.2.2	Requisitos no funcionales	5
1.3	Software para el desarrollo del sistema	5
1.3.1	Para el código fuente	5
1.3.2	Para la base de datos	5
1.3.3	Para el servidor web	5
1.3.4	Otros.....	5
1.4	Metodología de trabajo	6
1.4.1	Concepto	6
1.4.2	El Equipo Scrum	6
1.4.3	Artefactos de Scrum	8
1.4.4	Eventos de Scrum	9
2	PLANIFICACIÓN DEL PROYECTO.....	12
3	PLAZO DE CONCLUSIÓN DEL CONTRATO	13
4	PROPUESTA ECONÓMICA Y PLAN DE PAGOS	14
4.1	ESTIMACIÓN DE COSTOS DE PROYECTO	14
4.2	CÁLCULO DE LA COMPLEJIDAD	15
4.3	PRODUCTIVIDAD	16
4.4	COSTOS DE RECURSOS HUMANOS.....	16
4.5	COSTOS OPERATIVOS.....	17



ACTIONSOFT S.R.L

Actionsoft.corp@gmail.com

68444975

Cochabamba - Bolivia

4.6	COSTOS DE CAPACITACIÓN A USUARIOS FINALES.....	18
4.7	COSTO TOTAL	18
4.8	PLAN DE PAGOS.....	19
5	ADMINISTRACION DE RIESGO.....	19
5.1	IDENTIFICACION DE RIESGO	19
5.1.1	RIESGOS TECNOLOGICOS.....	20
5.1.2	RIESGOS ASOCIADOS AL EQUIPO DE TRABAJO	21
5.1.3	RIESGOS RELACIONADOS CON LOS REQUERIMIENTOS DEL CLIENTE.....	22
5.1.4	RIESGOS RELACIONADOS CON LA ESTIMACIÓN	23
5.1.5	PLAN DE CONTINGENCIA CONTRA RIESGOS	24



1 PROPUESTA DE SERVICIOS

1.1 Antecedentes

El presente documento da respuesta a la convocatoria pública CPTIS-9471-2020 y el pliego de especificaciones PETIS-9471-2020, presentadas por la empresa TIS, de esta manera “**ACTIONSOFT S.R.L.**” presenta una solución eficiente a la necesidad de un “Sistema de Administración de convocatoria de auxiliares”.

1.2 Identificación de los requerimientos del Sistema

1.2.1 Requisitos funcionales

Para la determinación de requisitos funcionales se tomó en cuenta los requerimientos emergentes del Pliego de Especificaciones, los cuales son:

- Publicación de la(s) convocatorias, en el que se tiene la información general por convocatoria.
- Registro de postulantes a la(s) convocatoria(s), que inicia desde el registro de un postulante a un ítem específico de una convocatoria, hasta la entrega de documentos de acuerdo al cronograma.
- Habilitación de postulantes que permite bajo los criterios especificados en la convocatoria habilitar/inhabilitar postulantes y genera el resultado de la habilitación.
- Calificación de méritos, que permite a la comisión registrar la evaluación de los documentos presentados por los postulantes y calificar acorde a la tabla respectiva de la convocatoria e ítem al que el postulante se presenta.
- Calificación de conocimientos, que permite a la comisión de conocimientos registrar los resultados de sus evaluaciones.
- Publicación de resultados que consolidan los resultados de las comisiones para así obtener una lista de auxiliares.

En otros casos, las especificaciones otorgadas por la descripción del desarrollo de software según la documentación **PEPTIS-9471-2020**, serán aplicadas mediante tareas dispuestas por las historias de usuario según la empresa, tomando en cuenta a todos los usuarios que cumplirán un rol específico en el sistema.



1.2.2 Requisitos no funcionales

Los requisitos no funcionales para el sistema son los siguientes:

- El sistema será desarrollado con una interfaz gráfica de usuario basada en Web.
- El sistema será accesible desde cualquier equipo de trabajo con navegadores Web, Google Chrome (17.0 o superior) y Mozilla Firefox (2.0 o superior).
- El sistema trabajará con el administrador de base de datos MySQL.
- El sistema contará con manual de usuario para su entendimiento y capacitación en el sistema.
- El sistema contará con manual técnico y de instalación.

1.3 Software para el desarrollo del sistema

Para satisfacer las características requeridas por el sistema, el equipo técnico ha decidido utilizar las siguientes tecnologías:

1.3.1 Para el código fuente

- El lenguaje de programación base utilizado será **PHP** por su facilidad de aprendizaje y porque va muy bien de la mano con el servidor apache.
- Para la interfaz de usuario se usará **HTML y CSS**.

1.3.2 Para la base de datos

- El gestor de base de datos utilizado será **MySQL**, por su rapidez y facilidad en el manejo de datos.

1.3.3 Para el servidor web

- El servidor web utilizado será **Apache** por su rápida configuración.

1.3.4 Otros

Como parte del control y administración en el proceso de desarrollo se utilizarán las siguientes herramientas:

- **Git** como herramienta de control de versiones y repositorio.



Como framework, se utilizará:

- **Laravel** por su expresividad, simplicidad y por su amplia y activa comunidad
- **Bootstrap** para la modelación de la vista, ya que es las factible y visiblemente atractivo para toda aplicación web.

1.4 Metodología de trabajo

El marco de trabajo que guiará el proceso de desarrollo del proyecto de trabajo será la metodología scrum personalizado según la empresa.

1.4.1 Concepto

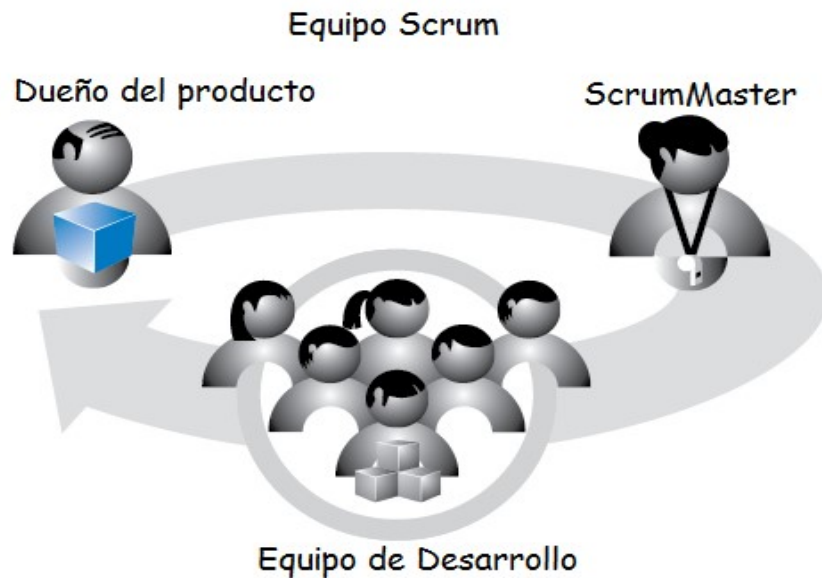
Scrum es un marco ágil para completar proyectos complejos. Scrum se puede utilizar para proyectos de desarrollo de software, así como para cualquier ámbito de trabajo complejo e innovador. Algunos de los beneficios son: flexibilidad a cambios; incremental e iterativo; cumplimiento de expectativas; reducción de riesgos; mayor productividad, y mayor calidad de software.

Sus mayores características son:

- Valorar a las personas del proyecto y su interacción (desarrolladores y clientes) por encima de los procesos y herramientas.
- La respuesta ante el cambio por encima del seguimiento de una planificación.
- La colaboración con el cliente por encima de la negociación del contrato y desarrollar software funcional de calidad.

1.4.2 El Equipo Scrum

El equipo en Scrum como en todo proceso ágil es uno de los elementos más importantes para el desarrollo, ya que estos son los que trabajarán de manera conjunta para lograr un objetivo. El Equipo Scrum está formado por un Dueño de Producto (Product Owner), un Facilitador (Scrum Master) y el Equipo de Desarrollo (Development Team).



- **DUEÑO DEL PRODUCTO (PRODUCT OWNER)**

Es el responsable de todo el proyecto, encargado de definir los objetivos del proyecto, mantener y priorizar las tareas en el backlog, asegurar que el equipo de desarrollo entienda la lista del producto (Product Backlog), esta persona también es el representante del Scrum Team y actúa como interlocutor único entre el equipo y el cliente, ya sea una persona individual o un conjunto de personas u organización interesada en la creación de un proyecto.

- **FACILITADOR (SCRUM MASTER)**

Es el responsable de velar que todos los integrantes del proyecto sigan los valores y principios ágiles, las reglas del proceso Scrum y guiar la colaboración entre el equipo y el cliente de manera que las colaboraciones sean beneficiadas al máximo.

El Scrum Master tiene como objetivo principal mantener la fluidez en la que el proyecto avance con el pasar del tiempo, brindando servicios o materiales al equipo que puedan facilitar en el desarrollo del proyecto.

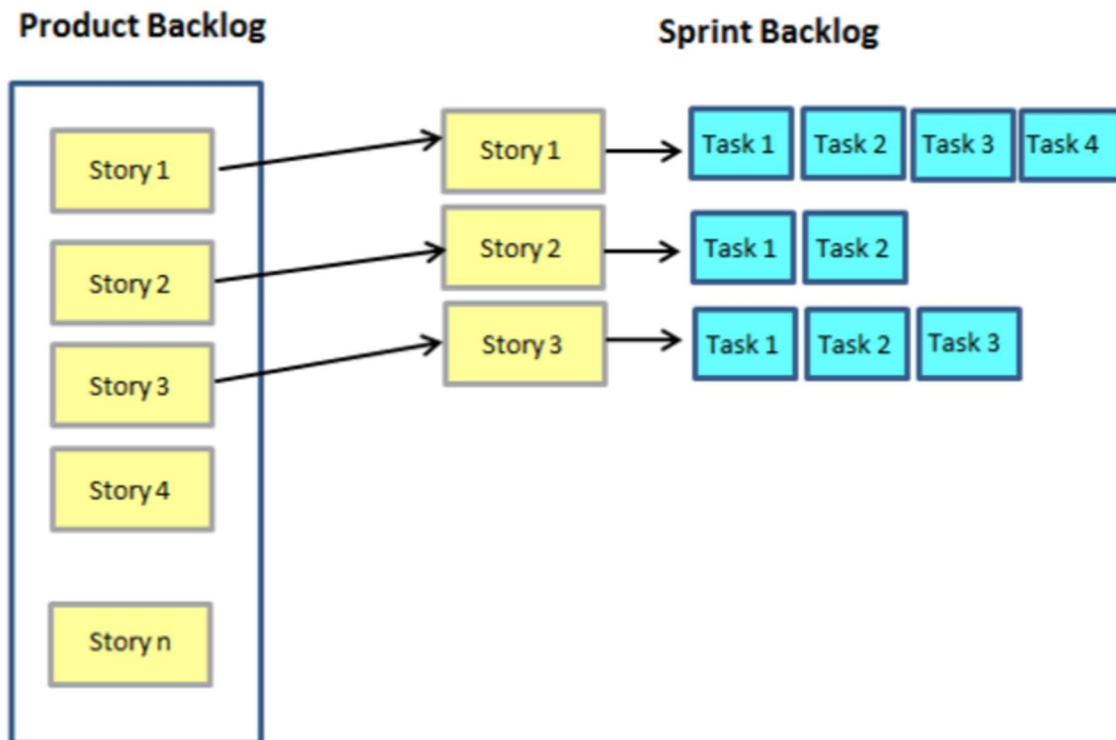
- **EQUIPO DE DESARROLLO (DEVELOPMENT TEAM)**

Es un conjunto de profesionales con conocimientos sólidos en las tecnologías para el desarrollo del proyecto. Este grupo desarrolla y prueba el producto del proyecto teniendo así un objetivo en común, compartiendo la responsabilidad del trabajo.

Los miembros del equipo de desarrollo son auto-organizados y está conformado normalmente de 3 a 9 personas.

1.4.3 Artefactos de Scrum

Los artefactos de Scrum sirven para capturar la comprensión compartida del equipo en un momento en particular. Es decir, ayudan al equipo a ver como hacen el trabajo para llegar a la meta de un Sprint. Son necesarios para asegurar que todos tengan el mismo entendimiento de lo que se tiene que hacer. A seguir están breves descripciones de los artefactos: Lista de Productos (Product Backlog), Lista de Pendientes del Sprint (Sprint Backlog) e Incremento (Increment).



- **LISTA DE PRODUCTOS (PRODUCT BACKLOG)**

Es una lista o documento de todo lo que se necesita en el producto. Es el único recurso de requerimientos para cualquier caso que el producto necesitaría. El dueño del producto es responsable de la lista de productos, incluyendo sus contenidos, disponibilidad y orden.

- **LISTA DE PENDIENTES DEL SPRINT (SPRINT BACKLOG)**

Es el conjunto de requerimientos que serán desarrollados por el equipo durante los Sprints, este es creado en el Sprint Planning. En este evento, normalmente se define las iteraciones y los requerimientos que serán parte de dichas iteraciones. Los requerimientos normalmente se dividen en historias de usuario, que contemplan el producto a entregar durante cada sprint, los cuales están subdivididos en tareas.

En esta lista se detalla las tareas del requerimiento, horas de trabajo (esfuerzo) y el responsable de cada tarea.

El Sprint Backlog sirve para hacer un seguimiento al equipo, permitiendo ver las tareas donde el equipo está teniendo problemas, con lo que se permite tomar decisiones al respecto.

- **INCREMENTO (INCREMENT)**

El incremento es el resumen de todos los ítems completados según la lista de productos (product backlog) durante un sprint y el valor de los incrementos de todos los sprint anteriores. Al final de un sprint, el nuevo incremento debe cambiar su estado a 'completo'. Un incremento es un cuerpo de inspecciones, trabajos hechos que colaboran el empirismo al final de un sprint. El incremento es un paso adelante hacia una visión o meta. Este tiene que estar en condiciones usables sin importar de que el dueño del producto decida publicarlo.

1.4.4 Eventos de Scrum

En Scrum, existen eventos predefinidos para asegurar que el proceso de desarrollo del proyecto está en el camino correcto. Está enfocado en diferentes aspectos del proceso. Estos eventos son bloques de tiempo (time-boxes), de tal modo que todos tienen una duración máxima.



- **SPRINT:**

El corazón de Scrum es el Sprint. Es un bloque de tiempo (time-box) con duración fija, durante el cual se crea un incremento de producto “Terminado”, utilizable y potencialmente desplegable. Cada Sprint tiene una definición de qué se va a construir, un diseño y un plan flexible que guiará la construcción y el trabajo y el producto resultante.

Algo importante que cabe resaltar es que durante un Sprint no se realizan cambios que afectan al proyecto, la composición del equipo de desarrollo se mantiene constante y los objetivos de calidad no disminuyen.

- **PLANIFICACION DEL SPRINT (SPRINT PLANNING)**

Esta reunión tiene como finalidad planear el trabajo a realizar durante un Sprint, el cual es creado por el trabajo de todo el equipo Scrum, teniendo como base el Product Backlog.

Cuando se realiza esta reunión de planificación se debe responder estas preguntas:

- ¿Qué es lo que se hizo desde la última reunión?
- ¿Qué es lo que se va a hacer hasta la siguiente reunión?
- ¿Cómo se va a llevar a cabo?

- **SCRUM DIARIO (DAILY SCRUM)**

El Scrum Diario es una reunión con un bloque de tiempo de 15 minutos para que el Equipo de Desarrollo sincronice sus actividades y cree un plan para las siguientes 24 horas. Esto se lleva a cabo inspeccionando el trabajo avanzado desde el último Scrum Diario y haciendo una proyección acerca del trabajo que podría completarse antes del siguiente. El Scrum Diario se realiza a la misma hora y en el mismo lugar todos los días, donde cada miembro del Equipo de Desarrollo explica:

- Lo que hizo el día anterior para ayudar al equipo a lograr el objetivo del sprint.
- Lo que hará el día presente para ayudar al equipo a lograr el objetivo del sprint.
- Qué impedimentos ha encontrado que evita al equipo a lograr el objetivo del sprint.



- **REVISION DE SPRINT (SPRINT REVIEW)**

Consiste en una reunión informal no superior a 4 horas para la revisión del Sprint. Durante esta revisión el equipo Scrum y los interesados analizan acerca de lo que se ha hecho durante el Sprint. En esta reunión se hace lo siguiente:

- El Product Owner identifica lo que ha sido terminado y lo que no.
- El equipo de desarrollo expone sobre las cosas positivas, los problemas que surgieron y cómo fueron solucionados, luego se muestra el trabajo terminado y responder cualquier inquietud.
- El Product Owner proyecta fechas de finalización probables basándose en el progreso obtenido hasta la fecha.
- Todo el equipo de desarrollo colabora acerca de qué es lo siguiente a hacer, de modo que la revisión del Sprint proporcione información de entrada valiosa para reuniones de planificación siguientes.

- **RETROSPECTIVA DEL SPRINT (SPRINT RETROSPECTIVE)**

Esta reunión es una oportunidad para que el equipo Scrum pueda examinarse a sí mismo y crear un plan de mejoras para el siguiente Sprint.

Este es un punto para determinar que se hizo bien y que se puede mejorar, esta reunión se debe realizar después del Sprint Review y antes del siguiente Sprint Planning, esta reunión al igual que todas las reuniones debe tener una duración de tiempo que depende del Sprint.

Los objetivos de esta reunión son:

- Revisar como fue el último Sprint en aspectos como procesos, herramientas, etc.
- Identificar los elementos más importantes que fueron positivos y crear posibles mejoras.
- Desarrollar un plan para implementar mejoras en el desempeño del trabajo del equipo.



2 PLANIFICACIÓN DEL PROYECTO

Id	Nombre de actividad	Inicio	Fin	Duración
1	Sprint 0	11/03/20	15/03/20	5 días
2	Sprint 1	16/03/20	31/03/20	15 días
3	Sprint 2	04/04/20	18/04/20	15 días
4	Sprint 3	22/04/20	06/05/20	15 días
5	Sprint 4	10/05/20	24/05/20	15 días
6	Sprint Release	29/05/20	01/06/20	4 días
7	Entrega de proyecto	12/06/20	12/06/20	1 día

De acuerdo a la metodología de trabajo Scrum Express definimos una planificación dinámica basada en sprints, que son iteraciones de tiempo fijo. En la reunión para la planificación del sprint se decide, junto con el cliente del producto, el conjunto de historias de usuarios (estos pueden ser vistos como requerimientos funcionales) que van a ser desarrolladas durante el Sprint, considerando solo los días laborales para esta planificación (lunes a viernes).



CRONOGRAMA DE ACTIVIDADE

Mes	Marzo				Abril				Mayo				Junio			
Semana	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Sprint 0																
Sprint 1																
Sprint 2																
Sprint 3																
Sprint 4																
Sprint Release																
Días	11/03-15/03		16/03 - 31/03		04/04 - 18/04			22/04 - 06/05			10/05 - 24/06		10/06 - 13/06			

4 PROPUESTA ECONÓMICA Y PLAN DE PAGOS

4.1 ESTIMACIÓN DE COSTOS DE PROYECTO

Se realizó la estimación de costos del proyecto en base a los **PUNTOS DE FUNCIÓN** dependiendo de la complejidad de los mismos

Tipo de Función	Descripción
Entrada Externa (EI)	Pantallas donde los usuarios ingresan datos
Salida Externa (EO)	Informes, gráficos, listado de datos
Consulta Externa (EQ)	Recuperar y mostrar datos del usuario (Buscar)
Archivo Lógico Interno (ILF)	Archivo del punto de vista lógico, no como en un sistema operativo, puede ser tablas en la base de datos
Archivo de Interfaz Externo (EIF)	Datos referenciados a otros sistemas, datos mantenidos por otros sistemas pero usados por el sistema actual

Cálculo de PFNA (Puntos de función no ajustados).

Tipo función	Complejidad			Complejidad esperada	Peso	Complejidad esperada
	Simple	Medio	Complejo			
(EI) Entrada Externa	3	4	6	6	13	78
(EO) Salida Externa	4	5	7	5	5	25
(EQ) Consulta Externa	3	4	6	6	6	36
(ILF) Archivo Lógico Interno	7	10	15	10	5	50
(EIF) Archivo de Interfaz externo	5	7	10	10	3	30
TOTAL PF NO AJUSTADOS (PFNA)						219

4.2 CÁLCULO DE LA COMPLEJIDAD

Para el cálculo de ajuste de complejidad se tomará en cuenta la siguiente tabla de puntuación:

Ausente	Insignificante	Moderada	Media	Significante	Fuerte
0	1	2	3	4	5

Grado de influencia del factor de la complejidad:

factor		Puntaje
1	Comunicación de datos	5
2	Procesamiento distribuido	3
3	Objetivos de Rendimiento	3
4	Configuración del equipamiento	2
5	Tasa de transacciones	2
6	Entrada de datos en línea	5
7	Interface con el usuario	5
8	Actualización en línea	3
9	Procesamiento complejo	3
10	Reusabilidad del código	4
11	Facilidad de implementación	3
12	Facilidad de operación	4
13	Instalaciones Múltiples	4
14	Facilidad de cambios	5
ΣFi		51

$$\text{Puntos de Función} = PFNA \times \left(0.65 + 0.01 \times \sum F_i \right)$$

$$\text{Puntos de Función} = 219 \times (0.65 + 0.01 \times 51)$$

$$\text{Puntos de Función} = 254$$

Según los cálculos realizados se tendrán 254 **PUNTOS DE FUNCIÓN** en el software a desarrollar.



4.3 PRODUCTIVIDAD

El tiempo estimado para el desarrollo del software será de 3 meses, tomando en cuenta de lunes a viernes, durante 13 semanas (65 días) a partir de la firma del contrato, se considerará como una jornada de trabajo 3 horas diarias y el número de miembros a trabajar es de 6 personas en el equipo de trabajo.

Para realizar el cálculo de productividad por meses se tomará en cuenta 3 meses de trabajo a partir de la fecha de firma del contrato

$$\text{Productividad} = 254 [PF] \div 6 \text{ Personas} \div 3 \text{ meses} = 14 \frac{PF}{\text{Persona}}$$

4.4 COSTOS DE RECURSOS HUMANOS

Costo Hora de Trabajo	Horas Trabajadas al Día	Días Trabajados a la Semana	Total en Semanas de Trabajo
3 \$us	3 horas	5 días	13 semanas

$$\text{Sueldo del Programador} = 3 \times 3 \times 5 \times 13 = 585 \$us$$

$$\text{Sueldo Programador por mes} = 585 \div 3 = 195 \$us$$

Tomando en cuenta los puntos de Función, el sueldo del programador y el esfuerzo requerido en el desarrollo del software se llegó a los siguientes resultados:

$$\text{Costo Punto de Funcion} = \frac{\text{Sueldo Programador por Mes}}{\text{Productividad}}$$

$$\text{Costo Punto de Función} = \frac{195}{14} = 14$$

$$\text{Costo Total} = \text{Puntos de Función} \times \text{Costo Punto de Función}$$

$$\text{Costo Total} = 254 \times 14 = 3556 \$us$$



PF	Productividad Esperada (PF/Persona)	Costo Persona / Mes (\$)	Costo de cada PF (\$)	Costo Total (\$)
254	14	195	14	3556

4.5 COSTOS OPERATIVOS

En la siguiente tabla se detalla el costo operativo mensual

Descripción	Monto
Luz	35 \$
Agua	10 \$
Internet	60 \$
Alquileres	265 \$
Mantenimiento	20 \$
Otros	20 \$
TOTAL	410 \$

Desglosando el Costo operativo mensual a costo operativo semanal (para esto suponemos que un mes comercial tiene 4 semanas), se tendría:

$$\frac{410}{4} = 102.5 \frac{\$us}{Semana}$$

Después de determinar nuestro costo operativo semanal obtendremos el costo operativo en el transcurso del desarrollo del software:

$$102.5 \times 13 = 1332.5 \$us$$



4.6 COSTOS DE CAPACITACIÓN A USUARIOS FINALES

Los costos de capacitación serán expuestos en la siguiente tabla:

Ítem	Días	Horas por día	Personas	Costo por hora	Costo total (\$)
Desarrollo de Manual de Usuario para El manejo del Software	1	2	1	3	6
Total					6

4.7 COSTO TOTAL

Teniendo en conocimiento los costos de recursos humanos, costos operativos tendremos el siguiente costo total sin incluir las utilidades de la empresa:

$$\text{Costo Total} = \text{Costo RRHH} + \text{Costo Operativo} + \text{Costo Capacitación}$$

$$\text{Costo Total} = 3356 + 1332.5 + 6$$

$$\text{Costo Total} = 4894.5 \text{ \$us}$$

En total la estimación de costos expresada en \$us y bolivianos será la siguiente:

	Costo en \$	Costo en Bs t/c 6.96
Costo Recursos Humanos	3556	24749.76
Costo Operativo	1332.5	9274.2
Costos de capacitación a usuarios	6	34.8
Costo Total	4894.5	34065.72

El costo total de proyectos según la estimación realizada asciende a **4894.5 \$us** (cuatro mil ochocientos noventa y cuatro 50/100 dólares americanos) este monto expresado en bolivianos ascendería a **34065.72 Bs** (treinta cuatro mil sesenta y cinco 72/100 bolivianos)



4.8 PLAN DE PAGOS

El plan de pagos se puede negociar con el cliente. La Empresa ACTIONSOFT S.R.L. a manera de facilitar el trabajo con la Empresa "TIS" plantea la siguiente propuesta: Realizar pagos parciales a medida que se vayan concluyendo los Sprints, definiendo las cuotas para cada pago.

La forma de pago tentativa y los montos estimados, se muestran en la Tabla.

Descripción	Porcentaje (%)	Monto (\$)	Monto (Bs)
1 Sprint de desarrollo	20	1036	7210.56
2 Sprint de desarrollo	25	1295	9013.2
3 Sprint de desarrollo	25	1295	9013.2
4 Sprint de desarrollo	30	1554	10815.84
TOTAL	100	5180	36052.8

5 ADMINISTRACION DE RIESGO

5.1 IDENTIFICACION DE RIESGO

La empresa ACTIONSOFT S.R.L. clasifica el impacto de la siguiente forma:

Impacto
Baja
Media
Moderada
Alta



5.1.1 RIESGOS TECNOLOGICOS

En esta categoría mencionaremos los riesgos de alguna falla de las herramientas de trabajo, tanto como el hardware, base de datos, los componentes que se reutilizarán, etc.

Código	Riesgo	Probabilidad Ocurrencia	Impacto	Causas probables
R1	Las Computadoras de trabajo presentan fallos en su funcionamiento	20%	MEDIO	<ul style="list-style-type: none">- Infección de Virus- Actualización de programas- No realización de mantenimiento
R2	Vulnerabilidad en la seguridad del software.	20%	MEDIO	<ul style="list-style-type: none">-Ataque de intruso al software.-Información que se almacena en el sistema
R3	-Pérdidas de la información	30%	MODERAD O	<ul style="list-style-type: none">-Cortes de energía eléctrica-No realización de copias de seguridad(backup).
R4	No Estar en constante actualización con las nuevas tecnologías	30%	MEDIO	<ul style="list-style-type: none">- Desempeño no deseado



5.1.2 RIESGOS ASOCIADOS AL EQUIPO DE TRABAJO

En cuanto a esta categoría de los riesgos asociados al equipo de trabajo mencionaremos los riesgos que el equipo de trabajo ocasiona durante el desarrollo del proyecto.

Código	Riesgo	Probabilidad Ocurrencia	Impacto	Causas probables
R5	- Falta de conocimiento o capacitación para desarrollar el proyecto	35%	ALTO	-Poco interés de los integrantes. -Personal inexperto en el desarrollo de software
R6	-Baja moral del personal, falta de confianza individual o grupal	30%	MODERADO	-Nunca se desarrolló un proyecto similar -Temor a los retos desconocidos
R7	-Abandono del integrante durante el desarrollo del software	50%	ALTO	-Irresponsabilidad de los integrantes -No tener compromiso con el equipo. -Causa externa más allá del alcance de los integrantes



5.1.3 RIESGOS RELACIONADOS CON LOS REQUERIMIENTOS DEL CLIENTE

Se podría presentar en este punto una mala interpretación de los principales requerimientos del software debido a cambios con alto impacto en el diseño

Código	Riesgo	Probabilidad Ocurrencia	Impacto	Causas probables
R8	- No tiene una idea clara de lo que se realizara.	40%	ALTO	-Cliente Indeciso -Falta de Comunicación con el cliente
R9	-Requerimientos demasiados complejos	30%	ALTO	-No hacer saber al cliente de las limitaciones. -No establecer limitaciones en el desarrollo de software.
R10	-Cambios o adición de nuevos requerimientos	30%	MODERADO	-No se logró capturar de manera óptima los requisitos del cliente. -No satisfacer las expectativas del cliente.



5.1.4 RIESGOS RELACIONADOS CON LA ESTIMACIÓN

Uno de los principales problemas en este punto sería tomar en cuenta la falta de financiamiento del proyecto puesto que este factor es de suma importancia para el desarrollo de este. Además, se tomará cuenta el factor del tiempo y dinero estimado. El factor de tiempo es importante ya que ahí se establecerá las fechas de entrega del proyecto.

Código	Riesgo	Probabilidad Ocurrencia	Impacto	Causas probables
R11	- Subestimación del tiempo de duración total del proyecto	35%	ALTO	-Mala Planificación de actividades. -No cumplir con las actividades en el tiempo establecido.
R12	-Subestimación a cerca de los recursos a utilizarse en dinero	20%	MEDIO	-Mala Planificación de estimación de costos del proyecto

5.1.5 PLAN DE CONTINGENCIA CONTRA RIESGOS

Para evitar los riesgos presentados anteriormente se tiene un plan para evitar riesgos que pueden ocurrir cuando se esté realizando el proyecto.

Código	Riesgo	Plan de contingencia
R1	Las Computadoras de trabajo presentan fallos en su funcionamiento	Instalar un antivirus Realizar mantenimiento Actualización de programas No usar herramientas de fuentes desconocidas
R2	Vulnerabilidad en la seguridad del software.	Negar el acceso a los datos a aquellas personas que no tengan derecho a ellos. Tener privilegios de acceso como ser administrador, usuario, cliente.
R3	Pérdidas de la información	Realizar copias de seguridad(backup). Utilizar un repositorio en la nube(GitHub)
R4	No Estar en constante actualización con las nuevas tecnologías	Estar en constante actualización. Saber que lenguajes de programación son más valoradas.
R5	Falta de conocimiento o capacitación para desarrollar el proyecto	Usar herramientas, lenguajes, frameworks conocidos o fáciles de aprender



R6	Baja moral del personal, falta de confianza individual o grupal	Tener los objetivos claros que se realizaran. Establecer una mejor comunicación entre los integrantes del equipo.
R7	Abandono del o los integrantes durante el desarrollo del software	Tener compromiso con la empresa
R8	No tiene una idea clara de lo que se realizara.	Tener Conversaciones con el cliente El cliente debe ser parte del equipo de desarrollo
R9	Requerimientos demasiados complejos	Dividir los requerimientos, en más específicos. El requerimiento debe ser claro.
R10	Cambios o adición de nuevos requerimientos	Conversar con el cliente cuando no esté de acuerdo. El cliente debe tener claro los requerimientos.
R11	Subestimación del tiempo de duración total del proyecto	Planificar de manera adecuada el tiempo en que se realizara el proyecto.
R12	Subestimación a cerca de los recursos a utilizarse en dinero	Conocer las capacidades de los miembros del equipo.