****

**UNIVERSIDAD PRIVADA DE TACNA**

**FACULTAD DE INGENIERIA**

**Escuela Profesional de Ingeniería de Sistemas**

**Informe Final**

**Proyecto Sistema Web de Registros de Salud - SIRESA**

Curso: Patrones de Software

Docente: *Mag. Patrick Cuadros Quiroga*

Integrantes:

***Ancco Suaña, Bruno Enrique (2023077472)  
Huaman Rivera, Roberto Carlos  (2021071077)***

***Rivera Muñoz, Augusto Joaquin (2022073505)***

**Tacna – Perú**

***2025***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CONTROL DE VERSIONES | | | | | |
| Versión | Hecha por | Revisada por | Aprobada por | Fecha | Motivo |
| 1.0 | EOLP | EOLP/RCHR | MPV | 19/06/2025 | Versión Original |
| 2.0 | EOLP | RCHR/EOLP | MPV | 23/06/2025 | Versión Final |

INDICE GENERAL

1. Antecedentes 1
2. Planteamiento del Problema 4
   1. Problema
   2. Justificación
   3. Alcance
3. Objetivos 6
4. Marco Teórico
5. Desarrollo de la Solución 9
   1. Análisis de Factibilidad (técnico, económica, operativa, social, legal, ambiental)
   2. Tecnología de Desarrollo
   3. Metodología de implementación

(Documento de VISION, SRS, SAD)

1. Cronograma 11
2. Presupuesto 12
3. Conclusiones 13

Recomendaciones 14

Bibliografía 15

Anexos 16

Anexo 01 Informe de Factibilidad

Anex0 02 Documento de Visión

Anexo 03 Documento SRS

Anexo 04 Documento SAD

Anexo 05 Manuales y otros documentos

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CONTROL DE VERSIONES | | | | | |
| Versión | Hecha por | Revisada por | Aprobada por | Fecha | Motivo |
| 1.0 | SNMY | EOLP/RHR/JLR | MPV | 07/05/2025 | Versión Original. |
| 2.0 | SNMY | EOLP/RHR/JLR | MPV | 12/05/2025 | Completar documento. |
| 3.0 | SNMY | RHR/JLR/EOLP | MPV | 22/05/2025 | Versión Final |

**Sistema Web de Registro de Salud - SIRESA**

**Documento de Especificación de Requerimientos de Software**

**Versión 1.0**

# Antecedentes

# La gestión de la configuración es clave para el éxito de cualquier proyecto de Tecnologías de la Información y Comunicación (TIC). Sin una gestión efectiva, se corre el riesgo de perder la integridad de los elementos configurados y no poder reportar su estado y configuración. Un plan de gestión de la configuración nos permite establecer un método consistente para identificar y controlar formalmente todos los elementos configurados del proyecto, incluyendo hardware, software, planes y presupuestos.

# La gestión de la configuración es fundamental para proteger los elementos configurados y comunicar los cambios que se han realizado sobre ellos. Al hacerlo de manera planificada y efectiva, contribuye a la producción de productos de alta calidad, evitando el doble trabajo y aumentando el valor de los activos informáticos. Además, ahorra costos y contribuye a la entrega de proyectos que satisfacen los requisitos establecidos en cuanto a costos, calendarios, calidad y requerimientos.

# Un plan de gestión de la configuración es una herramienta fundamental de planificación que describe los esfuerzos necesarios para implementar y ejecutar la gestión de la configuración a lo largo del ciclo de vida del proyecto. Proporciona visibilidad y control sobre el producto, incluyendo su desempeño, funcionalidad y atributos físicos. La gestión de la configuración garantiza la integridad de los elementos configurados y facilita la gestión ordenada de la información de las líneas base y los cambios que se realizan en ellas.

# Planteamiento del Problema

# En el desarrollo de proyectos de software, es común enfrentarse a diversos contratiempos, como retrasos en la elaboración de documentos y la necesidad de gestionar manualmente las actividades del proyecto. Además, la falta de comunicación por parte del líder del proyecto respecto a los cambios y versiones puede generar complicaciones. También, los retrasos en la gestión de cambios, debido a un proceso de preselección y diversificación, que son más frecuentes.

# En lo que respecta al cliente, la entrega tardía del estado y producto solicitado y su eventual insatisfacción son problemas recurrentes.

# Por ende, se requiere el desarrollo de un sistema de base de datos que centralice e integre los registros de salud relacionados con los establecimientos fiscalizados, permitiendo una gestión eficiente, transparente y accesible de la información, tanto para las autoridades sanitarias como para los propios establecimientos involucrados así como los usuarios que buscan información dirigido al rubro turismo.

# Problema

# Actualmente, tenemos la falta de un sistema capaz de controlar eficientemente las versiones y registrar los cambios, lo que genera ineficiencias en nuestros procesos. Nuestra propuesta es desarrollar un sistema que mejore la gestión de proyectos, agilizando su finalización y permitiendo un seguimiento transparente de las modificaciones efectuadas por el equipo.

# Este sistema ofrecerá la posibilidad de administrar múltiples proyectos a través de una interfaz intuitiva y fácil de usar. Además, garantizará el acceso inmediato y actualizado a la información desde cualquier lugar con conectividad a la base de datos.

# Justificación

# El desarrollo e implementación del Sistema para la Gestión y Administración de la Configuración de Software representa una inversión estratégica para la organización, ya que permitirá mejorar significativamente la eficiencia y la efectividad de los procesos relacionados con el desarrollo de proyectos de software. Al automatizar tareas clave como la gestión de usuarios, entregables y cambios, el sistema contribuirá a reducir los tiempos de ejecución y a mejorar la calidad del servicio ofrecido a los clientes.

# Alcance

El sistema web abarca las siguientes funcionalidades y características dentro de su primera fase de implementación:

* Registro de establecimientos: Ingreso de datos generales de cada local inspeccionado (nombre, dirección, tipo de servicio, etc.).
* Gestión de fiscalizaciones: Registro de visitas sanitarias con resultados detallados de la inspección.
* Asignación de estado sanitario: Clasificación visual del estado del establecimiento mediante un esquema de colores:
* Verde: Aprobado
* Rojo: Rechazado
* Amarillo: En proceso de revisión
* Consulta y búsqueda: Interfaz para que los usuarios autorizados puedan filtrar establecimientos por estado, ubicación, tipo, y fechas de inspección.
* Historial de visitas: Acceso al historial de fiscalizaciones por establecimiento.
* Control de acceso por roles: Módulos diferenciados para inspectores, supervisores y administradores.
* Soporte para reportes: Generación de reportes mensuales y exportación de datos relevantes para auditorías o análisis estadístico.

# 3. Objetivos

# Objetivo general

# El objetivo radica en la automatización de la Gestión de Configuración en proyectos de software, con la meta de reducir tanto los gastos como el tiempo invertido, asegurando la satisfacción del cliente y ofreciendo un servicio de calidad óptima. Esto implica mantener un estricto control sobre la seguridad de los documentos utilizados y garantizar su total accesibilidad en todo momento.

# Objetivos Específicos

* Crear una aplicación informática para facilitar la gestión de los proyectos asignados.
* Desarrollar un software capaz de llevar a cabo la administración de Metodologías.
* Elaborar un sistema que enumera de forma organizada los proyectos asignados.
* Diseñar un programa que se encargue de la gestión de usuarios de manera eficiente.
* Desarrollar una herramienta que proporcione la administración de documentos y sus respectivas versiones de manera efectiva.

**4. Marco Teórico**

# 4.1 Calidad de software

# Características propias del software aquellas que tú quieres controlar y asegurar, el software es un producto inmaterial que no se fabrica, tampoco se degradan físicamente, sino que se desarrolla. El software puede tener errores, incidencias, pero no son similares a lo que cualquier equipo de carácter físico.

# La calidad del software se encuentra casi a el par de la calidad tradicional, ligeramente detrás debido a que la calidad tradicional tiene varias décadas de historia, mientras que la calidad de software tiene entre 50 y 30 años de haber surgido.

# 4.2 Normativa ISO 9000

# Pone a disposición de un auditor o certificador los procesos internos, de forma que este indique si cumple o no la normativa al 100%, audita el sistema; Si los resultados son positivos se emite la certificación y cada cierto tiempo se tiene que renovar; La certificación es costosa, a consecuencia de costes que ocasionan la lejanía y el tiempo de duración de proceso (aprox. 6 meses). Se certifica la empresa y la metodología para el desarrollo de la aplicación.

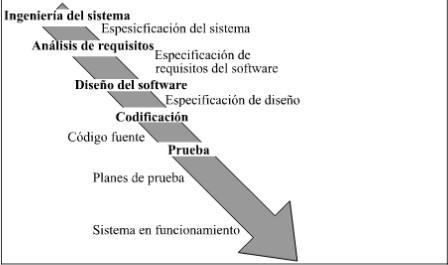
# 4.3 Medición del software

# En el software lo que se mide son atributos propios del mismo, se descompone un atributo general en otros más simples de medir, a veces se mide bien o mal ya que la descomposición del atributo genérico de calidad en otros sub atributos se torna irreal, se mide con datos estadísticos no avalados, es imposible decir que la medición se hace en forma correcta.

# El concepto de medida va de más a menos, va de lo general a lo concreto y lo concreto está asociado a la métrica, cuya combinación te daría el nivel de calidad o seguridad de tu producto. Las ciencias bien estructuradas se basan en medidas bien hechas, se basan en la matemática.

# 4.4 Líneas Base

# Una línea base es un concepto de gestión de la configuración que lleva a controlar los cambios sin impedir seriamente los cambios justificados. El Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) 610.12/1990 define una línea base como: Una especificación o producto que se ha revisado formalmente y sobre los que se ha llegado a un acuerdo, y que de ahí en adelante sirve como base para un desarrollo posterior y que puede cambiarse solamente a través de procedimientos formales de control de cambios.



# 4.5 Elementos de la Configuración

# Un elemento de la configuración del software es la información creada como parte del proceso de ingeniería un ECS (elemento de configuración de software) es un documento, un conjunto completo de casos de prueba o un componente de un programa 40 dado.

# Los ECS se organizan como objetos de configuración que deben ser catalogados por la base de datos del proyecto con un único nombre. Un ECS tiene un nombre y atributos, y está conectado a otros objetos mediante relaciones.

# La siguiente figura muestra el modelo de datos de los elementos de la configuración, cada objeto está relacionado con otro, si se lleva a cabo un cambio sobre un objeto las interrelaciones señalan a que otros objetos afectará

# 4.6 Gestión de Configuración de Software

# La GCS es un elemento importante de garantía de calidad y es responsable de controlar los cambios. Sin embargo, también se debe identificar los ECS individuales. El proceso se puede definir en cinco tareas de CGS:

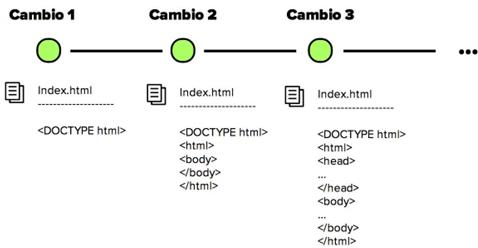
# Identificación.

* Control de versiones.
* Control de cambios.
* Auditorías de configuración.
* Generación de informes.

# 4.7 Proceso de Gestión de Configuración de Software

# Se denomina Gestión de la Configuración al conjunto de procesos destinados a asegurar la calidad de todo producto obtenido durante cualquiera de las etapas del desarrollo de un Sistema de Información (S.I.), a través del estricto control de los cambios realizados sobre los mismos y de la disponibilidad constante de una versión estable de cada elemento para toda persona involucrada en el citado desarrollo.

# Estos dos elementos (control de cambios y control de versiones de todos los elementos del S.I.) facilitan también el mantenimiento de los sistemas al proporcionar una imagen detallada del sistema en cada etapa del desarrollo. La gestión de la configuración se realiza durante todas las fases del desarrollo de un sistema de información, incluyendo el mantenimiento y control de cambios, una vez realizada la puesta en producción.



# 4.8 Identificación de Objetos en GCS

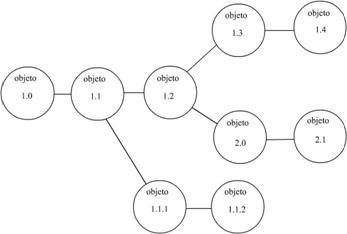
# Se pueden identificar dos tipos de objetos: los objetos básicos y los objetos compuestos. Un objeto básico es una unidad de texto creada durante el análisis, diseño, codificación o prueba. Un objeto compuesto es una colección de objetos básicos u objetos compuestos.

# Cada objeto tiene un conjunto de características que los identifican como únicos. El nombre del objeto es una cadena de caracteres que identifica al objeto sin ambigüedad. La descripción del objeto es una lista de elementos de datos que identifican:

# El tipo de ECS (documento, programa, datos) que está representado por el objeto.

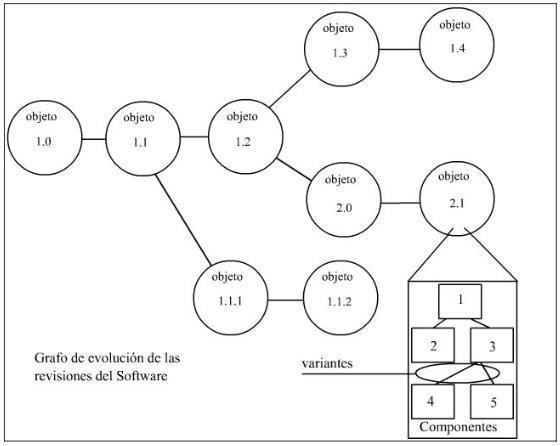
# Un identificador del proyecto; y la información de la versión y/o el cambio.

* El esquema de identificación de los objetos de software debe tener en cuenta que los objetos evolucionan a lo largo del proceso de ingeniería, por lo que se puede crear un grafo de evolución



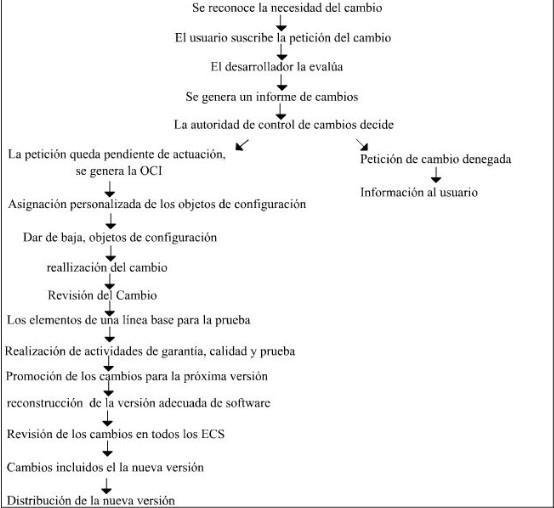
# 4.9 Control de Versiones

# El control de versiones combina procedimientos y herramientas para gestionar las versiones de los objetos de configuración creadas durante el proceso de ingeniería del software. "La gestión de configuración permite a un usuario especificar configuraciones alternativas del sistema de software mediante la selección de las versiones adecuadas. Esto se puede gestionar asociando atributos a cada versión del software y permitiendo luego especificar y construir una configuración describiendo el conjunto de atributos deseado. "Los atributos pueden ser tan sencillos como un número específico de versión asociado a cada objeto o tan complejos como una cadena de variables lógicas que especifiquen tipos de cambios funcionales aplicados al sistema.



# 4.10 Control de cambios

# En un gran proyecto de desarrollo de software, el cambio incontrolado lleva rápidamente al caos. El control de cambios combina los procedimientos humanos y las herramientas automáticas para proporcionar un mecanismo para el control de cambio.



# 4.11 Auditoría de la Configuración

# La autoridad de control de cambios (ACC) desempeña un papel activo en el segundo y tercer nivel de control. El papel de la ACC es el de tener una visión general, o sea, evaluar el impacto del cambio fuera del ECS en cuestión.

# ¿Cómo impactará el cambio en el hardware? ¿Cómo impactará en el rendimiento? ¿Cómo alterará el cambio la percepción del cliente sobre el producto?¿Cómo podemos asegurar que el cambio se ha implementado correctamente? La respuesta es doble: 1) revisiones técnicas formales y 2) auditorías de configuración del software.

# Las revisiones técnicas formales se centran en la corrección técnica del elemento de configuración que han sido modificados. Los revisores evalúan el ECS para determinar la consistencia con otros ECS, las omisiones o los posibles efectos secundarios.

# Una auditoría de configuración del software complementa la revisión técnica formal al comprobar características que generalmente no tiene en cuenta la revisión. La auditoría se plantea y responde con las siguientes preguntas:

* ¿Se ha hecho el cambio especificado en la OCI?
* ¿Se han incorporado modificaciones adicionales?
* ¿Se ha llevado a cabo una revisión técnica formal para evaluar la corrección técnica?
* ¿Se han seguido adecuadamente los estándares de ingeniería de software?
* ¿Se han "recalcado" los cambios en el ECS?
* ¿Se han especificado la fecha del cambio y el autor? ¿Reflejan los cambios los atributos del objeto de configuración?
* ¿Se han seguido procedimientos del GCS para señalar el cambio, registrarlo y divulgarlo?
* ¿Se han actualizado adecuadamente todos los ECS relacionados?

# 4.12 Informe de Estado

# La generación de informes de estado de la configuración es una tarea de GCS que responde a las siguientes preguntas:

1. ¿Qué pasó?
2. ¿Quién lo hizo?
3. ¿Cuándo pasó?
4. ¿Qué más se vio afectado?

La generación de informes de estado de la configuración desempeña un papel vital en el éxito del proyecto de desarrollo de software. Cuando aparece involucrada mucha gente es muy fácil que no exista una buena comunicación. Pueden darse errores entre las personas desarrolladoras del software. El IEC ayuda a eliminar esos problemas, mejorando la comunicación entre todas las personas involucradas.

# 5. Desarrollo de la Solución

# La implementación del proyecto tiene como objetivo desarrollar un sistema de gestión de configuración para proyectos. Se presenta una evaluación de la factibilidad del proyecto, incluyendo un análisis de los requisitos, la viabilidad técnica y la factibilidad financiera.

# Documento de Factibilidad:

* 1. **Análisis de Requisitos**
     + Requisitos funcionales
     + Requisitos no funcionales

# Análisis de Factibilidad Financiera

* + - El presupuesto asignado es suficiente para cubrir los costos del proyecto (desarrollo, implementación y mantenimiento).
    - La inversión en el proyecto generará un retorno sobre la inversión.

1. **Análisis de Factibilidad (técnico, económica, operativa, social, legal, ambiental)**

**Factibilidad Técnica:**

En esta sección, hemos examinado los componentes que el sistema utilizará. En lo que respecta al hardware, se están considerando equipos de escritorio con acceso a Internet, los cuales ya están disponibles para los miembros del equipo. En cuanto al software que se emplea para la implementación del sistema, se detalla a continuación.

En esta ocasión, se presume que el cliente dispone de un dispositivo como una laptop, por lo que este costo no será tomado en cuenta.

|  |  |  |
| --- | --- | --- |
| **Estructura** | **Recurso** | **Descripción** |
| Hardware | PC | Procesador: Intel Core i5 Memoria RAM: 4 GB  Almacenamiento: 500 GB HDD |
| Software | Sistema  Operativo | Windows 10 Home |

# Factibilidad Económica:

**Costos Generales**

Los costos relacionados con accesorios y material de oficina de uso diario son los siguientes:

|  |  |
| --- | --- |
| Costos Generales | Costo Total |
| Provisiones de Oficina | 1000.00 |
| TOTAL | 1000.00 |

# Costos operativos durante el desarrollo

# Estos gastos están asociados a los recursos que utilizaremos y otros pagos por servicios necesarios para llevar a cabo nuestro trabajo. A continuación, se presenta la tabla correspondiente:

|  |  |  |
| --- | --- | --- |
| Concepto de Pago | Actividades | Costo Total |
| Alquiler de Laptops (5) | Equipos para el desarrollo del  sistema | 2000.00 |
| Servicio de Energía Eléctrica (5) | Suministro de energía eléctrica | 1000.00 |

|  |  |  |
| --- | --- | --- |
| Servicio de Internet (5) | Servicio de conexión a internet | 1200.00 |
| TOTAL | | 4,200.00 |

# Factibilidad Operativa:

# Es crucial abordar con prontitud las dificultades prioritarias identificadas, dado que los procesos se llevan a cabo de forma cotidiana y, en muchos casos, de manera desigual. En este sentido, la creación de un sistema que automatice estas tareas se presenta como una solución imprescindible para mejorar la eficiencia y la equidad en la ejecución de los procesos.

# Factibilidad Social:

# La aceptación y la colaboración de los usuarios son elementos clave para el éxito del proyecto. En este sentido, se ha establecido un compromiso por parte de la empresa para entregar el sistema dentro del plazo acordado y dentro del presupuesto asignado. Además, se brindará capacitación sobre el uso del sistema y se ofrecerá mantenimiento sin costo durante el primer año, lo que contribuirá a garantizar una transición suave y una adopción efectiva por parte de los usuarios.

# Factibilidad Legal:

# Es fundamental asegurar el cumplimiento de las normativas legales pertinentes, en particular la Ley Orgánica de Protección de Datos (LOPD), para garantizar la confidencialidad y seguridad de los datos personales de los usuarios. Por tanto, se implementarán medidas de seguridad adecuadas y se adquirirán las licencias del software de manera legítima, evitando así posibles repercusiones legales en el futuro.

**Factibilidad Ambiental:**

Es esencial evaluar y considerar los posibles impactos ambientales del proyecto, tanto durante su desarrollo como en su implementación. Se llevará a cabo una evaluación exhaustiva de las condiciones ambientales, incluyendo aspectos como la geografía, el clima y el uso de recursos naturales, con el fin de minimizar cualquier efecto negativo sobre el entorno. Asimismo, se adoptarán prácticas sostenibles, como el uso de materiales reciclables, para reducir el impacto ambiental del proyecto.

# Tecnología de Desarrollo

El desarrollo del sistema web de Gestión de la Configuración de Software se ha llevado a cabo utilizando tecnologías modernas que permiten escalabilidad, facilidad de mantenimiento y una experiencia de usuario eficiente. A continuación, se detallan las herramientas, lenguajes y frameworks empleados en cada capa del sistema:

5.1. Lenguaje de Programación

C# (.NET Framework): Se utilizó C# en conjunto con el framework ASP.NET MVC para el desarrollo de la lógica de negocio y controladores del sistema, lo que permitió una separación clara entre lógica, vista y modelo.

5.2. Frameworks y Librerías

ASP.NET MVC 5: Framework principal para el desarrollo del backend web, permitiendo seguir el patrón MVC para una mejor organización del código.

Entity Framework: ORM utilizado para facilitar el acceso y manipulación de datos desde la base de datos SQL Server.

Bootstrap 5: Librería frontend utilizada para el diseño responsivo y moderno de las interfaces.

jQuery y DataTables: Usados para manejar interacciones del usuario y la visualización dinámica de tablas con búsqueda, paginación y exportación.

5.3. Base de Datos

SQLSever: Sistema de gestión de base de datos relacional, utilizado para almacenar la información del sistema, tales como usuarios, proyectos, elementos de configuración, fases, metodologías, cronogramas y solicitudes de cambio.

5.4. Herramientas de Desarrollo

Visual Studio 2022: Entorno de desarrollo integrado utilizado para la programación, depuración y administración del proyecto.

SQL Server Management Studio (SSMS): Herramienta empleada para la gestión de la base de datos.

Git y GitHub: Control de versiones y gestión colaborativa del código fuente.

Postman (opcional): Para pruebas de servicios y validación de endpoints.

5.5. Arquitectura del Sistema

El sistema sigue una arquitectura MVC (Modelo-Vista-Controlador) para garantizar la separación de responsabilidades. Además, se implementa una arquitectura cliente-servidor donde el cliente (navegador) se comunica con el servidor a través de HTTP, utilizando formularios y peticiones AJAX.

5.6. Otros Recursos

CSS personalizado: Para complementar el diseño visual del sistema más allá de las capacidades de Bootstrap.

FontAwesome: Librería de íconos para enriquecer visualmente los componentes del sistema.

1. **Metodología de implementación (Documento de VISIÓN, SRS, SAD)**

La implementación del proyecto seguirá un enfoque estructurado basado en los siguientes documentos y fases:

* Documento de Visión (VISIÓN): Este documento definirá los objetivos del proyecto, el público objetivo, los requisitos del usuario y la visión general del producto final.
* Especificación de Requisitos de Software (SRS): El SRS detalla todos los requisitos funcionales y no funcionales del sistema, incluyendo interfaces de usuario, requisitos de rendimiento, y restricciones.
* Diseño de Arquitectura de Software (SAD): El SAD describe la arquitectura del sistema, incluyendo el diseño de la base de datos, la estructura del código, y los componentes de software.

**6. Cronograma**

|  |  |  |  |
| --- | --- | --- | --- |
| ELEMENTO | FASE | ELEMENTO DE  CONFIGURACIÓN DE SOFTWARE | FECHA |
| ANÁLISIS PRELIMINAR | | Informe de Factibilidad | 18/04/2024 |
| Plan de Gestión de la  Configuración | 18/04/2024 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Informe de Visión | 23/04/2025 |
| Estándares de Programación | 25/04/2025 |
| Plan de Gestión de Riesgos | 04/05/2025 |
| RUP | INICIO | Especificación de  Requerimientos | 09/05/2025 |
| Diagrama de Caso de Uso | 11/05/2025 |
| Diagrama de Paquetes | 16/05/2025 |
| Narrativa de Caso de Uso | 16/05/2025 |
| Diagrama de Clases | 16/05/2025 |
| Diagrama de Entidad  Relación | 17/05/2025 |
| Diagrama de Componentes | 22/05/2025 |
| ELABORACIÓN | Diseño de Prototipos | 28/05/2025 |
| Diagrama de Secuencia | 29/05/2025 |
| Diagrama Físico de Base de  Datos | 31/05/2025 |
| Diagrama de Despliegue | 02/06/2025 |
| CONSTRUCCIÓN | Solución del Proyecto - Caso  de Usos |  |
| Pruebas del Sistema |  |
| TRANSICIÓN | Acta Informe Final |  |

**7. Presupuesto**

|  |  |
| --- | --- |
| Descripción | Monto |

|  |  |
| --- | --- |
| Costos generales | S/. 1000.00 |
| Costos operativos | S/.4200.00 |
| Costos del ambiente | S/.1000.00 |
| Costos del personal | S/.2100.00 |
| Costo total | S/.8300 .00 |

**8. Conclusiones**

* Se ha demostrado que el sistema propuesto cumple con todos los requisitos funcionales y no funcionales definidos, proporcionando soluciones escalables y flexibles que se alinean con las expectativas de los stakeholders.
* La implementación de este proyecto será beneficioso y cumplirá con los objetivos propuestos.
* Los desafíos encontrados en la conceptualización de la gestión de cambios y el control de versiones han llevado a la inclusión de un Comité de Control de Cambios, vital para la gestión eficaz de los ECS.
* Se definió la visión del propósito de este proyecto, así como también el alcance de este, que vendría a ser un sistema que agilice el proceso de inventario en el área de almacén.

# Recomendaciones

* Se sugiere implementar un enfoque colaborativo continuo con el cliente y otros interesados principales para refinar y ajustar el desarrollo del SGCS de manera que se alinee estrechamente con las expectativas y necesidades específicas.
* Se recomienda realizar varias entrevistas con el cliente para tener una visión clara de lo que desea que haga el sistema, así como también características no funcionales de la misma, la capacidad de los futuros usuarios y demás datos.

# Bibliografía

Atlassian. (s. f.). Gestión de configuración. <https://www.atlassian.com/es/microservices/microservices-architecture/configuration-management>

IBM. (s. f.). Gestión de configuración de software. <https://www.ibm.com/docs/es/rft/9.5.0?topic=assets-software-configuration-management>

Red Hat. (s. f.). ¿Qué es la gestión de la configuración?. <https://www.redhat.com/es/topics/automation/what-is-configuration-management>

NetApp. (s. f.). Funcionamiento de la gestión de la configuración. <https://www.netapp.com/es/devops-solutions/configuration-management/what-is-configuration-management/>

Guirado, R. (s. f.). Gestión de la Configuración del Software (GCS). Universidad de Almería. <https://w3.ual.es/~rguirado/posi/Tema5-Apartado5.pdf>