



Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

Módulos de planificación y ejecución de servicios científicos técnicos del INRH

Autores: Thalia Rosa Caballero Álvarez
Dasiel Torres Camejo

Tutora: MSc. Madelis Pérez Gil

Cotutora: Ing. Dayana Benítez Betancourt

La Habana, noviembre de 2023

Año 65 de la Revolución

Declaración de autoría

Los autores del trabajo de diploma con título “***Módulos de planificación y ejecución de servicios científicos técnicos del INRH.***”, conceden a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la investigación, con carácter exclusivo. De forma similar se declaran como únicos autores de su contenido. Para que así conste firman la presente a los 21 días del mes de noviembre del año 2023.

Dasiel Torres Camejo

Firma del Autor

Thalia Rosa Caballero Álvarez

Firma del Autor

Madelis Pérez Gil

Firma de la Tutora

Dayana Benítez Betancourt

Firma de la Cotutora

Frase

*“La tecnología por sí sola no basta. También tenemos que poner el corazón” –
Jane Goodall*

Dedicatoria

A nuestros padres...

Agradecimientos

A nuestros padres, tutores y profesores.

Resumen

La metodología previa de obtención de información en la Dirección de Ciencia, Tecnología e Innovación del Instituto Nacional de Recursos Hidráulicos (DCTI-INRH) para la gestión de servicios científico-técnicos del INRH comúnmente llamado proyectos de inversión se fundamentaba en documentos físicos y correos electrónicos. Este enfoque resultaba en la falta de disponibilidad en tiempo real, la recepción fragmentada de datos y la necesidad de un procesamiento manual por parte de los especialistas. Estos desafíos incidían negativamente en la disponibilidad y precisión de la información, afectando la operatividad y la toma de decisiones estratégicas. La investigación, centrada en el desarrollo de módulos específicos de planificación y ejecución, identificados como elementos críticos en el sistema de gestión de servicios científico-técnicos del INRH (proyectos de inversión), concluyó la necesidad urgente de implementar un software altamente configurable. El enfoque se orientó hacia la planificación, aprobación, ejecución y control eficiente del presupuesto asignado. La investigación destacó la importancia de cumplir con requisitos clave, proponiendo análisis detallados, un estudio exhaustivo del estado del arte, la adopción de tecnologías web y la selección estratégica de herramientas y metodologías. El último capítulo abordó el análisis del proceso de ejecución y planificación de los servicios científico-técnicos del INRH, proponiendo la automatización mediante una aplicación web. Se detalló la identificación de errores a través de diseños de casos de prueba y la validación del correcto funcionamiento mediante pruebas de software.

Palabras clave: ejecución, planificación, presupuesto, recursos hidráulicos, servicio.

Abstract

The previous methodology for obtaining information at the Directorate of Science, Technology, and Innovation of the National Institute of Hydraulic Resources (DCTI-INRH) for the management of scientific and technical services at the INRH, commonly referred to as investment projects, was based on physical documents and emails. This approach resulted in a lack of real-time availability, fragmented data reception, and the need for manual processing by specialists. These challenges negatively impacted the availability and accuracy of information, affecting operational efficiency and strategic decision-making. The research, focused on the development of specific planning and execution modules identified as critical elements in the scientific-technical service management system of the INRH for investment projects, concluded the urgent need to implement highly configurable software. The focus was directed towards planning, approval, execution, and efficient control of the allocated budget. The research emphasized the importance of meeting key requirements, proposing detailed analyses, a comprehensive study of the state of the art, the adoption of web technologies, and the strategic selection of tools and methodologies. The final chapter addressed the analysis of the execution and planning process of the scientific-technical services of the INRH, proposing automation through a web application. The identification of errors through test case designs and the validation of proper functioning through software tests were detailed.

Keywords: execution, planning, budget, hydraulic resources, service.

Índice

Introducción	1
Capítulo 1: Fundamentación teórica	6
1.1 Introducción	6
1.2 Gestión de la información	6
1.3 Sistemas de gestión de la información (SGI)	9
1.4 Estado del arte	14
1.4.1 Sistemas de gestión de información en el mundo	15
1.4.2 Sistemas de Gestión de Información en Cuba y en la UCI	17
1.4.3 Conclusiones del estudio del estado del arte	17
1.5 Metodología de desarrollo del software	18
1.5.1. Programación extrema	20
1.6 Herramientas y lenguajes informáticos	22
1.6.1 Marco de trabajo Zeolides 2.3	22
1.6.2 ExtJS 3.1	22
1.6.3 Zend Framework 2.0	23
1.6.4 Doctrine 2.0	23
1.6.5 PostgreSQL 13	23
1.6.6 PgAdmin versión 6.9	24
1.6.7 Visual Paradigm Form UML 8.0	25
1.6.8 Lenguaje de Modelado Unificado UML 5.0	25
1.6.9 TortoiseGit 2.14	26
1.6.10 Visual Studio Code 1.74.2	26
1.6.11 Apache 2.4	27
1.6.12 Apache JMeter 2019.1.1	27
1.6.13 PHP 5.6	27
1.6.14 HTML 5	28
1.6.15 CSS	28
1.6.16 JavaScript 1.6	28
1.6.17 GitLab	29
1.7 Conclusiones del capítulo	29
Capítulo 2: Diseño de la solución propuesta	30
2.1 Introducción	30
2.2 Descripción de la solución del Software	30
2.2.1 Características de la propuesta	31
2.2.2 Roles y funcionalidades	32
2.3 Lista de funcionalidades del sistema	33
2.4 Exploración	35
2.4.1 Historias de usuario (UH)	35
2.5 Planificación	42
2.6.1 Estimación del esfuerzo	42

2.6.2 Plan de iteraciones	43
2.6.3 Plan de Entregas	45
2.7 Diseño	46
2.7.1 Patrón arquitectónico	46
2.7.2 Patrones de diseño	47
2.7.3 Tarjetas CRC	50
2.7.4 Modelo de datos	52
2.8 Conclusiones del capítulo	53
Capítulo 3: Implementación y pruebas	54
3.1 Introducción	54
3.2 Implementación y pruebas	54
3.2.1 Tareas de ingeniería	54
3.3 Pruebas de caja blanca	59
3.3.1 Test unitarios	60
3.4 Pruebas de caja negra	62
3.4.1 Técnica: Partición de equivalencia	62
3.4.2 Pruebas de aceptación	64
3.5 Conclusiones del capítulo	67
Conclusiones	69
Recomendaciones	70
Referencias bibliográficas	72
Anexos	76
Anexo 1 acta de aceptación del proyecto	76
Anexo 2 Tareas de ingeniería	79
Anexo 3 Caso de prueba utilizando la técnica partición de equivalencia	94
Anexo 4 Modelo de datos	96

Índice de tablas

<i>Tabla 1 Tabla comparativa de sistemas homólogos</i>	17
<i>Tabla 2 Historia de Usuario 1</i>	36
<i>Tabla 3 Historia de Usuario 2</i>	36
<i>Tabla 4 Historia de Usuario 3</i>	37
<i>Tabla 5 Historia de Usuario 4</i>	37
<i>Tabla 6 Historia de Usuario 5</i>	38
<i>Tabla 7 Historia de Usuario 6</i>	38
<i>Tabla 8 Historia de Usuario 7</i>	39
<i>Tabla 9 Historia de Usuario 8</i>	39
<i>Tabla 10 Historia de Usuario 9</i>	39
<i>Tabla 11 Historia de Usuario 10</i>	40
<i>Tabla 12 Historia de Usuario 11</i>	40
<i>Tabla 13 Historia de Usuario 12</i>	41
<i>Tabla 14 Historia de Usuario 13</i>	41
<i>Tabla 15 Historia de Usuario 14</i>	41
<i>Tabla 16 Estimación de esfuerzo por Historia de Usuario</i>	43
<i>Tabla 17 Plan de duración de iteraciones</i>	45
<i>Tabla 18 Plan de entregas</i>	45
<i>Tabla 19 Tarjeta CRC #1</i>	51
<i>Tabla 20 Tarjeta CRC #2</i>	51
<i>Tabla 21 Tarjeta CRC #3</i>	51
<i>Tabla 22 Tarjeta CRC #4</i>	52
<i>Tabla 23 Tarjeta CRC #5</i>	52
<i>Tabla 24 Tarjeta CRC #6</i>	52
<i>Tabla 25 Tarea de ingeniería # 1</i>	55
<i>Tabla 26 Tarea de ingeniería # 2</i>	55
<i>Tabla 27 Tarea de ingeniería # 3</i>	56
<i>Tabla 28 Tarea de ingeniería # 4</i>	58
<i>Tabla 29 Tarea de ingeniería # 5</i>	59
<i>Tabla 31 Caso de prueba de aceptación # 1</i>	65
<i>Tabla 32 Caso de prueba de aceptación # 2</i>	65
<i>Tabla 33 Caso de prueba de aceptación # 3</i>	66
<i>Tabla 34 Caso de prueba de aceptación # 4</i>	66
<i>Tabla 36 Tarea de ingeniería # 6</i>	79
<i>Tabla 37 Tarea de ingeniería # 7</i>	79
<i>Tabla 38 Tarea de ingeniería # 8</i>	80
<i>Tabla 39 Tarea de ingeniería # 9</i>	80
<i>Tabla 40 Tarea de ingeniería # 10</i>	81
<i>Tabla 41 Tarea de ingeniería # 11</i>	81
<i>Tabla 42 Tarea de ingeniería # 12</i>	82
<i>Tabla 43 Tarea de ingeniería # 13</i>	82
<i>Tabla 44 Tarea de ingeniería # 14</i>	82
<i>Tabla 45 Tarea de ingeniería # 15</i>	83
<i>Tabla 46 Tarea de ingeniería # 16</i>	84
<i>Tabla 47 Tarea de ingeniería # 17</i>	85
<i>Tabla 48 Tarea de ingeniería # 18</i>	85
<i>Tabla 49 Tarea de ingeniería # 19</i>	86
<i>Tabla 50 Tarea de ingeniería # 20</i>	86
<i>Tabla 51 Tarea de ingeniería # 21</i>	86
<i>Tabla 52 Tarea de ingeniería # 22</i>	87
<i>Tabla 53 Tarea de ingeniería # 23</i>	87

Tabla 54 Tarea de ingeniería # 24	87
Tabla 55 Tarea de ingeniería # 25	88
Tabla 56 Tarea de ingeniería # 26	88
Tabla 57 Tarea de ingeniería # 27	89
Tabla 58 Tarea de ingeniería # 28	89
Tabla 59 Tarea de ingeniería # 29	90
Tabla 60 Tarea de ingeniería # 30	90
Tabla 61 Tarea de ingeniería # 31	91
Tabla 62 Tarea de ingeniería # 32	91
Tabla 63 Tarea de ingeniería # 33	91
Tabla 64 Tarea de ingeniería # 34	92
Tabla 65 Tarea de ingeniería # 35	92
Tabla 66 Tarea de ingeniería # 36	92
Tabla 67 Tarea de ingeniería # 37	93
Tabla 68 Tarea de ingeniería # 38	93
Tabla 69 Caso de prueba para la funcionalidad adicionar servicio ()	94

Índice de ilustraciones

Ilustración 1 Mapa de procesos del negocio	30
Ilustración 2 Patrones Grasp	48
Ilustración 3 Patrones Grasp	49
Ilustración 4 Patrones GoF	50
Ilustración 13 Resultados de pruebas al software	61
Ilustración 14 Resultados de pruebas al software	61
Ilustración 15 Resultado de pruebas funcionales de caja negra	64
Ilustración 16 Resultado de pruebas de aceptación	67
Ilustración 17 Acta de aceptación del proyecto por parte del cliente	77
Ilustración 18 Acta de aceptación del proyecto por parte del cliente	78
Ilustración 19 Modelo de datos	96

Introducción

En un contexto actual caracterizado por la creciente demanda de información y su adaptación a diversos entornos, surge la imperiosa necesidad de asegurar su perdurabilidad en el tiempo y su capacidad de difusión. En este escenario, el papel ha desempeñado un rol fundamental como medio y herramienta de trabajo en el ámbito corporativo. Sin embargo, es crucial tener en cuenta que la producción de papel conlleva la explotación de recursos no renovables en la naturaleza, lo que plantea una llamada de atención en términos de sostenibilidad.

El entorno digital, impulsado por el desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC), ha ganado relevancia con el tiempo. En el contexto empresarial, donde grupos buscan objetivos comunes, la necesidad de comunicación se intensifica. Sin embargo, en corporaciones con individuos en roles científicos, económicos o de interés general, surgen barreras culturales, sociales, tecnológicas, geográficas y temporales que dificultan la comunicación interpersonal. Superar estas barreras es un desafío destacado.

Cuba, como nación en desarrollo, se involucra activamente en la revolución tecnológica digital y busca mejorar constantemente su infraestructura tecnológica, especialmente en el ámbito empresarial. Con metas definidas para el desarrollo de las TIC y la informatización en diversos sectores, el avance tecnológico es el enfoque principal de los proyectos actuales en el país.

En este contexto, la Universidad de las Ciencias Informáticas (UCI) surge como uno de los proyectos emblemáticos de la Revolución cubana, con la misión de formar profesionales en el campo de la ingeniería informática, capacitados para liderar la informatización de la sociedad. La UCI se organiza en seis facultades, cada una de las cuales alberga centros de desarrollo de software especializados en diversas áreas.

En el campus de la misma se encuentra la Empresa de Tecnologías de la Información para la Defensa (Xetid), fundada en 2013. Xetid se dedica a proyectar, diseñar, desarrollar y comercializar productos y servicios basados en las TIC con un enfoque en soluciones tecnológicas integrales y la implementación de TIC en varias áreas de trabajo. Su misión principal es

contribuir a la informatización de la sociedad cubana mediante el uso adecuado de fuentes de información.

Xetid, con múltiples sedes en todo el territorio de la República de Cuba, proporciona soluciones informáticas que abarcan una amplia gama de sectores y necesidades en el país. Un ejemplo de su labor es el sistema de gestión empresarial "Distra", que permite informatizar la estructura organizativa de las empresas, abarcando desde la base organizacional hasta cada una de las áreas. Asimismo, Xetid colabora estrechamente con el Instituto Nacional de Recursos Hidráulicos (INRH).

En la actualidad, en la Dirección de Ciencia, Tecnología e Innovación del Instituto Nacional de Recursos Hidráulicos (DCTI-INRH), la obtención de información se realiza mediante documentos en formato Excel y Word. Estos documentos son completados por las diferentes delegaciones y posteriormente entregados de forma física o por correo. Esta metodología implica que la información no esté disponible en tiempo real, llegue de manera fragmentada y requiera un procesamiento manual por parte de los especialistas.

La forma en la que hoy se planifica el presupuesto en la DCTI-INRH puede generar dificultades en la empresa, conduciendo a procesos manuales propensos a errores y falta de precisión, lo que afecta tanto la operatividad como la toma de decisiones estratégicas. Además, la ausencia de una plataforma efectiva para consolidar datos financieros plantea desafíos significativos para obtener una visión holística de la situación económica de la empresa, impactando la capacidad de tomar decisiones informadas. Asimismo, la carencia de herramientas específicas para monitorear y controlar la ejecución de planes estratégicos y financieros puede complicar la toma de decisiones oportunas, afectando la implementación exitosa de estrategias empresariales. Estas carencias subrayan la importancia de adoptar soluciones tecnológicas que aborden de manera integral la planificación, consolidación de datos y control de planes en el ámbito financiero de una empresa.

Esta situación representa una brecha en el flujo de información, lo cual constituye una preocupación y un problema que nuestro estudio busca resolver con eficacia. El estudio concluye con la necesidad de ampliar la perspectiva de

desarrollo y adoptar la implementación de un software altamente configurable. Este software permitirá a los especialistas llevar a cabo la planificación, aprobación, ejecución y control del presupuesto asignado de manera eficiente y efectiva.

El sistema de gestión de información en desarrollo en Xetid consta de cinco módulos: configuración, planificación, ejecución, reportes y gráficas. Esta investigación se enfoca exclusivamente en el desarrollo de los módulos de planificación y ejecución, identificados como elementos críticos dentro del sistema de gestión de los servicios científico-técnicos del INRH (proyectos de inversión). El objetivo de este desarrollo es potenciar la funcionalidad del sistema en la gestión de los servicios científico-técnicos asociados a proyectos de inversión bajo la administración del INRH.

A raíz de la situación mencionada anteriormente, se define como **problema a resolver**: las limitaciones que presenta el INRH en la obtención y planificación de proyectos de inversiones, que influye y compromete negativamente a la integridad de la empresa y sus especialistas en los procesos de planificación, aprobación, ejecución y control del presupuesto aprobado.

El **objeto de estudio** de esta investigación se centra en los sistemas destinados a la gestión de información.

Para solucionar el problema planteado, se establece el siguiente **objetivo general**: Desarrollar los módulos de planificación y ejecución de servicios técnicos dentro del sistema del INRH en función de permitir a la empresa y a los especialistas la planificación, aprobación, ejecución y control del presupuesto aprobado en proyectos de inversionistas.

El **campo de acción** de esta investigación se centra en los sistemas de gestión de información en el proceso de planificación y ejecución de proyectos inversionistas para el INRH.

Se definen como **tareas de investigación**:

1. Revisión bibliográfica para generar un sólido marco teórico-conceptual relacionadas con sistemas de gestión inversionistas estableciendo similitudes con la investigación en curso.

2. Análisis de herramientas, tecnologías y lenguajes informáticos, además de las metodologías de desarrollo de software, para determinar cuáles son las más adecuadas para la implementación del sistema como solución al problema planteado en la investigación.
3. Estudio de sistemas homólogos, para comprender los aspectos comunes y regulares en el diseño de sistemas destinados a la gestión de información en función de conocer aspectos en el diseño de un sistema informático con de gestión.
4. Análisis de deficiencias y necesidades, para la identificación de las principales funcionalidades del sistema a desarrollar para la posterior implementación del mismo, teniendo en cuenta los requisitos definidos por el cliente.
5. Análisis de los diferentes tipos de pruebas de software existentes, para verificar el correcto funcionamiento del sistema desarrollado.

En el desarrollo de la presente investigación se emplearán de manera integrada los siguientes **métodos científicos**:

Métodos teóricos:

1. Analítico-sintético: Este método se utilizará en el análisis de la información recopilada durante la investigación en el INRH y en el estudio de información existente tanto a nivel nacional como internacional sobre sistemas de gestión de información. Permitirá distinguir conceptos, definiciones, avances y otros elementos relevantes para la investigación.
2. Inductivo-deductivo: Se aplicará este método para llegar a conclusiones a partir del análisis de la información recopilada sobre el proceso de planificación y ejecución de servicios técnicos, así como de las características generales y particulares relacionadas con los elementos, procesos o fenómenos de interés para la investigación.
3. Modelación: Este método se utilizará para representar la estructura, relaciones y características de la solución propuesta a través de diagramas. Facilitará el diseño y la comprensión de las clases necesarias para la implementación del sistema.

Métodos Empíricos:

1. Entrevista: Se aplicarán entrevistas a los trabajadores del INRH con el propósito de verificar las condiciones actuales del proceso de planificación y ejecución de servicios técnicos. También se investigará el nivel de informatización disponible para la gestión de la información generada en el contexto de su actividad laboral.
2. Observación: Este método se utilizará para realizar un seguimiento cercano del proceso de negocios y analizar los resultados de la gestión de información en el área del INRH.

El trabajo de diploma se ha estructurado en introducción, tres capítulos, conclusiones, recomendaciones, referencias bibliográficas. Estos capítulos incluyen:

Capítulo 1: Fundamentación teórica: Este capítulo establece el marco teórico y conceptual del estudio, incluyendo una revisión del estado del arte en sistemas de gestión de información en el proceso de planificación y ejecución de servicios técnicos del INRH. Además, se presenta la metodología de desarrollo de software que se aplicará en el proyecto, junto con las herramientas seleccionadas para su implementación.

Capítulo 2: Análisis y diseño: Este capítulo propone una solución basada en el análisis del proceso de planificación y ejecución de servicios técnicos del INRH. Se detalla el proceso y se crea un modelo representativo. Además, se plantea la estructura modular del sistema y se identifican los roles requeridos para su operación.

Capítulo 3: Implementación y pruebas: se centra en las etapas de obtención de requisitos del sistema, planificación temporal del proyecto y estructura general del software.

Capítulo 1: Fundamentación teórica

1.1 Introducción

A lo largo de la evolución de la informática, ha surgido una creciente tendencia hacia el análisis de la información. Esta tendencia busca facilitar la localización y el acceso a documentos, así como automatizar y supervisar los procesos de transferencia y eliminación de documentación. Este enfoque se ha convertido en una práctica común en la gestión empresarial, ya que es fundamental para el manejo eficiente de la información en instituciones que gestionan grandes cantidades de datos.

La comprensión y análisis de aspectos específicos son esenciales para respaldar el enfoque teórico-metodológico y conceptual de la investigación en curso. En este capítulo, se llevará a cabo un estudio del estado actual de la informática, lo que permitirá la selección de la metodología, herramientas y lenguajes informáticos adecuados para la implementación del proyecto.

1.2 Gestión de la información

El avance tecnológico ha revolucionado los sistemas de información en las empresas, dotándolos de mayor intuición y capacidad para almacenar, administrar e interpretar datos. En consecuencia, se han convertido en herramientas fundamentales para el progreso constante de la sociedad.

La Gestión de la información (GI) se refiere a un conjunto de procesos convencionales que supervisan el ciclo de vida de la información. Estos procesos abarcan desde la obtención de información (ya sea a través de su creación o captura) hasta su disposición final (archivado o eliminación). También comprenden operaciones como la extracción, combinación, depuración y distribución de la información a las partes interesadas. El propósito de la gestión de la información es garantizar la integridad, disponibilidad y confidencialidad de los datos (Reyes:, 2017).

Recursos hidráulicos: Recursos disponibles o potencialmente disponibles, en cantidad y calidad suficientes, en un lugar y en un período de tiempo apropiados para satisfacer una demanda identificable (Glosario internacional de hidrología, 2012).

El Instituto Nacional de Recursos Hidráulicos (INRH) fue creado por la Revolución el 10 de agosto de 1962 en aquel entonces respaldado por la ley No.1049. Se fundó la institución presidida por el Comandante Faustino Pérez Hernández, la cual en aquel momento tuvo la ardua tarea de colocar los cimientos del trabajo hidráulico al diseñar la primera estrategia nacional hidráulica en función del desarrollo económico del país e iniciar un ambicioso programa de construcciones, asumiendo además la atención de los servicios de acueducto y alcantarillado.

El 6 de junio de 1989, a través del decreto ley No. 114, se crea un nuevo organismo de la administración central del estado, el cual toma el mismo nombre que el fundado el 10 de agosto de 1962, y se mantiene el 10 de agosto como el día del trabajador hidráulico.

La dirección o gestión de proyectos es la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto para cumplir con los requisitos del mismo. Se logra mediante la aplicación e integración adecuadas de una serie de procesos agrupados, que conforman los cinco grupos de procesos. Estos grupos de procesos son: inicio, planificación, ejecución, seguimiento o control y cierre (García, 2016).

La finalidad de la GI es proporcionar mecanismos que permitan a una organización adquirir, producir y transmitir datos e información de manera eficiente, con la calidad, precisión y actualización necesarias para cumplir con los objetivos de la organización. En términos simples, se trata de obtener la información adecuada, en el momento preciso y al menor costo posible para tomar decisiones óptimas (Arévalo, 2007).

Desde esta perspectiva, la GI emerge como un nuevo concepto en el campo de la ciencia de la información. Está enfocado en la gestión de la inteligencia corporativa de una organización, permitiendo su estructuración interna y habilitándola para adaptarse a los cambios en su entorno mediante el uso eficiente de la información y los recursos de información disponibles (Salas, 2002).

La GI es un proceso que diversos autores han concluido en definir como la organización, evaluación, presentación y comparación de datos dentro de un

contexto específico, con un énfasis en el control de su calidad. El objetivo es asegurar que la información sea veraz, oportuna, significativa, exacta y útil, y que esté disponible en el momento en que se necesita. Este proceso involucra la gestión de diversos elementos, como información, documentos, metodologías, informes, publicaciones, soportes y flujos, todos alineados con los objetivos estratégicos de una organización (Gestión de la información y el conocimiento, 2012).

En síntesis, la GI se centra en garantizar que la información esté disponible, sea precisa y cumpla con los requisitos de calidad, contribuyendo así a la toma de decisiones efectivas y al logro de los objetivos de la organización.

Según Moreno González, las funciones de la GI abarcarían desde (Moreno González, 2002):

1. **Adquisición de información:** Esta función implica la recopilación de datos e información relevante tanto desde fuentes internas como externas a la organización. Esto incluye la captura de información de diversas fuentes, como bases de datos, documentos, informes, medios de comunicación, entre otros.
2. **Organización y clasificación:** Una vez adquirida, la información debe organizarse y clasificarse de manera eficiente. Esto implica la estructuración de datos para facilitar su búsqueda y recuperación cuando sea necesario.
3. **Almacenamiento y resguardo:** La información debe ser almacenada de manera segura para evitar pérdidas o daños. Esto puede implicar el uso de sistemas de almacenamiento físico o digitales, así como la implementación de medidas de seguridad adecuadas.
4. **Distribución y acceso:** La información debe estar disponible para las partes interesadas en el momento en que la necesiten. Esto puede incluir la distribución de informes, documentos o acceso a bases de datos a través de sistemas de gestión de información.
5. **Uso y aplicación:** La información adquirida y almacenada debe utilizarse de manera efectiva para respaldar la toma de decisiones, la ejecución de

tareas y el logro de los objetivos organizacionales. Esto implica analizar y aplicar la información de manera significativa.

6. Control de calidad: Se debe garantizar que la información sea precisa, veraz y actualizada. Esto implica la implementación de controles de calidad para verificar la integridad de los datos y su confiabilidad.
7. Disposición y eliminación: Cuando la información ya no es relevante o necesaria, se debe gestionar su disposición o eliminación de manera adecuada. Esto puede implicar la archivación de documentos o la eliminación segura de datos.
8. Auditoría y evaluación: Se deben realizar auditorías periódicas para evaluar la efectividad de los procesos de gestión de información y asegurarse de que se cumplan los estándares de calidad y seguridad.

Estas funciones abarcan todo el ciclo de vida de la información dentro de una organización y son fundamentales para optimizar su uso y contribuir al logro de los objetivos estratégicos.

La importancia de la GI radica en su capacidad para ayudar a la organización a reducir la incertidumbre y a adaptarse de manera efectiva a los cambios en su entorno. Además, la utilización adecuada de la información eleva la cultura y el desempeño de la entidad al respaldar la toma de decisiones informadas y la implementación de innovaciones.

En la actualidad, muchas de las soluciones empresariales se basan en teorías relacionadas con la GI y los sistemas de información (SI). Se considera que las organizaciones son SI en sí mismas, según las concepciones más recientes en el campo de la GI. Esto resalta la importancia de gestionar la información de manera estratégica y eficiente para lograr el éxito y la competitividad en un entorno empresarial en constante evolución. (Salas, 2002).

1.3 Sistemas de gestión de la información (SGI)

En la actualidad, las empresas dependen cada vez más de la información, la tecnología y las comunicaciones. La información se ha convertido en uno de los activos más críticos para las organizaciones, especialmente para aquellas que operan en sectores específicos de la economía.

Desde esta perspectiva, la información se define como cualquier dato o conjunto de datos que revela la situación del entorno o del sistema, sus objetivos o resultados, y que contribuye a reducir la incertidumbre al tomar decisiones (Hermoso, y otros, 2000).

Los **Sistemas de Información** (SI) se reconocen como herramientas fundamentales para el acceso y uso de la información, además de facilitar el proceso de toma de decisiones en las organizaciones. Estos sistemas desempeñan un papel crucial al proporcionar a las empresas la capacidad de recopilar, procesar, almacenar y distribuir información relevante para sus operaciones y estrategias comerciales. En un entorno empresarial cada vez más competitivo y orientado hacia la toma de decisiones informadas, los SI son esenciales para el éxito organizacional.

Un SI es un conjunto de elementos orientados al tratamiento y administración de datos e información, organizados y listos para su posterior uso, generados para cubrir una necesidad (Elexey, 2013).

El concepto de SI a menudo se confunde con el de sistema de información informático, pero no son idénticos. El último pertenece al campo de la tecnología de la información y puede ser parte de un SI como un recurso material. En general, los SI abordan el desarrollo y la administración de la infraestructura tecnológica de una organización (Elexey, 2013).

Para comprender mejor este tema, es importante tener en cuenta algunos conceptos clave:

- **Dato:** Es una expresión general que describe características de entidades y no constituye información por sí mismo.
- **Información:** Está compuesta por un conjunto de datos que han sido supervisados y ordenados de manera que puedan utilizarse para construir un mensaje. La información permite resolver problemas y tomar decisiones. Para que los datos se conviertan en información relevante para la toma de decisiones, deben ser procesados de alguna manera (Emery, 1990).

- **Organización:** Se puede entender como un sistema sistemático compuesto por personas y tecnología con el propósito de lograr un objetivo. Este enfoque sistémico de las organizaciones es fundamental para esta investigación (HAMPTON, 1983).

En resumen, un SI no se limita a la tecnología de la información, sino que engloba el desarrollo y la gestión de la infraestructura tecnológica de una organización. La información se deriva de datos procesados y organizados de manera que sean útiles para la toma de decisiones, y las organizaciones se consideran sistemas sistemáticos con un propósito definido.

Las características principales de un sistema de información, según Emery, se resumen en los siguientes puntos (Emery, 1990):

1. Parte integral de las actividades de la organización: Un sistema de información bien diseñado forma parte integrante de las actividades de la organización en todos sus niveles. No es una entidad independiente, sino que está integrado en los procesos operativos y de toma de decisiones.
2. Basado en tecnología de computación: Los SI dependen en gran medida de la tecnología informática. Esto significa que utilizan sistemas de hardware y software para recopilar, almacenar, procesar y distribuir información de manera eficiente.
3. Interacción hombre-máquina: Un SI eficaz facilita la interacción eficiente entre las tareas realizadas por humanos y las tareas ejecutadas por máquinas. Esto implica que los usuarios pueden interactuar de manera efectiva con el sistema para lograr sus objetivos.
4. Colección de subsistemas: Un SI consta de una colección de subsistemas interconectados. La forma en que estos subsistemas se conectan puede variar según el nivel de integración técnica y económica que sea más adecuado para la organización. La integración de datos electrónicos (EDI) es un ejemplo de un sistema que puede ser útil en diversas situaciones.
5. Adaptabilidad a cambios: Un SI bien diseñado debe ser capaz de adaptarse continuamente a las necesidades cambiantes y a los avances

tecnológicos. Esto significa que debe ser flexible y capaz de evolucionar con el tiempo.

Por otro lado, según Bootello, los sistemas de información cumplen tres funciones fundamentales (BOOTELLO, 2008):

1. Automatizar los procesos operativos: Los sistemas de información automatizan y agilizan las operaciones comerciales y administrativas de una organización, lo que contribuye a una mayor eficiencia en el trabajo diario.
2. Proporcionar información para la toma de decisiones: Estos sistemas generan datos y análisis que ayudan a las personas a tomar decisiones informadas, lo que es esencial para la gestión eficaz de una organización.
3. Lograr ventajas competitivas: La implementación efectiva de un sistema de información puede brindar ventajas competitivas a una organización en su mercado, ya que mejora la eficiencia y la capacidad de respuesta.

Los SGI desempeñan un papel esencial en la adquisición y el manejo de la información dentro de una organización. Además de automatizar procesos y proporcionar apoyo para la toma de decisiones, pueden ser una fuente de ventaja competitiva en el mercado si se utilizan y comprenden eficazmente.

Un **Sistema de Gestión Integrado** (SGI), según la definición de la Organización Internacional de Normalización (ISO), es un sistema único diseñado para administrar múltiples aspectos de las operaciones de una organización en cumplimiento con diversas normas y estándares. Estos aspectos pueden incluir la gestión de la calidad, el cuidado del medio ambiente y la salud y seguridad laboral. Un SGI tiene como objetivo guiar a una organización en el análisis de los requisitos de los clientes, asegurando que el personal esté motivado y bien capacitado, definiendo los procesos para la producción y la prestación de servicios, y manteniéndolos bajo control.

En términos simples, un SGI opera en el núcleo de una organización, mejorando y optimizando su funcionamiento a través del flujo de trabajo interno. Los trabajadores aportan datos al SGI, que generalmente son tareas y registros de rutina, y el sistema procesa esta información para devolver datos útiles y análisis

(salidas). El análisis de estas salidas proporciona información valiosa que puede utilizarse para tomar decisiones estratégicas y competitivas.

En resumen, un SGI es una herramienta integral que permite a una organización administrar eficazmente múltiples aspectos de sus operaciones, cumplir con las normas y estándares aplicables y utilizar la información recopilada para mejorar continuamente su desempeño y competitividad.

El Sistema de Gestión de Información Integrada (SGII) es responsable de seleccionar, procesar y distribuir información de diversas fuentes en una organización. Estas fuentes de información pueden clasificarse en tres categorías principales:

- a.** Información interna: Esta categoría abarca la información generada en la actividad cotidiana de la institución. Incluye datos y registros producidos por los procesos y operaciones internos de la organización.
- b.** Información externa: Aquí se incluye la información adquirida por la institución desde fuentes externas. Esta información se obtiene para obtener datos relevantes sobre temas de interés para la organización, como datos de mercado, tendencias de la industria, informes de competidores, etc.
- c.** Información corporativa o pública: Esta categoría abarca la información que la institución emite o divulga hacia el exterior. Puede incluir informes, comunicados de prensa, documentos corporativos y otros materiales que la organización comparte con el público.

Es importante destacar que el desarrollo de un SGI es fundamental para el éxito de una organización. La eficacia de estos sistemas puede marcar la diferencia entre el éxito y el fracaso de una empresa. No se trata solo de contar con la última tecnología en hardware y software, sino de diseñar un sistema de información que se adapte a las necesidades y recursos específicos de la organización. En ocasiones, sistemas más modestos en términos de tecnología de la información pueden satisfacer de manera efectiva las necesidades de la

empresa. Por lo tanto, las organizaciones deben considerar los sistemas de información como una parte integral de su estrategia de negocio y adaptarlos de manera coherente a sus objetivos y recursos disponibles.

Después de haberse elegido la organización de un sistema en su totalidad, es necesario decidir la aproximación para descomponer los subsistemas en módulos.

Un subsistema es un sistema en sí mismo, cuyo funcionamiento no depende de los servicios proporcionados por otro subsistema. Los subsistemas se componen por otros módulos y tienen interfaces definidas, las cuales se usan para comunicarse con otros subsistemas.

Un módulo de un sistema es normalmente un componente de un subsistema que proporciona uno o más servicios a otros módulos. A su vez este usa los servicios proporcionados por otros módulos. Los módulos se componen normalmente de varios componentes del sistema más simples. (Sommerville, 2005).

Un proyecto de inversión es un plan al cual se le asignan recursos financieros con el propósito de generar un bien o un servicio que atienda y satisfaga una necesidad humana.

La importancia de los proyectos de inversión consiste en invertir los excedentes del flujo de efectivo en un bien, servicio, o ambas que será redituable a mediano o largo plazo, considerando factores de riesgo que puedan afectar la generación de valor y riqueza a quienes aportan el capital del proyecto (Serrano, 2020).

1.4 Estado del arte

Es un “tipo de evaluación descriptiva; seria, sistematizada y consistente. Supone el revivir de una mínima parte de la memoria científica de la humanidad en aquel campo dentro del cual enmarcamos nuestro proyecto investigativo” (EL ESTADO DEL ARTE: UNA METODOLOGÍA DE INVESTIGACIÓN, 2015).

El uso extendido de los SGI ha abarcado una amplia gama de aplicaciones. No obstante, resulta fundamental destacar que su implementación en el contexto del desarrollo de software de gestión de sistemas hidráulicos no ha alcanzado las expectativas previamente establecidas. Dentro de este ámbito, se identifican

instancias aisladas de sistemas informáticos creados con la finalidad específica de abordar esta categoría de aplicación. En el marco de esta investigación, procederemos a examinar y analizar dichos ejemplos, poniendo énfasis en aquellos aspectos que puedan contribuir al enriquecimiento de nuestro conocimiento sobre la materia en cuestión.

1.4.1 Sistemas de gestión de información en el mundo

- **Microsoft Project:**

Microsoft Project es una herramienta de gestión de proyectos que ha desempeñado un papel fundamental en la planificación y ejecución de proyectos en diversas industrias durante décadas. Aquí te proporciono una breve historia y el quehacer de Microsoft Project:

Microsoft Project se lanzó por primera vez en 1984 bajo el nombre de "Project Manager" como una aplicación independiente para MS-DOS. Fue desarrollado por la empresa de software de gestión de proyectos de Seattle, Aldus Corporation, antes de ser adquirido por Microsoft en 1985. La primera versión para Windows, Microsoft Project 1.0, se lanzó en 1990. A lo largo de los años, ha experimentado múltiples versiones y mejoras, convirtiéndose en una herramienta completa de gestión de proyectos.

En 2000, Microsoft Project se convirtió en parte de la suite Microsoft Office, lo que permitió una mayor integración. Con el tiempo, Microsoft Project ha evolucionado para incluir características avanzadas de programación, seguimiento de tareas, gestión de recursos, análisis de costos y comunicación en equipo, lo que lo convierte en una herramienta esencial para la gestión de proyectos de diversa complejidad en industrias como la construcción, tecnología, salud y muchas otras.

Microsoft Project se utiliza para planificar proyectos, definir tareas y recursos, asignar responsabilidades, establecer fechas límite y presupuestos, realizar un seguimiento del progreso del proyecto y generar informes de estado y análisis de costos. Permite crear diagramas de Gantt, facilita la colaboración en equipo mediante la sincronización de proyectos en línea y la integración con otras aplicaciones de Microsoft y servicios de almacenamiento en la nube. Proporciona

herramientas para gestionar riesgos, realizar análisis de recursos y evaluar el impacto de los cambios en el proyecto (Francisco Javier Cruz Castañón, 2014).

- **Kanbanize by Businessmap:**

Kanbanize es una plataforma Kanban para la gestión ágil de proyectos con la que puedes dividir tus proyectos y visualizarlos en múltiples niveles de jerarquía. Es el software líder para la gestión de proyectos ágiles y de cartera. Proporciona visibilidad en todos los proyectos y carteras, conecta la planificación y la ejecución y ayuda a los equipos a hacer entregas más rápido. Basado en la idea de que todos los procesos evolucionan, Kanbanize puede adaptarse rápidamente a los cambios de una organización, independientemente de que se trate de un startup¹ o una corporación.

Kanbanize ofrece un rendimiento increíble desde la visualización de las iniciativas clave de la empresa hasta su división en varios niveles de elementos de trabajo jerárquicos. El software combina un conjunto avanzado de herramientas Kanban, incluyendo herramientas de colaboración simples y fáciles de usar que pueden ayudar a tu equipo a evaluar el progreso de tus proyectos.

Funciones claves:

- Línea de tiempo de proyecto.
- Tableros Kanban.
- Seguimiento de tiempo.
- Automatización del flujo de trabajo.
- Previsión de proyectos.
- Informes automatizados del estado del proyecto.
- Gestión de dependencias.
- Potente módulo de análisis (Businessmap, 2023).

¹ Empresa de nueva creación que, gracias a su modelo de negocio escalable y al uso de las nuevas tecnologías, tiene grandes posibilidades de crecimiento.

1.4.2 Sistemas de Gestión de Información en Cuba y en la UCI

- **Sistema para la planificación y control del presupuesto en el Centro de Geoinformática y Señales Digitales (GEYSED):**

Sistema informático desarrollado en la UCI para planificar y controlar la ejecución del presupuesto de gasto en el centro GEYSED. Se encarga además de insertar y eliminar elementos del gasto, para realizar el presupuesto y exportar el mismo a PDF. Para la construcción del software se utilizaron diferentes tecnologías, el Lenguaje Unificado de Modelado Visual Paradigm, lenguaje de programación C++ y como sistema gestor de base de datos PostgreSQL.

El sistema da cumplimiento a la necesidad del asesor económico del Centro GEYSED de planificar y controlar el presupuesto haciendo uso de esta herramienta informática que genera los reportes asociados a las tareas de manera automática. Este software es capaz de realizar el presupuesto de gasto de GEYSED. El mismo se utiliza por el asesor económico para realizar el control de los gastos en los que ha incurrido el centro a modo general y cada uno de sus proyectos de manera específica, así como el análisis de los mismos a través de gráficos (Elizabeth Salazar Laffita, 2013).

1.4.3 Conclusiones del estudio del estado del arte

En la tabla siguiente se detallan algunos aspectos que resultaron determinantes para la aprobación de la implementación de los módulos de planificación y ejecución en el sistema de gestión de presupuesto del INRH, orientado a la planificación y supervisión de proyectos de inversión.

Tabla 1 Tabla comparativa de sistemas homólogos

Parámetros	Microsoft Project	Kanbanize	GEYSED
Planifica presupuesto	Si	Si	Si
Desaprobar y Aprobar plan	No	No	No
Adjuntar documentos	Si	No	No
Generar reporte	Si	Si	Si
Calcular total de presupuesto	Si	Si	Si
Control de presupuesto	No	Si	Si

Por todo lo antes mencionado se decidió implementar un sistema que cumpliera con los requisitos principales mostrados en la tabla anterior. En el desarrollo de estos módulos se debe lograr la planificación y ejecución del presupuesto de gasto en el INRH en cada uno de sus proyectos, así como los elementos mostrados en la tabla anterior. Además, se considera que sería una solución confiable y concreta que cubra las necesidades del cliente. Para su correcta elaboración se estudiarán las metodologías existentes, el lenguaje de modelado y la herramienta CASE a utilizar, y otros elementos de gran importancia para su desarrollo.

El análisis de los diversos casos examinados previamente ha llevado a la identificación de conclusiones significativas para esta investigación. Específicamente, se ha constatado que actualmente no se ha hallado ningún sistema existente que pueda ser ajustado de manera apropiada para cumplir con los requisitos específicos relacionados con el procesamiento de información y la gestión de nuevos proyectos en el INRH. En lugar de ello, se exploraron las funcionalidades ya existentes en dichos sistemas con el propósito de implementar los módulos de planificación y ejecución en los servicios técnicos del INRH.

Por lo tanto, se confirma la relevancia de este proyecto de investigación, ya que está encaminado a llenar una brecha existente en la gestión de proyectos en el INRH, contribuyendo de manera significativa a mejorar la eficiencia y eficacia de la institución en el procesamiento de información y la administración de nuevos proyectos en el ámbito de los recursos hídricos. Estas conclusiones parciales respaldan la necesidad imperante de desarrollar una solución a medida para abordar los desafíos específicos identificados en el INRH.

1.5 Metodología de desarrollo del software

Una **metodología de software** es un conjunto de enfoques, técnicas, procedimientos y herramientas estandarizados que se utilizan en el ciclo de vida del desarrollo de software para planificar, diseñar, implementar, probar y mantener sistemas de software de manera sistemática y efectiva (Pressman, 2014).

La tarea de clasificar las metodologías es un desafío arduo debido a la abundancia de propuestas existentes y las notables diferencias en cuanto al grado de detalle, la información disponible y el alcance que presentan. A grandes rasgos, considerando su filosofía de desarrollo, es posible agruparlas en dos categorías principales: las metodologías ágiles o ligeras y las metodologías pesadas o tradicionales.

Existen numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo. Por una parte, tenemos aquellas propuestas más tradicionales que se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, las herramientas y notaciones que se usarán. Estas propuestas han demostrado ser efectivas y necesarias en un gran número de proyectos, pero también han presentado problemas en otros muchos.

Una posible mejora es incluir en los procesos de desarrollo más actividades, más artefactos y más restricciones, basándose en los puntos débiles detectados. Sin embargo, el resultado final sería un proceso de desarrollo más complejo que puede incluso limitar la propia habilidad del equipo para llevar a cabo el proyecto. Otra aproximación es centrarse en otras dimensiones, como por ejemplo el factor humano o el producto software. Esta es la filosofía de las metodologías ágiles, las cuales dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas.

Este enfoque está mostrando su efectividad en proyectos con requisitos muy cambiantes y cuando se exige reducir drásticamente los tiempos de desarrollo, pero manteniendo una alta calidad. Las metodologías ágiles están revolucionando la manera de producir software, y a la vez generando un amplio debate entre sus seguidores y quienes por escepticismo o convencimiento no las ven como alternativa para las metodologías tradicionales (Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP), 2006).

Dada la premura temporal, la naturaleza de los requisitos de alto nivel proporcionados y el objetivo de optimizar la eficiencia en la construcción de software, se ha tomado la determinación de adoptar un enfoque de desarrollo ágil. Este enfoque se elige por su capacidad para agilizar la implementación del

software, minimizar la generación de documentación y gestionar la variabilidad de los requisitos de manera efectiva a lo largo del ciclo de desarrollo.

1.5.1. Programación extrema

EXtreme Programming (XP) es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre. Kent Beck (BECK, 2000), el padre de XP, describe la filosofía de XP en sin cubrir los detalles técnicos y de implantación de las prácticas. Posteriormente, otras publicaciones de experiencias se han encargado de dicha tarea (Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP), 2006).

XP está centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje del equipo de desarrollo. Se basa en la retroalimentación (feedback) continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, donde existe un alto riesgo técnico (BECK, 1999).

Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento historias de usuario pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas

o ser modificadas. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas (BECK, 1999).

XP como metodología al fin, presenta determinados artefactos para guiar el desarrollo del proyecto. Estos artefactos son las Historias de Usuario (HU), las Tareas de Ingeniería (TI) y las tarjetas Clase-Responsabilidad Colaboradores (CRC).

Según Beck (BECK, 1999) Aunque en otras fuentes de información aparecen algunas variaciones, XP contiene varios roles: programador, cliente, encargado de pruebas (tester), encargado de seguimiento (tracker), entrenador (coach), consultor y gestor (big boss).

El ciclo de desarrollo consiste a grandes rasgos en los siguientes pasos (José H. Canós):

1. El cliente define el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor de negocio.
5. Vuelve al paso 1.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el software o no se cumplirán los plazos. De la misma forma el cliente tiene la obligación de manejar el ámbito de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración.

Un proyecto XP tiene éxito cuando el cliente selecciona el valor de negocio a implementar basado en la habilidad del equipo para medir la funcionalidad que puede entregar a través del tiempo (Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP), 2006). Las etapas señaladas por Beck en su libro Extreme Programming Explained son: exploración, planificación de

entregas, iteraciones, producción, mantenimiento y muerte del proyecto. Pero otros autores como (Calero, 2003)Manuel Calero Solís y Doug Wallace, Isobel Raggett y Joel Aufgang (WALLACE, y otros, 2002) agrupan las prácticas definidas por Beck en cuatro etapas: planificación, diseño, implementación y pruebas.

1.6 Herramientas y lenguajes informáticos

La selección de herramientas y tecnologías en un proyecto es un aspecto de importancia crítica, ya que influye directamente en la funcionalidad, el rendimiento y la viabilidad del producto final. La elección precisa y la justificación de estas herramientas son aspectos fundamentales en un entorno académico, ya que demuestran la toma de decisiones informadas y la comprensión profunda de las implicaciones tecnológicas en el desarrollo de proyectos, lo que es esencial en una tesis de grado o trabajo de diploma relacionado con la informática o la tecnología.

1.6.1 Marco de trabajo Zeolides 2.3

Es un conjunto de librerías, herramientas, tecnologías y componentes de software integrados en un marco de trabajo para desarrollar aplicaciones de múltiples propósitos, gran tamaño y grandes volúmenes de datos, que permiten el desarrollo ágil, basado en componentes, centrado en los requerimientos del usuario, las interfaces de usuario y la lógica del negocio de las aplicaciones que con el mismo se desarrollen, aunque puede utilizarse con múltiples propósitos, por ejemplo para aplicaciones de tiempo real, su objetivo fundamental es el desarrollo de soluciones web de gestión empresarial (Desarrollo del componente Análisis del Sistema de Planificación de Producciones., 2018).

1.6.2 ExtJS 3.1

Ext JS (pronunciado como "ekst"¹) es una biblioteca de JavaScript para el desarrollo de aplicaciones web interactivas usando tecnologías como AJAX, DHTML y DOM. Fue desarrollada por Sencha.

Originalmente construida como una extensión de la biblioteca YUI por Jack Slocum, en la actualidad puede usarse como extensión para la biblioteca jQuery

y Prototype. Desde la versión 1.1 puede ejecutarse como una aplicación independiente.

1.6.3 Zend Framework 2.0

Zend Framework es un marco MVC basado en PHP 5 para desarrollo web. Reemplazando el código manual inconsistente con una biblioteca de componentes extensible y liviana. ZF acelera su tiempo de desarrollo y ayuda a aumentar su estabilidad y mantenibilidad de las aplicaciones. ZF cuenta con el respaldo de Zend y está disponible bajo una licencia de código abierto compatible con empresas (Zend Framework in Action, 2009).

Zend Framework y su evolución en el proyecto Laminas siguen siendo relevantes en el desarrollo de aplicaciones web PHP, especialmente en casos en los que se requiere un enfoque modular y se aprovechan los componentes individuales para construir aplicaciones personalizadas y seguras.

1.6.4 Doctrine 2.0

Doctrine es un marco de mapeo objeto-relacional (ORM) para PHP que se utiliza ampliamente en el desarrollo de aplicaciones web y de bases de datos.

Una de las fortalezas más destacadas de Doctrine es su capacidad para gestionar esquemas de bases de datos de manera eficiente y coherente. A través de su sistema de migraciones, Doctrine permite a los equipos de desarrollo realizar cambios en la estructura de la base de datos de forma controlada y colaborativa, lo que facilita la gestión de la evolución del esquema a lo largo del ciclo de vida de una aplicación. Es una herramienta poderosa para el desarrollo de aplicaciones PHP que interactúan con bases de datos relacionales. Su capacidad para simplificar el acceso y la manipulación de datos, junto con su enfoque en la seguridad y la gestión de esquemas, lo convierten en una elección sólida para proyectos web y aplicaciones empresariales.

1.6.5 PostgreSQL 13

POSTGRESQL fue desarrollado en la Universidad de California en Berkeley (1977–1985). El código Ingres fue posteriormente mejorado por Relational Technologies/Ingres Corporation, que produjo uno de los primeros servidores de

bases de datos relacionales comercialmente exitosos. También en Berkeley, Michael Stonebraker dirigió un equipo para desarrollar un servidor de base de datos relacional de objetos llamado Postgres (1986-1994). Illustra2 tomó el código Postgres y lo desarrolló hasta convertirlo en un producto comercial. Dos Los estudiantes de posgrado de Berkeley, Jolly Chen y Andrew Yu, agregaron posteriormente capacidades SQL a Postgres. El proyecto resultante se llamó Postgres95 (1994-1995). Los dos más tarde abandonaron Berkeley, pero Chen continuó manteniendo Postgres95, que tenía una lista de correo activa (Momjian, 2001).

Es un sistema de gestión de bases de datos relacionales (RDBMS) de código abierto que ha ganado reconocimiento y respeto en la comunidad de desarrollo de software y bases de datos. Su robustez, capacidad de extensión y soporte para características avanzadas lo hacen especialmente atractivo en aplicaciones empresariales y proyectos que requieren un almacenamiento de datos fiable y escalable.

Otra característica destacada de PostgreSQL es su compromiso con la disponibilidad y la tolerancia a fallos. Ofrece opciones de replicación, clústeres de alta disponibilidad y restauración continua, lo que garantiza que los datos estén protegidos y disponibles incluso en entornos empresariales exigentes.

1.6.6 PgAdmin versión 6.9

PgAdmin es una herramienta de gestión para PostgreSQL y bases de datos relacionales derivadas, como de EnterpriseDB EDB Advanced Server. Puede ejecutarse como una aplicación web o de escritorio. Se puede implementar como una aplicación web configurando la aplicación para que se ejecute en modo servidor. Se puede consultar la implementación del servidor sobre cómo ejecutar pgAdmin en modo servidor (pgAdmin, 2023).

Una de las ventajas más notables de pgAdmin es su capacidad para gestionar múltiples servidores PostgreSQL desde una única interfaz. Esto es especialmente útil en entornos donde se administran varias bases de datos o instancias de PostgreSQL, ya que permite una administración centralizada y eficiente. Es una herramienta esencial para administradores y desarrolladores que trabajan con bases de datos PostgreSQL. Su conjunto completo de

características y su interfaz de usuario eficiente lo convierten en una elección confiable y valiosa para la gestión de bases de datos PostgreSQL.

1.6.7 Visual Paradigm Form UML 8.0

Visual Paradigm es un proveedor líder y reconocido mundialmente de soluciones de software de transformación de TI y negocios. Permite a las organizaciones mejorar la agilidad empresarial y de TI y fomentar la innovación a través de estándares abiertos populares (visual-paradigm, 2023).

Visual Paradigm for UML es una solución integral y altamente efectiva para el modelado UML que ha demostrado su valía en el ámbito del desarrollo de software y la ingeniería de sistemas. Su interfaz de usuario intuitiva y su conjunto completo de herramientas de modelado hacen que sea una elección atractiva para equipos de desarrollo que buscan una herramienta sólida y versátil para la visualización y el diseño de sistemas complejos. Es una herramienta sólida y completa para el modelado UML que se adapta bien a proyectos de desarrollo de software y sistemas de envergadura. Su capacidad para generar código, la colaboración en equipo y su compatibilidad con múltiples tipos de diagramas UML lo convierten en una elección sólida para profesionales y equipos de desarrollo orientados a la calidad y la eficiencia en la visualización y el diseño de sistemas.

1.6.8 Lenguaje de Modelado Unificado UML 5.0

El lenguaje de modelado unificado (UML) es un lenguaje de modelado estandarizado de propósito general en el campo de Ingeniería de software orientada a objetos. UML incluye un conjunto de técnicas de notación gráfica para crear Modelos visuales de sistemas de software orientados a objetos. UML combina técnicas de modelado de datos, modelado de negocios, modelado de objetos, modelado de componentes y se puede utilizar en todo el ciclo de vida del desarrollo de software y en diferentes tecnologías de implementación (UNIFIED MODELING LANGUAGE (UML) OVERVIEW, 2012).

UML ofrece una variedad de diagramas, desde diagramas de clases que modelan la estructura de objetos y sus relaciones hasta diagramas de secuencia que representan el flujo de interacciones entre componentes. La versatilidad de

UML lo convierte en una herramienta poderosa para describir sistemas de cualquier tamaño o complejidad, desde aplicaciones pequeñas hasta sistemas empresariales críticos. Es una herramienta esencial para profesionales de la ingeniería de software y arquitectura de sistemas. Su capacidad para modelar sistemas de manera visual, su universalidad y su apoyo a las mejores prácticas de diseño lo convierten en un estándar invaluable para aquellos que buscan diseñar, comunicar y construir sistemas de software sólidos y confiables.

1.6.9 TortoiseGit 2.14

TortoiseGit es una interfaz de Windows Shell para Git y está basada en TortoiseSVN. Es de código abierto y se puede construir completamente con software disponible gratuitamente. Como no es una integración para un IDE específico como Visual Studio, Eclipse u otros, se puede usar con cualquier herramienta de desarrollo que se desee y con cualquier tipo de archivo. La interacción principal con TortoiseGit será mediante el menú contextual del explorador de Windows. TortoiseGit lo apoya con tareas regulares, como confirmar, mostrar registros, diferenciar dos versiones, crear ramas y etiquetas, crear parches, etc.

Está desarrollado bajo la GPL. Lo que significa que su uso es completamente gratuito para cualquiera, incluso en un entorno comercial, sin ninguna restricción. El código fuente también está disponible gratuitamente, por lo que incluso se puede desarrollar una propia versión si se desea (colaboradores, 2023).

1.6.10 Visual Studio Code 1.74.2

Visual Studio Code es un editor de código optimizado compatible con operaciones de desarrollo como depuración, ejecución de tareas y control de versiones. Su objetivo es proporcionar justo las herramientas que un desarrollador necesita para un ciclo rápido de creación de código y depuración y deja flujos de trabajo más complejos a IDE con funciones más completas, como Visual Studio IDE. Se ejecuta en macOS, Linux y Windows. (visualstudio, 2023).

1.6.11 Apache 2.4

Apache es un servidor web de código abierto, conocido como "Apache HTTP Server," que se desarrolló en la década de 1990. Es altamente popular y confiable, con características clave que incluyen:

- Funciona como servidor web, entregando contenido web a través de solicitudes HTTP.
- Es de código abierto, lo que permite a la comunidad de desarrolladores mejorar y evolucionar el software.
- Es compatible con varios sistemas operativos, lo que lo hace versátil.
- Utiliza módulos para agregar funcionalidades adicionales y es altamente personalizable.
- Ofrece características de seguridad y control de acceso.
- Está diseñado para alto rendimiento y escalabilidad.
- Permite una configuración flexible mediante archivos de configuración.

1.6.12 Apache JMeter 2019.1.1

La aplicación Apache JMeter™ es un software de código abierto, una aplicación Java 100% pura diseñada para cargar pruebas de comportamiento funcional y medir el rendimiento. Fue Diseñado originalmente para probar aplicaciones web, pero tiene desde entonces se expandió a otras funciones de prueba. Apache JMeter se puede utilizar para probar el rendimiento tanto en modo estático como dinámico. recursos, aplicaciones web dinámicas. Se puede utilizar para simular una carga pesada en un servidor, grupo de servidores, Red u objeto para probar su resistencia o analizar el rendimiento general. bajo diferentes tipos de carga (Apache, 2023).

1.6.13 PHP 5.6

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. Fue creado inicialmente por el programador danés-canadiense Rasmus Lerdorf en 1994. En la actualidad, la implementación de referencia de PHP es producida por The PHP Group. PHP 5 fue lanzado en

Julio del 2004 después de un largo desarrollo y varios pre-releases. Está básicamente impulsado por su núcleo, Zend Engine 2.0 que contiene un nuevo modelo de objetos y docenas de nuevas opciones. El equipo de desarrollo de PHP incluye docenas de desarrolladores, así como docenas de otras personas trabajando en proyectos relacionados y de soporte para PHP, como PEAR, PECL, y documentación, y una infraestructura en red subyacente de más de cien servidores web individuales en seis de los siete continentes del mundo. Aunque es solo una estimación basada en estadísticas de años anteriores, es seguro suponer que PHP ahora está instalado en diez o quizá cien millones de dominios en todo el mundo (Group, 2023).

1.6.14 HTML 5

HTML (HyperText Markup Language) Lenguaje compuesto de una serie de etiquetas o marcas que permiten definir el contenido y la apariencia de las páginas web. Aunque se basa en SGML, no se puede considerar que sea un subconjunto. Existen cientos de etiquetas con diferentes atributos. W3C se encarga de su estandarización. El futuro sustituto de HTML es XHTML (Mora, 2001).

1.6.15 CSS

CSS (Cascading Style Sheets, u Hojas de Estilo en Cascada) es la tecnología desarrollada por el W3C con el fin de separar la estructura de la presentación. Es un lenguaje constituido a base de selectores, propiedades y valores que tiene como objetivo dar aspecto visual al código HTML (HUMBERTO, 2003).

1.6.16 JavaScript 1.6

JavaScript se introdujo en 1995 como una forma de agregar programas a páginas web en el navegador Netscape Navigator. El lenguaje ha sido desde entonces adoptado por todos los otros navegadores webs principales. Ha hecho que las aplicaciones web modernas sean posibles: aplicaciones con las que puedes interactuar directamente, sin hacer una recarga de página para cada acción. JavaScript también es utilizado en sitios web más tradicionales para proporcionar diversas formas de interactividad e ingenio. Es importante tener en

cuenta que JavaScript casi no tiene nada que ver con el lenguaje de programación llamado Java (Haverbeke, 2018).

1.6.17 GitLab

GitLab es una plataforma integral de gestión de ciclo de vida de aplicaciones que ofrece un conjunto completo de herramientas para el desarrollo, la implementación y la supervisión de proyectos de software. Ofrece una amplia gama de características técnicas que facilitan la colaboración, la automatización y la administración de proyectos de desarrollo de software.

1.7 Conclusiones del capítulo

Luego de llevar a cabo la investigación y una revisión bibliográfica, se obtuvieron las siguientes conclusiones:

- El análisis de sistemas homólogos proporcionó información valiosa sobre las tendencias en sistemas de gestión de información, enfocado en Sistemas de recursos hídricos.
- Se optó por la metodología XP dado por su capacidad para facilitar entregas de software funcional en cada iteración, adaptándose de manera efectiva a las necesidades específicas del cliente.
- Una cuidadosa elección de lenguajes de programación, herramientas y metodología lo cual es esencial para garantizar eficiencia, productividad y calidad del producto final.

Capítulo 2: Diseño de la solución propuesta

2.1 Introducción

En este capítulo, se inicia la conceptualización del sistema propuesto, tras un análisis de los elementos fundamentales de la investigación. Este capítulo sienta las bases fundamentales para la propuesta de solución al ofrecer una comprensión detallada de los procesos de gestión de proyectos en el INRH, representados visualmente a través de la notación BPMN ("Business Process Model and Notation" Modelo y Notación de Procesos de Negocio, en español), junto con una estructura modular bien definida y roles identificados.

Estos elementos son fundamentales para el desarrollo del sistema de gestión de información propuesto en la presente investigación. Además, este capítulo se dedica a la identificación de las principales funcionalidades de la solución propuesta. A lo largo de las secciones que lo componen, se expondrán los artefactos generados durante la aplicación de la metodología de desarrollo de software XP en sus fases de Exploración, Planificación y Diseño.

2.2 Descripción de la solución del Software

La solución de software se enfocará en la informatización de dos subprocesos críticos, comenzando desde la planificación del plan y llegando hasta la presentación de la ejecución del presupuesto aprobado. Esto se puede apreciar claramente en la Ilustración 1, que representa visualmente la secuencia de estos subprocesos (Betancourt, 2020).

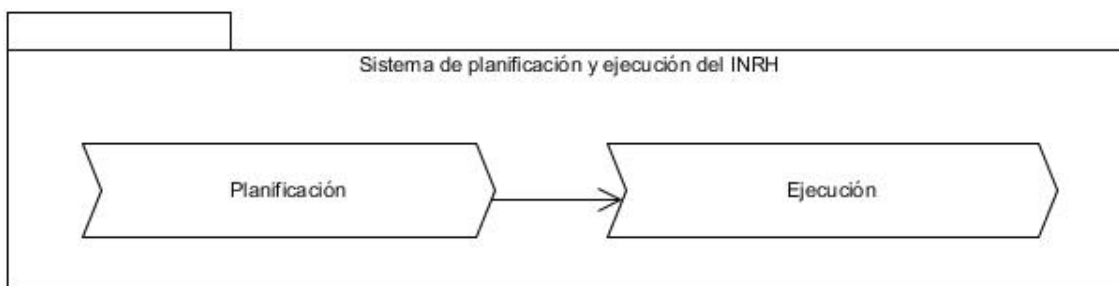


Ilustración 1 Mapa de procesos del negocio

A continuación, se describen detalladamente cada uno de los procesos previamente ilustrados:

- **Planificación:** este proceso parte cuando el director técnico de la delegación elabora el listado de fichas que desea incluir en el plan del año. Las fichas contienen todos los datos de los SCTI (Servicios en Ciencia Tecnología e Innovación) que se están solicitando y el presupuesto en MT (Moneda total) y CL (Capacidad de liquidez). Los proyectos pasan por un proceso de aprobación por parte del Dir. técnico y Dir. económico de la delegación. Estas fichas aprobadas son enviados a la DCTI donde se hace una revisión y aprobación o no de cada ficha. Como resultado de las fichas aprobadas sale el presupuesto aprobado para cada delegación provincial. Por otra parte, el nivel central también se elaboran sus propias fichas que arrojan el presupuesto del plan anual del nivel central. Finalmente se obtiene el Total general conformado por el presupuesto de las delegaciones provinciales, el presupuesto del nivel central y la reserva. En el caso del nivel central también presentan el desglose mensual del presupuesto contratado (Betancourt, 2020).
- **Ejecución:** este proceso parte del presupuesto aprobado para cada SCT (Servicios Científicos Técnicos) desde la planificación. Tanto el nivel central como las delegaciones registraran mes a mes el valor ejecutado para cada SCT, este valor estará respaldado por una o varias facturas. El total ejecutado nunca es mayor al presupuesto aprobado. El valor ejecutado por las delegaciones pasará por un proceso de aprobación donde intervienen el Dir. técnico de la delegación, el Dir. económico de la delegación y finalmente será revisado y aprobado por la DCTI. En este proceso de ejecución las delegaciones podrán crear nuevos proyectos con presupuestos que sobran de proyectos que estaban planificados, pero por diferentes afectaciones no pueden ser ejecutados completamente o de presupuesto aprobado por el DCTI que se toma de la reserva (Betancourt, 2020).

2.2.1 Características de la propuesta

El presente trabajo tiene como objetivo el desarrollo de una aplicación web con una interfaz intuitiva y de fácil acceso para usuarios de diversos niveles de competencia tecnológica. Se utilizarán tecnologías de código abierto en su implementación. Los módulos principales a desarrollar incluyen planificación y

ejecución, con un énfasis en la importancia de los módulos de configuración y reportes, los cuales son fundamentales para el funcionamiento integral del sistema. A continuación, se proporciona una breve descripción de los componentes y funcionalidades esenciales que deberá incluir el sistema:

Planificación de Servicio Científico Técnico e Innovación: Este módulo será el encargado de realizar todo el proceso de registro de las fichas a nivel de delegación o a nivel central, además será el encargado de llevar el proceso de aprobación desde que surge la ficha; la aprobación por director técnico de la delegación, el delegado y DCTI-INRH. A la ficha se podrá adjuntar cualquier tipo de documentos y observaciones. El presupuesto aprobado para una ficha podrá ser desagregado por meses.

Ejecución de Servicio Científico Técnico e Innovación: Este módulo será el encargado de llevar la ejecución mensual de todas las fichas aprobadas en plan, la ejecución mensual estará acompañada por un número de factura. Para casos excepcionales el sistema brindara la posibilidad de registrar nuevas fichas en tiempo de ejecución y finalizar otras fichas que no consumieron todo el presupuesto que se le aprobó.

2.2.2 Roles y funcionalidades

1. Director técnico de la delegación se encarga de:

- Gestionar banco de problemas.
- Gestionar entidad ejecutora.
- Gestionar plan SCTI.
- Gestionar ejecución SCTI.
- Reportes.

2. Delegado se encarga de:

- Revisar Plan SCTI.
- Revisar ejecución SCTI.
- Reportes.

3. Especialista (DCTI-INRH) se encarga de:

- Gestionar objetivos.
- Gestionar año de planificación.
- Gestionar estado de meses.
- Aprobar servicios.
- Reportes.

2.3 Lista de funcionalidades del sistema

Una lista de funcionalidades se define como las especificaciones de la función que el sistema de software o sus componentes deben llevar a cabo. Cada función se caracteriza por un conjunto de entradas, comportamientos y salidas específicos. Las funcionalidades pueden abordar cálculos, detalles técnicos, manipulación de datos y otras funcionalidades concretas que se espera que el sistema cumpla (Sommerville, 2005).

A continuación, se enuncian el listado de funcionalidades que contemplaran los módulos:

Plan de servicios científico-técnico: funcionalidad que permitirá la presentación de los servicios científico-técnico de las delegaciones provinciales de recursos hidráulicos y el nivel central. Los SCT serán sometidos a un proceso de aprobación. El plan total de las delegaciones o el nivel central quedará compuesto por la sumatoria del presupuesto CUP de cada ficha aprobada por el especialista del nivel central. A continuación, se enuncia el listado de funcionalidades que contemplará este módulo:

1. Listar entidades dado dominio
2. Buscar entidad
3. Limpiar búsqueda
4. Expandir árbol de entidades
5. Contraer árbol de entidades
6. Listar servicios
7. Adicionar servicio
8. Modificar servicio

9. Eliminar servicio
10. Desglose mensual del presupuesto
11. Revisado por director técnico de la delegación
12. Revisado por delegado
13. Aprobado por Dirección CTI-INRH
14. Cancelar por director técnico de la delegación
15. Cancelar por delegado
16. Desaprobar por Dirección CTI-INRH
17. Gestionar observación:
 - 17.1 Adicionar observación
 - 17.2 Eliminar observación
 - 17.3 Ver observación
18. Adjuntar documento (Ejemplo: Contrato o cualquier otro documento)
19. Generar reporte ficha
20. Generar reporte Plan de Servicios Científicos Técnicos (SCT)
21. Buscar servicio
22. Limpiar búsqueda
23. Consultar ayuda
24. Listar ejercicio
25. Calcular total plan insertado
26. Calcular total plan aprobado por nivel central

Ejecución del plan de servicios científico-técnico: esta funcionalidad permitirá registrar la ejecución mensual de cada servicio, la ejecución de cada mes estará respaldada por al menos una factura. Un servicio podrá ser terminado sin haber ejecutado todo su presupuesto, al terminar el servicio se pueden adjuntar notas y documentos que justifiquen el fin del servicio, un servicio pudiera terminar antes porque el proveedor no cumplió con el contrato o porque el servicio se pudo ejecutar con menos presupuesto del planificado. En la funcionalidad los especialistas podrán crear nuevos servicios, este nuevo servicio pasará por un proceso de aprobación y solo entonces se podrá ejecutar.

1. Adicionar servicio
2. Modificar servicio
3. Eliminar servicio
4. Revisar servicio por director técnico de la delegación

5. Revisar servicio por delegado
6. Aprobar servicio por dirección CTI-INRH
7. Cancelar servicio por director técnico de la delegación
8. Cancelar servicio por delegado
9. Desaprobar servicio por dirección CTI-INRH
10. Desglosar presupuesto
 - 10.1 Gestionar presupuesto mensual
 - 10.2 Gestionar factura mensual
 - 10.3 Gestionar fecha de la facturación
 - 10.4 Gestionar documento de factura
 - 10.5 Generar orden de pagos
11. Terminar ejecución de servicio
12. Gestionar observación
13. Gestionar documento (ejemplo, documento del proveedor con motivos de cancelación)
14. Calcular total plan (Presupuesto MT aprobado desde la funcionalidad plan de SCT)
15. Calcular total ejecutado (sumatoria de los desgloses mensuales)
16. Calcular total por ejecutar (total ejecución-total ejecutado)
17. Exportar ficha

2.4 Exploración

En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología (BECK, 2000).

2.4.1 Historias de usuario (UH)

Las historias de usuario representan una breve descripción del comportamiento del sistema, se realizan por cada característica principal del sistema y son utilizadas para cumplir estimaciones de tiempo y el plan de lanzamientos, así mismo reemplazan un gran documento de requisitos y presiden la creación de

las pruebas de aceptación. Cada historia de usuario debe ser lo suficientemente comprensible y delimitada para que los programadores puedan implementarlas en unas semanas. (SINTYA MILENA MELÉNDEZ VALLADAREZ, 2016)

A raíz del análisis realizado a los procesos relacionados con los módulos de planificación y ejecución de los servicios técnicos del sistema del INRH, se logró obtener una comprensión completa de todos los aspectos involucrados. Esto facilitó la elaboración de todas las historias de usuario durante la fase de exploración, sin necesidad de realizar modificaciones ni agregar nuevas historias de usuario a lo largo de las iteraciones. A continuación, se presentan las historias de usuario definidas en colaboración entre el cliente:

Tabla 2 Historia de Usuario 1

Historia de Usuario	
Número:1	Nombre: Gestionar Entidad.
Usuario: Dir. Técnico de la delegación.	
Prioridad en negocio: Alta.	Riesgo en desarrollo: Media.
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: Este proceso parte cuando el director técnico de la delegación elabora el listado de fichas que desea incluir en el plan del año. La gestión incluye: listar entidades dado dominio, buscar entidad, limpiar búsqueda, expandir árbol de entidades y contraer árbol de entidades.	
Observaciones:	

Tabla 3 Historia de Usuario 2

Historia de Usuario	
Número:2	Nombre: Gestionar Servicios.
Usuario: Dir. Técnico de la delegación.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 1	Iteración asignada: 1
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: El director podrá gestionar los servicios generales. La gestión incluye las acciones de Listar servicios, Adicionar servicio, Modificar servicio y Eliminar servicio.	

Observaciones:

Tabla 4 Historia de Usuario 3

Historia de Usuario	
Número: 3	Nombre: Gestionar presupuesto.
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1.4	Iteración asignada: 1
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: Los directivos podrán dar respuesta de aprobado o no el presupuesto propuesto. La gestión contiene: Desglose mensual del presupuesto, Revisado por director técnico de la delegación, Revisado por delegado, Aprobado por Dirección CTI-INRH, Cancelar por director técnico de la delegación, Cancelar por delegado, Desaprobar por Dirección CTI-INRH.	
Observaciones: Los proyectos pasan por un proceso de aprobación por parte del Dir. Técnico y Dir. Económico de la delegación. Estas fichas aprobadas son enviadas a la DCTI donde se hace una revisión y aprobación o no de cada ficha. Como resultado de las fichas aprobadas sale el presupuesto aprobado para cada delegación provincial.	

Tabla 5 Historia de Usuario 4

Historia de Usuario	
Número: 4	Nombre: Gestionar observación.
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 0.6	Iteración asignada: 1
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: Los directivos pueden adjuntar sus observaciones a la propuesta. La gestión tiene como acciones: Adicionar observación, Eliminar observación, Ver observación y Adjuntar documento (Ejemplo: Contrato o cualquier otro documento).	
Observaciones:	

Tabla 6 Historia de Usuario 5

Historia de Usuario	
Número: 5	Nombre: Gestionar reportes
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Puntos estimados: 0.6	Iteración asignada: 3
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: El sistema le permitirá al directivo generar reportes a partir de la información estadística e histórica almacenada en el mismo sobre el proceso de aprobación o no de los servicios en SCTI. Dentro de sus acciones se encuentran: Generar reporte ficha, Generar reporte Plan de Servicios Científicos Técnicos (SCT), Buscar servicio, Limpiar búsqueda, Consultar ayuda, Listar ejercicio, Calcular total del plan insertado y Calcular total plan aprobado por nivel central.	
Observaciones: Por otra parte, el nivel central también se elabora sus propias fichas que arrojan el presupuesto del plan anual del nivel central. Finalmente se obtiene el Total general conformado por el presupuesto de las delegaciones provinciales, el presupuesto del nivel central y la reserva. En el caso del nivel central también presentan el desglose mensual del presupuesto contratado.	

Tabla 7 Historia de Usuario 6

Historia de Usuario	
Número: 6	Nombre: Gestionar Servicios planificados.
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 0.4	Iteración asignada: 2
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: Una vez planificado los servicios el Dir. Técnico, Dir. Económico de la delegación o DCTI podrá Adicionar servicio, Modificar servicio, Eliminar servicio, Revisar servicio por director técnico de la delegación, Revisar servicio por delegado, Aprobar servicio por Dirección CTI-INRH, Cancelar servicio por director técnico de la delegación, Cancelar servicio por delegado, Desaprobar servicio por Dirección CTI-INRH y Desglosar presupuesto.	
Observaciones: Ninguna	

Tabla 8 Historia de Usuario 7

Historia de Usuario	
Número: 7	Nombre: Gestionar presupuesto mensual.
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 1.6	Iteración asignada: 2
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: Tanto el nivel central como las delegaciones registraran mes a mes el valor ejecutado para cada SCT, este valor estará respaldado por una o varias facturas. El total ejecutado nunca es mayor al presupuesto aprobado.	
Observaciones:	

Tabla 9 Historia de Usuario 8

Historia de Usuario	
Número: 8	Nombre: Gestionar factura mensual
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 0.2	Iteración asignada: 2
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: después de gestionar el presupuesto mensual el valor ejecutado está respaldado por facturas. El total ejecutado nunca es mayor al presupuesto aprobado. Dentro de las acciones de gestión se encuentran: gestionar factura mensual, gestionar fecha de la facturación, gestionar documento de factura y generar orden de pagos.	
Observaciones:	

Tabla 10 Historia de Usuario 9

Historia de Usuario	
Número: 9	Nombre: Gestionar observación.
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Puntos estimados: 0.6	Iteración asignada: 3
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	

Descripción: Los directivos pueden adjuntar sus observaciones a la propuesta planificada. La gestión tiene como acciones: adicionar observación, eliminar observación, ver observación y adjuntar documento.

Observaciones:

Tabla 11 Historia de Usuario 10

Historia de Usuario	
Número: 10	Nombre: Gestionar documento.
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: los directivos podrán gestionar todo lo relativo a documentos adjuntados, así como eliminar, adjuntar y adicionar ejemplo: documento del proveedor con motivos de cancelación.	
Observaciones:	

Tabla 12 Historia de Usuario 11

Historia de Usuario	
Número: 11	Nombre: Calcular total plan presupuestado.
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 4
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: En este proceso de ejecución las delegaciones podrán crear nuevos proyectos con presupuestos que sobran de proyectos que estaban planificados, pero por diferentes afectaciones no pueden ser ejecutados completamente o de presupuesto aprobado por el DCTI que se toma de la reserva. Se calcula el presupuesto MT aprobado desde la funcionalidad Plan de SCT.	
Observaciones:	

Tabla 13 Historia de Usuario 12

Historia de Usuario	
Número: 12	Nombre: Calcular Total ejecutado.
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 4
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: Sumatoria de los desgloses mensuales.	
Observaciones:	

Tabla 14 Historia de Usuario 13

Historia de Usuario	
Número: 13	Nombre: Calcular Total por ejecutar.
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 1	Iteración asignada: 3
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	
Descripción: En este proceso los directivos pueden visualizar el presupuesto total sobrante o no Total ejecución-total ejecutada.	
Observaciones:	

Tabla 15 Historia de Usuario 14

Historia de Usuario	
Número: 14	Nombre: Exportar ficha
Usuario: Dir. Técnico, Dir. Económico de la delegación, DCTI.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 0.8	Iteración asignada: 3
Programador responsable: Dasiel Torres Camejo y Thalía R. Caballero Álvarez	

Descripción: Los proyectos pasan por un proceso de aprobación por parte del Dir. Técnico y Dir. Económico de la delegación. Estas fichas aprobadas son enviadas a la DCTI donde se hace una revisión y aprobación o no de cada ficha.
Observaciones:

2.5 Planificación

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días. Las estimaciones de esfuerzo asociado a la implementación de las historias la establecen los programadores utilizando como medida el punto. Un punto, equivale a una semana ideal de programación. Las historias generalmente valen de 1 a 3 puntos. Por otra parte, el equipo de desarrollo mantiene un registro de la “velocidad” de desarrollo, establecida en puntos por iteración, basándose principalmente en la suma de puntos correspondientes a las historias de usuario que fueron terminadas en la última iteración.

La planificación se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de historias. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Al planificar según alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación (Wells, 2013).

2.6.1 Estimación del esfuerzo

Con el propósito de determinar el tiempo requerido para que el equipo de desarrollo complete todas las HU y cuántos puntos deben ser desarrollados, se procede a realizar la estimación del esfuerzo por HU. Considerando que la

velocidad del proyecto no se clarifica por completo hasta después de transcurrir varias iteraciones, se estimó inicialmente en aproximadamente 4 puntos por iteración. Las 14 HU se organizaron en tres iteraciones. La determinación de la cantidad de iteraciones se basó en dos indicadores fundamentales: la experiencia previa del equipo de desarrollo (analogía) y el ajuste permitido por la planificación guiada por el alcance del sistema (cantidad total de puntos dividida por la velocidad del proyecto), además de las necesidades inmediatas del cliente.

Tabla 16 Estimación de esfuerzo por Historia de Usuario

Iteración	Historias de Usuario		Puntos Estimados (semanas)
1	1	Gestionar Entidad	1
	2	Gestionar Servicios	1
	3	Gestionar presupuesto	1.4
	4	Gestionar Observación	0.6
2	6	Gestionar servicios planificados	0.4
	7	Gestionar presupuesto mensual	1.6
	8	Gestionar factura mensual	0.2
	10	Gestionar documento	1
3	13	Calcular total por ejecutar	1
	14	Exportar ficha	0.8
	5	Gestionar reportes	0.6
	9	Gestionar observación planificada	0.4
4	12	Calcular Total ejecutado	1
	11	Calcular Total del plan presupuestado	1
Total			12

2.6.2 Plan de iteraciones

Una vez que se han identificado y descrito las Historias de Usuario (HU) y se ha estimado el esfuerzo requerido para su implementación en XP (Extreme Programming), se procede a la generación del artefacto denominado "Plan de Iteraciones". Este plan detalla cuáles HU se implementarán en cada iteración, lo que contribuye a optimizar el desempeño del equipo de desarrollo. Al concluir la última iteración, el sistema estará preparado para su lanzamiento en producción.

Posteriormente, se procede a la planificación de la etapa de implementación, en la cual las HU se desarrollarán y probarán en un ciclo iterativo, transformándose en entregas incrementales (TI). Para garantizar un desempeño organizado, el equipo ha establecido un total de cuatro iteraciones.

- **Iteración 1**

En esta iteración se implementarán HU de nivel de prioridad alta que estén relacionadas con el subproceso de planificación de los servicios técnicos del INRH, creando con ella la línea base de la arquitectura del sistema. Además, se obtendrá una primera versión del producto en la que el cliente podrá probar algunas de las funcionalidades solicitadas como gestionar servicios, gestionar entidad, gestionar presupuesto y gestionar Observación HU 1, 2, 3 y 4.

- **Iteración**

En esta iteración se implementarán el resto de las HU con prioridad alta en el negocio y se corregirán errores o no conformidades de los usuarios con los componentes implementados en la iteración anterior. Como resultado se obtendrá una versión más completa sobre las funcionalidades del sistema. Al culminar esta iteración los usuarios tendrán acceso entonces a funcionalidades tanto del subsistema de planificación como ejecución.

- **Iteración 3**

Esta iteración es la dedicada a la implementación HU de nivel de prioridad media. Se corregirán errores o no conformidades de los usuarios con las funcionalidades desarrolladas en la iteración anterior. Como resultado se obtendrá un avance bastante significativo en la implementación de las funcionalidades más objetivas.

- **Iteración**

En esta iteración se implementarán HU de baja prioridad, completando el desarrollo de los módulos de planificación y ejecución de los servicios técnicos del INRH. Se obtendrán dos subsistemas que responderán a las necesidades inmediatas del cliente en el cual los usuarios podrán ejecutar todas las funcionalidades existentes en ambos módulos y explotarlo a plenitud.

Tabla 17 Plan de duración de iteraciones

Iteración	Historias de Usuario		Duración (semanas)
1	1	Gestionar Entidad	4.0
	2	Gestionar Servicios	
	3	Gestionar presupuesto	
	4	Gestionar Observación	
2	1	Gestionar servicios planificados	3.2
	2	Gestionar presupuesto mensual	
	3	Gestionar factura mensual	
	4	Gestionar documento	
3	1	Calcular total por ejecutar	2.8
	2	Exportar ficha	
	3	Gestionar reportes	
	4	Gestionar observación planificada	
4	1	Calcular Total ejecutado	2.0
	2	Calcular Total del plan presupuestado	
	Total		12.0

2.6.3 Plan de Entregas

Con el propósito de mantener al cliente centrado en la mejora del producto y permitir que sea el propio cliente quien realice las pruebas de campo, se llevan a cabo varias entregas del sistema. A continuación, se presenta una tabla que refleja lo mencionado, detallando las fechas en las que se llevan a cabo las reuniones entre el equipo de desarrollo y los propietarios del software para la liberación de las versiones del producto:

Tabla 18 Plan de entregas

Iteración	Fecha de entrega
Iteración 1	24 de julio de 2023
Iteración 2	14 de agosto de 2023
Iteración 3	4 de septiembre de 2023

2.7 Diseño

En la fase de diseño el ciclo de vida de desarrollo de sistemas, el analista utiliza la información recopilada en las primeras fases para realizar el diseño lógico del sistema de información. El analista diseña procedimientos precisos para la captura de datos que aseguran que los datos que ingresen al sistema de información sean correctos. También incluye el diseño de archivos y base de datos que almacenarán gran parte de los datos indispensables para los encargados de tomar decisiones en la organización. Una base de datos bien organizada es el crecimiento de cualquier sistema de información. (Kenneth E. Kendall, 2005).

XP construye un proceso de diseño evolutivo que se basa en refactorizar un sistema simple en cada iteración. Todo el diseño se centra en la iteración actual y no se hace nada anticipadamente para necesidades futuras. El resultado es un proceso de diseño disciplinado, que combina la disciplina con la adaptabilidad de una manera que la hace una de las más desarrolladas entre todas las metodologías ágiles. (CANÓS, y otros, 2011).

2.7.1 Patrón arquitectónico

Un patrón es un diseño que ofrece soluciones a problemas de arquitectura en ingeniería de software. Da una descripción de los elementos y el tipo de relación que tienen junto con un conjunto de restricciones sobre cómo pueden ser usados. Este expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, con sus responsabilidades e interrelaciones.

El patrón **MVC (Modelo, Vista, Controlador)** es un patrón de arquitectura de software que separa los datos de una aplicación, separando así la interfaz que ve el usuario de la lógica del negocio en tres componentes distintos. El patrón MVC permite facilitar el problema de la reutilización del código, ya que al ser utilizado el código es separado en varias partes que pueden ser reutilizadas sin modificaciones. A continuación, se enuncia una breve descripción de dicho modelo:

- **Vista (view):** representa la interfaz de usuario y todas las herramientas con las cuales el usuario hace uso del programa.
- **Modelo (model):** es donde está toda la lógica del negocio, la representación de todo el sistema incluido la interacción con una base de datos, si es que el programa así lo requiere.
- **Controlador (controller):** este componente es el que responde a la interacción (eventos) que hace el usuario en la interfaz y realiza las peticiones al modelo para pasar éstos a la vista.

Al dividirse en varias capas permite que se logre un mejor entendimiento a la hora de dar soporte, debido a que cada código se sitúa en su carpeta correspondiente. La independencia que tiene el modelo de la vista permite que la interfaz de usuario pueda mostrar múltiples vistas de los mismos datos simultáneamente y además agregar nuevas opciones de presentación que generalmente no afecta al modelo (Dayana Benitez Betancourt, 2022).

2.7.2 Patrones de diseño

Un patrón de diseño de software es una técnica para resolver problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Un patrón de diseño resulta ser una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias.

Los patrones de diseño hacen que sea más fácil reutilizar buenos diseños y arquitecturas. Los patrones de diseño ayudan a elegir las alternativas de diseño que hacen que un sistema sea reutilizable, y evitan aquellas que dificultan dicha reutilización. Un patrón de diseño nombra, abstrae e identifica los aspectos clave de una estructura de diseño común, lo que los hace útiles para crear un diseño orientado a objetos reutilizable. El patrón de diseño identifica las clases e instancias participantes, sus roles y colaboraciones, y la distribución de

responsabilidades. En definitiva, los patrones de diseño ayudan a un diseñador a lograr rápidamente un buen diseño (Erich Gamma, 1995).

Patrones GRASP:

GRASP *General Responsibility Assignment Software Patterns* (en español Patrones Generales de Software para la Asignación de Responsabilidades) son una serie de buenas prácticas enfocadas a la calidad del software, calidad "estructural" por llamarla de alguna manera, ya que no se ha centrado en pruebas sino en la estructura y las responsabilidades de las clases que componen el software que desarrollamos. GRASP hay que considerarlo como una serie de buenas prácticas, unos pocos consejos, que hay que seguir si se quiere hacerlo bien (Carmona, 2012).

Ejemplo de patrón GRASP identificado:

1. Controlador

La clase “*EjecutorasModel*” actúa como un controlador, gestionando las operaciones relacionadas con la manipulación y obtención de datos. Este enfoque sigue el principio de diseño GRASP de asignar la responsabilidad de control a una clase especializada.

```
1  <?php
2
3  11 references | 0 implementations | Comment Code | Comment Code
4  class EjecutorasModel extends ZendExt_Model {
    0 references | 0 overrides
```

Ilustración 2 Patrones Grasp

2. Experto en Información

La clase “*EjecutorasModel*” tiene la responsabilidad de agregar datos representativos (“*addDatosRep*”). Este sigue el principio de Experto en Información, que sugiere asignar la responsabilidad a la clase que tiene la información necesaria para cumplir con dicha tarea de manera eficiente.

```

246 + 1 reference | 0 overrides
247 public function addDatosRep($params) {
248     $dm = Doctrine_Manager::getInstance();
249     $conn = $dm->getConnection();
250
251     $query = "DELETE FROM mod_serviciosct.conf_representante WHERE identidad = ". $params['identidad'];
252     $conn->execute($query);
253     $query = "INSERT INTO mod_serviciosct.conf_representante(identidad, representante, cargo) VALUES (".$params['identidad'].",".$params['rep
254     $conn->execute($query);
255     return array(codM => 1);
256 }

```

Ilustración 3 Patrones Grasp

Patrones GoF:

Los patrones Gang of Four (en español Banda de los cuatro) (GoF) surgen como una forma indispensable de enfrentar los problemas propios de la codificación. Fueron dados a conocer en el libro “Design Pateterns–Elements of Reusable Software” de Erich Gamma, Richard Helm, Ralph Jonson y John Vlissides, a partir de entonces estos patrones son conocidos como los patrones de la pandilla de los cuatro (GoF, gang of four). El empleo de los mismos garantiza una solución efectiva y a prueba de fallos para cada problema en cuestión, que no puede ser solucionado por las vías normales (Erich Gamma, 1995).

Ejemplo de patrón GoF identificado:

1. Singleton

- Este patrón garantiza que una clase tenga solo una instancia y proporciona un punto de acceso global a ella. En este fragmento de código, se visualiza en “ZendExt_IoC::getInstance()” que sigue el diseño del patrón Singleton, asegurando que solo haya una instancia de la clase “ZendExt_IoC”.

```

1  <?php
2
3  11 references | 0 implementations | Comment Code | Comment Code
4  class EjecutorasModel extends ZendExt_Model {
5
6      0 references | 0 overrides
7      public function listarDiagnostico($params) {
8          return DatDiagnostico::listarDiagnostico($params);
9      }
10
11      0 references | 0 overrides
12      public function listarDeficiencia($params) {
13          $datos = DatDiagnosticodeficiencia::listarDeficiencia($params);
14          for ($i = 0; $i < count($datos->data); $i++) {
15              $integrator = ZendExt_Ioc::getInstance();
16              $idummayor = $datos->data[$i]['idummayor'];
17              $datNomummayor = $integrator->metadatos->DameEstructura($idummayor);
18              $datos->data[$i]['denominacionummayor'] = $datNomummayor[0]->denominacion;
19              $idummenor = $datos->data[$i]['idummenor'];
20              $datNomummenor = $integrator->metadatos->DameEstructura($idummenor);
21              $datos->data[$i]['denominacionummenor'] = $datNomummenor[0]->denominacion;
22          }
23          return $datos;
24      }
25  }

```

Ilustración 4 Patrones GoF

2.7.3 Tarjetas CRC

En el proceso de diseño de aplicaciones según la metodología XP, no se exige la representación del sistema a través de diagramas de clases utilizando el Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language). En cambio, se emplean otras técnicas, como las tarjetas CRC, que actúan como una extensión informal de UML.

Las tarjetas CRC (Class Responsibility Collaborator) es una actividad en grupo de modelado orientado a objetos en la cual el equipo puede manifestar y debatir ideas acerca del diseño de un sistema. Hace especial énfasis en la simplicidad, comunicación y límites de un sistema. Se suele utilizar en las primeras fases del desarrollo de una historia como paso previo a la implementación o escritura de un esquema UML. Esta herramienta fue creada por Ward Cunningham and Kent Beck.

La técnica de las tarjetas CRC se puede usar para estructurar el sistema a través de análisis guiados por la responsabilidad. Las clases se examinan, se filtran y se refinan sobre la base de sus responsabilidades con respecto al sistema, y las clases con las que necesitan colaborar para completar sus responsabilidades. La elaboración de las tarjetas CRC es una técnica orientada a establecer un planteo inicial de los objetos que participan en el sistema. El producto de la aplicación de esta técnica, es un conjunto de tarjetas asociadas a las clases, en

las que se especifican las responsabilidades y sus colaboraciones con otras clases.

Las tarjetas CRC identifican y organizan las clases bajo el paradigma orientado a objetos, lo que incluye asignación de responsabilidades. Cada tarjeta contiene el nombre de la clase (que representa una o más historias de usuario), una descripción de las responsabilidades o métodos asociados con la clase, así como la lista de las clases con que se relaciona o que colaboran con ella. Las tarjetas CRC son el único trabajo de diseño que se genera como parte del proceso de XP (Barry J. Holmes, 2001).

Tabla 19 Tarjeta CRC #1

Tarjeta CRC	
Clase: PlanificaciónServicio	
Responsabilidad: Almacenar la información referente a las tareas de planificación de los nuevos proyectos y Generar formularios para el registro y actualización de clases entidades.	Colaboración: • EntidadEjecutora • Título • Presupuesto • AprobadoxDirector

Tabla 20 Tarjeta CRC #2

Tarjeta CRC	
Clase: EjecuciónServicio	
Responsabilidad: Almacenar la información referente a las tareas de ejecución.	Colaboración: • PlanificaciónServicio • TerminarEjecucion

Tabla 21 Tarjeta CRC #3

Tarjeta CRC	
Clase: DesgloseEjecución	
Responsabilidad: Desglosar la información referente a las tareas de ejecución.	Colaboración: • EjecucionServicio • Ejercicio

	• Importe
--	---

Tabla 22 Tarjeta CRC #4

Tarjeta CRC	
Clase: DesglosePresupuesto	
Responsabilidad: Desglosar la información referente a las tareas de Presupuesto.	Colaboración: • PlanificacionServicio • Enero • Febrero

Tabla 23 Tarjeta CRC #5

Tarjeta CRC	
Clase: PlanificaciónServicioObservación	
Responsabilidad: Mostrar la información referente a las tareas de planificación de los nuevos proyectos.	Colaboración: • PlanificaciónServicio • Usuario • Fecha • Asunto

Tabla 24 Tarjeta CRC #6

Tarjeta CRC	
Clase: Entidad Ejecutora	
Responsabilidad: Almacenar la información referente a las entidades que ejecutan los nuevos proyectos.	Colaboración: • Entidad • Provincia • Municipio

2.7.4 Modelo de datos

Una vez que se han diseñado las tarjetas CRC y definidas las clases de entidades, se procede a diseñar el modelo físico de la base de datos que el sistema utilizará para almacenar la información. El modelo de datos permite describir las principales clases que componen la capa del modelo del sistema, los tipos de datos de los atributos que contendrá la base de datos y las relaciones

entre las entidades. Facilita la descripción de los elementos de la realidad que participan en un problema específico y cómo se relacionan entre sí. En la [ilustración 16](#), se presenta el diagrama entidad-relación que representa el modelo físico de datos diseñado para el sistema

2.8 Conclusiones del capítulo

En este capítulo, se ha llevado a cabo un análisis de cómo se desarrolla actualmente el proceso de Gestión de Proyectos de Inversiones dentro del INRH. Basándose en este análisis, se ha propuesto una solución que cumple con el segundo objetivo de esta investigación, que es la automatización de dicho proceso mediante una aplicación web.

En la etapa de diseño, se definieron los elementos de la estructura del sistema. El uso de patrones generales de software para asignar responsabilidades garantizó una solución que se basa en la reutilización de código y resuelve problemas en contextos similares dentro del desarrollo de software. Se elaboraron las tarjetas CRC, las cuales permitieron definir las clases, sus relaciones a través de colaboradores y sus responsabilidades. Esto posibilitó la reducción del acoplamiento y el aumento de la reutilización en la herramienta. Este capítulo contribuyó al cumplimiento del tercer objetivo específico de la investigación.

Capítulo 3: Implementación y pruebas

3.1 Introducción

Una vez completada la fase de diseño, que establece las bases estructurales del sistema, se procede a la etapa de implementación y pruebas. En esta fase, se desglosan las HU en TI para simplificar su entendimiento y ejecución. Se lleva a cabo la codificación de las pruebas unitarias, la implementación de las TI y la ejecución de las pruebas de aceptación. Esta etapa destaca por ser el período con resultados concretos y significativos, culminando en un sistema finalizado y validado por el cliente.

Este capítulo aborda aspectos como los estándares de codificación aplicados al código generado, las iteraciones realizadas durante la implementación de las HU, así como las pruebas realizadas y sus resultados.

3.2 Implementación y pruebas

En la fase de implementación y pruebas se realizan las iteraciones planificadas. Durante el transcurso de las iteraciones se realiza la implementación de las historias de usuario seleccionadas para cada una de estas. Al inicio de las mismas, se lleva a cabo una revisión del plan de iteraciones y se modifica de ser necesario. Como parte de este plan, se descomponen las HU en TI o tareas de desarrollo. Estas tareas son para el uso de los programadores, pueden escribirse utilizando un lenguaje técnico y no necesariamente deben ser entendibles para el cliente (BECK, 2000).

Antes de llevar a cabo la implementación de las tareas, se desarrollan pruebas unitarias para cada una de ellas, con el propósito de evaluar la funcionalidad y detectar posibles fallos. Una vez finalizada cada iteración, se entrega una versión del producto al propietario del software y se procede a realizar pruebas de aceptación, asegurando la validación de la satisfacción del cliente con el sistema. Siguiendo la planificación establecida, se llevaron a cabo un total de cuatro iteraciones de desarrollo en el sistema, culminando con un producto listo para su despliegue.

3.2.1 Tareas de ingeniería

En la fase de implementación y prueba, las Tareas de Ingeniería (TI) desempeñan un papel importante al almacenar los detalles necesarios para

llevar a cabo cada historia de usuario. La complejidad de una HU puede determinar la presencia de múltiples TI asociadas. El equipo de desarrollo especifica claramente a qué HU pertenecen las TI, representándolas mediante tablas divididas en secciones que incluyen el número de la tarea, el número incremental de la HU, el nombre de la tarea, el tipo de tarea, los puntos estimados, las fechas de inicio y fin según la HU, el programador responsable y una descripción detallada de la funcionalidad a implementar.

Las TI es el artefacto en el que se apoya el equipo desarrollador para dejar plasmado la manera en que se le da solución a la HU a la que está asignada la TI (Villegas, 2018).

A continuación de enumeran las tareas de ingeniería generadas en esta fase:

Tabla 25 Tarea de ingeniería # 1

Tarea TI1UH1	
Número de la tarea:1	Número de la UH:1
Nombre de la tarea: Buscar entidad	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 10/06/23	Fecha fin: 15/06/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: El usuario en el campo vacío de búsqueda, busca la entidad ejecutora y selecciona una de ellas en el árbol de entidades. Esta acción le debe permitir realizar un filtrado en el árbol, una vez insertado un criterio de búsqueda en el campo de texto. La búsqueda se efectuará en dependencia de la pestaña que se encuentre enmarcada en ese momento. En este caso la búsqueda se efectuaría sobre el árbol de entidades ejecutoras. Esta acción le debe permitir buscar un objetivo del sistema. Esta funcionalidad solo se habilitará si se selecciona un objetivo del listado. Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Búsqueda avanzada de objetivo(s)".	

Tabla 26 Tarea de ingeniería # 2

Tarea TI2HU2	
Número de la tarea:2	Número de la UH:2
Nombre de la tarea: Listar servicios	

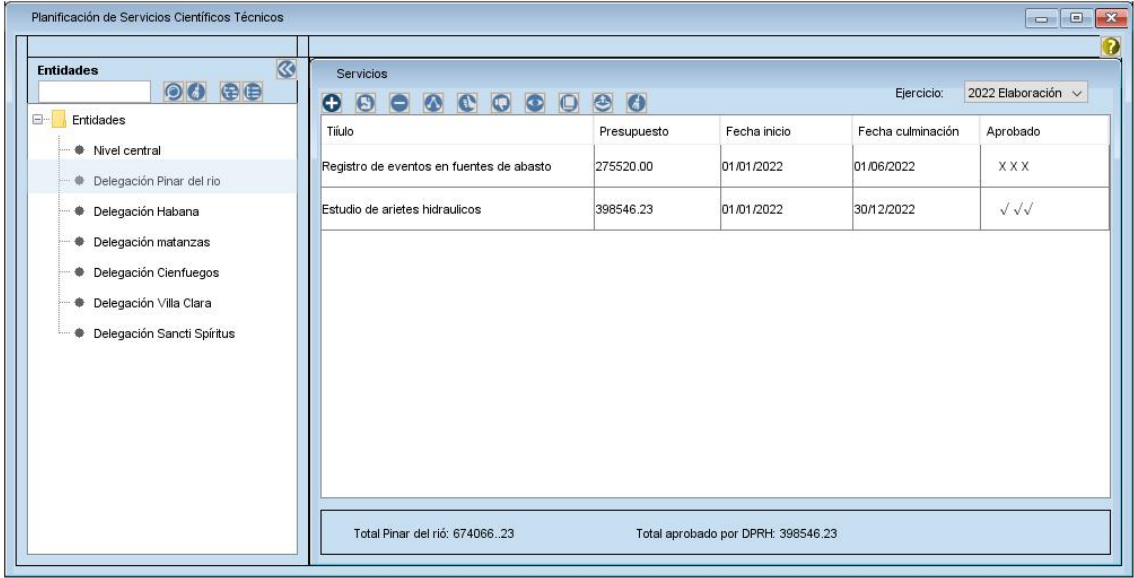
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 16/06/23	Fecha fin: 18/06/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: Después de adicionar los servicios a planificar en la interfaz: gestionar servicios tiene la opción de listar todos los servicios a planificar visualizando el presupuesto, fecha de inicio y fin y si está aprobado o no.	
	

Tabla 27 Tarea de ingeniería # 3

Tarea TI3UH2	
Número de la tarea: 3	Número de la UH: 2
Nombre de la tarea: Adicionar servicio	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 18/06/23	Fecha fin: 20/06/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	

Descripción: El usuario ingresa los datos de la entidad ejecutora con nombre, representante, cargo, teléfono, correo, fax, dirección, ciudad y provincia.

Aquí se especifican los datos del proyecto o trabajo, se rellenan los campos de título del proyecto, presupuesto requerido, fecha de inicio y fecha de fin.

Se selecciona la prioridad de la política nacional del agua que tributa el trabajo, con las siguientes opciones (se pueden seleccionar más de una): uso racional y productivo del agua, uso eficiente de la infraestructura hidráulica, gestión de riesgos a la calidad del agua y gestión de eventos asociados a eventos extremos del clima.

Se debe seleccionar en un campo aparte los problemas a resolver del banco de problemas (se pueden seleccionar todos o ninguno).

En esta ventana se brindará la posibilidad de guardar los datos especificados y cerrar la ventana. Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Guardar datos y cerrar ventana".

Una vez invocado este botón, el sistema confirma el registro de los datos y muestra el mensaje de información: "La fila ha sido adicionada satisfactoriamente.", quedando actualizados los datos de la matriz con cada una de las filas previamente insertadas.

- Es importante tener en cuenta que el sistema no debe permitir el registro de una fila con igual denominación que otra, de suceder, se debe mostrar el mensaje de información: "La fila que desea adicionar ya existe.", manteniéndose así en la misma interfaz para especificar otra denominación de la fila que desea registrar en el sistema.
- Este botón deberá aparecer deshabilitado hasta tanto no se hayan especificado los datos obligatorios.

Al cerrar la ventana sin guardar los datos.

Nota: al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Cerrar ventana sin guardar los datos".

Adicionar servicio

Datos
Observaciones

DATOS ENTIDAD EJECUTORA

Nombre entidad:

Representante:
Cargo:
Correo:

Teléfono:
Móvil:
FAX:

Dirección:

Ciudad:
Provincia:

DATOS DEL TRABAJO

Título:
Registro de eventos en fuentes de abasto

Presupuesto CUP:
275520.00
Fecha inicio:
01/01/2022
Fecha fin:
30/12/2022

Prioridad de la política nacional del agua que tributa el trabajo:

☒ Uso racional y productivo del agua
☐ Gestión de riesgos asociados a la calidad del agua

☐ Uso eficiente de la infraestructura hidráulica
☒ Gestión de eventos asociados a eventos extremos del clima

Banco de problemas:

☒ Problema 1
☒ Problema 2
☐ Problema 3

1 Objetivo 1

1.1 Indicador

☒ 1.1.1 Indicador complementario1
☐ 1.1.2 Indicador complementario2

1.2 Indicador 2

☐ 1.2.1 Indicador complementario 3
☒ 2.1.2 Indicador complementario 4

2 Objetivo 2

2.1 Indicador

Cancelar
Aplicar
Aceptar

Tabla 28 Tarea de ingeniería # 4

Tarea TI4UH2	
Número de la tarea:4	Número de la UH:2
Nombre de la tarea: Modificar servicio	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 22/06/23	Fecha fin: 24/06/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: El usuario tiene la opción de modificar el servicio previamente insertado mediante un botón en la esquina superior izquierda, el cual le permitirá editar cualquiera de los campos anteriormente descritos.	

Al cerrar la ventana sin guardar los datos.

Nota: al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Cerrar ventana sin guardar los datos".

Tabla 29 Tarea de ingeniería # 5

Tarea TI5UH2	
Número de la tarea: 5	Número de la UH: 2
Nombre de la tarea: Eliminar servicio	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 24/06/23	Fecha fin: 26/06/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: El usuario tiene la opción de eliminar el servicio previamente insertado mediante un botón en la esquina superior izquierda.	

Estas tareas de ingeniería están asociadas a las historias de usuarios catalogadas con alta prioridad, mientras que las restantes se encuentran detalladas en los [anexos](#) correspondientes.

3.3 Pruebas de caja blanca

La prueba de caja blanca, en ocasiones llamada prueba de caja de vidrio, es una filosofía de diseño de casos de prueba que usa la estructura de control descrita como parte del diseño a nivel de componentes para derivar casos de prueba. Al usar los métodos de prueba de caja blanca, puede derivar casos de prueba que:

- 1) Garanticen que todas las rutas independientes dentro de un módulo se revisaron al menos una vez.
- 2) Revisen todas las decisiones lógicas en sus lados verdadero y falso.
- 3) Ejecuten todos los bucles en sus fronteras y dentro de sus fronteras operativas.
- 4) Revisen estructuras de datos internas para garantizar su validez (Pressman, 2014).

Roger S. Pressman ha destacado la importancia de las pruebas de caja blanca en el desarrollo de software. Según Pressman, las pruebas de caja blanca, también conocidas como pruebas estructurales, se centran en examinar la lógica

interna y la estructura del código fuente. Afirmó que este enfoque es crucial para lograr una cobertura del código y garantizar la calidad del software.

Pressman abogó por el uso de diversas técnicas de prueba de caja blanca, como la prueba de camino básico, la prueba de bucles, y la prueba de condiciones, para evaluar diferentes aspectos del código. Además, señaló que estas pruebas son fundamentales para identificar y corregir posibles errores en la implementación del software, asegurando así un producto más robusto y fiable.

3.3.1 Test unitarios

Las pruebas unitarias consisten en aislar una parte del código y comprobar que funciona a la perfección. Son pequeños test que validan el comportamiento de un objeto y la lógica. El unit testing suele realizarse durante la fase de desarrollo de aplicaciones de software o móviles. Normalmente las llevan a cabo los desarrolladores (Yeeply, 2021).

Con ellas se detectan antes errores que, sin las pruebas unitarias, no se podrían detectar hasta fases más avanzadas como las pruebas de sistema, de integración e incluso en la beta. Realizar pruebas unitarias con regularidad supone, al final, un ahorro de tiempo y dinero.

Se empleó Apache JMeter para la realización de pruebas de carga y estrés, herramienta encargada de desempeñar un papel fundamental en el proceso de evaluación y optimización de sistemas de software en términos de su capacidad de rendimiento y resistencia. Apache JMeter, una herramienta ampliamente reconocida y utilizada en el campo de la ingeniería de rendimiento de software, se empleó para llevar a cabo pruebas de carga y estrés en el contexto de esta investigación de tesis.

La elección de Apache JMeter se basó en su capacidad para simular múltiples usuarios concurrentes, generando cargas controladas y escalables en el sistema en estudio. Esta herramienta, respaldada por su estructura altamente personalizable, permitió la generación de escenarios de prueba que imitaban situaciones reales, lo que resultó en la obtención de datos fiables y significativos relacionados con el rendimiento del sistema.

En la Ilustración 3 se exponen los resultados derivados de la ejecución de los test unitarios aplicados a la base de datos del proyecto. Como criterio de evaluación, se definió la expectativa de obtener respuestas por parte de la base de datos en un intervalo de tiempo inferior a los 10 segundos.

Ilustración 5 Resultados de pruebas al software

Informe Agregado

Nombre: Informe Agregado

Comentarios: Pruebas a la BD

Escribir todos los datos a Archivo

Nombre de archivo: Navegar Log/Mostrar sólo: ☐ Escribir en Log Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
Petición JDBC	10000	817	760	1234	0	3290	0.00%	1036.8/sec	42.5
Total	10000	817	760	1234	0	3290	0.00%	1036.8/sec	42.5

En la Ilustración 4 se presentan los resultados obtenidos tras la ejecución de los test unitarios aplicados directamente al proyecto en cuestión.

Ilustración 6 Resultados de pruebas al software

Summary Report

Name: Reporte resumen

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: ☐ Errors ☐ Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Petición HTTP	200	15295	0	26743	7775.19	0.00%	7.2/sec	3.93	1.87	560.0
TOTAL	200	15295	0	26743	7775.19	0.00%	7.2/sec	3.93	1.87	560.0

☐ Include group name in label? ☒ Save Table Header

3.4 Pruebas de caja negra

Las pruebas de caja negra, también llamadas pruebas de comportamiento, se enfocan en los requerimientos funcionales del software; es decir, las técnicas de prueba de caja negra le permiten derivar conjuntos de condiciones de entrada que revisarán por completo todos los requerimientos funcionales para un programa. Las pruebas de caja negra no son una alternativa para las técnicas de caja blanca. En vez de ello, es un enfoque complementario que es probable que descubra una clase de errores diferente que los métodos de caja blanca. Las pruebas de caja negra intentan encontrar errores en las categorías siguientes (Pressman, 2014):

- 1) Funciones incorrectas o faltantes.
- 2) Errores de interfaz.
- 3) Errores en las estructuras de datos.
- 4) Errores en acceso a las bases de datos externas.
- 5) Errores de comportamiento o rendimiento.
- 6) Errores de inicialización y terminación.

3.4.1 Técnica: Partición de equivalencia

La técnica de partición de equivalencia es una estrategia utilizada en las pruebas de software, específicamente dentro del enfoque de prueba de caja negra. Su objetivo es simplificar la creación de casos de prueba al dividir el conjunto de datos de entrada en grupos o clases equivalentes, donde se espera un comportamiento similar.

Esta técnica se basa en la premisa de que, si un conjunto de datos de entrada funciona correctamente para una clase de equivalencia, se espera que funcione para cualquier otro dato de esa misma clase. Por lo tanto, no es necesario probar cada dato individualmente, lo que ahorra tiempo y recursos (Jaramillo, 2014).

La metodología de partición de equivalencia constituye un enfoque esencial en el análisis del dominio de entrada de un programa. A través de la identificación y segmentación de clases de datos representativas, esta técnica facilita la generación eficiente de casos de prueba. El propósito fundamental es la elaboración de casos de prueba que revelen específicas clases de errores, con

el objetivo de optimizar la eficacia del proceso de pruebas y, al mismo tiempo, minimizar la carga de trabajo asociada.

En la práctica, esta técnica busca definir casos de prueba selectivos que sean capaces de descubrir fallos particulares, contribuyendo así a la reducción significativa del número total de casos de prueba necesarios.

Casos de pruebas de partición de equivalencia

Un ejemplo de los escenarios de los casos de pruebas generados se puede observar en la Tabla 68 en los [anexos](#).

A continuación, se muestran los resultados obtenidos de la prueba de caja negra utilizando la técnica de partición de equivalencia realizado al software en 4 iteraciones, la Tabla 30 brinda información del total de no conformidades encontradas de tipo funcionales, de ortografía y solicitudes de cambios. También se muestra las que fueron resueltas en cada iteración. Con un total de 43 requisitos funcionales se detectaron 12 no conformidades en la primera iteración resolviéndose satisfactoriamente 9, en la segunda iteración se detectaron 3 no conformidades resolviéndose satisfactoriamente 2, en la tercera iteración se detectó 1 no conformidad resolviéndose satisfactoriamente y en la cuarta iteración no se detectaron no conformidades, con un total de 12 no conformidades.

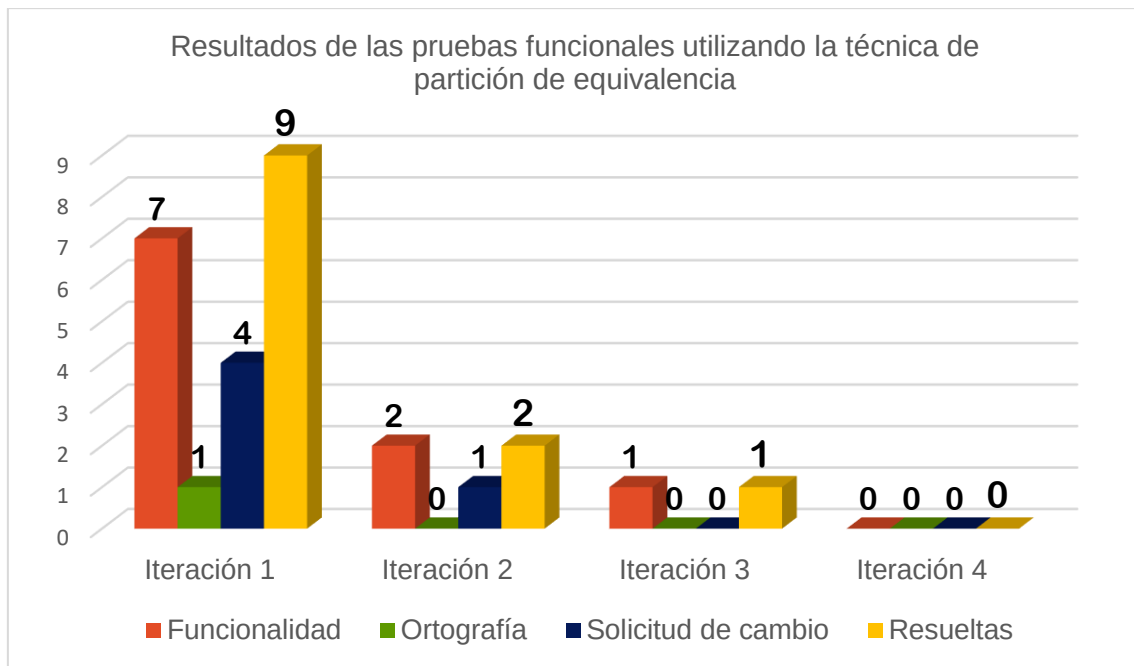


Ilustración 7 Resultado de pruebas funcionales de caja negra

3.4.2 Pruebas de aceptación

El propósito fundamental de las pruebas de aceptación radica en la validación del cumplimiento de un sistema, asegurando su funcionamiento conforme a lo esperado. Esto permite al usuario del sistema determinar si lo acepta, considerando tanto su funcionalidad como su rendimiento. Las pruebas de aceptación son diseñadas por el usuario del sistema y preparadas por el equipo de desarrollo, si bien su ejecución y aprobación final recaen en el usuario. Estas pruebas se centran en verificar que el sistema satisface los requisitos de funcionamiento esperados, los cuales se encuentran definidos en el catálogo de requisitos y en los criterios de aceptación del sistema de información. Esto culmina en la aceptación final del sistema por parte del usuario. El responsable de los usuarios debe supervisar los criterios de aceptación previamente establecidos en el plan de pruebas del sistema y, posteriormente, dirigir las pruebas de aceptación finales.

Las pruebas de aceptación son representadas mediante tablas divididas por las siguientes secciones:

- **Código:** Código identificativo para cada prueba.
- **Historia de usuario:** Número de la HU que se prueba.

- **Nombre:** Identifica la HU que se prueba.
- **Descripción:** Breve descripción de la prueba.
- **Condiciones de ejecución:** Requerimientos que debe cumplir el sistema para poder ejecutar la funcionalidad.
- **Pasos de ejecución:** Pasos a tener en cuenta para ejecutar la funcionalidad.
- **Resultados esperados:** Resultado obtenido por el sistema luego de ejecutar la funcionalidad.

Tabla 30 Caso de prueba de aceptación # 1

Caso de prueba de aceptación	
Código: HU1_P1	Historia de Usuario: 1
Nombre: Gestionar Entidad.	
Descripción: Prueba para la funcionalidad de gestionar una entidad	
Condiciones de ejecución: Debe estar autenticado	
Pasos de ejecución: El usuario en el campo vacío de búsqueda, busca la entidad ejecutora y selecciona una de ellas en el árbol de entidades.	
Resultados esperados: El sistema muestra un listado de los objetivos que cumplan con los criterios de búsqueda especificados. Permite realizar un filtrado en el árbol, una vez insertado un criterio de búsqueda en el campo de texto. En este caso la búsqueda se efectuó sobre el árbol de entidades ejecutoras.	

Tabla 31 Caso de prueba de aceptación # 2

Caso de prueba de aceptación	
Código: HU2_P2	Historia de Usuario: 2
Nombre: Gestionar servicios	
Descripción: Prueba para la funcionalidad gestionar servicios	
Condiciones de ejecución: Deben existir entidades ejecutoras	
Pasos de ejecución: El usuario ingresa los datos de la entidad ejecutora con nombre, representante, cargo, teléfono, correo, fax, dirección, ciudad y provincia. Se especifican los datos del proyecto o trabajo, se rellenan	

los campos de título del proyecto, presupuesto requerido, fecha de inicio y fecha de fin. Y por último guardar los datos especificados y cerrar la ventana.

Resultados esperados:

Una vez invocado el botón de guardar, el sistema confirmó el registro de los datos y muestra el mensaje de información: "La fila ha sido adicionada satisfactoriamente.", quedando actualizados los datos de la matriz con cada una de las filas previamente insertadas.

El sistema no permitió el registro de una fila con igual denominación que otra, y se mostró el mensaje de información: "La fila que desea adicionar ya existe.", manteniéndose así en la misma interfaz para especificar otra denominación de la fila que desea registrar en el sistema.

Después de adicionar los servicios a planificar en la interfaz se listó todos los servicios a planificar visualizando el presupuesto, fecha de inicio y fin y si está aprobado o no de manera satisfactoria

Tabla 32 Caso de prueba de aceptación # 3

Caso de prueba de aceptación	
Código: HU3_P3	Historia de Usuario: 3
Nombre: Desglose mensual del presupuesto	
Descripción: Prueba para el desglose mensual del presupuesto	
Condiciones de ejecución: Debe existir al menos un servicio adicionado	
Pasos de ejecución: Después de añadido el servicio puede desglosar en meses el presupuesto presionando el botón desglosar presupuesto.	
Resultados esperados: Permitió desglosar un servicio planificado por meses en el sistema.	

Tabla 33 Caso de prueba de aceptación # 4

Caso de prueba de aceptación	
Código: HU4_P4	Historia de Usuario: 4
Nombre: Gestionar observación	
Descripción: Prueba para gestionar observación del servicio	
Condiciones de ejecución: Para poder adicionar una observación al plan debe haber al menos un servicio adicionado para poder añadir alguna observación.	
Pasos de ejecución: El usuario en la pestaña añadir observación rellena los campos vacíos con asunto de la observación y observación con no más de 255 caracteres.	

Resultados esperados: Se presiona el botón aceptar o aplicar y el sistema muestra el mensaje de información: "La observación ha sido adicionada satisfactoriamente. "en el caso de cancelar la operación Se presiona el botón Cancelar el sistema y cancela la operación y se cierra la ventana. En el caso de que el usuario deje algún campo vacío el sistema deshabilita el botón Aceptar o Aplicar y subraya el campo incorrecto mostrando el mensaje: "Este campo es obligatorio." al pasar el mouse por encima del campo subrayado.

A continuación, se muestran los resultados obtenidos en cuatro iteraciones de pruebas realizadas al sistema, durante la primera iteración se recibieron solicitudes de modificación por parte del cliente en 2 de ellos. Sin embargo, los casos restantes fueron validados y aceptados por el cliente al estar en conformidad con los resultados esperados. Posteriormente, después de abordar las 2 modificaciones requeridas y las 3 no conformidades, se aprobó la propuesta de solución presentada (consultar Ilustración en [anexos](#) Acta de aceptación del proyecto por parte del cliente). En la figura siguiente se representa gráficamente los resultados de las pruebas de aceptación realizadas mediante un gráfico de barras.

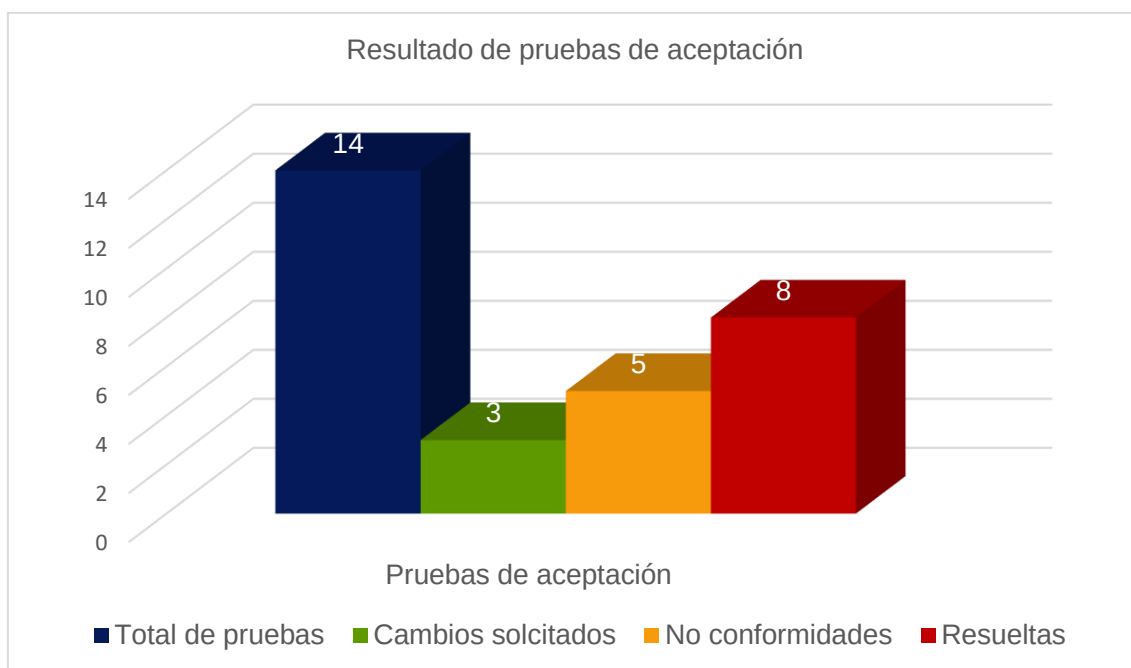


Ilustración 8 Resultado de pruebas de aceptación

3.5 Conclusiones del capítulo

En este capítulo se reflejó la ejecución de la fase de implementación y pruebas, arribando a las siguientes conclusiones:

- Se realizaron las 4 iteraciones planificadas, implementándose un total de 14 HU a través de 38 TI.
- La implementación de las TI hizo posible el cumplimiento del plan de entrega de las HU.
- La realización de pruebas unitarias y de aceptación permitió comprobar que las funcionalidades fueron implementadas correctamente y que el sistema satisface las necesidades del cliente.

Conclusiones

La investigación ha culminado en el desarrollo de una solución innovadora orientada a optimizar la gestión de proyectos de inversión en el INRH. La aplicación exitosa de la metodología XP que ha demostrado su eficacia al posibilitar entregas iterativas de software funcional, adaptándose de manera ágil a los requisitos cambiantes del cliente. La identificación de historias de usuario y la precisa estimación del tiempo han desempeñado un papel crucial en la planificación y ejecución efectiva del proyecto, como lo evidencian las pruebas unitarias y de aceptación que validaron la conformidad del sistema con las exigencias del cliente, así como la correcta implementación de sus funcionalidades.

La integración de estos módulos en el sistema proporcionará al INRH y a sus especialistas la capacidad de realizar de manera remota, confiable y continua la planificación, ejecución y control de los servicios científicos técnicos. El desafío de implementar servicios informáticos con el propósito de mejorar la calidad de los procesos en las empresas se sitúa en el corazón del campo informático. De esta manera, la transformación de la información, actualmente accesible solo de forma física, mediante la aplicación de las nuevas TIC, constituye un pilar estratégico del programa de informatización, representando así otro avance significativo en la informatización de la sociedad respaldado por el país.

Recomendaciones

Estas recomendaciones ayudarán a garantizar que los módulos de planificación y ejecución de servicios científicos técnicos del INRH sean efectivos y se mantengan alineados con los requisitos cambiantes a lo largo del tiempo:

1. Capacitación continua:

- Se recomienda proporcionar capacitación continua a los usuarios y al personal encargado de administrar el sistema. Esto garantizará que estén al tanto de las últimas características y funcionalidades, maximizando así la eficiencia y la productividad del sistema a medida que evoluciona.

2. Mantenimiento y actualización:

- Es esencial establecer un programa de mantenimiento y actualización regular para el sistema. Esto incluye la corrección de errores, la optimización de rendimiento y la incorporación de nuevas características y mejoras. Mantener el sistema actualizado asegura su longevidad y eficacia continua a lo largo del tiempo.

3. Integración con otros sistemas:

- Si es relevante, se debe considerar la integración con otros sistemas utilizados en el instituto. Esto puede mejorar la eficiencia y la coherencia de los procesos al evitar la duplicación de datos y la entrada manual. La interoperabilidad con sistemas existentes fortalecerá la eficacia global.

4. Escalabilidad:

- El sistema debe estar diseñado teniendo en cuenta la posibilidad de crecimiento en el número de servicios. Asegurarse de que el sistema sea escalable facilitará la expansión futura sin problemas, adaptándose a las necesidades cambiantes del INRH.

5. Evaluación periódica:

- Realizar evaluaciones periódicas del sistema, sus objetivos y su alineación con las necesidades del INRH. Esto permitirá tomar medidas proactivas para garantizar que el sistema siga siendo relevante y eficaz. La evaluación constante es clave para adaptarse a cambios en los requisitos y entorno operativo.

Estas prácticas sugeridas contribuirán a mantener la efectividad y la relevancia a largo plazo de los módulos de planificación y ejecución de servicios científicos técnicos en el INRH.

Referencias bibliográficas

Angel Arias, Alicia Durango. 2016. *Ingeniería y arquitectura de software 2da Edición*. s.l. : It Campus Academy , 2016. ISBN.

Apache, Fundación de software. 2023. JMeter. *JMeter*. [Online] Fundación de software Apache , 2023. [Citado em: 1 de octubre de 2023.] jmeter.apache.org. ISBN.

Arévalo, Julio Alonso. 2007. *Gestión de la Información, gestión de contenidos y conocimientos*. Salamanca : Universidad de Salamanca, 2007.

Barry J. Holmes, Daniel T. Joyce. 2001. *Object-oriented programming with Java. Sudbury.2da edición*. Massaguetts : Jones and Bartlett Publishers international, 2001. ISBN.

BECK, K. 2000. *Extreme Programming Explained*. s.l. : Pearson Education, 2000. ISBN.

BECK, K. 1999. *Extreme Programming Explained. Embrace Change , "Una explicación de la programación extrema. Aceptar el cambio"*. s.l. : Pearson Education., 1999. ISBN.

Betancourt, Dayana Rodríguez. 2020. *Sistema de planificación, aprobación, ejecución y control del presupuesto INRH*. s.l. : Xetid, 2020.

BOOTELLO, R. 2008. . *Sistemas de Información*. s.l. : MBA Edición,, 2008. ISBN.

Businessmap. 2023. Businessmap. *businessmap*. [Online] 2023. [Citado em: 16 de Noviembre de 2023.] <https://businessmap.io>. ISBN.

Calero, Mannuel. 2003. *Una explicación de la programación extrema (XP).. V Encuentro usuarios xBase*. Madrid : s.n., 2003. ISBN.

CANÓS, J. H. e LETELIER, P. y PENADÉS, M. C. 2011. *Métodologías Ágiles en el Desarrollo de Software*. ESPAÑA : Universidad Politécnica de Valencia. , 2011. ISBN.

Carmona, Juan García. 2012. *SOLID y GRASP buenas practicas hacia el éxito en el desarrollo del software*. Sevilla : s.n., 2012. ISBN.

Cillero, Manuel. 2023. manuel.cillero. *manuel.cillero*. [Online] 2023. [Citado em: 2023 de octubre de 2023.] manuel.cillero.es. ISBN.

colaboradores, TortoiseGit y. 2023. TortoiseGit. *TortoiseGit*. [Online] 2023. [Citado em: 1 de octubre de 2023.] tortoisegit.org. ISBN.

Dayana Benitez Betancourt, Fernando Sera Griñán. 2022. *Proyecto técnico del software Sistema de planificación, aprobación, ejecución y control del presupuesto INRH*. La Habana , Cuba : s.n., 2022.

Desarrollo del componente Análisis del Sistema de Planificación de Producciones. **Rodríguez, Alejandro Hernández. 2018.** La Habana : s.n., 2018. ISBN.

EL ESTADO DEL ARTE: UNA METODOLOGÍA DE INVESTIGACIÓN. **Gómez Vargas, Maricelly, Galeano Higueta, Catalina e Jaramillo Muñoz, Dumar Andrey. 2015.** 2, Colombia : Fundación Universitaria Luis Amigó, 2015, Vol. 6. E-ISSN: 2216-1201.

Elexey, Hernández Lugones. 2013. *Los sistemas de información y su implicación para Cuba*. La Habana : s.n., 2013.

Elizabeth Salazar Laffita, Fabián Guerra Padrón. 2013. repositorio.uci.cu. *Repositorio*. [Online] UCI, Junio de 2013. [Citado em: 16 de Noviembre de 2023.] <https://repositorio.uci.cu/jspui/handle/ident/8558>. ISBN.

Emery, James. C. 1990. *Sistemas de información. para la dirección. El recurso estratégico crítico*. s.l. : Ediciones Diaz de Santos, 1990. ISBN 42636440.

Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. 2002. *Design Patterns: Elements of Reusable Object*. 2002. ISBN.

—. **1995.** *Patrones de Diseño Elementos de software orientado a objetos reutilizable*. 1995. ISBN.

Espinosa, Cecilia Esther Hernández. 2011. *Módulos de Administración y Gestión de Información del SIG INRH*. La Habana : s.n., 2011.

Francisco Javier Cruz Castañón, David Borge Díez, Manuel Castro Gil y Antonio Colmenar Santos. 2014. *Gestión de proyectos con Microsoft Project 2013*. Madrid, España : RA-MA, 2014. ISBN.

García, Laura Ameijide. 2016. *Gestión de proyectos según el PMI*. Barcelona : Universitat Oberta de Catalunya, 2016. ISBN.

Gauchat, J.D. 2017. *El gran libro de HTML5, CSS3 y JavaScript*. Barcelona : Mink Books, 2017. ISBN.

Gestión de la información y el conocimiento. **MsC. María Josefina Vidal Ledo, MsC. Ana Bárbara Araña Pérez. 2012.** La Habana, Cuba : Escuela Nacional de Salud Pública, 2012. ISBN.

Glosario internacional de hidrología.], Organización Meteorológica Mundial [1002. 2012. s.l. : UNESCO, 2012. ISBN.

Group, PHP. 2023. PHP Group. *PHP*. [Online] 2023. [Citado em: 1 de octubre de 2023.] <https://www.php.net/manual/es/history.php.php>. ISBN.

HAMPTON, D.R. 1983. *“Administración Contemporanea”*. México : Mc Gaw-Hill, 1983. ISBN.

Haverbeke, Marijn. 2018. *Eloquent JavaScript*. 2018. ISBN.

Hermoso, José Joaquín López, et al. 2000. *Informática Aplicada a la Gestión de Empresas* . Madrid : ESIC EDITORIAL, 2000. ISBN.

HOLMES, B. y T. JOYCE, D. 2000. *Object-oriented programming with Java. Sudbury*. London , New york : s.n., 2000. ISBN.

HUMBERTO, BARCIA PÉREZ EDWIN. 2003. *DESARROLLO DEL MÓDULO DE INTELIGENCIA DE NEGOCIOS PARA EL SISTEMA DE GESTION DE EMBARQUES*. España : s.n., 2003. ISSN.

2000. *Informática Aplicada a la Gestión de Empresas* . [A. do livro] José joaquin López, et al. *Informática Aplicada a la Gestión de Empresas* . Madrid : ESIC EDITORIAL, 2000.

Informatización y dirección de empresas en Cuba: evolución y desafíos. **Encinosa, Lázaro J. Blanco. 2017.** 1, La Habana : Universidad de La Habana, 2017, Vol. 11. ISBN.

Instituto Universitario de Investigación de Ingeniería del Agua y Medio Ambiente • Ciudad Politécnica de la Innovación - Edificio 8G Acceso D, 5ª planta • Camino de Vera S/N. 2023.

<https://www.iiama.upv.es.iiama>. [Online] 2023. [Citado em: 1 de octubre de 2023.]
<https://www.iiama.upv.es/iiama/es/sala-prensa/noticias/nueva-version-del-software-para-la-planificacion-y-gestion-de-recursos-hidricos-aquatoool.html>. ISBN.

Jaramillo, Nando. 2014. *prezi. prezí*. [Online] 4 de octubre de 2014. [Citado em: 19 de noviembre de 2023.] <https://prezi.com>. ISBN.

José H. Canós, Patricio Letelier y M^a Carmen Penadés. *Métodologías Ágiles en el Desarrollo de Software*. Valencia : Universidad Politécnica de Valencia. ISBN.

Kenneth E. Kendall, Julie E. Kendall. 2005. *Análisis y diseño de sistemas 6ta edición*. México : PEARSON EDUCACIÓN, 2005. ISBN.

La gestión de la información como base a la gestión del conocimiento y del aprendizaje en las organizaciones en las universidades. . **Moreno González, Nancy and Rodríguez Gonzalez, Fermín Orestes. 2002.** n., La Habana : "Educación Superior (Universidad de la Habana), 2002, Vol. vol.2.

Ledo, MsC. María Josefina Vidal e Pérez, MsC. Ana Bárbara Araña. 2012. Gestión de la información y el conocimiento. *Mi SciELO*. ISSN 0864-2141, 2012, Vol. vol.26, no.3.

Lévy, Pierre. 1999. *Qu'est-ce que le virtual?* París : Éditions de la Découverte, 1999. ISBN: 84-493-0585-3.

LOS SISTEMAS DE INFORMACIÓN: EVOLUCIÓN Y DESARROLLO . [A. do livro] Alejandro HERNANDEZ TRASOBARES. *LOS SISTEMAS DE INFORMACIÓN: EVOLUCIÓN Y DESARROLLO* . Universidad de Zaragoza : Departamento de Economía y Dirección de Empresas.

Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP). **Patricio Letelier, M^a Carmen Penadés. 2006.** 26, Buenos Aires : Departamento de Sistemas Informáticos y Computación (DSIC), 2006, Vol. 05. ISSN 1666-1680.

Momjian, Bruce. 2001. *PostgreSQL Introduction and Concepts*. s.l. : ADDISON–WESLEY, 2001. ISBN 0-201-70331-9.

Mora, Sergio Luján. 2001. *Programación en Internet: Clientes Web*. Alicante España : Editorial Club Universitario, 2001. ISBN.

Moreno González, Nancy and Rodríguez Gonzalez, Fermín Orestes. 2002. La gestión de la información como base a la gestión del conocimiento y del aprendizaje en las organizaciones en las universidades. . *Educación Superior (Universidad de la Habana)*. 2002, Vol. vol.2, no.2.

pgAdmin. 2023. *pgAdmin. pgAdmin*. [Online] 2023. [Citado em: 1 de octubre de 2023.] www.pgadmin.org. ISBN.

Pressman, R. S. 2014. *Software Engineering: A Practitioner's Approach*. s.l. : McGraw-Hill Education, 2014. ISBN.

Reyes, Lic. Arturo Roelvis Fernández. 2017. ¿QUE ES GESTIÓN DE LA INFORMACIÓN? *infomed instituciones* . [Online] Centro de Toxicología y Biomedicina de Santiago de Cuba Universidad de Ciencias Médicas., 16 de abril de 2017. [Citado em: 14 de junio de 2023.] <https://instituciones.sld.cu/toximed/2017/04/16/que-es-gestion-de-la-informacion/>. ISBN/ISSN.

Rivero, Eblis Manuel Caro. 2015. *Sistema para la gestión de información del Departamento de Reproducción de Documentos de la Universidad de las Ciencias Informáticas (DRDuci).* LA HABANA CUBA : UCI, 2015. ISBN.

s.r.o., JetBrains. 2000-2023. JetBrains. *JetBrains*. [Online] 2000-2023. [Citado em: 1 de octubre de 2023.] www.jetbrains.com/es-es/phpstorm/. ISBN.

Salas, Karla Rodríguez. 2002. *GESTIÓN DE LA INFORMACIÓN EN LAS ORGANIZACIONES.* s.l. : Bibliotecas Vol. XX, No.1y No. 2., 2002.

Serrano, Francisco Gerardo. 2020. *Proyectos de inversión* . ciudad de México : Grupo Editorial Patria S.A, 2020. ISBN.

SINTYA MILENA MELÉNDEZ VALLADAREZ, MARIA ELIZABETH GAITAN , NELDIN NOEL PÉREZ REYES. 28 de enero del 2016. *METODOLOGIA ÁGIL DE DESARROLLO DE SOFTWARE PROGRAMACION Extrema.* Nicaragua : RECINTO UNIVERSITARIO RUBÉN DARÍO (RURD), 28 de enero del 2016. ISBN.

SINTYA MILENA MELÉNDEZ VALLADAREZ, MARIA ELIZABETH GAITAN , NELDIN NOEL PÉREZ REYES. 2016. *METODOLOGIA ÁGIL DE DESARROLLO DE SOFTWARE PROGRAMACION Extrema.* Nicaragua : RECINTO UNIVERSITARIO RUBÉN DARÍO, 2016. ISBN.

Sommerville, Ian. 2005. *Ingeniería de software 7ma edición.* MADRID España : Pearson Education S.A, 2005. ISBN.

TRASOBARES, Alejandro HERNANDEZ. LOS SISTEMAS DE INFORMACIÓN: EVOLUCIÓN Y Desarrollo. Universidad de Zaragoza : Departamento de Economía y Dirección de Empresas.

UNIFIED MODELING LANGUAGE (UML) OVERVIEW. **Padmanabhan, Bharath. 2012.** 2012. ISBN.

Valladarez. 2016. *SINTYA MILENA MELÉNDEZ VALLADAREZ, MARIA ELMETODOLOGIA ÁGIL DE DESARROLLO DE SOFTWARE PROGRAMACION Extrema.* Nicaragua : RECINTO UNIVERSITARIO RUBÉN DARÍO, 2016. ISBN.

VALLADAREZ, SINTYA MILENA MELÉNDEZ. 2016. *Metodología ágil de desarrollo de software programación extrema* . Nicaragua : Universidad Rubén Darío, 2016. ISBN.

Vázquez, Anaely Saunders. 2012. De los Sistemas de Gestión de Información a los Sistemas de Gestión Estratégica de Información. *Revista española de Documentación Científica* . junio 2012, 2012, Vol. 1, 1.

Villegas, Adrián Anaya. 2018. *A propósito de programación extrema XP (eXtreme Programming).* Cauca : s.n., 2018. ISBN.

visual-paradigm. 2023. visual-paradigm. *visual-paradigm*. [Online] 2023. [Citado em: 1 de Octubre de 2023.] www.visual-paradigm.com. ISBN.

visualstudio. 2023. visualstudio. *visualstudio*. [Online] 2023. [Citado em: 1 de octubre de 2023.] code.visualstudio.com. ISBN.

WALLACE, D. e RAGGETT, I. y AUFANGANG, J. 2002. *Extreme Programming for Web Projects.* Boston : Addison Wesley, 2002. ISBN.

Wells, Donovan. Last modified October 8, 2013.. *Extreme Programming: A gentle introduction.* Estados Unidos : All Rights reserved., Last modified October 8, 2013. ISBN.

—. **2013.** *Extreme Programming: A gentle introduction*. Estados Unidos : s.n., 2013. ISBN.

XETID. 2023. hidro. *hidro*. [Online] XETID, 2023. [Citado em: 1 de Octubre de 2023.]
www.hidro.gob.cu. ISBN.

YeePLY. 2021. YeePLY. *YeePLY*. [Online] 2021. [Citado em: 29 de Octubre de 2023.]
www.yeePLY.com. ISBN.

Zend Framework in Action. **Rob Allen, Nick Lo, Steven Brown. 2009.** United States of America :
Oldham country, 2009. ISBN.

Anexos

Anexo 1 Acta de aceptación del proyecto

La imagen a continuación evidencia como fue aceptada por parte del cliente la propuesta de solución planeada:

Acta de aceptación

En la Habana, a los 19 días del mes de Octubre del año 2023

DE UNA PARTE: La Empresa de Tecnologías de la Información para la Defensa, en forma abreviada XETID representada en acto por **Fernando Sera Griñán** en su carácter de **Jefe de proyecto** con facultades para firmar la presente Acta de aceptación, quien en lo sucesivo se denominará **EJECUTORA**.

DE OTRA PARTE: INSTITUTO NACIONAL DE RECURSOS HIDRÁULICOS, representada en acto por **Rodobaldo José Rodríguez** en su carácter de **Jefe de proyecto** con facultades para firmar la presente Acta de aceptación, quien en lo sucesivo se denominará **CLIENTE**.

Primero: Que en cumplimiento del Contrato No. 2037-N-CV-22 con nombre **Desarrollo del Sistema de Planificación, aprobación, ejecución y control del presupuesto para la demanda y contratación de servicios científicos técnicos para el Instituto Nacional de Recursos Hidráulicos**; han sido efectuadas las actividades que se describen, Las Partes DECLARAN:

CONSIDERANDO: Que se han efectuado las actividades siguientes:

Hito 7: A la aceptación de los siguientes trabajos:

- Componente Ejecución
 - Desarrollo de componente Ejecución de Servicio Científico Técnico e Innovación.

Hito 8: A la aceptación de los siguientes trabajos:

- Desarrollo de componente Reportes.

Hito 9: A la aceptación de los siguientes trabajos:

- Desarrollo de componente Gráfico Analítico.

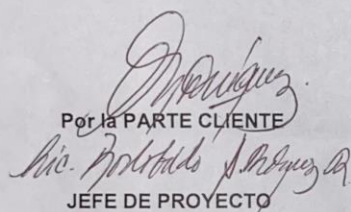
CONSIDERANDO: Que las actividades realizadas han sido desarrolladas con la calidad requerida y bajo las condiciones pactadas y aprobadas por Las Partes.

CONSIDERANDO: Que las actividades que se han ejecutado cumplen con los requerimientos de La Parte CLIENTE.

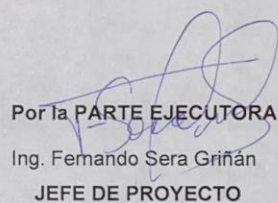
CONSIDERANDO: Que La Parte EJECUTORA ha entregado la documentación que avala la ejecución de este acto a La Parte CLIENTE.

POR TANTO: Las Partes acuerdan formalizar mediante la presente Acta, Aceptadas las actividades que han sido ejecutadas en esta fecha.

Y para que así conste, se extiende la presente Acta en tres (2) ejemplares, rubricados por Las Partes.



Por la PARTE CLIENTE
Lic. Rolando J. Rodríguez
JEFE DE PROYECTO



Por la PARTE EJECUTORA
Ing. Fernando Sera Griñán
JEFE DE PROYECTO

Ilustración 10 Acta de aceptación del proyecto por parte del cliente

Anexo 2 Tareas de ingeniería

Tabla 34 Tarea de ingeniería # 6

Tarea TI6HU3	
Número de la tarea: 6	Número de la UH: 3
Nombre de la tarea: Desglose mensual del presupuesto	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 26/06/23	Fecha fin: 28/06/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: El usuario tiene la opción de observar después de agregado el servicio, el desglose mensual del presupuesto propuesto. Esta acción le debe permitir desglosar un grupo de objetivos seleccionados en el sistema. Se mostrará deshabilitado por defecto, solo se habilitará si se selecciona un objetivo de los listados en el sistema. También se podrá acceder a esta funcionalidad a través del clic derecho (secundario), sobre uno de los objetivos listados, de forma tal que pueda definir un grupo de objetivos y dentro de este especificar los objetivos que debe contener dicha agrupación. Nota: al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Desglosar objetivo"	
	

Tabla 35 Tarea de ingeniería # 7

Tarea TI7HU3	
Número de la tarea: 7	Número de la UH: 3
Nombre de la tarea: Revisado por director técnico de la delegación y por delegado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 28/06/23	Fecha fin: 30/06/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	

Descripción: En la interfaz se encuentra la opción de estar revisado por el director técnico de la delegación o delegado.

También se permitirá actualizar los datos listados en la interfaz, de forma tal que una vez invocada a la misma, el sistema recargue la interfaz y actualice cada uno de los datos mostrados en la misma. Esta acción solo mostrará cambios en caso que se encuentre a la espera de alguna respuesta.

Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Actualizar".

Tabla 36 Tarea de ingeniería # 8

Tarea TI8HU3	
Número de la tarea: 8	Número de la UH: 3
Nombre de la tarea: Aprobado por el DCTI- INRH	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 01/07/23	Fecha fin: 02/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En la interfaz se encuentra la opción de estar aprobado por el DCTI-INRH. También se permitirá actualizar los datos listados en la interfaz, de forma tal que una vez invocada a la misma, el sistema recargue la interfaz y actualice cada uno de los datos mostrados en la misma. Esta acción solo mostrará cambios en caso que se encuentre a la espera de alguna respuesta. Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Actualizar".	

Tabla 37 Tarea de ingeniería # 9

Tarea TI9HU3	
Número de la tarea: 9	Número de la UH: 3
Nombre de la tarea: Cancelar por Director técnico de la delegación y por delegado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 02/07/23	Fecha fin: 04/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	

Descripción: En la interfaz se encuentra la opción de cancelar el servicio por el técnico de la delegación y por el delegado. También se permitirá actualizar los datos listados en la interfaz, de forma tal que una vez invocada a la misma, el sistema recargue la interfaz y actualice cada uno de los datos mostrados en la misma. Esta acción solo mostrará cambios en caso que se encuentre a la espera de alguna respuesta.

Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Actualizar".

Tabla 38 Tarea de ingeniería # 10

Tarea TI10HU3	
Número de la tarea: 10	Número de la UH: 3
Nombre de la tarea: Desaprobado por DCTI-INRH	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 04/07/23	Fecha fin: 06/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: en la interfaz se encuentra la opción de desaprobar por el DCTI-INRH. También se permitirá actualizar los datos listados en la interfaz, de forma tal que una vez invocada a la misma, el sistema recargue la interfaz y actualice cada uno de los datos mostrados en la misma. Esta acción solo mostrará cambios en caso que se encuentre a la espera de alguna respuesta. Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Actualizar".	

Tabla 39 Tarea de ingeniería # 11

Tarea TI11HU4	
Número de la tarea: 11	Número de la UH: 4
Nombre de la tarea: Adicionar observación	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 06/07/23	Fecha fin: 08/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En esta ventana el usuario puede adicionar una determinada observación. En el campo de asunto se dispone a poner el asunto de la observación y en el campo de la descripción se describe dicha observación.	

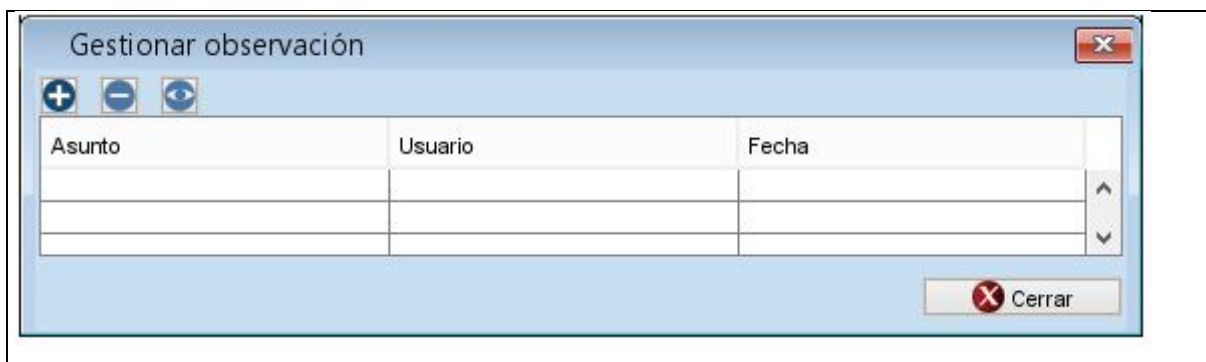


Tabla 40 Tarea de ingeniería # 12

Tarea TI12HU4	
Número de la tarea: 12	Número de la UH: 4
Nombre de la tarea: Eliminar observación	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 08/07/23	Fecha fin: 10/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En la ventana de gestionar observación se dispone de un botón (esquina superior izquierda) capaz de eliminar una determinada observación, ya hecha, en el listado de observaciones previamente elaboradas.	

Tabla 41 Tarea de ingeniería # 13

Tarea TI13HU4	
Número de la tarea: 13	Número de la UH: 4
Nombre de la tarea: Ver observación	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 10/07/23	Fecha fin: 12/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En la ventana de gestionar observación se dispone de un botón (esquina superior izquierda) capaz de ver, sin poder editar, una determinada observación, ya hecha, en el listado de observaciones previamente elaboradas.	

Tabla 42 Tarea de ingeniería # 14

Tarea TI14HU4	
Número de la tarea: 14	Número de la UH: 4
Nombre de la tarea: Adjuntar documento	

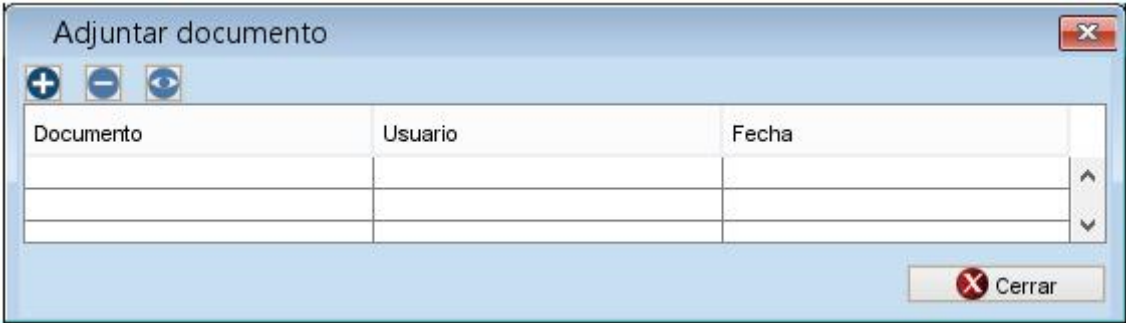
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 12/07/23	Fecha fin: 14/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En el caso de que el usuario necesite adjuntar algún documento podrá ir a la opción y tendrá la opción de adjuntar cualquier documento que se encuentre a disposición, con la opción de un botón en la esquina superior izquierda.	
	

Tabla 43 Tarea de ingeniería # 15

Tarea TI15HU5	
Número de la tarea: 15	Número de la UH: 5
Nombre de la tarea: Generar reporte ficha	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 14/07/23	Fecha fin: 16/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: El usuario podrá adjuntar el logo de la ficha en configurar encabezado y pie de firma, esta opción tiene la posibilidad de ser cancelado o aceptado después de ser previamente subido el logo (sección de encabezado). El usuario deberá añadir un pie de firma donde rellena los campos de elaborado por, revisado por y aprobado por (sección pie de firma).	

Descripción: El usuario podrá adjuntar el logo de la ficha en configurar encabezado y pie de firma, esta opción tiene la posibilidad de ser cancelado o aceptado después de ser previamente subido el logo (sección de encabezado).

El usuario deberá añadir un pie de firma donde rellena los campos de elaborado por, revisado por y aprobado por (sección pie de firma).

Tabla 45 Tarea de ingeniería # 17

Tarea TI17HU5	
Número de la tarea:17	Número de la UH: 5
Nombre de la tarea: Buscar servicio	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 18/07/23	Fecha fin: 20/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: El usuario en el campo vacío de búsqueda, busca el servicio antes ya registrado. Esta acción le debe permitir realizar un filtrado en el árbol, una vez insertado un criterio de búsqueda en el campo de texto. La búsqueda se efectuará en dependencia de los servicios que se encuentren en ese momento. En este caso la búsqueda se efectuaría sobre el árbol de servicios. Esta acción le debe permitir buscar un objetivo del sistema. Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Búsqueda avanzada de objetivo(s)".	

Tabla 46 Tarea de ingeniería # 18

Tarea TI8HU5	
Número de la tarea: 18	Número de la UH: 5
Nombre de la tarea: Consultar ayuda	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 20/07/23	Fecha fin: 22/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En la ventana el usuario podrá disponer de consultar ayuda para trabajar con el sistema. Botón que se encuentra en la esquina superior derecha con símbolo de pregunta. También se puede mostrar ayuda detallada de la funcionalidad. Se puede acceder de forma rápida a esta funcionalidad con la combinación de teclas (Alt +H).	

Tabla 47 Tarea de ingeniería # 19

Tarea TI19HU5	
Número de la tarea:19	Número de la UH:5
Nombre de la tarea: Listar ejercicio	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 22/07/23	Fecha fin: 24/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En la ventana el usuario podrá disponer de consultar el año en el que desea trabajar con el sistema. Botón que se encuentra en la esquina superior derecha.	

Tabla 48 Tarea de ingeniería # 20

Tarea TI20HU5	
Número de la tarea: 20	Número de la UH: 5
Nombre de la tarea: Calcular total del plan insertado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 24/07/23	Fecha fin: 26/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: La suma del total de los planes insertados se muestran en la parte inferior izquierda.	

Tabla 49 Tarea de ingeniería # 21

Tarea TI21HU5	
Número de la tarea:21	Número de la UH: 5
Nombre de la tarea: Calcular total del plan aprobado por el nivel Central	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 26/07/23	Fecha fin: 29/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: La suma del total de los planes insertados se muestran en la parte inferior izquierda.	

Tabla 50 Tarea de ingeniería # 22

Tarea TI22HU6	
Número de la tarea: 22	Número de la UH: 6
Nombre de la tarea: Adicionar servicio antes planificado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 29/07/23	Fecha fin: 31/07/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: El usuario incluso después de haber planificado, puede, en el proceso de ejecución, adicionar más servicios, con el presupuesto sobrante, el cual igual requiere de un proceso de aprobación. El usuario ingresa los datos de la entidad ejecutora con nombre, representante, cargo, teléfono, correo, fax, dirección, ciudad y provincia. Aquí se especifican los datos del proyecto o trabajo, se rellenan los campos de título del proyecto, presupuesto requerido, fecha de inicio y fecha de fin. Se selecciona la prioridad de la política nacional del agua que tributa el trabajo, con las siguientes opciones (se pueden seleccionar más de una): uso racional y productivo del agua, uso eficiente de la infraestructura hidráulica, gestión de riesgos a la calidad del agua y gestión de eventos asociados a eventos extremos del clima. Se debe seleccionar en un campo aparte los problemas a resolver del banco de problemas (se pueden seleccionar todos o ninguno).	

Tabla 51 Tarea de ingeniería # 23

Tarea TI23HU6	
Número de la tarea: 23	Número de la UH: 6
Nombre de la tarea: Modificar servicio antes planificado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.4
Fecha de inicio: 31/07/23	Fecha fin: 2/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: El usuario tiene la opción de modificar el servicio antes adicionado en una opción de la interfaz en la esquina superior izquierda.	

Tabla 52 Tarea de ingeniería # 24

Tarea TI24HU6

Número de la tarea: 24	Número de la UH: 6
Nombre de la tarea: Eliminar servicio antes planificado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 2/08/23	Fecha fin: 3/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: El usuario tiene la opción de eliminar el servicio antes adicionado tanto en el proceso de planificación como ejecución, en un botón de la interfaz que se encuentra en la esquina superior izquierda.	

Tabla 53 Tarea de ingeniería # 25

Tarea TI25HU6	
Número de la tarea: 25	Número de la UH: 6
Nombre de la tarea: Revisar servicio por director técnico de la delegación y delegado antes planificado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 3/08/23	Fecha fin: 4/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En la interfaz se encuentra la opción de estar revisado por el director técnico de la delegación o delegado. También se permitirá actualizar los datos listados en la interfaz, de forma tal que una vez invocada a la misma, el sistema recargue la interfaz y actualice cada uno de los datos mostrados en la misma. Esta acción solo mostrará cambios en caso que se encuentre a la espera de alguna respuesta. Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Actualizar".	

Tabla 54 Tarea de ingeniería # 26

Tarea TI26HU6	
Número de la tarea: 26	Número de la UH: 6
Nombre de la tarea: Aprobar servicio por DCTI-INRH antes planificado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 4/08/23	Fecha fin: 5/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	

Descripción: En la interfaz se encuentra la opción de estar aprobado por el DCTI-INRH También se permitirá actualizar los datos listados en la interfaz, de forma tal que una vez invocada a la misma, el sistema recargue la interfaz y actualice cada uno de los datos mostrados en la misma. Esta acción solo mostrará cambios en caso que se encuentre a la espera de alguna respuesta. Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Actualizar".

Tabla 55 Tarea de ingeniería # 27

Tarea TI27HU6	
Número de la tarea: 27	Número de la UH: 6
Nombre de la tarea: Cancelar servicio por director técnico de la delegación y delegado antes planificado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 5/08/23	Fecha fin: 6/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En la interfaz se encuentra la opción de cancelar el servicio por el director técnico de la delegación y delegado. También se permitirá actualizar los datos listados en la interfaz, de forma tal que una vez invocada a la misma, el sistema recargue la interfaz y actualice cada uno de los datos mostrados en la misma. Esta acción solo mostrará cambios en caso que se encuentre a la espera de alguna respuesta. Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Actualizar".	

Tabla 56 Tarea de ingeniería # 28

Tarea TI28HU6	
Número de la tarea: 28	Número de la UH: 6
Nombre de la tarea: Desaprobar servicio por DCTI-INRH antes planificado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 06/08/23	Fecha fin: 07/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	

Descripción: En la interfaz se encuentra la opción de desaprobar por el DCTI-INRH. También se permitirá actualizar los datos listados en la interfaz, de forma tal que una vez invocada a la misma, el sistema recargue la interfaz y actualice cada uno de los datos mostrados en la misma. Esta acción solo mostrará cambios en caso que se encuentre a la espera de alguna respuesta. Al pasar el cursor por encima de este campo se debe mostrar el siguiente texto en el tooltip: "Actualizar".
--

Tabla 57 Tarea de ingeniería # 29

Tarea TI29HU7	
Número de la tarea: 29	Número de la UH: 7
Nombre de la tarea: Desglosar presupuesto mensual	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 07/08/23	Fecha fin: 08/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En este módulo la opción de desglosar el presupuesto por mes es diferente en este caso el usuario selecciona el mes a desglosar y se observa el importe el número de cada factura utilizada la fecha de emisión y recepción si existe o no documento de factura y el consecutivo autorizo de pago.	

Tabla 58 Tarea de ingeniería # 30

Tarea TI30HU8	
Número de la tarea: 30	Número de la UH: 8
Nombre de la tarea: Insertar fecha de facturación	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 08/08/23	Fecha fin: 09/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Este debe ser un campo obligatorio, por lo que no es opcional la especificación de datos en el mismo, de dejar el campo vacío, el sistema deberá marcar el campo en rojo y mostrar el mensaje de error: "Este campo es obligatorio.", al pasar el cursor del mouse por encima del mismo. El formato de la fecha debe ser: dd/mm/yyyy, de establecer cualquier otro tipo de valor, el sistema deberá marcar el campo en rojo y mostrar el mensaje de error: "El formato establecido: dd/mm/yyyy. ", al pasar el cursor del mouse por encima del campo. Es importante tener en cuenta que la fecha de fin no debe ser posterior a la fecha de inicio, por lo que, de ocurrir, el sistema deberá marcar el campo en rojo y mostrar el mensaje de	

error: "La fecha de inicio debe ser anterior o igual a la fecha de fin.", al pasar el cursor por encima del campo.

Tabla 59 Tarea de ingeniería # 31

Tarea TI32HU8	
Número de la tarea: 31	Número de la UH: 8
Nombre de la tarea: Generar documento de factura	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 09/08/23	Fecha fin: 10/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: Al generar este documento, se deberá mostrar en la parte superior del mismo la fecha del día en que se genera el modelo. En el pie de firma del documento el campo de fecha se deberá quedar vacío para según se vaya firmando se coloque su fecha correspondiente.	

Tabla 60 Tarea de ingeniería # 32

Tarea TI32HU9	
Número de la tarea: 32	Número de la UH:9
Nombre de la tarea: Adicionar observación	
Tipo de tarea: Desarrollo	Puntos estimados:0.2
Fecha de inicio: 10/08/23	Fecha fin: 11/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En este módulo después de haber pasado el proceso de planificación se pueden ver las observaciones y agregar observaciones en los botones que se encuentran en la esquina superior izquierda del panel.	

Tabla 61 Tarea de ingeniería # 33

Tarea TI33HU9	
Número de la tarea: 33	Número de la UH: 9
Nombre de la tarea: Eliminar observación	
Tipo de tarea: Desarrollo	Puntos estimados:0.2
Fecha de inicio: 11/08/23	Fecha fin: 12/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	

Descripción: En este módulo después de haber pasado el proceso de planificación se puede, agregar observaciones y eliminar observaciones en los botones que se encuentran en la esquina superior izquierda del panel.

Tabla 62 Tarea de ingeniería # 34

Tarea TI34HU9	
Número de la tarea:34	Número de la UH: 9
Nombre de la tarea: Ver observación	
Tipo de tarea: Desarrollo	Puntos estimados:0.2
Fecha de inicio: 12/08/23	Fecha fin: 13/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En este módulo (Ejecución), después de haber pasado el proceso de planificación se pueden ver las observaciones, agregar observaciones y eliminar en los botones que se encuentran en la esquina superior izquierda del panel.	

Tabla 63 Tarea de ingeniería # 35

Tarea TI35HU9	
Número de la tarea: 35	Número de la UH: 9
Nombre de la tarea: Adjuntar documento	
Tipo de tarea: Desarrollo	Puntos estimados:0.2
Fecha de inicio: 12/08/23	Fecha fin: 13/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: En el caso de que el usuario necesite adjuntar algún documento podrá ir a la opción y tendrá la opción de adjuntar cualquier documento que se encuentre a disposición, con la opción de un botón en la esquina superior izquierda.	

Tabla 64 Tarea de ingeniería # 36

Tarea TI36HU10	
Número de la tarea: 36	Número de la UH: 10
Nombre de la tarea: Adicionar documento del proveedor	
Tipo de tarea: Desarrollo	Puntos estimados:0.2
Fecha de inicio: 13/08/23	Fecha fin: 14/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	

Descripción: En el caso de que el usuario necesite adjuntar algún documento del proveedor podrá ir a la opción y tendrá la opción de adjuntar cualquier documento que se encuentre a disposición, con la opción de un botón en la esquina superior izquierda.

Tabla 65 Tarea de ingeniería # 37

Tarea TI37HU11	
Número de la tarea: 37	Número de la UH: 11
Nombre de la tarea: Calcular total del plan presupuestado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 14/08/23	Fecha fin: 15/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: La suma del total de los planes insertados se muestran en la parte inferior izquierda.	

Tabla 66 Tarea de ingeniería # 38

Tarea TI38HU12	
Número de la tarea: 38	Número de la UH:12
Nombre de la tarea: Calcular total ejecutado	
Tipo de tarea: Desarrollo	Puntos estimados: 0.2
Fecha de inicio: 16/08/23	Fecha fin: 17/08/23
Programador responsable: Dasiel Torres Camejo y Thalia Caballero Álvarez	
Descripción: La suma del total de los planes insertados se muestran en la parte inferior izquierda.	

Anexo 3 Caso de prueba utilizando la técnica partición de equivalencia

Tabla 67 Caso de prueba para la funcionalidad adicionar servicio ()

Escenario	Denominación de entidad	Representante	Cargo	Correo	Móvil	Dirección	Ciudad	Provincia	Presupuesto CUP	Respuesta del sistema	Resultado
Introducir datos correctos	(V)GEOCUBA	(V)Lianet	(V)Jefe	(V)Lianet@gmail.com	(V)54313131	(V)calle 35	(V)San Luis	(V)Pinar del Río	(V)55000	SMMN	(S)
Introducir datos incorrectos	(F) 11*&%@ 32 +-	(V)Lianet	(V)Jefe	(V)Lianet@gmail.com	(V)54313131	(V)calle 35	(V)San Luis	(V)Pinar del Río	(V)55000	SMM*	(S)
	(V)GEOCUBA	(F)1263622	(V)Jefe	(V)Lianet@gmail.com	(V)54313131	(V)calle 35	(V)San Luis	(V)Pinar del Río	(V)55000	SMM#	(S)
	(V)GEOCUBA	(V)Lianet	(F)123	(V)Lianet@gmail.com	(V)54313131	(V)calle 35	(V)San Luis	(V)Pinar del Río	(V)55000	SMM#	(S)
	(V)GEOCUBA	(V)Lianet	(V)Jefe	(F)@@@	(V)54313131	(V)calle 35	(V)San Luis	(V)Pinar del Río	(V)55000	SMM@	(S)
	(V)GEOCUBA	(V)Lianet	(V)Jefe	(V)Lianet@gmail.com	(F)letra	(V)calle 35	(V)San Luis	(V)Pinar del Río	(V)55000	SMMletras	(S)
	(V)GEOCUBA	(V)Lianet	(V)Jefe	(V)Lianet@gmail.com	(V)54313131	(F)****	(V)San Luis	(V)Pinar del Río	(V)55000	SMM*	(S)
	(V)GEOCUBA	(V)Lianet	(V)Jefe	(V)Lianet@gmail.com	(V)54313131	(V)calle 35	(F)980	(V)Pinar del Río	(V)55000	SMM#	(S)
	(V)GEOCUBA	(V)Lianet	(V)Jefe	(V)Lianet@gmail.com	(V)54313131	(V)calle 35	(V)San Luis	(F)*+91	(V)55000	SMM*	(S)
	(V)GEOCUBA	(V)Lianet	(V)Jefe	(V)Lianet@gmail.com	(V)54313131	(V)calle 35	(V)San Luis	(V)Pinar del Río	(F)letras	SMMletras	(S)

Dejar todos los campos vacíos	(F)	(F)	(F)	(F)	(F)	(F)	(F)	(F)	(F)	SMM_	(S)
-------------------------------	-----	-----	-----	-----	-----	-----	-----	-----	-----	------	-----

Leyenda

SMM*: El sistema muestra un mensaje no debe contener caracteres especiales.

SMM#: El sistema muestra un mensaje no debe contener números.

SMMletras: El sistema muestra un mensaje no debe contener letras.

SMMN: El sistema muestra un mensaje se ha añadido un nuevo servicio.

SMM@: El sistema muestra un mensaje debe ser un correo válido.

SMM_: El sistema muestra un mensaje los campos son obligatorios.

Anexo 4 Modelo de datos

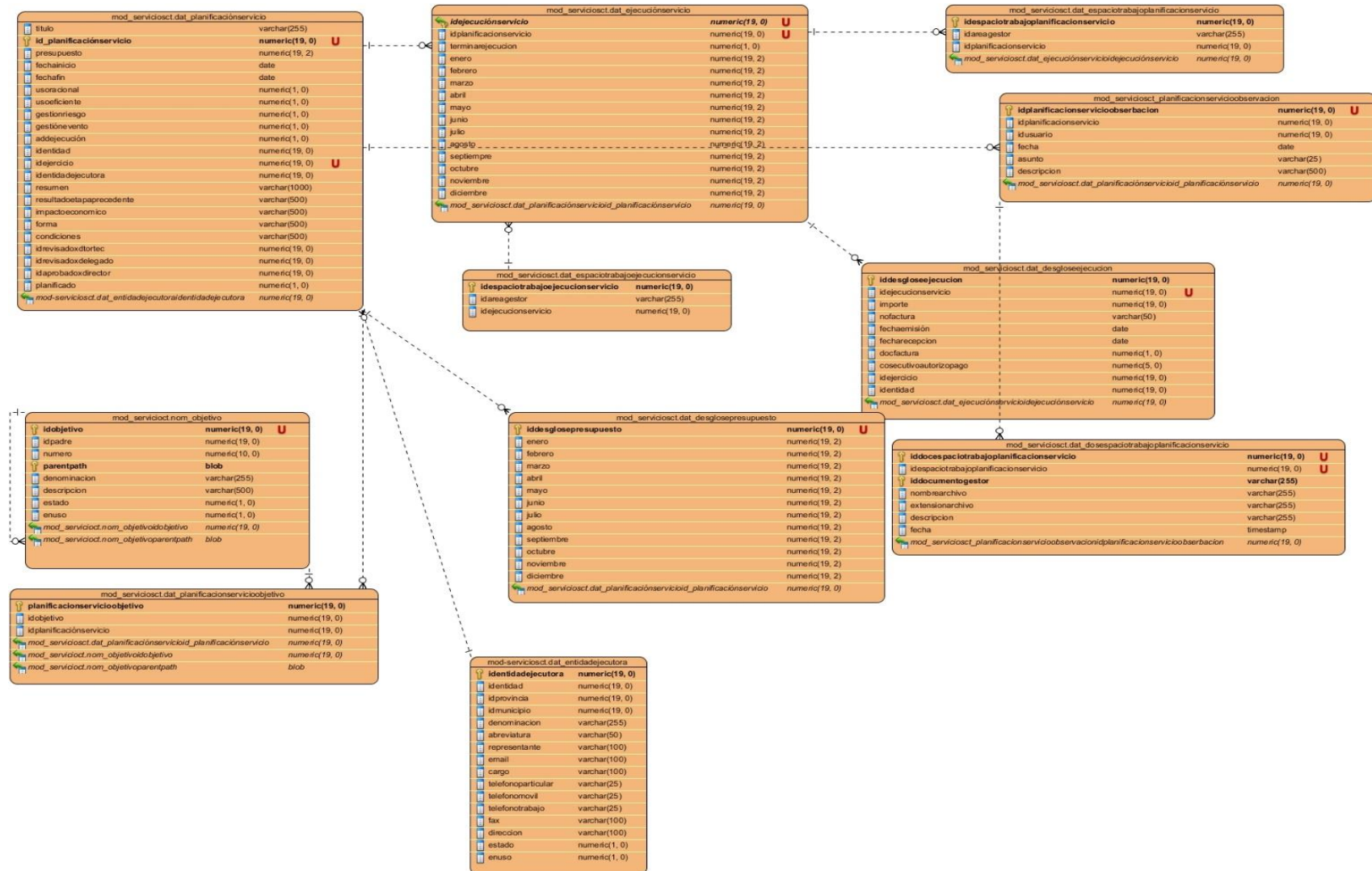


Ilustración 11 Modelo de datos