



Universidad de Cienfuegos “Carlos Rafael Rodríguez”

Facultad de Informática

Departamento de Informática

Sistema de gestión para Eventos de ciencia-técnica e innovación.

Trabajo de diploma para optar por el título de Ingeniería en Informática

Autor(es):

Frank David Cruz Enríquez

Tutor(es):

Lic. Bunny Prieto Santana

Dr. C. Eduardo R. Concepción Morales

Cienfuegos, Cuba

Curso 2023

Declaración de autoría

Declaro que soy el único autor (Declaramos que somos los únicos autores) del trabajo de diploma titulado "<Título del Trabajo de Diploma>", y autorizo (autorizamos) al <Nombre del Departamento> de (de la, del) <Nombre de la Entidad> y al Departamento de Informática de la Facultad de Informática en la Universidad de Cienfuegos "Carlos Rafael Rodríguez" para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo (firmamos) la presente a los 12 días del mes de Junio del 2021

(Si procede)

Nombre completo del primer autor

Nombre completo del segundo autor

(Si procede)

Nombre completo del primer tutor

Nombre completo del segundo tutor

OPINIÓN DEL USUARIO

El Trabajo de Diploma, titulado <Título>, fue realizado en nuestra entidad <Nombre completo de la entidad que utilizará el sistema>. Se considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

☐ Totalmente

☐ Parcialmente en un ____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes (cuantificar):

Como resultado de la implantación de este trabajo se reporta un efecto económico que asciende a <valor> MN y/o <valor> CUC. (Este valor debe ser REAL, no indica lo que se reportará, sino lo que reporta a la entidad. Puede desglosarse por conceptos, tales como: cuanto cuesta un software análogo en el mercado internacional, valor de los materiales que se ahorran por la existencia del software, valor anual del (de los) salario(s) equivalente al tiempo que se ahorra por la existencia del software).

Y para que así conste, se firma la presente a los ____ días del mes de _____ del año _____

Nombre del representante de la entidad

Cargo

Firma

Cuño

OPINIÓN DEL TUTOR

Título: <Título del trabajo de diploma>

Autor: <Nombres y apellidos del autor o los autores>

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan.

<Aquí el tutor debe expresar cualitativamente su opinión y medir (usando la escala: muy alta, alta, adecuada) entre otras las cualidades siguientes:

- Independencia
- Originalidad
- Creatividad
- Laboriosidad
- Responsabilidad>

<Además, debe evaluar la calidad científico-técnica del trabajo realizado (resultados y documento) y expresar su opinión sobre el valor de los resultados obtenidos (aplicación y beneficios) >

Por todo lo anteriormente expresado considero que el estudiante está apto para ejercer como Ingeniero Informático; y propongo que se le otorgue al Trabajo de Diploma la calificación de <nota 2-Desaprobado, 3-Aprobado, 4-Bien, 5-Excelente>. <Además, si considera que los resultados poseen valor para ser publicados, debe expresarlo también>

(Si procede)

Nombre completo del primer tutor

<Grado científico, Categoría docente y/o investigativa>

Nombre completo del segundo tutor

<Grado científico, Categoría docente y/o investigativa>

Fecha: _____

AGRADECIMIENTOS

A mis padres y hermano que me han guiado por el camino del bien y han estado siempre a mi lado. A ustedes gracias por darme la oportunidad de agradecerle hoy en día la persona que soy. Todo en mí se los debe a ustedes y quiero aprovechar la ocasión para decirles que los quiero mucho.

A mi novia por su cariño y su infinito amor, por estar siempre a mi lado en los momentos malos y buenos. Por su eterna compañía. Por su apoyo incondicional.

A todos mis profesores, y compañeros de aula, gracias por su ayuda.

DEDICATORIA

(Opcional)

Pequeños textos que el (los) autor(es) redacta(n) a su gusto y estilo personal, dedicados a las personas o instituciones que en su opinión lo ameriten.

RESUMEN

El presente proyecto de tesis consiste en desarrollar e implementar una Sistema Web para la División Territorial de Etecsa en Cienfuegos centrado en la automatización de los procesos de gestión de eventos para la innovación de la ciencia, técnica y medioambiente de esta Empresa, la cual presenta cierta dificultad a la hora de realizar estos eventos debido a que no posee una herramienta informática capaz de llevar a cabo dichos eventos. El objetivo fundamental de esta investigación es la creación de una aplicación web para informatizar estos procesos y consigo mejorar la gestión de información generada y posibilitar una mejor organización en los procesos llevados a cabo en el centro laboral.

El proyecto está basado en tres ramas. La primera es la Programación Web siendo esta la base principal del proyecto, se apoyará en la rama de Diseño Web y sustentará toda su información en la rama de Base de Datos.

Para el desarrollo de este trabajo se utilizó ICONIX como metodología de desarrollo, y la herramienta CASE Enterprise Architect versión 16.1 para el diseño de la ingeniería de software. Para la programación se utilizó Visual Studio Code y framework Laravel versión 10 junto a php versión 8.1, y como sistema de gestión de bases de datos se utilizó MySQL.

Palabras clave:

Aplicación Web, informatizar, desarrollar

RESUMEN EN INGLES

The present thesis project consists of developing and implementing a Web System for the Territorial Division of Etecsa in Cienfuegos focused on the automation of event management processes for the innovation of science, technology and the environment of this Company, which presents certain difficulty when carrying out these events because it does not have a computer tool capable of carrying out these events. The fundamental objective of this research is the creation of a web application to automate these processes and improve the management of information generated and enable better organization of the processes carried out in the workplace.

The project is based on three branches. The first is Web Programming, this being the main basis of the project, it will be supported by the Web Design branch and will support all its information in the Database branch.

For the development of this work, ICONIX was used as the development methodology, and the CASE Enterprise Architect version 16.1 tool was used for the software engineering design. For programming, Visual Studio Code and Laravel framework version 10 were used along with PHP version 8.1, and MySQL was used as the database management system.

Keywords:

Web application, computerize, develop

Indices

INTRODUCCIÓN	12
CAPITULO I – Fundamentación Teórica	16
1.1-Introducción	16
1.2-Objeto de estudio	16
Objetivos estratégicos de la organización	16
1.3-Flujo actual de los procesos y Análisis crítico de la ejecución de los procesos	16
Procesos objeto de automatización	17
1.4-Sistemas automatizados existentes vinculados al campo de acción	17
1.5-Metodología y herramientas utilizadas en la solución propuesta	17
1.5.1 Metodología de desarrollo de software y Lenguaje de Modelaje Unificado	17
1.5.2 Lenguaje Unificado de Modelado	19
1.5.3 Lenguaje de programación PHP	20
1.5.4- Framework de Desarrollo.	21
1.5.5-Base de datos:	23
1.6-Conclusiones.....	24
CAPITULO II – Análisis y diseño de la solución propuesta	25
2.1-Introducción	25
2.1-Requisitos.....	25
2.1.1-Requisitos funcionales.....	25
2.1.2-Definición de los requisitos no funcionales.....	27
2.2 Modelamiento del Dominio.....	28
2.2.1-Reglas del negocio a considerar	30
2.3-Modelación de los Casos de Uso	30
2.4-Paquetes y sus relaciones.....	35
2.5 Descripción de los casos de uso	35
2.6-Análisis de Robustez	39
2.7-Arquitectura técnica	50
2.8-Conclusiones.....	52
Capítulo III – Descripción de la solución propuesta.....	53
3.1-Introducción	53
3..2-Diagrama de clases del diseño	53
Diagrama 20. Diagrama de clases de diseño.....	53
3.3-Principios de diseño	54
3.4-Tratamiento de errores	54

3.5-Diseño de la base de datos	55
3.6-Diagrama de despliegue	56
3.7-Estimación por casos de uso	56
3.8-Cálculo de los puntos de casos de uso sin ajustar	57
3.8.1-Factor de peso de los actores del sistema	58
3.8.2-Factor de peso sin ajustar de los casos de uso del sistema.....	59
3.8.3-Factores de complejidad técnica.....	60
3.8.4-Factores ambientales	61
3.9-Estimación del esfuerzo.....	61
3.9.1-Resultados de los cálculos.....	62
3.10-Distribución del esfuerzo.....	63
3.10.1-Estimación del esfuerzo	63
3.11-Determinación de los costos.....	64
3.11.1-Estimación del costo de realización del proyecto	64
3.12-Beneficios Tangibles e Intangibles	65
3.13-Análisis de costos y beneficios	66
3.14-Conclusiones.....	66
<i>Conclusiones:</i>	<i>67</i>
<i>Recomendaciones:</i>	<i>68</i>
<i>Referencias Bibliográficas</i>	<i>69</i>
<i>Anexos:.....</i>	<i>71</i>
Anexo 1-Pantalla Inicio de sesión	71
Anexo 2-Pantalla Registro	72
Anexo 3-Pantalla Principal	72
Anexo 4-Pantalla Memorias	73
Anexo 5-Pantalla de Eventos	73
Anexo 6-Pantalla de las Ponencias.....	74
Anexo 7-Pantalla del Banco de Problemas	74
Anexo 8-Pantalla de las estadísticas de los eventos por fecha	75
Anexo 9-Pantalla de todos los Usuarios	75
Anexo 10-Pantalla Crear Evento	76

INTRODUCCIÓN

El rápido avance de la tecnología, los revolucionarios sistemas de cómputo, la inesperada velocidad de expansión de las redes de computadoras y el intercambio de datos que estas provocaron, propiciaron la proliferación de nuevas fuentes de información digital, revolucionando los conceptos tradicionales que se tienen de libros, bibliotecas, investigaciones y aprendizaje.

No es difícil imaginar que en un futuro toda la información será procesada y adquirida a través de medios electrónicos. A criterio de muchos, el libro tradicional será complementado, y posiblemente reemplazado, por alguna nueva tecnología que permita al lector un mejor aprendizaje, o simplemente, una lectura más amena.[1]

La información es un recurso necesario y de gran valor para las empresas que la gestionan adecuadamente. La información implica un proceso de interpretación y transformación cuyo principal objetivo es la minimización de la incertidumbre en la toma de decisiones. A pesar de que todos los empleados de una organización precisan información, no la necesitan del mismo tipo. El tipo de información requerido dependerá de factores como el nivel jerárquico, la labor que se está realizando, la confidencialidad y la urgencia, entre otros.[1]

La gestión de datos y la utilización de la información en una cuestión fundamental para el hombre como actor y beneficiado del surgimiento de la computación y todos los adelantos científicos-técnicos, desempeñando este un importante papel en el desarrollo de diferentes métodos y herramientas para gestionar toda la información que se presenten en las empresas; conociéndose así que todo sistema constituye una ayuda en la informatización de las tareas necesarias de una institución.[2]

Un sistema de gestión es una herramienta que permite controlar, planificar, organizar y, hasta cierto punto, informatizar las tareas de una empresa. Su objetivo es unificar en un único software todas las operaciones de la compañía con el fin de facilitar la toma de decisiones y el análisis de los datos.[3]

En los últimos años, los sistemas de gestión empresarial han ido ganando popularidad como herramientas fundamentales para las organizaciones gracias a sus prestaciones cada vez más completas, especialmente en un proceso de transformación digital como el que se está viviendo actualmente.[3]

En Cuba, una de las prioridades claves para el desarrollo económico y social lo constituye el papel que juegan la ciencia, la tecnología y la innovación. Esto es una exigencia del proceso de cambio al que llama la actualización del modelo económico cubano y a la necesidad de mejorar los impactos de la ciencia y la tecnología, mediante el desarrollo en todos los ámbitos de la actividad productiva, la vida social y la cultura.

El país actualmente se encuentra en un proceso de informatización social y empresarial que entre sus objetivos está la creación de sistemas informáticos para la gestión de procesos empresariales, de almacenamiento de datos, de relación con el cliente para mediante estos sistemas automatizar, mejorar y ordenar la mayor cantidad de procesos, estructuras de negocio e información que actualmente existe en duro (papel) para poder alcanzar un mayor desarrollo en marcos sociales, políticos y económicos.[4]

Situación Problemática:

Actualmente en la División Territorial de Etecsa Cienfuegos el proceso de gestión para la Innovación de la Ciencia y la Técnica se efectúa de manera manual y basándose en los Office (Excel, Word, etc.) existentes en esta empresa, sin apoyarse en ninguna metodología o uso de software, siendo este un proceso muy difícil de gestionar y organizar su información por toda la documentación que genera. Por esto el Especialista General para la Ciencia-Técnica y Medio Ambiente tiene la necesidad de integrar todas las actividades que ejecuta su cargo y poder realizar un proceso de informatización y gestión de almacenamiento de datos.

Debido a la falta de un software:

- 1-La información de las ponencias entregadas se guardan de manera desorganizada y hace muy difícil su gestión y revisión posteriormente.
- 2-Los trabajadores entregan sus ponencias con ciertos errores, al no saber por cual plantilla guiarse o no conocer la existencia de estas.

Para resolver lo anteriormente planteado es necesaria la creación de un sistema web con una base de datos de respaldo que mejore la experiencia de los trabajadores, que cree las bases para la informatización de los procesos y organice de la mejor manera posible toda la documentación que se genera agregando así un plus en la innovación de la ciencia de esta empresa.

Se identifica como Problema de investigación: ¿Cómo gestionar los eventos para la Innovación de la ciencia y la técnica en Etecsa de manera eficiente?

Objeto de estudio: Los procesos de gestión de los eventos para Innovación de la Ciencia y la Técnica en ETECSA.

Campo de acción: Los procesos y subprocesos de gestión de los eventos para Innovación de la Ciencia y la Técnica en ETECSA.

Idea a defender: La elaboración de una aplicación web para informatizar y gestionar los procesos de los eventos para Innovación de la Ciencia y la Técnica en ETECSA.

Objetivo General: Desarrollar una aplicación web que permita integrar e informatizar varios procesos que actualmente están separados y se efectúan de manera manual.

Del objetivo general se desprenden los siguientes Objetivos Específicos:

- ✓ Analizar el proceso actual de la gestión de información de los eventos en Etecsa.
- ✓ Especificar los requisitos funcionales y no funcionales a partir de los procesos de modelado actual.
- ✓ Diseñar un sistema informático que agilice el acceso a la información.
- ✓ Implementar el sistema informático propuesto.
- ✓ Realizar la estimación de esfuerzo necesario para la ejecución del proyecto.
- ✓ Implementar el sistema informático diseñado.

Aporte práctico

La investigación encamina su aporte al mejoramiento de los procesos de la gestión comercial del departamento, lo cual permite un impacto positivo en la Empresa de Telecomunicaciones de Cuba S.A (ETECSA). El sistema permitirá resolver la carencia de una aplicación web, al cual los participantes en los procesos de innovación y la ciencia tendrán acceso para así informatizar y agilizar dichos procesos.

El presente trabajo está estructurado en 3 capítulos, a continuación, se muestra una breve descripción de cada uno de ellos:

Capítulo 1. Fundamentación Teórica

En el mismo se abordan los fundamentos teóricos donde se analizan el flujo actual de los procesos que se mantienen en el Banco de Sangre Cienfuegos y las necesidades con las que cumplen, seguido de la revisión bibliográfica de antecedentes nacionales e internacionales, analizando los sistemas existentes y se dan a conocer la metodología y herramientas utilizadas en la solución del problema de investigación.

Capítulo 2. ANALISIS Y DISEÑO DE LA PROPUESTA

Se realiza una caracterización de la concepción general del sistema a desarrollar, donde se especifican sus requisitos funcionales y no funcionales, se conceptualizan los objetos del modelo del dominio y se representan. Se lleva a cabo la modelación de los casos de uso y sus descripciones detalladas junto a los requerimientos, así como sus análisis de robustez. Por último, se enmarca el diseño detallado del sistema a desarrollar y sus principios.

Capítulo 3.

Teniendo en cuenta el capítulo 2 se realiza el estudio de factibilidad a través de la estimación por casos de uso, donde se realiza un seguimiento de los factores de cálculo de esfuerzo entre los que están, el factor de peso de los actores del sistema, el factor de peso sin ajustar de los casos de uso, los factores de complejidad técnica y los factores ambientales, permitiendo obtener los resultados de la estimación de esfuerzo y esclarecer los beneficios tangibles e intangibles y el análisis de costo del proyecto.

CAPITULO I – Fundamentación Teórica

1.1-Introducción

En el presente capítulo se abordan aspectos teóricos sobre la gestión de la información relacionada con el proceso de ciencia y técnica. Es el resultado de la búsqueda y análisis de la información vinculada al objeto de estudio, sistemas existentes asociados al campo de acción y se definen las técnicas, metodologías y herramientas a utilizar en la elaboración del sistema informático.

1.2-Objeto de estudio

Objetivos estratégicos de la organización

- Diseñar y desarrollar una Aplicación Web de fácil navegación y funcional para los futuros usuarios.
- Brindar a la División Territorial de Etecsa Cienfuegos una mejora para poder gestionar los eventos que se desarrollan en el centro con una mayor organización y facilidad.
- Implementar la Aplicación Web y capacitar a los usuarios.

1.3-Flujo actual de los procesos y Análisis crítico de la ejecución de los procesos

La especialista general de la ciencia-técnica y medio ambiente realiza una convocatoria masiva a los trabajadores de Etecsa y a algún invitado interesado en participar. Se da a conocer las plantillas para las entregas de las futuras ponencias. Tiempo antes de la realización de los eventos los trabajadores tienen que entregar sus ponencias en donde deben especificar el título de la ponencia, autores y entregar un archivo pdf con su trabajo a exponer en el evento. Es necesario hacer un listado con la cantidad de participantes del evento. Una vez culminado el evento en cuestión se da a conocer los reconocimientos y evaluaciones de los trabajos entregados, generando así los diplomas a entregar.

Todas las ponencias son guardadas ya sean en la computadora con la entrega de pdf o en físico mediante impresiones.

Procesos objeto de automatización

El sistema está basado en la automatización de los procesos de gestión de la información en la División territorial de Etecsa en Cienfuegos. Agrupando los requerimientos funcionales del sistema. Permite a la Especialista General para la Ciencia y la Técnica de esta empresa crear los nuevos eventos, gestionar cada uno de estos de una manera fácil, permite visualizar el proceso y estar al tanto de algún cambio. Permite a los trabajadores e invitados de la empresa a participar en los eventos creados según el rol asignado por el Gestor CT. Tiene una base de datos actualizada mensualmente de los trabajadores del centro, de los fórums realizados anteriormente, de las ponencias presentadas en años anteriores y un banco de problemas ya sean resueltos.

1.4-Sistemas automatizados existentes vinculados al campo de acción

Después de investigar acerca de la existencia de un sistema que diera solución a la problemática que afecta al departamento de coordinación general en la parte de ciencia y técnica, se arribó a la conclusión que no existe ningún software a nivel nacional que le solucione lo planteado anteriormente.

1.5-Metodología y herramientas utilizadas en la solución propuesta

A continuación, se describen la metodología y las herramientas utilizadas.

1.5.1 Metodología de desarrollo de software y Lenguaje de Modelaje Unificado

ICONIX es una metodología de desarrollo de software que media entre lo complejo y documentado (pesada) del RUP (Rational Unified Process) y lo simple y práctico (ágil) del XP (Extreme Programming), sin eliminar las tareas de análisis y de diseño que XP no contempla. Esta es la principal razón por la que fue seleccionada esta metodología para desarrollar el proyecto.

Esta metodología fue desarrollada por Doug Rosenberg y Matt Stephens. ICONIX es un proceso simplificado en comparación con otros procesos más tradicionales; unifica un conjunto de métodos de análisis y diseño orientados a objetos con el objetivo de abarcar todo el ciclo de vida de un proyecto. Presenta claramente las actividades de cada fase y exhibe una secuencia de pasos que deben ser seguidos. Además, está adaptado a los patrones y utiliza varios artefactos de UML; se guía a través de casos de uso y sigue un ciclo de vida iterativo e incremental.[5], [6]

El proceso ICONIX cuenta con tres características fundamentales:

- Iterativo e Incremental: Durante el desarrollo del modelo del dominio y la definición de los casos de uso se producen varias iteraciones. El ciclo de vida incremental consiste en desarrollar por partes el producto de manera que puedan ser integradas funcionalmente. Ciclo de vida iterativo, en cada ciclo de iteración se revisa y mejora el producto.
- Trazabilidad: Cada paso que se realiza está definido por un requisito, se define la trazabilidad como la capacidad de seguir una relación entre los diferentes artefactos de software producidos.
- Uso racional de UML: representa un enfoque minimalista y utiliza un subconjunto de los artefactos de UML. ICONIX se organiza en cuatro pasos fundamentales con sus correspondientes hitos. Cada paso del proceso ICONIX culmina con un hito de revisión. La primera de ellas es la definición de requisitos, seguida del análisis y diseño preliminar; a continuación, le sigue el diseño y finaliza con su implementación. En cada una de las fases se ejecutan tareas específicas, según se describe a continuación:

1. Requisitos

- Requisitos funcionales: definen lo que hará el sistema. La definición de los requisitos funcionales queda un poco fuera del proceso ICONIX. Se asume que los requisitos funcionales ya han sido definidos o se están definiendo.
- Modelado del dominio: identificar los objetos del dominio, así como sus interrelaciones (agregación, generalización-especialización), y obtener un diagrama de clases de alto nivel.
- Requisitos de comportamiento: definir cómo los usuarios interactúan con el sistema; escribir los casos de uso y sus diagramas. Los casos de usos pueden ser refinados posteriormente. Se recomienda utilizar prototipos de interfaces gráficas de usuarios (GUI). Organizar los casos de uso en grupos capturando esta estructura en un diagrama de paquetes. o Asignar los requerimientos funcionales a los casos de uso.
- Hito 1: Revisión de los requerimientos: asegurarse que las descripciones de los casos de uso se correspondan con las expectativas de los clientes.

2. Análisis/Diseño preliminar

- Análisis de robustez: elaborar el diagrama de robustez (un diagrama de los pasos de un caso de uso). Actualizar los casos de uso según corresponda. o Identificar los objetos frontera, controladores y entidades. Actualizar el modelo del dominio con posibles nuevas clases, corrigiendo ambigüedades y agregando atributos a los objetos del dominio.
- Nombrar todas las funciones del software (controladores) necesarias para el trabajo de los casos de uso.
- Hito 2: Revisión del diseño preliminar (PDR).

En esta fase se debe comenzar a formular la arquitectura técnica con el objetivo de tener una idea general del sistema que se desarrollará (¿será una aplicación Web?, ¿o una aplicación de escritorio?) ¿Se utilizará un framework de aplicación específico? Sin embargo, no se debe cruzar la línea hacia un diseño detallado, solamente asumir ciertos supuestos acerca del diseño, y asegurarse que éste funcionará en la arquitectura seleccionada.

3. Diseño detallado

➤ Diagramas de secuencia: elaborar un diagrama de secuencia por cada caso de uso que muestre los detalles de su implementación. La función principal de los diagramas de secuencia es asignar comportamientos a las clases.

➤ Actualizar el modelo del dominio agregando las operaciones a los objetos del dominio. En este punto, los objetos del dominio se convierten en realidad en clases del dominio, o entidades, y el modelo del dominio se va convirtiendo en un modelo estático, o diagrama de clases que es una parte esencial del diseño detallado.

➤ Hito 3: Revisión crítica del diseño (CDR). Verificar que el diseño se corresponde con todos requisitos identificados.

4. Implementación

➤ Codificación / pruebas unitarias: Escribir el código y las pruebas unitarias (o en orden inverso).

➤ Pruebas de integración: Basar las pruebas de integración en los casos de uso de manera que se chequee tanto el curso normal como los alternativos.

➤ Pruebas de sistema y de aceptación basadas en casos de uso.

1.5.2 Lenguaje Unificado de Modelado

El proceso ICONIX utiliza el método orientado a objetos con enfoque unificado basado en el Lenguaje Unificado de Modelado (UML, por sus siglas en inglés), el cual es el lenguaje gráfico de modelaje para especificar, visualizar, construir y documentar los elementos de los sistemas de software. UML se utiliza para especificar o para describir métodos o procesos. Es útil para definir un sistema, para detallar los artefactos en el sistema, para documentar y construir. El Lenguaje Unificado de Modelado permite una comunicación sencilla y rápida entre desarrolladores y clientes del software que se desarrolla. No se necesitan conocimientos profundos de ingeniería del software para que los clientes comprendan lo que los desarrolladores muestran, de modo que rápidamente pueden expresar su conformidad con el producto o las nuevas mejoras que desean ver introducidas.[7], [8]

1.5.3 Lenguaje de programación PHP

PHP es un lenguaje de programación de uso general que se adapta especialmente al desarrollo web.

El código PHP suele ser procesado en un servidor web por un intérprete PHP implementado como un módulo, un Daemon o como un ejecutable de interfaz de entrada común (CGI). En un servidor web, el resultado del código PHP interpretado y ejecutado que puede ser cualquier tipo de datos, como el HTML generado o datos de imágenes binarias formaría la totalidad o parte de una respuesta HTTP. Existen diversos sistemas de plantillas, sistemas de gestión de contenidos y frameworks que pueden emplearse para organizar o facilitar la generación de esa respuesta.

Características:

- Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- Es considerado un lenguaje fácil de aprender, ya que en su desarrollo se simplificaron distintas especificaciones, como es el caso de la definición de las variables primitivas, ejemplo que se hace evidente en el uso de php arrays.
- El código fuente escrito en PHP es invisible al navegador web y al cliente, ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando módulos (llamados extensiones).
- Posee una amplia documentación en su sitio web oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite aplicar técnicas de programación orientada a objetos.
- No requiere definición de tipos de variables, aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar, aun haciéndolo, el programador puede aplicar en su trabajo cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del patrón de diseño Modelo Vista Controlador (MVC), que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes.

- Debido a su flexibilidad, ha tenido una gran acogida como lenguaje base para las aplicaciones WEB de manejo de contenido, y es su uso principal.[9]

1.5.4– Framework de Desarrollo.

Laravel 10

Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web que utiliza PHP 8 o superior. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti". Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC.

Laravel tiene como objetivo ser un framework que permita el uso de una sintaxis elegante y expresiva para crear código de forma sencilla y permitiendo multitud de funcionalidades. Intenta aprovechar lo mejor de otros frameworks y aprovechar las características de las últimas versiones de PHP.

Características

- 1-Sistema de enrutamiento.
- 2-Blade, Motor de plantillas.
- 3-Peticiones Fluent.
- 4-Eloquent ORM
- 5-Basado en Composer.
- 6-Soporte para el caché
- 7-Soporte para MVC

Patrón MVC

Laravel propone en el desarrollo usar 'Routes with Closures', en lugar de un MVC tradicional con el objetivo de hacer el código más claro. Aun así, permite el uso de MVC tradicional.

Modelo

Laravel incluye un sistema de mapeo de datos relacional llamado Eloquent ORM que facilita la creación de modelos. Este ORM se funda en un patrón active record. Es opcional el uso de Eloquent, pues también dispone de otros recursos que facilitan la interacción con los datos, o específicamente la creación de modelos.⁷

El código del modelo puede ser más simple en el caso que el nombre de la tabla coincida con el nombre de la clase, ya que Laravel, al igual que Ruby on Rails, usa el paradigma de programación donde se favorece "la convención sobre la configuración".

Vista

Laravel incluye de paquete un sistema de procesamiento de plantillas llamado Blade. Este sistema de plantillas favorece un código mucho más limpio en las Vistas, además de incluir un sistema de Caché que lo hace mucho más rápido. El sistema Blade de Laravel, permite una sintaxis mucho más reducida en su escritura.

Lo cual no es una gran ventaja, siempre es posible usar una expresión resumida en PHP. No obstante, lo que, si es una gran ventaja, es el modo en que Blade maneja las plantillas.

Plantillas

Las plantillas en Blade son archivos de texto plano que contiene todo el HTML de la página con etiquetas que representan elementos o zonas a incluir en la plantilla, o vistas parciales como se conocen en otros frameworks PHP. Sin embargo, en Blade estos elementos incrustados se organizan en un solo archivo. Esta es una idea muy interesante de Laravel que mejora la organización de las vistas y su rendimiento. Sobre todo, cuando las vistas pueden llegar a ser muy complejas incluso con elementos anidados. En el render de una Vista completa en Laravel se usan dos archivos: la plantilla definiendo el HTML global y las zonas a incluir. Un solo archivo, la Vista, con los elementos (partial views).

Controlador

Los controladores contienen la lógica de la aplicación y permiten organizar el código en clases sin tener que escribirlo todo en las rutas. Todos los controladores deben extenderse de la clase BaseController .

Influencia

La influencia de Laravel ha crecido rápidamente desde su lanzamiento. En la comunidad de desarrolladores es considerado como alternativa sencilla de usar pero que tiene todas las funcionalidades que debe tener un framework. Ha sido descargado más de 20'500,000 de veces, y se espera que supere en popularidad a otros frameworks ya establecidos más antiguos.[10]

1.5.5-Base de datos:

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, todo para entornos de desarrollo web.[11]

1. Código abierto

MySQL utiliza la Licencia Pública General de GNU, por lo que se puede descargar, utilizar y modificar a voluntad. Esto facilita su uso tanto académico como profesional.

2. Uso multiplataforma

Una de sus características principales y de mayor ventaja es que puede instalarse en entornos con sistemas operativos diversos como Windows, Mac y la mayoría de distribuciones Linux, así como en ambientes Unix.

3. Escalabilidad

Tiene soporte para 40-50 millones de registros, 150.000-200.000 tablas y 5000 millones de filas.

4. Tipos de datos

Soporta una amplia gama de tipos de datos, lo que permite tener una gran versatilidad en cuanto a las situaciones, industrias o casos de uso donde puede implementarse una base de datos MySQL. Puede emplearse para la industria financiera, al manejar datos con mucha precisión; por otro lado, también puede utilizarse en ámbitos de geolocalización por sus datos de tipo espacial. De igual forma puede competir, en ciertas situaciones, con las bases de datos no relacionales con su tipo de dato JSON.

5. Conjuntos de caracteres

Es compatible con un gran listado de conjuntos de caracteres e idiomas, lo que le permite adaptarse a cualquier parte del mundo. Sin duda alguna, es un aspecto que le ha ayudado a posicionarse en los sistemas de internet a lo largo y ancho del planeta.

6. Clientes gráficos

Si bien MySQL utiliza su propio lenguaje para administrar los datos almacenados, existen diversas herramientas o clientes gráficos que nos permiten interactuar con las bases de datos, ayudando a que dicha interacción sea más sencilla y, por lo tanto, más rápida. Algunas herramientas son:

- PHPMysqlAdmin
- MySQL Workbench

7. Soporte para lenguajes de programación

Las características y ventajas de MySQL son muchas, pero sin duda todas ellas son mejor explotadas cuando están integradas dentro de un sistema de información. Para ello existe un amplio abanico de API nativas, librerías, paquetes, etc. que permiten integrar una base de datos MySQL en un sistema desarrollado en cualquier lenguaje de programación.

8. Documentación actualizada

Al ser muy popular y utilizado, permite que exista una documentación oficial muy amplia, además de una comunidad enorme siempre dispuesta a ayudar, colaborar y aportar al conocimiento compartido.[12]

1.6-Conclusiones

En este capítulo se realizó un estudio de los conceptos asociados al dominio del problema, así como un análisis de las tecnologías Web, las metodologías de desarrollo, los lenguajes de programación y gestor de bases de datos. Seleccionándose la metodología ICONIX como guía para el desarrollo y documentación del sistema propuesto. Se utilizó como lenguaje de programación PHP en su versión 8.1 con el Entorno de Desarrollo Integrado (IDE) Visual Code Studio basándose en la arquitectura cliente/servidor y MySQL para la gestión de la base de datos.

CAPITULO II – Análisis y diseño de la solución propuesta

2.1-Introducción

El modelado del sistema constituye uno de los procesos fundamentales para la automatización de procesos. Comienza estableciendo requisitos de todos los elementos del sistema y asignando al software un subgrupo de estos requisitos los cuales puede ser funcionales (funcionalidades del sistema) y no funcionales (características no funcionales del producto).

La ingeniería de requisitos comprende todas las tareas relacionadas con la determinación de las necesidades o de las condiciones a satisfacer para un software tomando en cuenta los diversos requerimientos de las partes interesadas El propósito es hacer que los mismos alcancen un estado óptimo antes de alcanzar la fase de diseño en el proyecto.

2.1-Requisitos

La etapa análisis de requisitos es la más importante en el desarrollo del proyecto informático porque define lo que será capaz de hacer el sistema. Se deben identificar de manera clara las descripciones de los casos de uso del dominio, así como puntualizar con el usuario el comportamiento y funcionalidad que necesita el programa. A partir de los datos analizados se construye el modelo del dominio con los requerimientos funcionales y requerimientos no funcionales.

2.1.1-Requisitos funcionales

Mediante la definición de los requisitos funcionales se precisa el comportamiento o función particular de un sistema cuando se cumplen ciertas condiciones. Por lo general, deben incluir funciones específicas, descripciones de los flujos de trabajo a ser desempeñados por el sistema y otros requerimientos de negocio, cumplimiento, seguridad u otra índole. Se debe asegurar que todos los involucrados, principalmente los clientes, comprendan claramente el significado de cada uno. Una especificación deficiente de los requisitos del software puede conducir a proyectos fallidos, por tanto, esta disciplina adquiere cada día importancia y se dedica más tiempo al estudio previo de los requisitos del usuario, al nivel de comunicación y al modelado del negocio antes de desarrollar el software.[13]

A continuación, se exponen y se enumeran los requerimientos funcionales del sistema:

RF001- Autenticación:

RF001- 01-Iniciar Sesión: El sistema debe permitir iniciar su sesión de trabajo en la web.

RF001- 02-Registrarse: El sistema debe permitir registrarse en la web para poder tener un rol.

RF001- 03-Cerrar sesión: El sistema debe permitir cerrar su sesión de trabajo.

RF002- Gestionar Usuarios. El sistema debe permitir al Gestor gestionar los usuarios.

RF002- 01-Asignar Rol.

RF002- 02-Agregar usuario.

RF002- 03-Eliminar Usuario.

RF003-Gestionar Eventos: El sistema debe permitir gestionar los eventos.

RF003-01-Crear Evento.

RF003-02-Editar Evento.

RF003-03-Eliminar Evento.

RF004- Inscribirse al Evento: El sistema debe permitir al ponente inscribirse al evento en proceso.

RF004-01-Agregar Ponencia al Evento: El sistema debe permitir al ponente agregar su ponencia.

RF004-02-Almacenar ponencia en la Biblioteca: El sistema debe permitir guardar las ponencias entregadas en la base de datos.

RF005- Descargar Archivos Necesarios: El sistema debe permitir descargar los archivos necesarios.

RF005-01-Descargar preforma PPT del trabajo.

RF005-02-Descargar Aval de Certificado.

RF006- Visualizar Banco de Problemas: El sistema debe permitir visualizar el banco de problemas.

RF007- Gestionar Banco de Problemas: El sistema debe permitir al Gestor gestionar el banco de problemas.

RF007-01-Agregar Problema al Banco de problemas.

RF007-02-Editar un Problema del banco.

RF007-03-Eliminar un Problema del Banco.

RF008- Visualizar módulo de Distinciones: El sistema debe permitir visualizar el módulo distinciones.

RF009- Editar Distinciones. El sistema debe permitir agregar usuarios a las distinciones.

RF010- Acceder al módulo de Memorias: El sistema debe permitir visualizar el módulo memorias.

RF011- Acceder a Estadísticas: El sistema debe permitir visualizar el módulo Estadísticas.

2.1.2-Definición de los requisitos no funcionales

Los requerimientos no funcionales del sistema propuesto son los siguientes:

1- Apariencia o interfaz externa

El sistema presenta una interfaz sencilla, de fácil acceso para los usuarios y con un menú de opciones bien detallado; predomina el color azul y naranja que identifica a la empresa a la que está destinado el producto; resalta las letras en contraste con el fondo de la aplicación y permite a este leer con facilidad.

2- Usabilidad

El sistema indica directamente los datos que se deben introducir, además de los formularios que deben ser completados.

3- Confiabilidad

EL producto estará protegido contra fallos e inconsistencia de datos, pues antes de guardar información en la base de datos se realizarán varios chequeos para que la información sea la correcta.

4- Rendimiento

El sistema debe procesar las transacciones en poco tiempo, además, ejecutar de manera exitosa las operaciones que implican un elevado nivel de procesamiento y retornan una cantidad elevada de datos. Debe recuperarse en un corto período de tiempo ante cualquier falla. La eficiencia de la aplicación está determinada en gran medida por el aprovechamiento de los recursos que se disponen en la arquitectura por capas y la velocidad de las consultas a la base de datos.

5- Restricciones en el diseño y la implementación

El SGBD es MySQL

6- Ayuda y documentación en línea

El producto posee enunciados de derechos de autor, logos corporativos, iconos y otros elementos de la interfaz de usuario que deban mantenerse consistente con el resto de la documentación.

7- Seguridad

La confidencialidad de la información con que trabajará el sistema estará protegida de acceso no autorizado pues el sistema debe tener diferentes niveles de acceso.

Solo ciertos usuarios tendrán acceso a modificar la información sobre la que se basa funcionamiento el sistema, garantizándose la seguridad e integridad de los datos almacenados y de esa forma la confiabilidad de la información de los resúmenes y procesamientos realizados. Cada usuario solo controlará la información correspondiente a los sectores y zonas que maneja, lo cual es obtenido en el momento de autenticación del usuario en el sistema.[14], [15]

2.2 Modelamiento del Dominio

Un modelo de dominio en la resolución de problemas e ingeniería de software, es un modelo conceptual de todos los temas relacionados con un problema específico. En él se describen las distintas entidades, sus atributos, papeles y relaciones, además de las restricciones que rigen el dominio del problema.

El modelo del dominio define el alcance del proyecto y permite formar la base sobre la cual construir los casos de uso. En este proceso se ejecutan las tareas de transformaciones de los objetos y conceptos relacionados con el problema en clases. A partir de los datos que se asocian con los requerimientos se elabora la parte estática que describe la estructura y constituye el modelo del dominio. También proporciona un vocabulario común para que exista una comunicación clara entre los miembros del equipo de desarrollo y los clientes.

Para la construcción de este diagrama se establecen los siguientes procedimientos:

- Tomar documentos disponibles y hacer una lectura rápida, subrayando los sustantivos y notando frases posesivas y verbos (uso posterior). Los sustantivos y frases nominales se convertirán en objetos y atributos. Los verbos y frases verbales se convertirán en operaciones y relaciones. Las frases posesivas indican los sustantivos que son atributos y no objetos.
- Formar una lista con los sustantivos y frases nominales identificados, evitando los plurales y las repeticiones y ordenándola alfabéticamente.
- Revisar la lista eliminando los elementos innecesarios o incorrectos. Volver a revisar textos, leyendo entre líneas.

Construir relaciones de generalización. Establecer asociaciones entre clases. Establecer relaciones de agregación. Los objetos del dominio identificados son los siguientes:

Evento: Proceso en el que participa los trabajadores de la empresa con vistas a la innovación de la ciencia y la técnica en el centro.

Gestor: Encargado del gestionar el Sistema.

Usuario: Trabajador de la Empresa que no participa en el Evento

Ponente: Trabajador o invitado de la Empresa que tiene intenciones de participar en los Eventos.

Ponencia: Documentos desarrollados por los trabajadores.

Preforma del trabajo: Documentos que sirven como plantillas para realizar las ponencias.

Problema: Contratiempo guardados en la base de datos al que se le debe dar cierta solución en el evento.

Banco de problemas: conjunto de problemas almacenados en la base de datos.

2.2.1-Reglas del negocio a considerar

- ✓ El Especialista General para la Ciencia y la Técnica en la división territorial de Etecsa en Cienfuegos es el encargado de crear un evento y de monitorear todo el proceso.
- ✓ Es necesario brindar la información sobre la preforma de la ponencia y la fecha exacta en que se realizara el Evento.
- ✓ Permitir a los trabajadores que se sumen al evento y que puedan entregar sus ponencias relacionadas con el evento de una manera fácil y segura.
- ✓ Permitir siempre que el usuario pueda descargar el aval de certificado, la plantilla PPT.
- ✓ Todas las ponencias son almacenadas y organizadas.
- ✓ No puede existir ponencias con el mismo nombre almacenadas.

2.3-Modelación de los Casos de Uso

En la metodología ICONIX, la identificación de los casos de uso es un proceso que juega un papel esencial para el desarrollo de un sistema informático, principalmente porque se usan para descubrir, capturar y presentar los requerimientos de usuario en una forma accesible a todos los involucrados.

El modelo de los casos de uso comprende los actores, el sistema y los propios casos de uso. El conjunto de funcionalidades de un sistema se determina examinando las necesidades funcionales de cada actor. Los actores representan entidades externas al sistema, personas o sistemas, y son análogos a un rol del usuario.

Los casos de uso se deben organizar en tantos paquetes como sea necesario, con un paquete por cada subsistema, con el objetivo de facilitar la comprensión de los mismos y de sus relaciones.

En la Figuras siguientes se muestra el diagrama de paquetes de los casos de usos identificados:

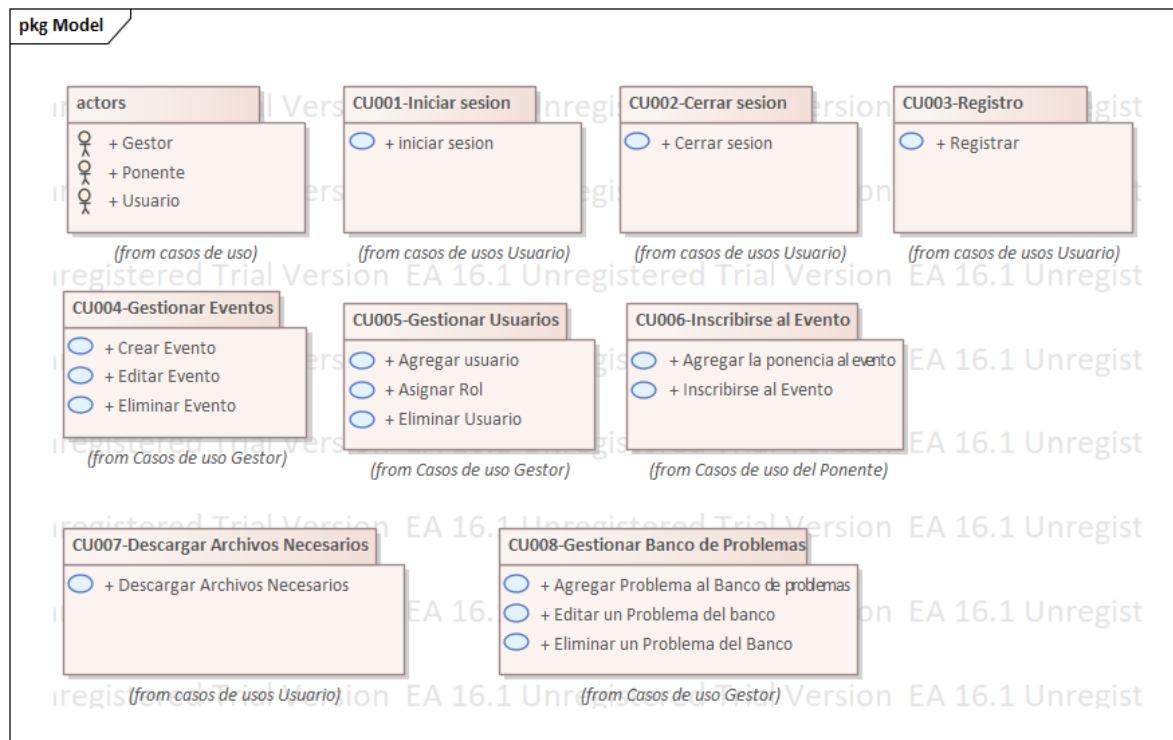


Diagrama 1. Paquete de casos de usos.

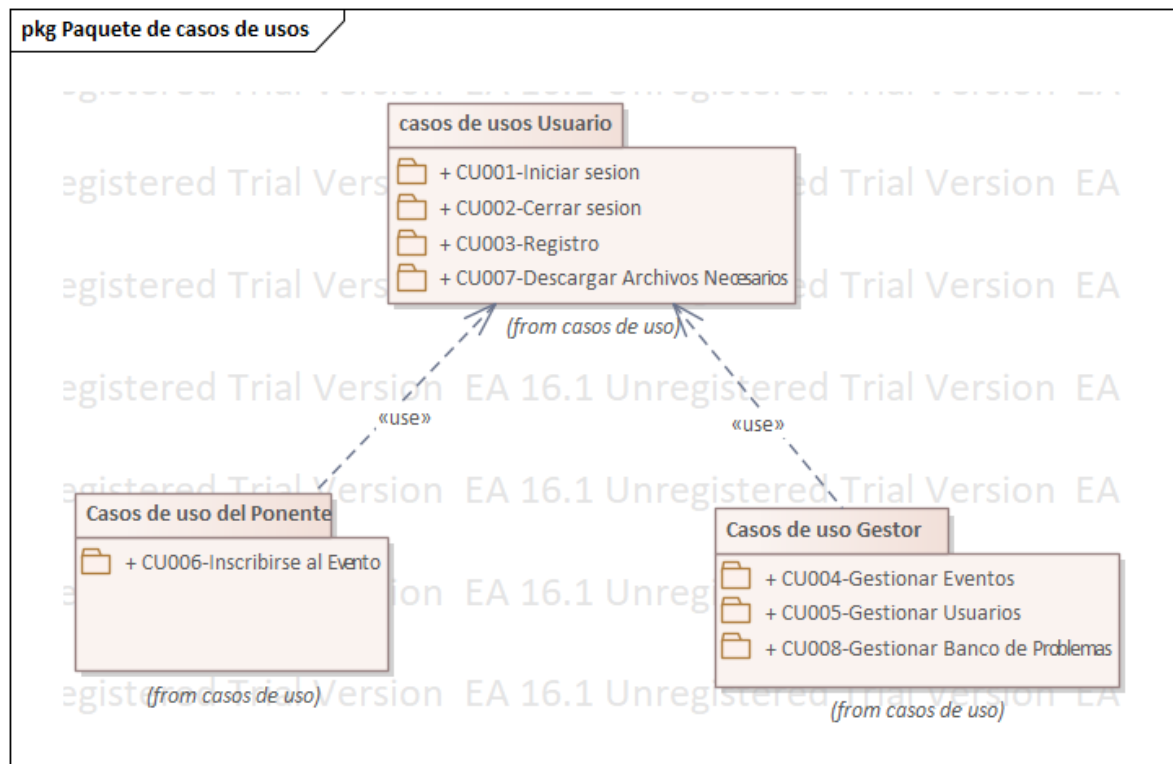


Diagrama 2. Paquete de casos de usos por Actores.

Actores del sistema:

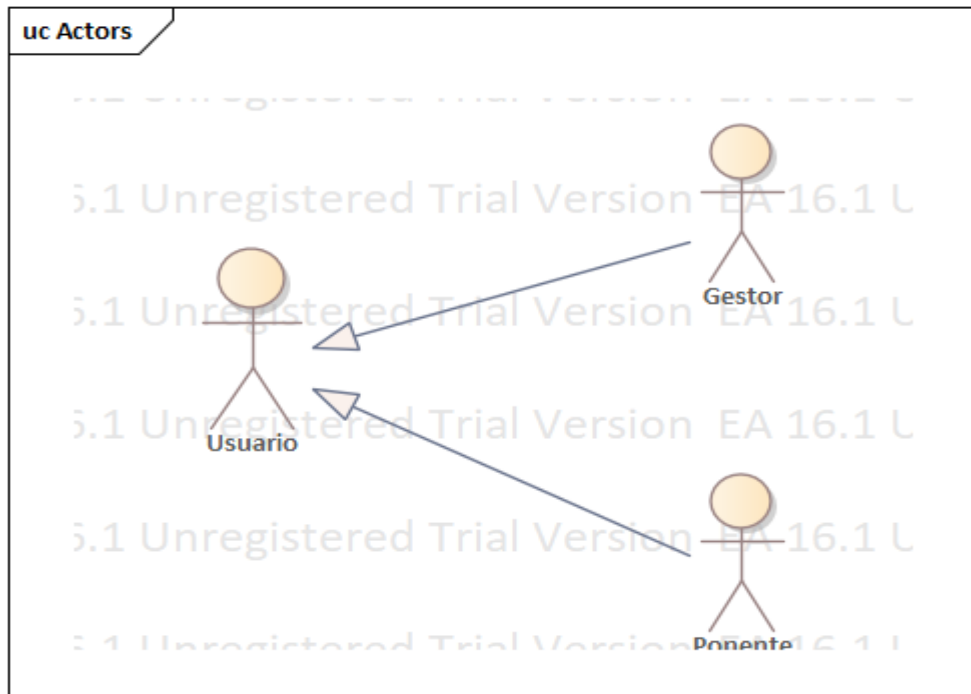


Diagrama 3. Modelo de actores del sistema.

A continuación, se muestran los diagramas de casos de usos para cada rol del sistema:

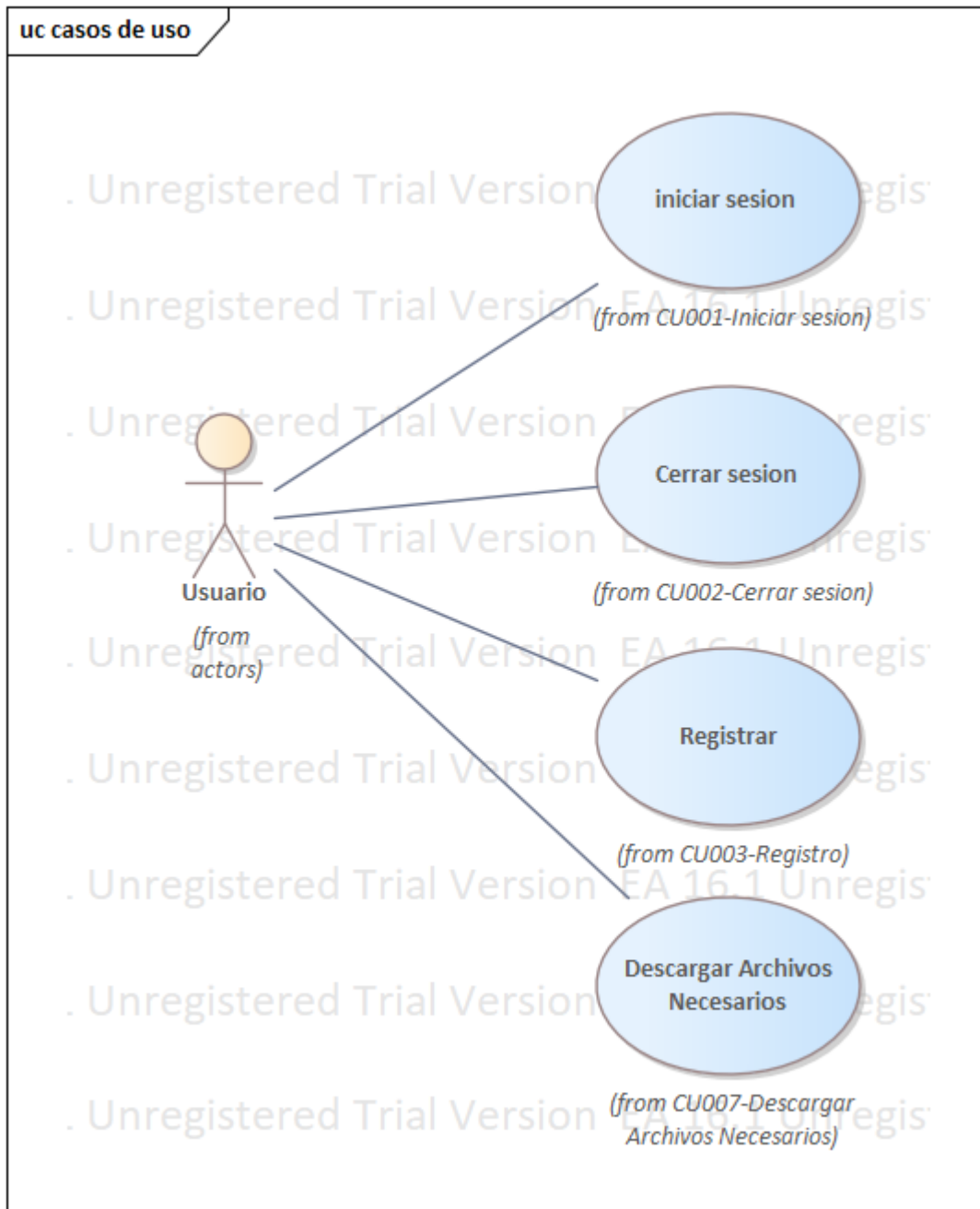


Diagrama 4. Casos de usos de un Usuario.

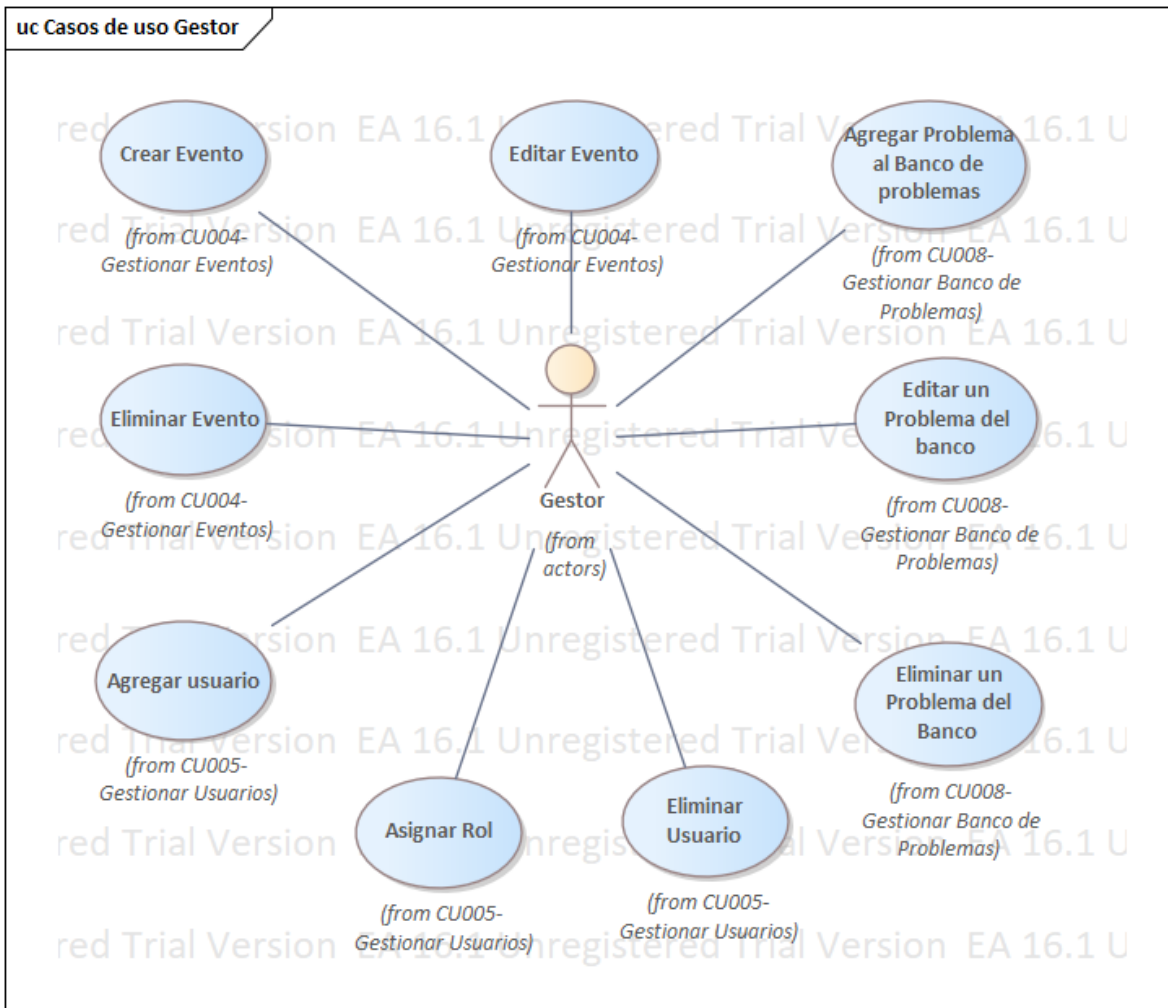


Diagrama 5. Casos de usos del Gestor.

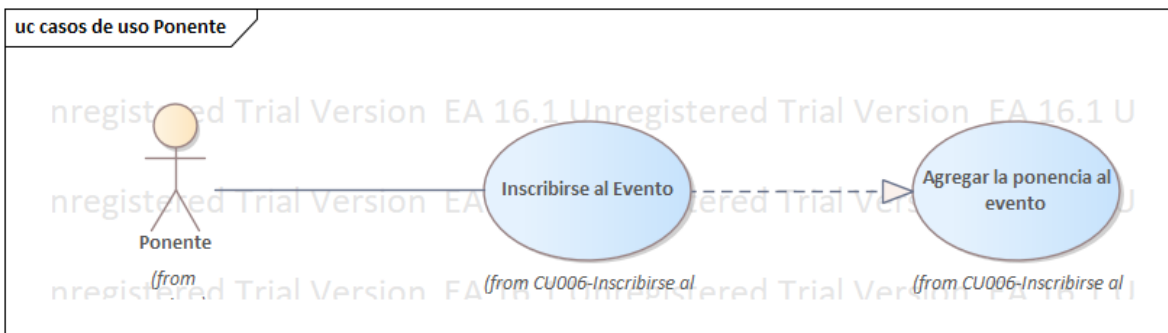


Diagrama 6. Casos de usos del Ponente.

2.4-Paquetes y sus relaciones

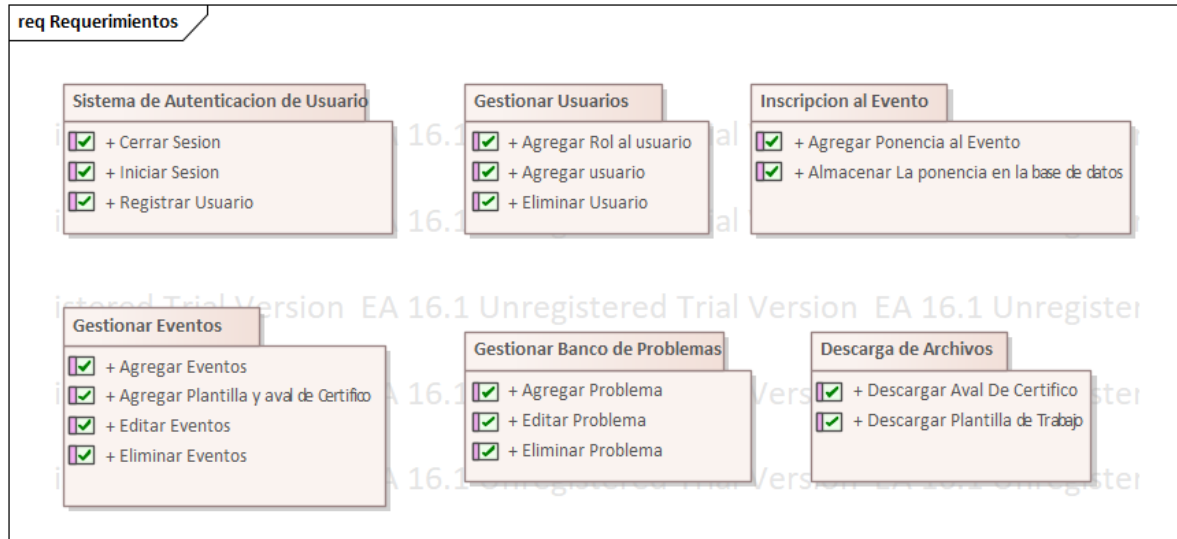


Diagrama 7. Paquete de Requerimientos

2.5 Descripción de los casos de uso

La labor de construir casos de uso, se basa principalmente en la identificación del texto que los describe y su posterior refinamiento para especificar el comportamiento de un sistema. Por ende, en este subepígrafe se realiza la descripción textual de los principales casos de uso y su relación con los actores del sistema.

La metodología ICONIX recomienda para la descripción textual una serie de instrucciones. Se debe aplicar la regla de los dos párrafos, un curso básico que muestra el camino óptimo del proceso y un curso alternativo para describir las verificaciones que realiza el sistema o los posibles errores que pueden ocurrir. Se deben escribir los casos de uso en voz activa con oraciones bien estructuradas utilizando sustantivos y verbos y hacer uso del flujo evento respuesta el cual refleja una acción del actor y la correspondiente respuesta del sistema.

TABLA 1. DESCRIPCIÓN DEL CASO DE USO < INICIAR SESIÓN >

Nombre del caso de uso	CU001-Iniciar sesión
Actores	Usuario
Resumen	El caso de uso inicia cuando el usuario desea iniciar en sesión.
Precondiciones	El usuario debe estar registrado en el sistema.
Poscondiciones	El usuario accede a su sesión de trabajo en la web.

TABLA 2. DESCRIPCIÓN DEL CASO DE USO < CERRAR SESIÓN >

Nombre del caso de uso	CU002-Cerrar sesión
Actores	Usuario
Resumen	Cierra la sesión.
Precondiciones	EL usuario debe haber iniciado sesión anteriormente.
Poscondiciones	Cierra la sesión de trabajo.

TABLA 3. DESCRIPCIÓN DEL CASO DE USO < REGISTRO >

Nombre del caso de uso	CU003-Registro
Actores	Usuario
Resumen	Te registras a la web como rol de ponente
Precondiciones	EL usuario no debe estar registrado en la web.
Poscondiciones	Crea una sesión de trabajo para un usuario con rol de ponente

TABLA 4. DESCRIPCIÓN DEL CASO DE USO < GESTIONAR USUARIOS >

Nombre del caso de uso CU004-Gestionar Usuarios	
Actores	Gestor
Resumen	Permite asignar roles, agregar usuarios y eliminar usuarios.
Precondiciones	Debe estar autenticado en el sistema.
Poscondiciones	Elimina usuarios de la base de datos de forma permanente, asigna roles a usuarios registrados y permite agregar usuarios nuevos a la base de datos.

TABLA 5. DESCRIPCIÓN DEL CASO DE USO < GESTIONAR EVENTOS >

Nombre del caso de uso CU005-Gestionar Eventos	
Actores	Gestor
Resumen	Permite crear nuevos eventos, modificarlos y eliminarlos.
Precondiciones	Debe estar autenticado en el sistema.
Poscondiciones	Inicia un nuevo Evento a decisión del Gestor, Permite modificarlo y eliminarlo de ser necesario.

TABLA 6. DESCRIPCIÓN DEL CASO DE USO < INSCRIBIRSE AL EVENTO >

Nombre del caso de uso CU006-Inscribirse al Evento	
Actores	Ponente
Resumen	Permite a un usuario registrarse a un evento.
Precondiciones	Debe estar autenticado en el sistema, el evento debe estar creado con anterioridad.
Poscondiciones	El sistema guarda la información introducida y almacena la ponencia en la base de datos.

TABLA 7. DESCRIPCIÓN DEL CASO DE USO < DESCARGAR ARCHIVOS NECESARIOS >

Nombre del caso de uso	CU007-Descargar Archivos Necesarios
Actores	Usuario
Resumen	Permite descargar los archivos relacionados con los eventos
Precondiciones	
Poscondiciones	Descarga los documentos relacionados al evento

TABLA 8. DESCRIPCIÓN DEL CASO DE USO < GESTIONAR BANCO DE PROBLEMAS >

Nombre del caso de uso	CU008-Gestionar Banco de Problemas
Actores	Gestor
Resumen	Permite agregar problemas al banco de problemas y poder modificarlos y eliminarlos posteriormente.
Precondiciones	Estar autenticado en el sistema.
Poscondiciones	Agregar, Modificar y Eliminar problemas en el banco de Problemas.

2.6-Análisis de Robustez

El análisis de robustez facilita el proceso de refinamiento de las descripciones textuales de los casos de uso y el descubrimiento de nuevos objetos en el modelo del dominio. Estos objetos forman parte de los diagramas de robustez y se clasifican de la siguiente manera:

- Objetos interfaz: usados por los actores para comunicarse con el sistema, generalmente son: menús, pantallas, diálogos y ventanas.
- Objetos entidad: usualmente son objetos del modelo del dominio, pueden ser tablas de una base de datos o ficheros que contienen toda la información que necesita permanecer almacenada después de la ejecución de un caso de eso.
- Objetos de control (controladoras): agrupan la lógica de la aplicación, son intermediarios entre la interfaz y los objetos entidad.

A la hora de realizar diagramas de robustez se deben tener en cuenta algunas reglas básicas:

- Los actores solo pueden comunicarse con objetos interfaz.
- Las interfaces solo con controladoras y actores.
- Los objetos entidad solo con las controladoras.
- Por último, las controladoras se comunican con todos menos con los actores.

Un diagrama de robustez representa y asegura que se cubra el camino básico y los posibles caminos alternos que corresponden a cada caso de uso. Son la primera vista preliminar del nuevo sistema.[16]

A continuación, se muestran los diagramas de robustez de cada caso de uso:

CU001-Iniciar sesión:

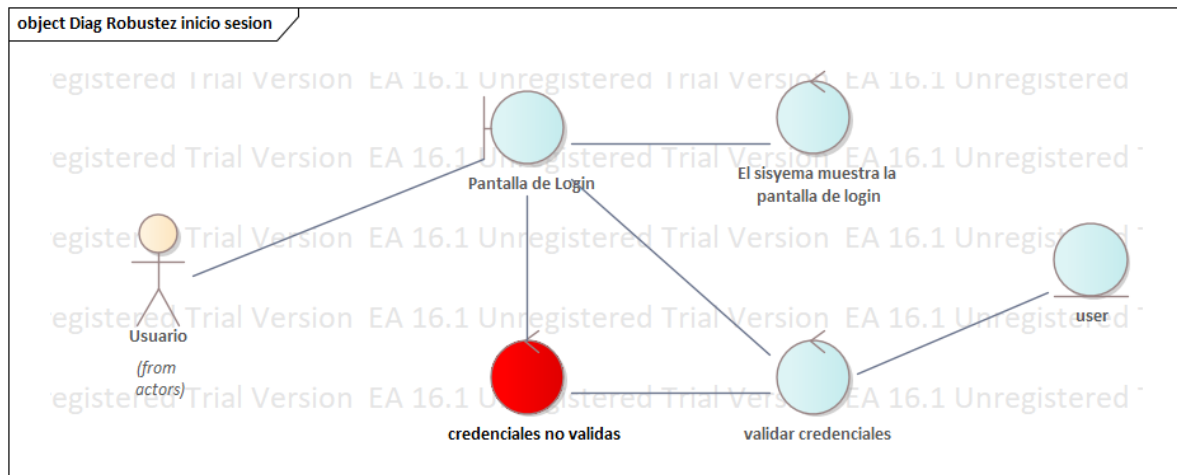


Diagrama 5. Diagrama de robustez del caso de uso Iniciar sesión.

CU002-Cerrar sesión

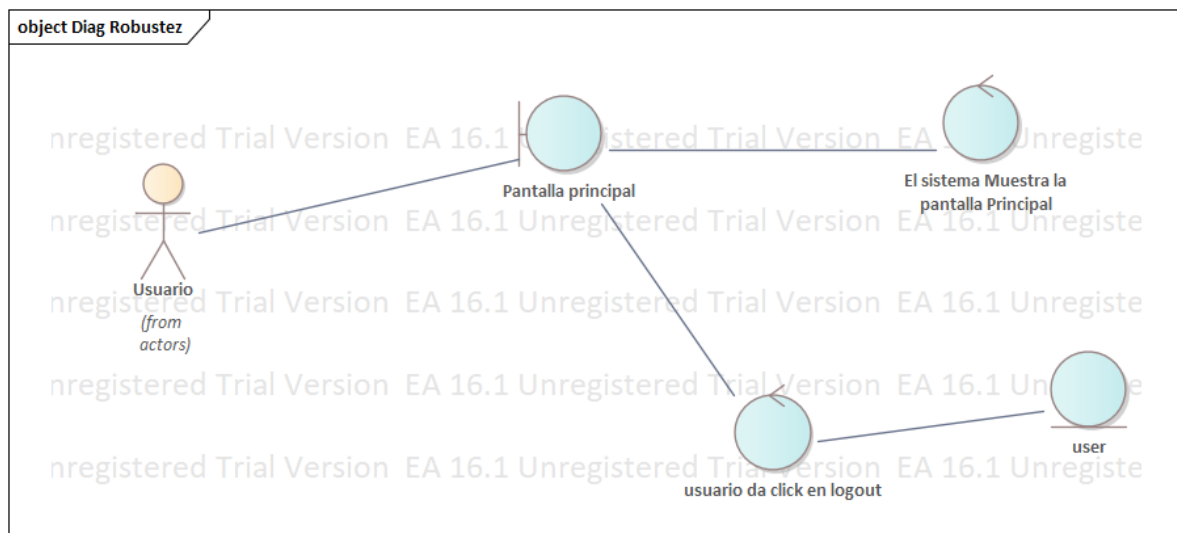


Diagrama 6. Diagrama de robustez del caso de uso Cerrar sesión.

CU003-Registro

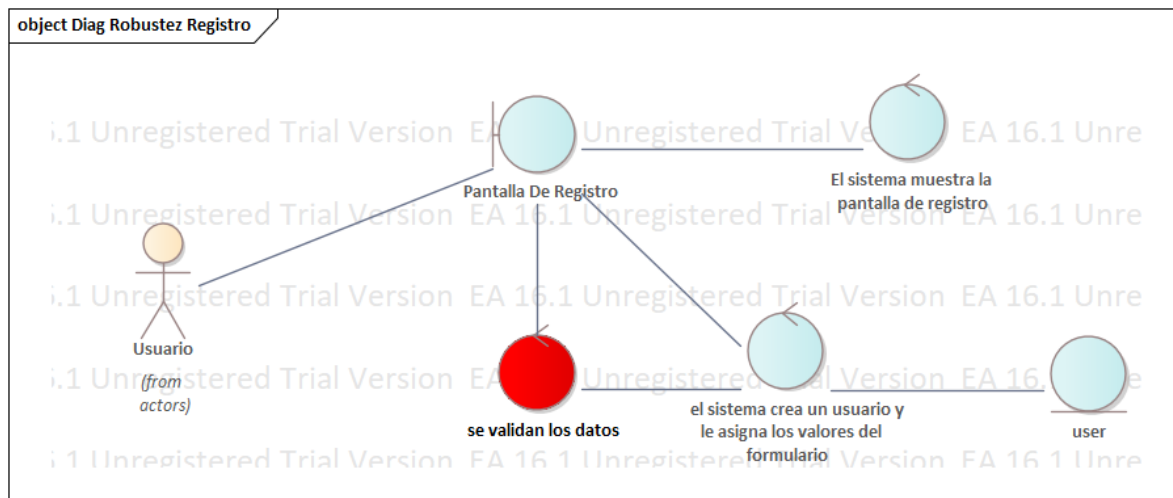


Diagrama 7. Diagrama de robustez del caso de uso Registrar.

CU004-Gestionar Usuarios

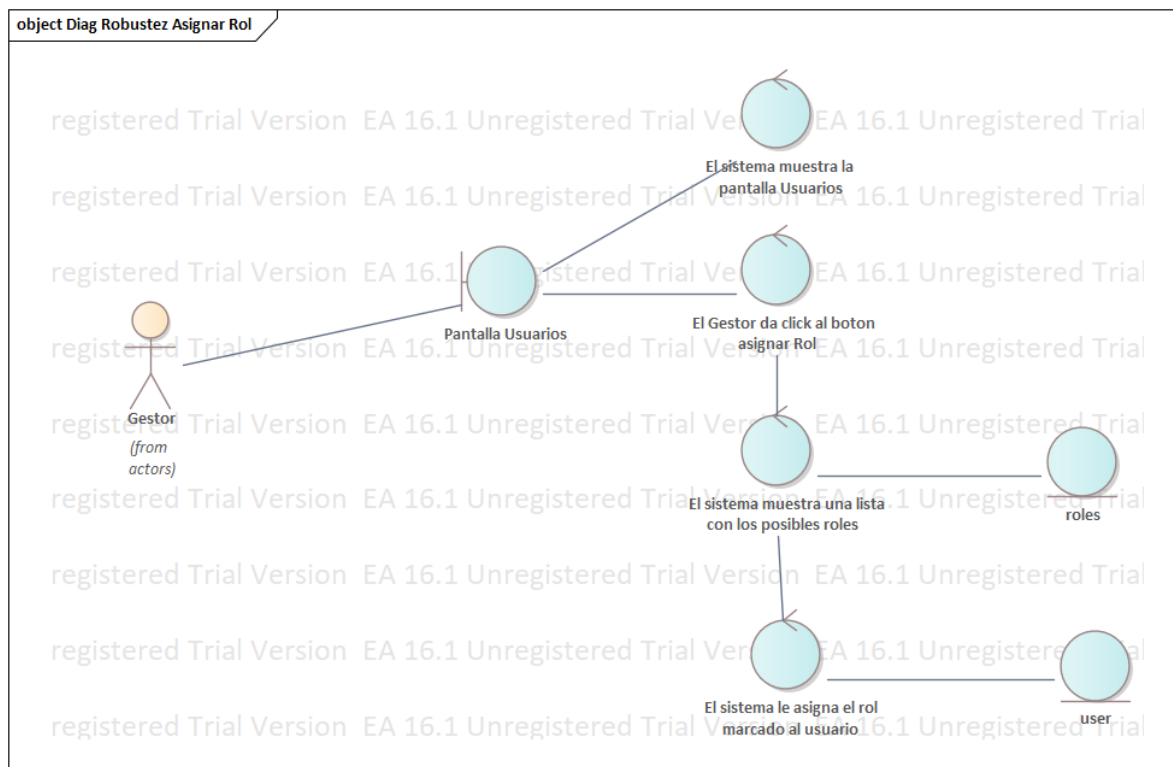


Diagrama 8. Diagrama de robustez del caso de uso Asignar Rol.

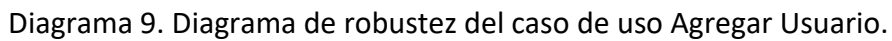


Diagrama 9. Diagrama de robustez del caso de uso Agregar Usuario.

CU005-Gestionar Eventos

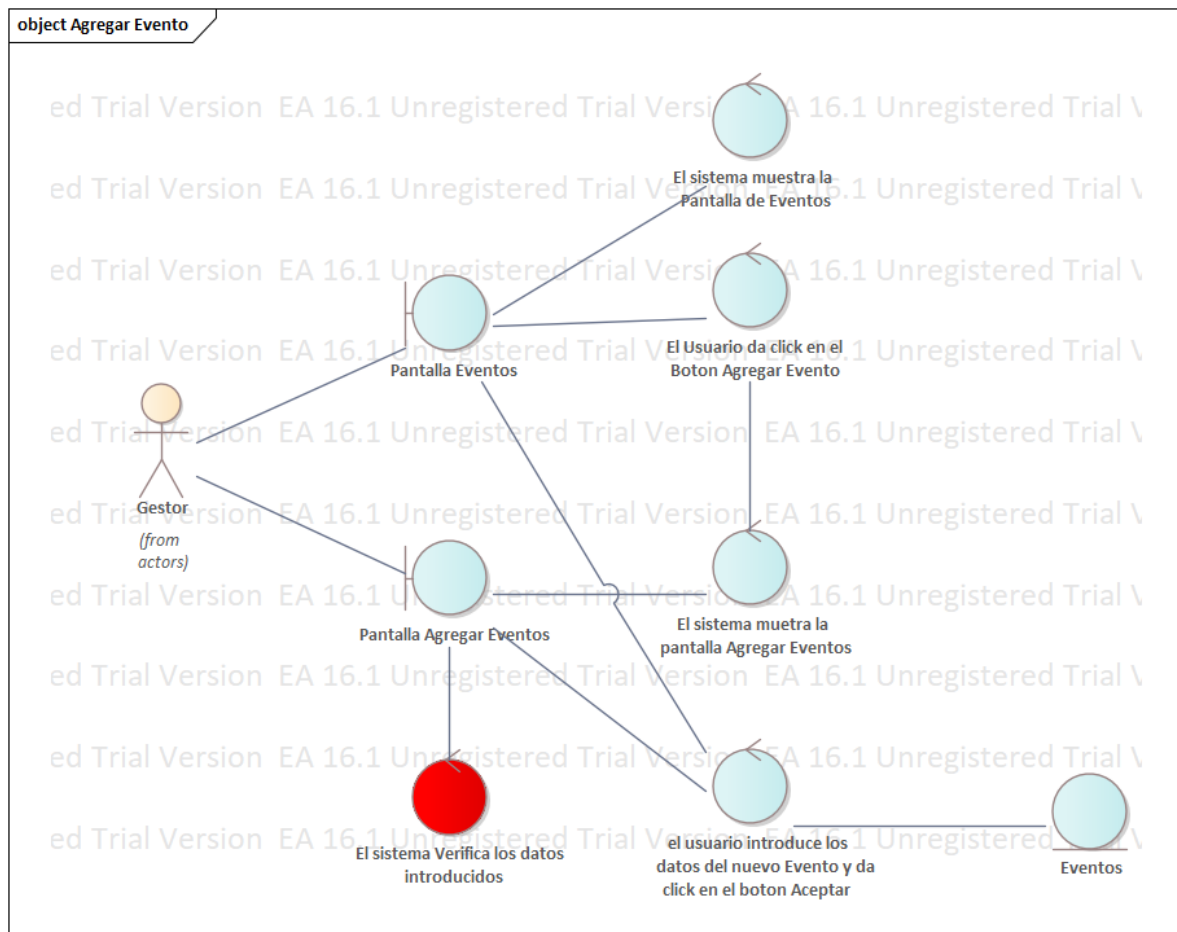


Diagrama 11. Diagrama de robustez del caso de uso Agregar Evento.

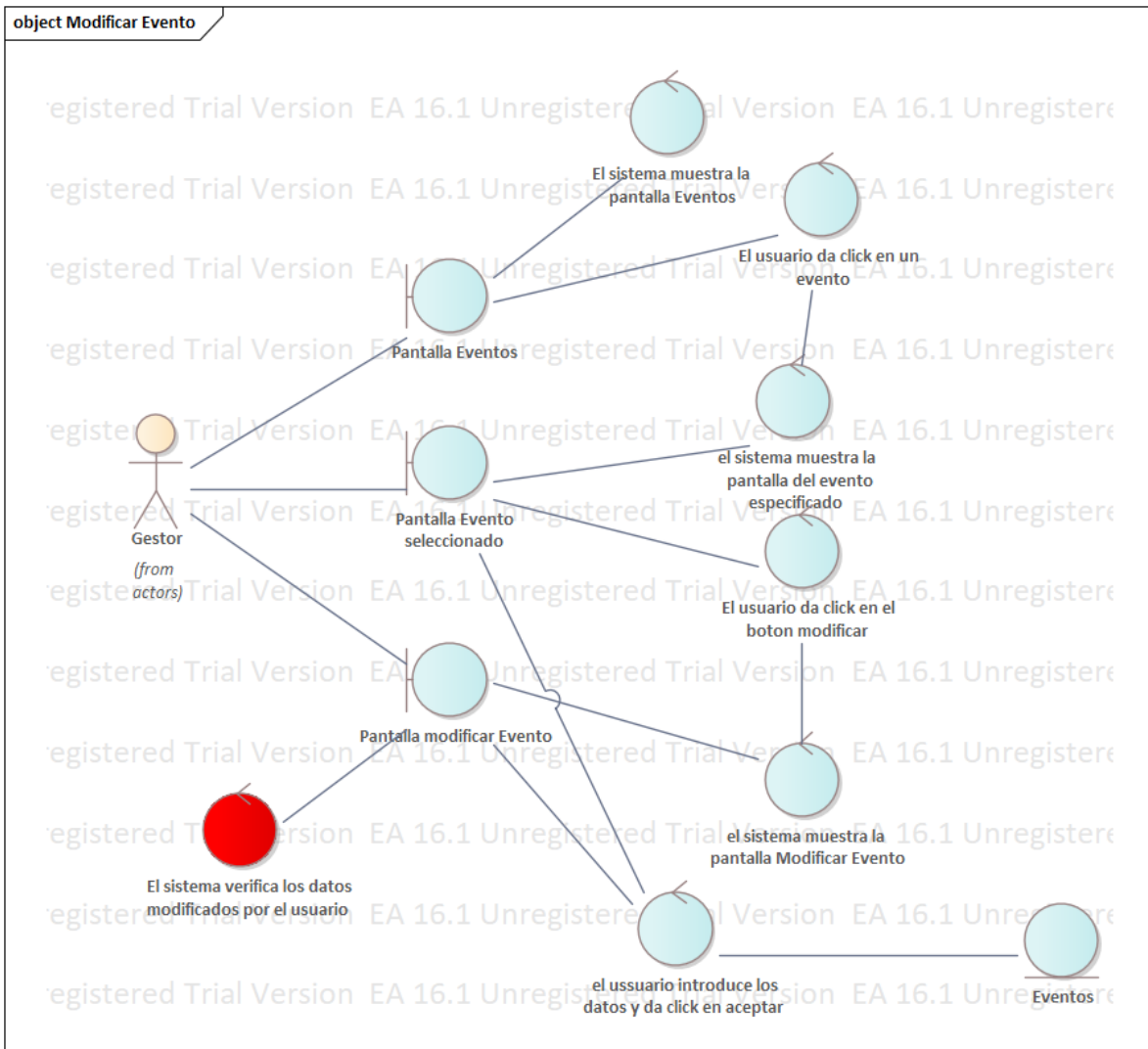


Diagrama 12. Diagrama de robustez del caso de uso Modificar Evento.

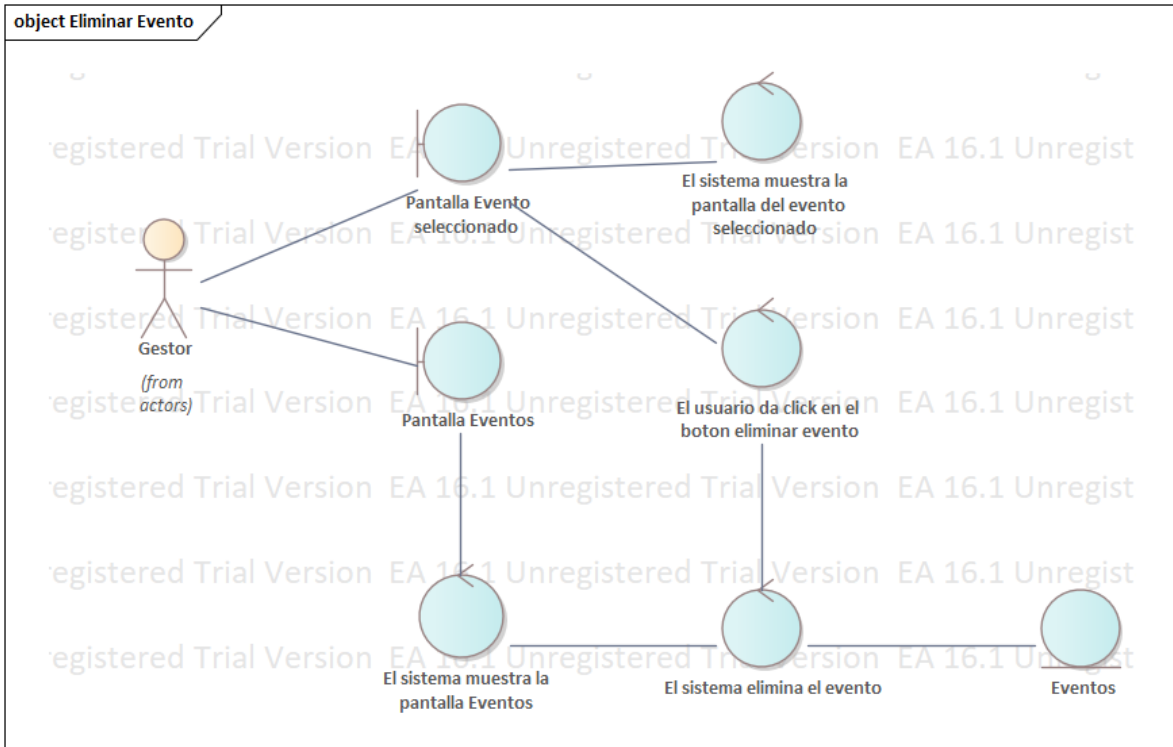


Diagrama 13. Diagrama de robustez del caso de uso Eliminar Evento.

CU006-Inscribirse al Evento

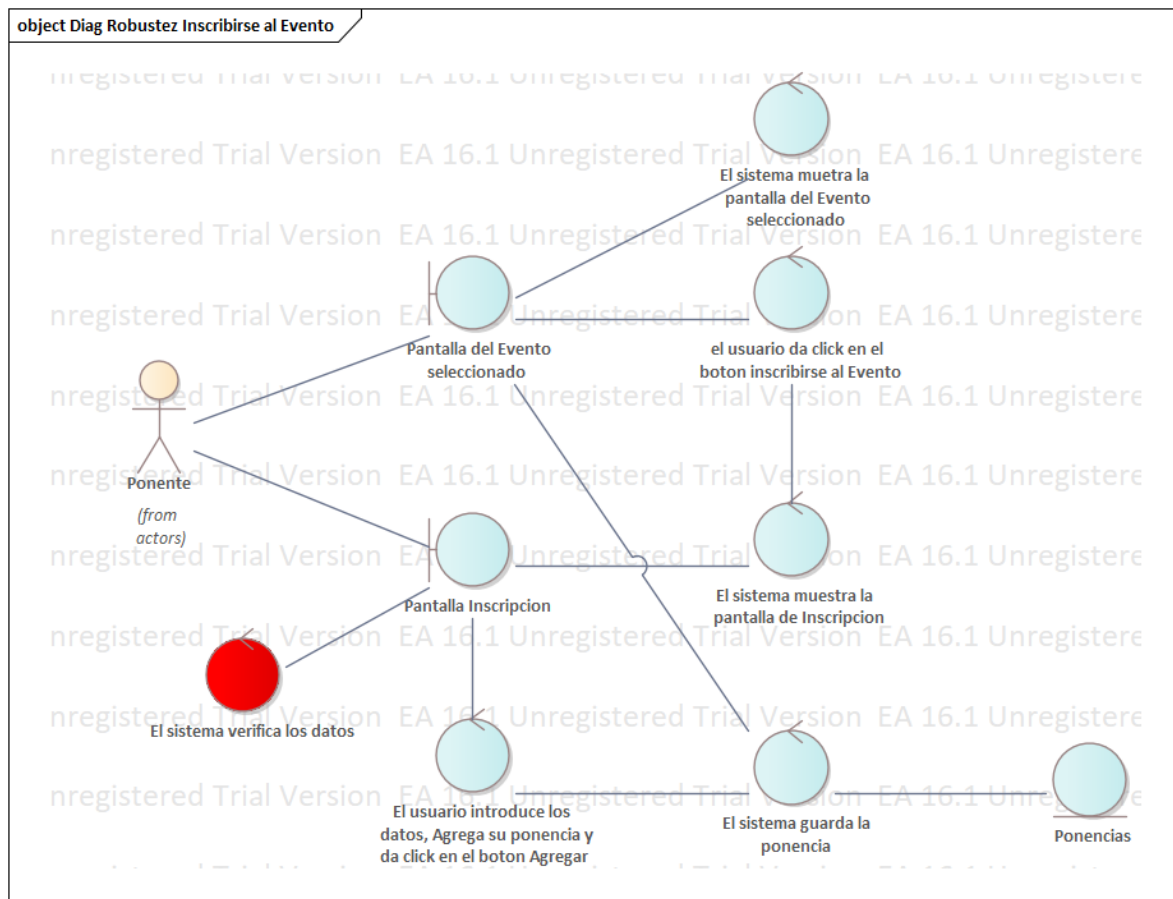


Diagrama 14. Diagrama de robustez del caso de uso Inscribirse Evento.

CU007-Descargar Archivos Necesarios

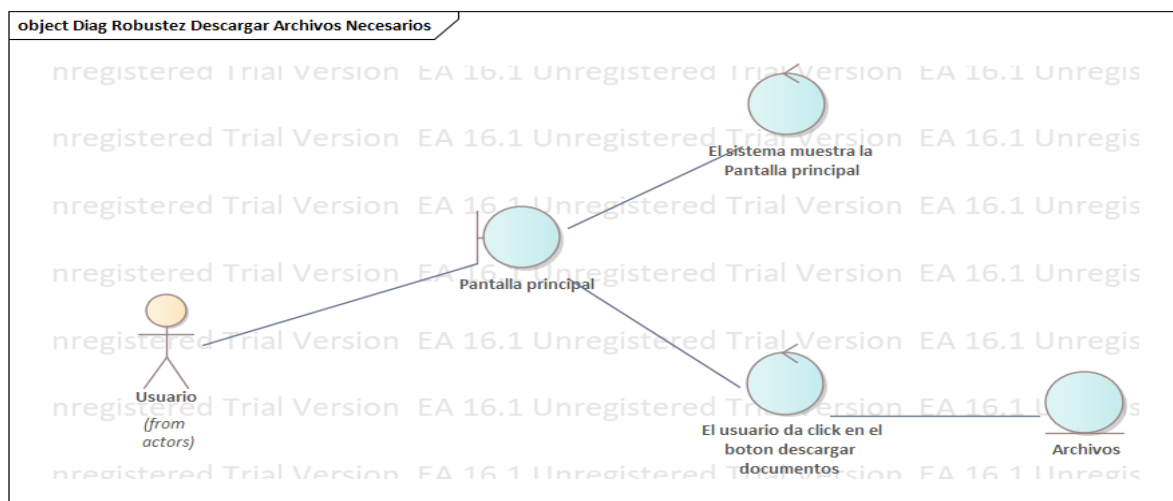


Diagrama 15. Diagrama de robustez del caso de uso Descargar Archivos.

CU008-Gestionar Banco de Problemas

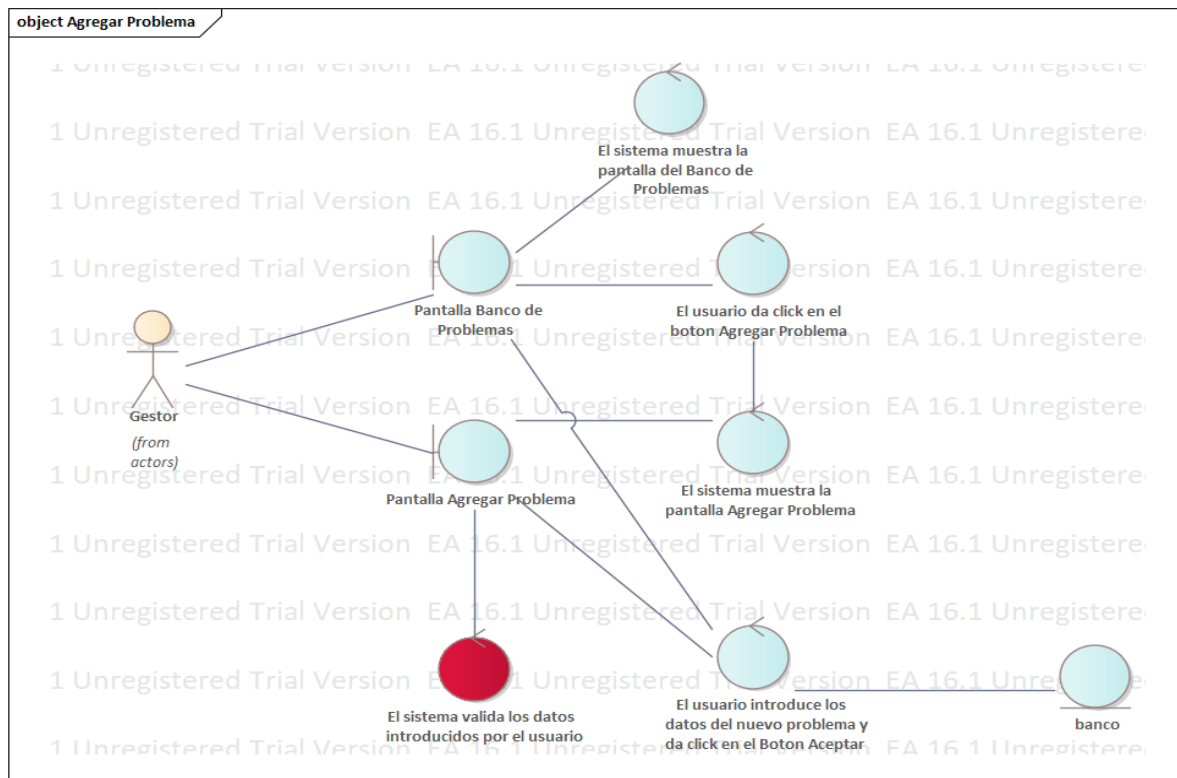


Diagrama 16. Diagrama de robustez del caso de uso Agregar Problema.

object Editar Problema

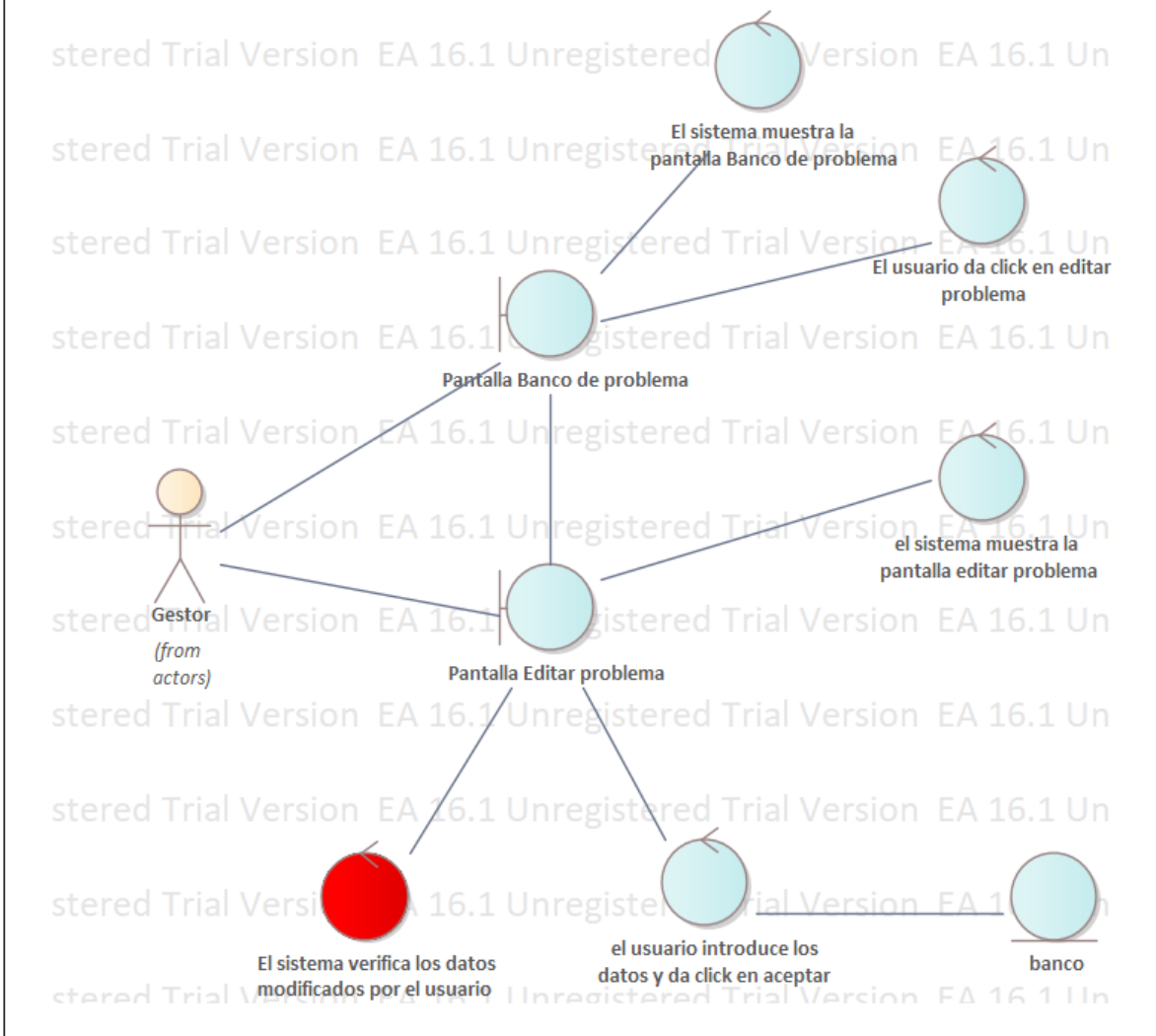


Diagrama 17. Diagrama de robustez del caso de uso Editar Problema.

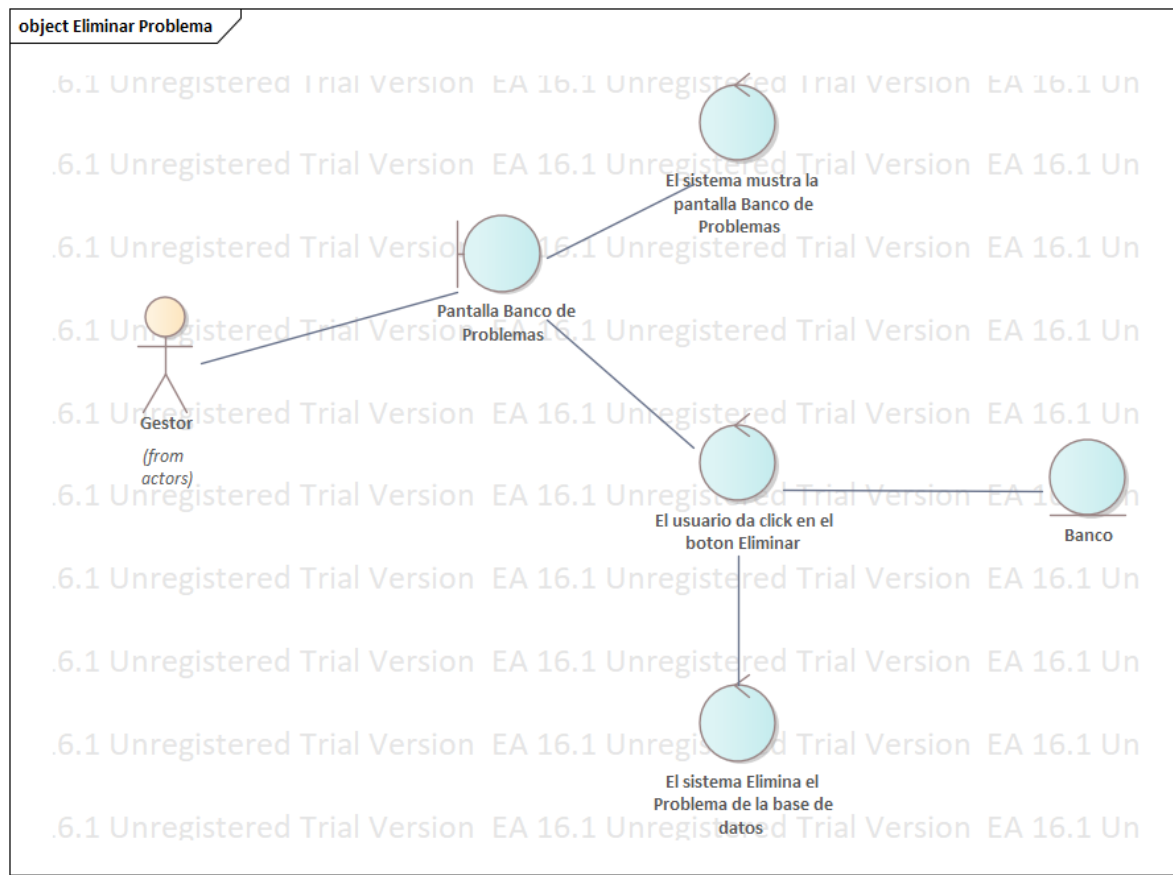


Diagrama 18. Diagrama de robustez del caso de uso Editar Problema.

2.7-Arquitectura técnica

El análisis de robustez permite ir llenando la brecha entre el análisis y el diseño. Una vez culminada la revisión del diseño preliminar, se sugiere comenzar a formular la arquitectura técnica, o arquitectura del sistema, o arquitectura de software, la cual se puede ir relacionando con el framework específico que se vaya a utilizar.

deployment Arquitectura

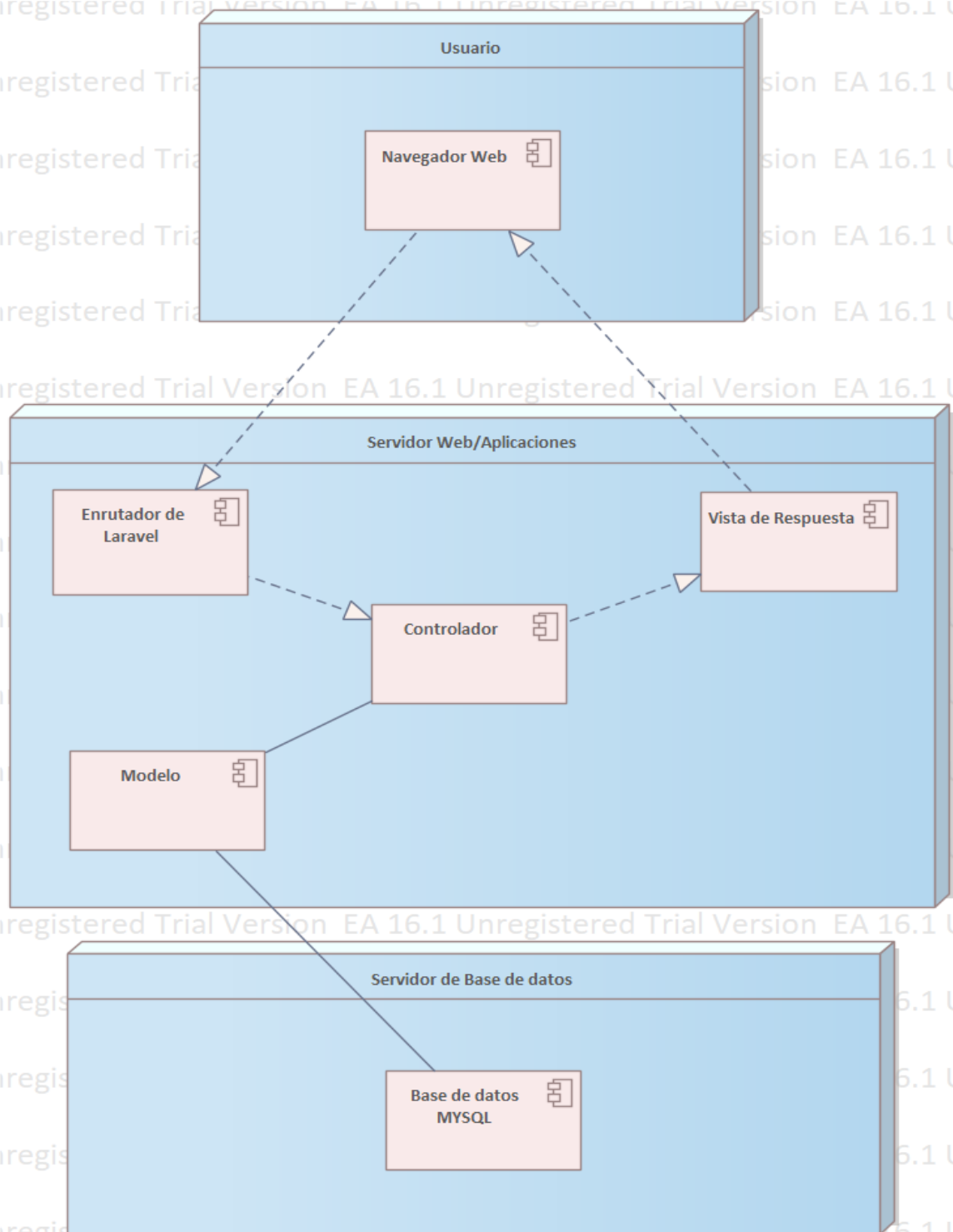


Diagrama 19. Arquitectura

La utilización del framework Laravel impone cierta estructura a la aplicación Web. El diagrama de la arquitectura propuesta refleja, básicamente, el flujo de procesamiento de una petición del cliente hasta la elaboración de una respuesta por el servidor. El procesamiento de una petición ocurre de la siguiente manera en Laravel. La petición que proviene del cliente es enrutada hacia un determinado controlador. El enrutador utiliza los patrones descritos en los ficheros web.php, que establecen la relación entre las urls que se reciben y los controladores que las procesan. El controlador procesa la petición que recibe interactuando, posiblemente, con los modelos. Los modelos son definidos, generalmente, en la carpeta Models que se encuentra dentro de App\Html. Como resultado final, el controlador elabora una respuesta que se envía hacia el cliente mediante una vista. Laravel se encarga de crear las tablas de la base de datos en el sistema de gestión de bases de datos seleccionado mediante las migraciones creadas y ejecutadas previamente.

2.8-Conclusiones

En este capítulo se ha descrito la propuesta de sistema mediante sus requerimientos funcionales y no funcionales. A partir de estos requerimientos se pudo dar paso al modelado del sistema, identificando los actores del sistema y los casos de usos existentes. Se realizó una descripción detallada de cada caso de uso del sistema y se mostró sus respectivos diagramas de robustez. Se pudo observar el diagrama de la Arquitectura de la web regida por el framework utilizado. Con todo esto se logró plasmar detalladamente la forma en que funciona todo el sistema.

Capítulo III – Descripción de la solución propuesta

3.1-Introducción

En este capítulo se presenta el diseño detallado donde se visualiza el diagrama de clases del diseño y sus principios, tratamiento de errores y el análisis de factibilidad de la propuesta. En el desarrollo de un software es recomendable realizar la estimación antes del análisis, diseño, implementación y prueba, para tener una idea de si el software es o no factible; esto tiene como finalidad indicar los objetivos, alcances, restricciones y disponibilidad de los recursos necesarios para lograr dichos objetivos.

3.2-Diagrama de clases del diseño

En este capítulo se muestran las clases obtenidas durante la fase de diseño del sistema modelado y analizado anteriormente.

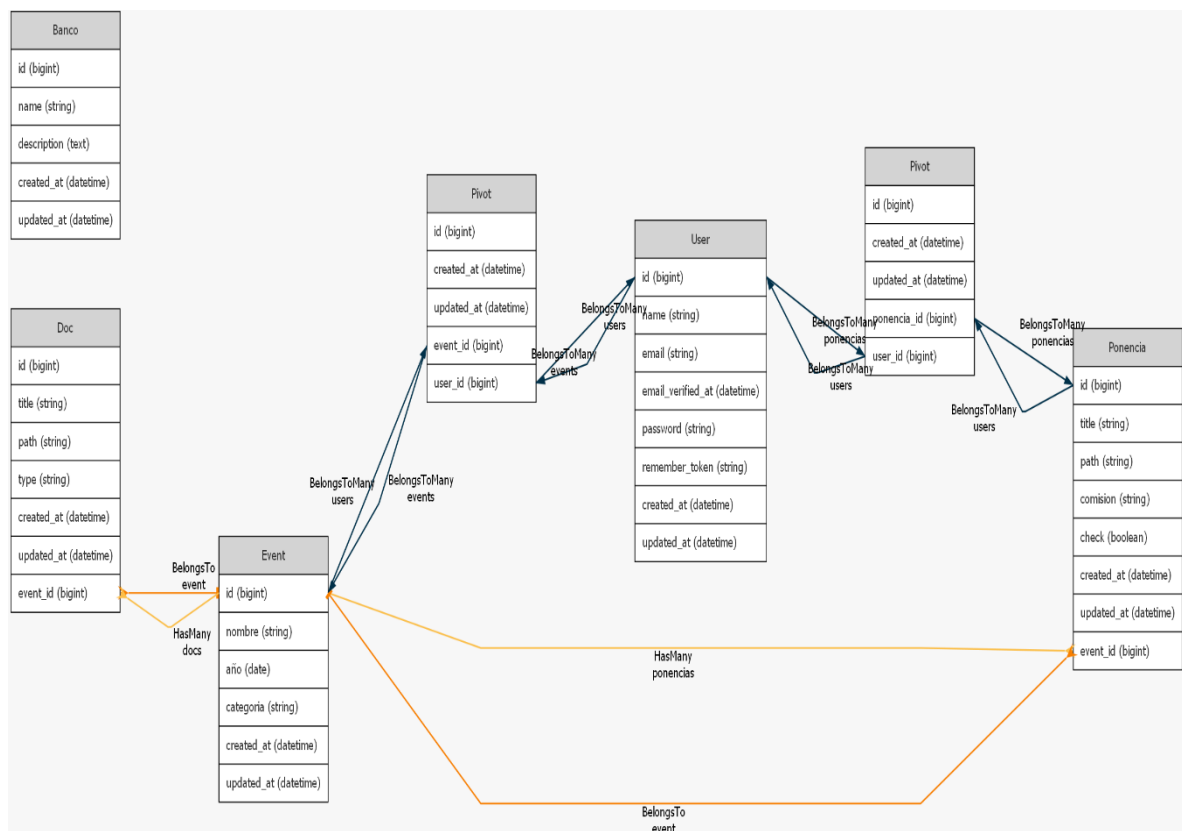


Diagrama 20. Diagrama de clases de diseño.

3.3-Principios de diseño

- La interfaz cuenta con un diseño simple.
- La web dispone de:
 - 1-Una barra de navegación.
 - 2-Varias Tablas que proporcionan la información requerida.
 - 3-Los formularios necesarios para introducir los datos a la base de datos.
- Se utilizo el framework de diseño Tailwind CSS como herramienta para el desarrollo frontend.

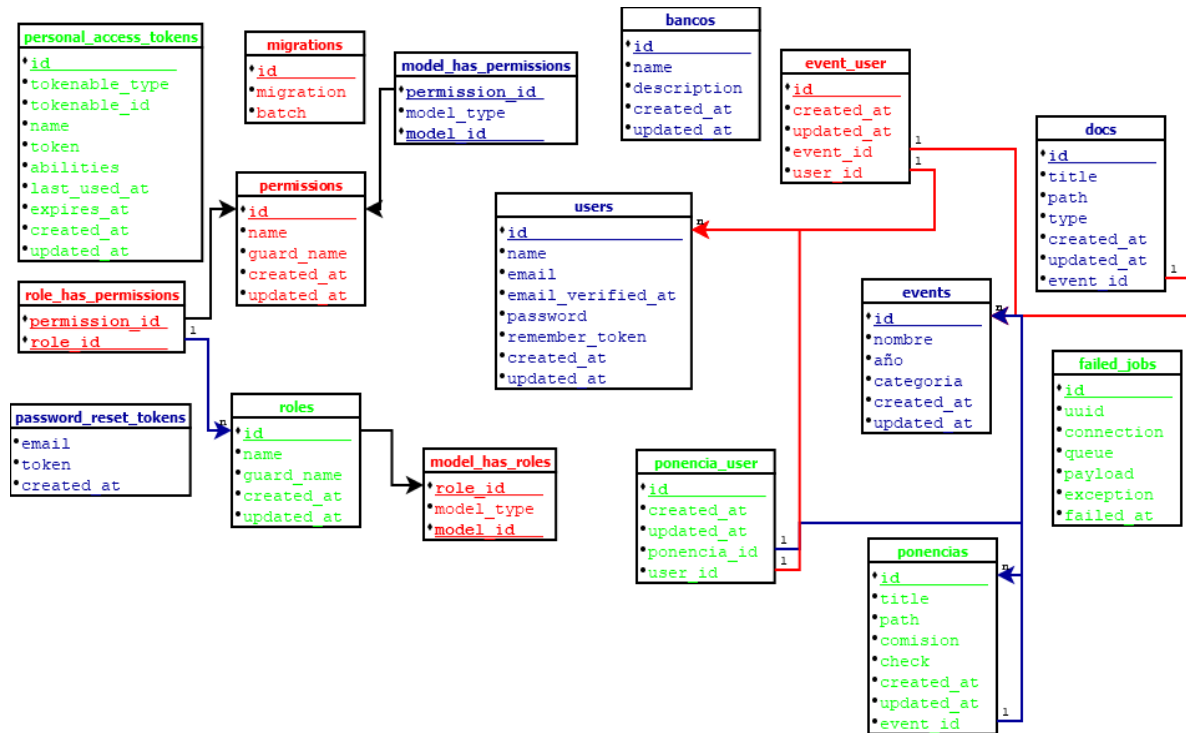
3.4-Tratamiento de errores

El sistema propuesto presenta una interfaz diseñada, implementada y dirigida a manejar las excepciones y errores. El mismo tiene la obligación de detectar problemas en el proceso de autenticación por parte del usuario y en el control de acceso a las funcionalidades.

El sistema utiliza los mecanismos de validación de la información que ofrece el framework Laravel, además de utilizar la validación de los datos directamente en el navegador como una característica extra de usabilidad, con el propósito de minimizar la entrada de información errónea, e indicarle al usuario el tipo de información que debe manipular. El sistema está diseñado para que el usuario escriba solo lo necesario con el fin de disminuir el margen de error.

3.5-Diseño de la base de datos

La base de datos se gestiona a través del gestor de base de datos MySQL, la misma aun no se encuentra Implementada en el servidor de Etecsa.



3.6-Diagrama de despliegue

Representación gráfica del diagrama. Explicar la estructura de nodos que se presentan. Indicar que arquitectura capas se va implementar y su correspondencia con los nodos.

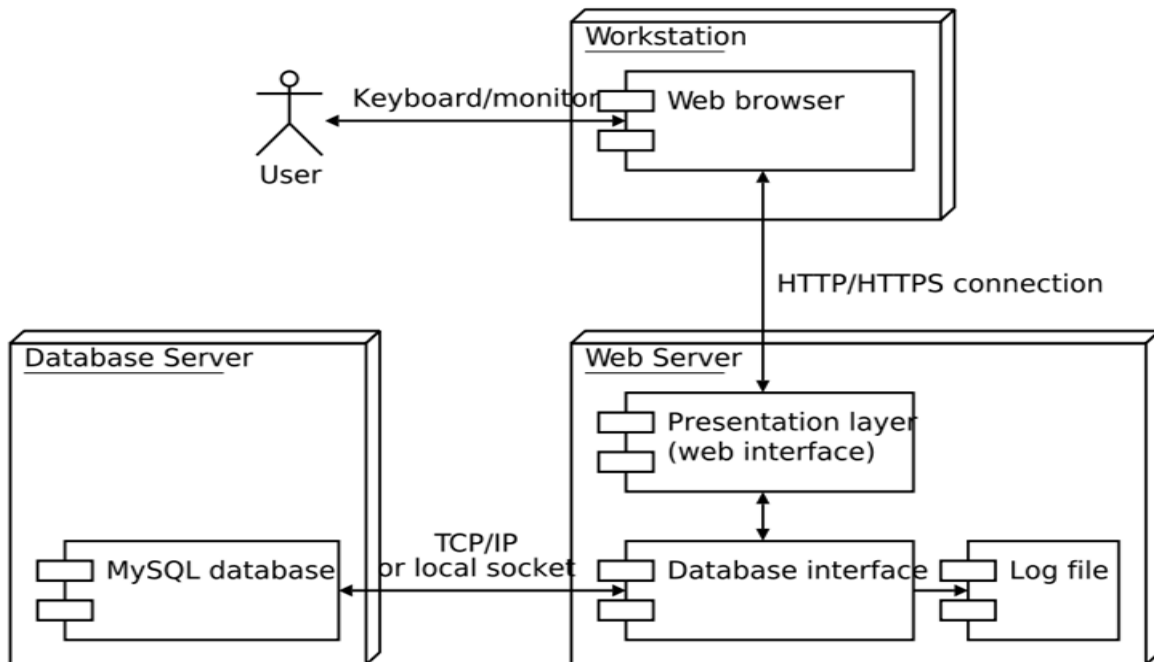


FIGURA 1. DIAGRAMA DE DESPLIEGUE

3.7-Estimación por casos de uso

Existen varias alternativas posibles para la estimación del esfuerzo basada en caso de uso: Análisis de puntos de función y análisis de puntos de casos de uso. En este trabajo se utiliza la estimación mediante el análisis de Puntos de Casos de Uso es un método propuesto originalmente por Gustav Karner de Objectory AB, y posteriormente refinado por muchos otros autores.

El análisis de Puntos de Casos de Uso es un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de pesos a varios factores que lo afectan. A partir de los valores asignados a estos factores, y de sus pesos, se estima el tiempo para el desarrollo del proyecto.[17]

3.8-Cálculo de los puntos de casos de uso sin ajustar

El método de punto de casos de uso consta de varias etapas. En primer lugar, se calculan los puntos de casos de uso sin ajustar (UUCP). Para ello, es necesario determinar el factor de peso sin ajustar de los actores (UAW), así el

factor de pesos sin ajustar de los casos de uso (UUCW).

$$UUCP = UAW + UUCW$$

- UUCP: Puntos de Casos de Uso sin ajustar.
- UAW: Factor de peso de los actores sin ajustar.
- UUCW: Factor de peso de los casos de uso sin ajustar.

En la segunda etapa se calculan los puntos de casos de uso ajustados (UCP), según la siguiente fórmula:

$$UCP = UUCP * TCF * EF$$

- UUC: Puntos de Casos de Uso ajustados.
- UUCP: Puntos de Casos de Uso sin ajustar.
- TCF: Factor de complejidad técnica.
- EF: Factor de ambiente

A continuación, se especifican los detalles de cálculos de cada de los elementos que intervienen en los cálculos.

3.8.1-Factor de peso de los actores del sistema

Para el cálculo del factor de peso de los actores del sistema se utiliza la siguiente asignación de pesos:

Tipo De Actor	Descripción	Factor
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API).	1
Medio	Otro sistema interactuando a través de un protocolo (ej. TCP/IP) o una persona interactuando a través de una interfaz en modo texto.	2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica (GUI).	3

Tabla 9. Asignación de peso sin ajustar según el tipo de actor.

En la Tabla 9 se presenta la asignación realizada para los actores del sistema propuesto.

Actor	Tipo de Actor	Factor de Peso
Usuario	Complejo	3
Gestor	Complejo	3
Ponente	Complejo	3

Tabla 9. Factor de peso de los actores del sistema

$$UAW = (Cantidad\ de\ actores) * Peso$$

$$UAW = 3 * 3 = 9$$

3.8.2-Factor de peso sin ajustar de los casos de uso del sistema

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo según la siguiente tabla:

Tipo de caso de uso	Descripción	Factor
Simple	3 transacciones o menos	5
Medio	4 a 7 transacciones	10
Complejo	Más de 7 transacciones	15

Tabla 10. Pesos de los casos de uso

El factor de pesos de casos de uso sin ajustar se calcula como la sumatoria de los pesos asignados a cada de uso. En el sistema propuesto la asignación realizada es la siguiente:

Paquete	Nombre	Complejidad
Usuario	Iniciar Sesión	Simple
Usuario	Registrar	Simple
Usuario	Cerrar Sesión	Simple
Usuario	Descargar Archivos	Simple
Gestor	Crear Evento	Medio
Gestor	Editar Evento	Simple
Gestor	Eliminar Evento	Simple
Gestor	Asignar Rol	Simple
Gestor	Agregar Usuario	Simple
Gestor	Eliminar Usuario	Simple
Gestor	Agregar Problema al Banco de problemas	Simple

Gestor	Editar un Problema del banco	Simple
Gestor	Eliminar un Problema del banco	Simple
Ponente	Inscribirse al evento	Simple

Tabla 11. Complejidad de los casos de uso del sistema

El sistema propuesto incluye 14 casos de uso, de ellos, 13 simples, 1 medio.

Factor de peso de los casos de uso sin ajustar(UUCW) = 75

3.8.3-Factores de complejidad técnica

El factor de complejidad técnica se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. A continuación, se muestra una tabla donde se encuentran los valores y el peso utilizados.

Métrica	Descripción	Peso	Valor	TCF
TCF01	Sistema distribuido	2,00	0,00	0,00
TCF02	Objetivos de performance o tiempo de respuesta	1,00	5,00	5,00
TCF03	Eficiencia del usuario final	1,00	5,00	5,00
TCF04	Procesamiento interno complejo	1,00	3,00	3,00
TCF05	El código debe ser reutilizable	1,00	5,00	5,00
TCF06	Facilidad de instalación	0,50	5,00	2,50
TCF07	Facilidad de uso	0,50	5,00	2,50
TCF08	Portabilidad	2,00	5,00	10,00
TCF09	Facilidad de cambio	1,00	5,00	5,00
TCF10	Concurrencia	1,00	5,00	5,00
TCF11	Incluye objetivos especiales de seguridad	1,00	3,00	3,00
TCF12	Provee acceso directo a terceras partes	1,00	2,00	2,00
TCF13	Se requieren facilidades especiales de entrenamiento a los usuarios	1,00	1,00	1,00
			Total:	49,00

Tabla 12. Factores de complejidad técnica

3.8.4-Factores ambientales

Los factores ambientales tienen que ver con las habilidades y el entrenamiento del grupo involucrado en el desarrollo. El cálculo del mismo es similar al cálculo del factor de complejidad técnica, es decir, se trata de un conjunto de factores que se cuantifican con valores de 0 a 5.

En la Tabla 13 se muestran los valores y pesos utilizados.

Métrica	Descripción	Peso	Valor	TCF
ECF01	Familiaridad con el modelo de proyecto utilizado	1,50	5,00	7,50
ECF02	Experiencia en la aplicación	0,50	3,00	1,50
ECF03	Experiencia en orientación a objetos	1,00	4,00	4,00
ECF04	Capacidad del analista líder	0,50	5,00	2,50
ECF05	Motivación	1,00	5,00	5,00
ECF06	Estabilidad de los requerimientos	2,00	5,00	10,00
ECF07	Personal part-time	-1,00	2,00	-2,00
ECF08	Dificultad del lenguaje de programación	-1,00	3,00	-3,00
			Total:	26,50

Tabla 13. Factores de ambiente

3.9-Estimación del esfuerzo

El esfuerzo en horas-hombre viene dado por:

$$E = UCP * CF$$

- E: Esfuerzo en horas – hombre
- UCP: Puntos de casos de uso ajustados
- CF: Factor de conversión.

El factor de conversión toma los valores 20 o 28 según las reglas descritas. En nuestro caso, el factor de conversión asumido fue de 22, según los valores de los factores de ambiente.

Se debe tener en cuenta que este método proporciona una estimación del esfuerzo en horas-hombre contemplando sólo el desarrollo de la funcionalidad especificada en los casos de uso.

3.9.1-Resultados de los cálculos

Los datos anteriores se ingresaron a la herramienta Enterprise Architect y se obtuvieron los siguientes resultados:

Factor de complejidad técnica	Valor
Valor TCF no ajustado (UTV)	49,00
Ponderación de TCF (TWF)	0,01
Constante TCF (TC)	0,60
Factor de complejidad técnica (TCF) = $TC + (UTV * TWF)$	1,09

Tabla 14. Resultados del cálculo del factor de complejidad técnica

Factor de complejidad ambiental	Valor
Valor ECF no ajustado (UEV)	26,50
Ponderación de ECF (EWF)	-0,03
Constante de ECF (EC)	1,40
Factor de complejidad ambiental (ECF) = $EC + (UEV * EWF)$	0.60

Tabla 15. Resultados del cálculo del factor de complejidad ambiental

Los resultados se muestran en la Tabla siguiente:

Item	Valor
Cantidad de casos de uso	14
Puntos de casos de uso sin ajustar (UUCP)	84
Factor de complejidad técnica (TCF)	1,09
Factor de complejidad ambiental (ECF)	0.60
Puntos de casos de uso ajustados ($UUCP * TCF * ECF$) = UCP	55

Horas estimadas por UUCP (HRS)	22
Total de horas (HRS * UCP)	1,210

Tabla 16. Resultados del cálculo del esfuerzo de desarrollo.

A partir de las estimaciones obtenidas, se puede concluir que para la implementación de las funcionalidades de los casos de uso se requieren 1,210 horas – hombre.

3.10-Distribución del esfuerzo

El esfuerzo estimado anteriormente representa una parte del total del esfuerzo de todo el proyecto, en este caso un 50%. Este 50% se refiere al esfuerzo total para la implementación, o sea, para el desarrollo de las funcionalidades especificadas en los Casos de Uso. En la siguiente tabla se detallan la distribución en porcentaje, para el esfuerzo total en el desarrollo del proyecto:

Actividad	Porcentaje
Análisis	10%
Diseño	20%
Programación	50%
Pruebas	10%
Sobrecarga	10%
Total	100%

Tabla 17. Porcentajes de actividad por etapas.

3.10.1-Estimación del esfuerzo

Actividad	Porcentaje	Cantidad - horas
Análisis	5%	100.8
Diseño	15%	302.4
Programación	60%	1,210

Pruebas	10%	201.6
Sobrecarga	10%	201.6
Total		

Tabla 18. Estimación del esfuerzo.

3.11-Determinación de los costos

Como se calculó anteriormente el esfuerzo realizado es de 2,016 Horas-Hombre, de aquí que se considere que un trabajador emplea en su trabajo 24 días de cada mes y que labora diariamente 8 horas.

$$CMT = E \text{ (Total)} / (CT * CDL * CHL)$$

- CMT: cantidad de meses para la realización del trabajo

- CT: cantidad de trabajadores

- CDL: cantidad de días laborales por mes

- CHL: cantidad de horas laborables por día

Por lo tanto,

$$CMT = 2420 / (1 * 24 \text{ días} * 8 \text{ horas})$$

$$CMT = 2,016 / 192$$

$$CMT = 10.5 \text{ meses}$$

$$CMT \approx 10 \text{ meses}$$

$$TDES \text{ (total)} = E \text{ (total)} / CH \text{ (total)}$$

- TDES: Tiempo de Desarrollo

- CH: Cantidad de Hombres

$$TDES \text{ (total)} = 2016 \text{Hombres-Hora} / 1 \text{ Hombre} = 2016 \text{Horas}$$

3.11.1-Estimación del costo de realización del proyecto

$$C \text{ (total)} = E \text{ (total en HH)} * CHH$$

- C: Costo total

- CHH: Costo por Hombre Horas

$$CHH = K * THP$$

- K: Coeficiente que tiene en cuenta los costos indirectos (1.5 y 2.0)

- THP: Tarifa Horaria Promedio

$$C \text{ (total)} = E \text{ (total en HH)} * K * THP$$

$$SP = \$ 3800$$

$$THP = \$ 3800 / 160 = \$ 23.75$$

Entonces:

$$C \text{ (total)} = 2016 \text{ Horas-Hombre} * 1 * \$ 23.75 = \$ 47,880$$

Por todos los cálculos anteriores se determinó que el costo de la realización de este proyecto

es de \$ 47,880.

3.12-Beneficios Tangibles e Intangibles

Los beneficios tangibles son las ventajas económicas cuantificables que obtiene la organización

a través del uso del sistema y se pueden estimar en recursos, pesos y tiempo ahorrado. Como

beneficio tangible de este proyecto está:

El desarrollo de la aplicación tendría un costo de \$ 47 880MN. Este valor representa un ahorro

al no tener que pagar la realización de este software.

Los beneficios intangibles son aquellos beneficios difíciles de cuantificar que obtiene una organización a través del uso de un sistema de información, pero por ello no dejan de ser menos importantes. Dentro de los beneficios intangibles del trabajo se destacan:

- El sistema sustituirá un proceso engorroso, permitiendo desarrollarse de manera más eficiente.
- Garantiza la seguridad y acceso a la información

3.13-Análisis de costos y beneficios

El proyecto no presenta costo para la identidad, pero su desarrollo está valorado en \$ 47 880

MN, por lo que el sistema reporta un ahorro económico a la empresa, no solo por sus prestaciones, sino también por su proceso de desarrollo. Un producto de similares características, de encargarse a una empresa de desarrollo de software, o incluso su producción por parte de los desarrolladores de la empresa, implicaría un costo para la entidad por concepto de salario.

3.14-Conclusiones

Se representan el diseño detallado donde se visualiza el diagrama de clases del diseño y principios y tratamiento de errores de la aplicación. El resultado del estudio de factibilidad realizado determina que el proyecto es factible, con un costo de desarrollo de la aplicación de \$ 47 880 MN, el cual representa un ahorro al centro, en un tiempo de desarrollo de aproximadamente 10 meses.

Conclusiones:

Teniendo en cuenta los objetivos planteados al inicio del proyecto, se arriban a las siguientes conclusiones:

1. Se obtuvo una especificación completa de los procesos de gestión de la información en la División Territorial de Etecsa Cienfuegos, y se identificaron las principales problemáticas.
 2. Se realizó la modelación de los procesos de la aplicación, se identificaron 11 requisitos funcionales y 2 actores.
 3. Se diseñó un sistema informático siguiendo el flujo de trabajo de la metodología ICONIX.
 4. Se determinó que la investigación es factible de realizar a partir del análisis de esfuerzo y costos realizado, lo que resultó en que tendría un costo de \$ 47 880 MN en un tiempo de desarrollo de aproximadamente 10 meses.
 5. Se implementó la aplicación propuesta para la División Territorial de Etecsa Cienfuegos.
-

Recomendaciones:

Los objetivos trazados con la realización de este trabajo fueron cumplidos, se sugiere tomar esta propuesta solo como la una fase de un proyecto mucho más ambicioso.

Se recomienda entonces:

- Finalizar el período de prueba, permitiendo la validación por parte de la empresa de este nuevo sistema.
- Profundizar en el análisis de cada uno de los procesos que intervienen en la gestión de eventos, con el objetivo de agregarle nuevas funcionalidades al sistema y así enriquecer su valor práctico y profesional.

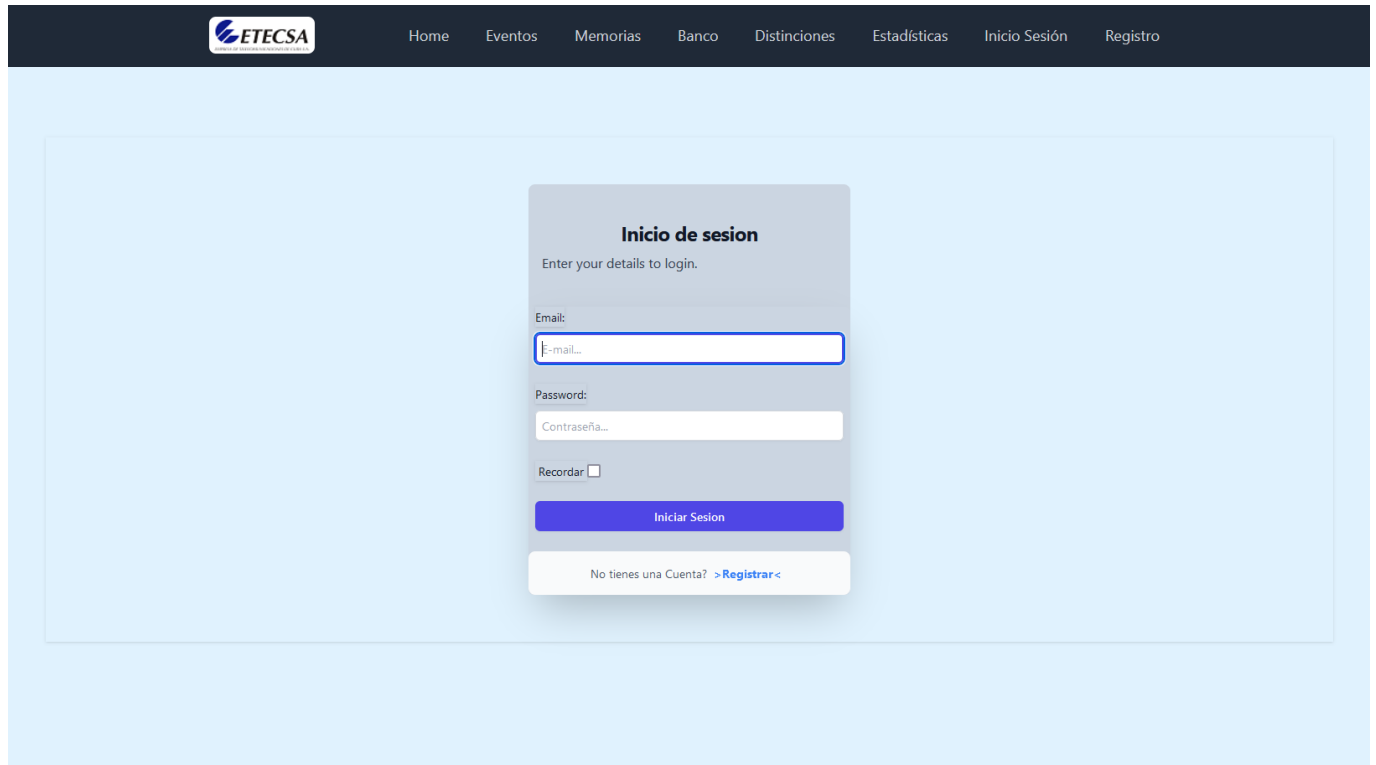
Referencias Bibliográficas

- [1] «Tecnologías de la información y la comunicación», *Wikipedia, la enciclopedia libre*. 1 de diciembre de 2023. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Tecnolog%C3%ADas_de_la_informaci%C3%B3n_y_la_comunicaci%C3%B3n&oldid=155747279
- [2] «Gestión de datos», *Wikipedia, la enciclopedia libre*. 3 de octubre de 2023. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Gesti%C3%B3n_de_datos&oldid=154274936
- [3] «¿Qué es un sistema de gestión y para qué sirve?» Accedido: 6 de noviembre de 2023. [En línea]. Disponible en: <https://www.ekon.es/blog/sistemas-de-gestion-integral-para-el-funcionamiento-optimo-de-la-empresa/>
- [4] Y. P. Meneses, «Informatización de la sociedad cubana, un proceso impostergable», Presidencia de Cuba. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: <https://www.presidencia.gob.cu/es/noticias/informatizacion-de-la-sociedad-cubana-un-proceso-impostergable/>
- [5] «ICONIX - Wikipedia». Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: <https://en.wikipedia.org/wiki/ICONIX>
- [6] «ICONIX - EcuRed». Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: <https://www.ecured.cu/ICONIX>
- [7] «Qué es el lenguaje unificado de modelado (UML)», Lucidchart. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: <https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>
- [8] «Lenguaje unificado de modelado», *Wikipedia, la enciclopedia libre*. 4 de noviembre de 2023. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Lenguaje_unificado_de_modelado&oldid=155071458
- [9] «PHP», *Wikipedia, la enciclopedia libre*. 6 de diciembre de 2023. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: <https://es.wikipedia.org/w/index.php?title=PHP&oldid=155853607>
- [10] «Laravel», *Wikipedia, la enciclopedia libre*. 4 de diciembre de 2023. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: <https://es.wikipedia.org/w/index.php?title=Laravel&oldid=155810518>
- [11] «MySQL», *Wikipedia, la enciclopedia libre*. 26 de noviembre de 2023. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: <https://es.wikipedia.org/w/index.php?title=MySQL&oldid=155623564>
- [12] P. Londoño, «Qué es MySQL, para qué sirve y características principales». Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: <https://blog.hubspot.es/website/que-es-mysql>
- [13] «Requisito funcional», *Wikipedia, la enciclopedia libre*. 7 de marzo de 2023. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Requisito_funcional&oldid=149734810
- [14] «Requisito no funcional», *Wikipedia, la enciclopedia libre*. 15 de junio de 2023. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Requisito_no_funcional&oldid=151870844
- [15] «Requisitos no funcionales: ¿Por qué son importantes?», Gluo. Accedido: 6 de diciembre de 2023. [En línea]. Disponible en: <https://gluo.mx/blog/requisitos-no-funcionales-por-que-son-importantes>

- [16] «Guía de Usuario de Enterprise Architect». Accedido: 6 de diciembre de 2023. [En línea]. Disponible en:
http://www.sparxsystems.com.ar/eauserguide/index.html?generated_robustness_diagram.htm
- [17] «Estimacion del esfuerzo basada en casos de usos (1) by Rocio Arevalo - Issuu». Accedido: 7 de diciembre de 2023. [En línea]. Disponible en: <https://issuu.com/roci-arevalo/docs/estimacion-del-esfuerzo-basada-en-c>

Anexos:

Anexo 1-Pantalla Inicio de sesión



The image shows a screenshot of the ETECSA website's login page. The header is dark blue with the ETECSA logo on the left and navigation links (Home, Eventos, Memorias, Banco, Distinciones, Estadísticas, Inicio Sesión, Registro) on the right. The main content area has a light blue background. In the center, there is a white login form with a grey header that says "Inicio de sesion" and "Enter your details to login." The form includes fields for "Email:" (with a placeholder "E-mail...") and "Password:" (with a placeholder "Contraseña..."). Below these is a "Recordar" checkbox. A blue button labeled "Iniciar Sesion" is positioned below the checkbox. At the bottom of the form, a link says "No tienes una Cuenta? > Registrar <".

Inicio de sesion
Enter your details to login.

Email:

Password:

Recordar ☐

[Iniciar Sesion](#)

No tienes una Cuenta? [> Registrar <](#)

Anexo 2-Pantalla Registro

 [Home](#) [Eventos](#) [Memorias](#) [Banco](#) [Distinciones](#) [Estadísticas](#) [Inicio Sesión](#) [Registro](#)

Registro

Nombre y Apellidos:

Correo:

Password:

Password Confirm:

[Registrarse](#)

Ya tienes una cuenta? [>Inicio de sesion<](#)

Anexo 3-Pantalla Principal

 [Home](#) [Eventos](#) [Memorias](#) [Banco](#) [Distinciones](#) [Estadísticas](#) [Usuarios](#) [Cerrar Sesión→](#)

Portal para la Innovación y la Ciencia

División Territorial Cienfuegos

Lorem ipsum dolor sit amet. Et earum eius eos voluptatum mollitia hic ratione eius quo reprehenderit asperiores a quibusdam quia. Cum facilis nulla aut debitis inventore rem sint eius aut harum cumque? Ut ratione illo est iusto quae qui quae aperiam. In mollitia soluta At molestias quod ab voluptatem ipsum hic quam magni sed natus autem eum debitis quia. Qui quisquam amet et quis ratione in provident natus. Sed aperiam voluptas sed beatae deserunt ut itaque iusto aut repellat sequi vel necessitatibus similique? Eum sapiente natus est commodi ipsa sit sunt autem aut accusamus voluptatem 33 voluptate recusandae At illo sint est perferendis explicabo! Quo perspicatis beatae non temporibus provident qui dolorum quas qui impedit nostrum vel iusto ipsam non eaque fugit.



"Somos una empresa en constante crecimiento e innovación, y orientada a satisfacer las expectativas de nuestros clientes. Logramos que nuestra sociedad este mas y mejor conectada con soluciones del mundo digital, Tributando al incremento de la calidad de vida del cubano. Sustentamos nuestro trabajo en la Profesionalidad, integridad moral, sentido de pertenencia y cultura de servicio que nos distingue."

Anexo 4-Pantalla Memorias



[Home](#) [Eventos](#) [Memorias](#) [Banco](#) [Distinciones](#) [Estadísticas](#) [Usuarios](#) [Cerrar Sesión→](#)

Memorias

Lorem ipsum dolor sit amet. Est odio repellat At ipsum quibusdam et perspiciatis nulla aut quod voluptas ut repudiandae excepturi in reprehenderit eius vel possimus accusantium. Ut sint iure et illo magnam et omnis asperiores in maxime alias et obcaecati delectus. Ut recusandae ullam eos mollitia voluptatum et rerum accusantium et repellat dolores. Qui quibusdam quasi sed molestias dolor est similique maxime aut consequatur recusandae quo fuga quibusdam. Id quaerat doloremque a molestiae rerum aut repellendus quisquam est adipisci illo.



Eliminar

Eliminar

Agregar Imagen

Anexo 5-Pantalla de Eventos



[Home](#) [Eventos](#) [Memorias](#) [Banco](#) [Distinciones](#) [Estadísticas](#) [Usuarios](#) [Cerrar Sesión→](#)

Eventos

+ Nuevo

Id	Nombre Del Evento	Año	Categoría	Acciones
3	Forum de Tecnica	2023	BTJ	Abrir Editar Eliminar
2	Forum de Ciencia	2022	Mujer Creadora	Abrir Editar Eliminar
1	Forum para la innovacion	2020	Fórum	Abrir Editar Eliminar

Anexo 6-Pantalla de las Ponencias



[Home](#) [Eventos](#) [Memorias](#) [Banco](#) [Distinciones](#) [Estadísticas](#) [Usuarios](#) [Cerrar Sesión→](#)

Evento Forum de Tecnica del año 2023

[Regresar](#) [+ Add](#)

Ponencias del evento Forum de Tecnica

Title	Comision	Participantes	Acciones
podgfmndsfmg	Mejora	Frank Cruz Enriquez Manuel Perez	👁 Ver 📄 Descargar 🗑 Eliminar ☑ Check Pendiente

Anexo 7-Pantalla del Banco de Problemas



[Home](#) [Eventos](#) [Memorias](#) [Banco](#) [Distinciones](#) [Estadísticas](#) [Usuarios](#) [Cerrar Sesión→](#)

Banco de Problemas

[+ Agregar Problema](#)

Id	Nombre	Descripcion	Acciones
1	idfhvgdhnfyndfvd	dokfvjdfigodmfgodmfogjdfgjdfj dfgoidsjfgjdfsgj ogfjdlsjfgosdfogjds dfg90[wdhfeuthgiodef	✎ Editar 🗑 Eliminar

Anexo 8-Pantalla de las estadísticas de los eventos por fecha

 HomeEventosMemoriasBancoDistincionesEstadísticasUsuariosCerrar Sesión→					
Eventos Ordenados por Fecha					
Eventos del 2023					
ID	Nombre del Evento	Categoría	Cantidad de ponencias	Cantidad de Participantes	Acciones
3	Forum de Técnica	BTJ	1	2	 Abrir
Eventos del 2022					
ID	Nombre del Evento	Categoría	Cantidad de ponencias	Cantidad de Participantes	Acciones
2	Forum de Ciencia	Mujer Creadora	0	0	 Abrir
Eventos del 2021					
ID	Nombre del Evento	Categoría	Cantidad de ponencias	Cantidad de Participantes	Acciones
Eventos del 2020					
ID	Nombre del Evento	Categoría	Cantidad de ponencias	Cantidad de Participantes	Acciones

Anexo 9-Pantalla de todos los Usuarios

 HomeEventosMemoriasBancoDistincionesEstadísticasUsuariosCerrar Sesión→					
Usuarios					+ Agregar Usuario
Id	Nombre	Correo	Rol	Acciones	
1	Frank Cruz Enriquez	frankcruz@example.com	["admin"]	Editar Rol	Eliminar
2	Manuel Perez	manuel@example.com	["gestor"]	Editar Rol	Eliminar
3	Joselu Martinez	joselu@example.com	["ponente"]	Editar Rol	Eliminar

Anexo 10-Pantalla Crear Evento

[Home](#)[Eventos](#)[Memorias](#)[Banco](#)[Distinciones](#)[Estadísticas](#)[Usuarios](#)[Cerrar Sesión→](#)

Este Formulario creara un Evento

Nombre:

Año:

Categoría:

Fórum

▼

Plantilla:

Examinar...

No se ha seleccio...o ningún archivo.

Aval de Certificado:

Examinar...

No se ha seleccio...o ningún archivo.

Regresar

Crear Evento