

# 02

---

## **DISEÑO DEL SISTEMA AUTOMATIZADO**

**PARA LA GENERACIÓN Y GESTIÓN DE DOCUMENTOS EN LA ETAPA PRECONTRACTUAL DE  
COMPRAS PÚBLICAS PARA INSTITUCIONES PÚBLICAS EN EL ECUADOR**

**DESIGN OF THE AUTOMATED SYSTEM FOR THE GENERATION AND MANAGEMENT OF DOCUMENTS IN  
PRE-CONTRACTUAL STAGE OF PUBLIC PURCHASES FOR PUBLIC INSTITUTIONS IN ECUADOR**

Tonysé de la Rosa-Martín<sup>1</sup>

E-mail: [tdelarosa@umet.edu.ec](mailto:tdelarosa@umet.edu.ec)

ORCID: <https://orcid.org/0000-0002-0881-6034>

Jorge Luis León-González<sup>2</sup>

E-mail: [joshuamashiaj92@gmail.com](mailto:joshuamashiaj92@gmail.com)

ORCID: <https://orcid.org/0000-0003-2092-4924>

<sup>1</sup> Universidad Metropolitana. Ecuador.

<sup>2</sup> Editorial Exced. Ecuador.

Cita sugerida (APA, séptima edición)

De la Rosa-Martín, T., & León-González, J. L. (2023). Diseño del sistema automatizado para la generación y gestión de documentos en la etapa precontractual de compras públicas para instituciones públicas en el Ecuador. *Revista Transdisciplinaria de Estudios Sociales y Tecnológicos*, 3(1), 14-25.

**Fecha de presentación:** septiembre, 2022

**Fecha de aceptación:** diciembre, 2022

**Fecha de publicación:** enero, 2023

---

## **RESUMEN**

La presente investigación se centra en el Diseño de un sistema para la generación de documentos en la etapa preparatoria y precontractual de compras públicas para las instituciones públicas del Ecuador. Se realizó una investigación completa de las falencias en los procesos actuales que se manejan para las compras públicas, se determinó realizar distintos módulos para la generación de documentos, además de módulos de administración. Este sistema fue desarrollado en un ambiente web como lenguaje principal PHP, además se utilizaron dos frameworks: Laravel y Bootstrap, también se utilizará la librería de JQuery, como almacenamiento de los datos se utilizó el gestor de base de datos MariaDB. Este Sistema se apoya en la metodología de desarrollo de Software RUP (Rational Unified Process), se utilizó el lenguaje unificado de modelado UML, estas son herramientas necesarias en el análisis y diseño de sistemas POO (Programación orientada a objetos). El sistema resuelve el problema de la gestión de las compras públicas para instituciones inscritas al gobierno en el Ecuador.

### **Palabras clave:**

Compras, oferta pública, gobierno, institucionalidad.

## **ABSTRACT**

The present investigation focuses on the Design of a system for the generation of documents in the pre-contractual and pre-contractual stage of public purchases for public institutions in Ecuador. A complete investigation of the shortcomings in the current processes that are managed for public purchases was carried out, it was determined to carry out different modules for the generation of documents, in addition to administration modules. This system was developed in a web environment as the main language PHP, in addition two frameworks were used: Laravel and Bootstrap, the JQuery library will also be used, as data storage the MariaDB database manager was used. This Systems is based on the RUP (Rational Unified Process) Software development methodology, the unified modeling language UML was used, these are necessary tools in the analysis and design of OOP (Object Oriented Programming) systems. The system solves the problem of managing public purchases for institutions registered with the government in Ecuador.

### **Keywords:**

Purchases, public offering, government, institutionality.

## INTRODUCCIÓN

Las instituciones públicas en el Ecuador son responsables de llevar un comportamiento limpio y claro en todos los procesos que estos realicen. En el decreto N° 744 en aquel entonces el Expresidente Rafael Correa Delgado establece: *“La creación, administración y desarrollo del portal del sistema oficial de información de contratación pública y consultoría del Ecuador, para garantizar una política de transparencia, una administración pública libre de corrupción en el sector público ecuatoriano”*. (Ramírez & Herrera, 2013)

El Servicio Nacional de Contratación Pública es una entidad pública que se encarga por ley de liderar y regular la gestión transparente y efectiva del servicio de contratación pública. Optimiza los recursos del estado y dinamizando el desarrollo económico y social del Ecuador. Además, nos asegura la transparencia en los procesos de contratación pública y la inclusión de empresas o proveedores nacionales.

Las instituciones públicas se adaptan a los lineamientos que el SERCOP establece hasta el día de hoy, deben regirse a la ley orgánica del sistema nacional de contratación pública, que establece principios y normas para la contratación pública ecuatoriana.

Cada institución ha creado, optimizado, mejorado sus procesos internos para dar cumplimiento a las normas que el SERCOP solicita.

Estos procesos se los han trabajado hasta poder perfeccionarlos de acuerdo a sus necesidades. Cada institución implementa las herramientas necesarias para poder perfeccionar sus procesos y optimizar los recursos que emplean en cada proceso.

El Ministerio de Gobierno gestiona, aprueba y controla todas las actividades encaminadas a la elaboración de la cartografía oficial y del archivo de datos geográficos y cartográficos del país, elabora especies valoradas y documentos de seguridad como único organismo autorizado y difunde las ciencias geoespaciales; coadyuvando a la defensa de la soberanía e integridad territorial, seguridad integral, apoyo al desarrollo nacional y contribución a la paz regional y mundial.

Esta investigación se realiza para mejorar y optimizar los recursos, procesos y administración, para la contratación pública en la institución. Ministerio de Gobierno se encuentra ubicado en la ciudad de Quito en el Palacio de Carondelet. La institución está a cargo de todas las solicitudes de las áreas que conforman en si la institución y de departamentos que se manejan a nivel nacional. Se debe garantizar la transparencia y atención a los pedidos que estos tengan brindando calidad en sus servicios.

El registro de actas, consiste en obtener la información de las reuniones realizadas por los actores, que están involucrados en cada proceso de adquisición. En estas reuniones se establece acuerdos o estatutos que regulan las calificaciones y postulaciones de los proveedores que participan en cada proceso.

El registro de resoluciones permite obtener los estatutos iniciales o finales de los procesos establecidas por el director del Instituto Geográfico Militar. Estas resoluciones además permiten el registro de artículos establecidos por la institución que deben ser publicados en el portal del SERCOP.

El registro de informes permite obtener la información resumida de todas las actas, además este informe es generado al final del registro todas las actas, aquí los delegados a cargo del proceso dan una recomendación del mejor proveedor que hay en el concurso.

Estos procesos son los principales que manejan internamente el departamento de servicios institucionales, estos generan una amplia documentación, además que estos sufren cambios hasta culminar cada adquisición, aumentando el consumo de recursos en cada formulación de los documentos.

Los documentos generados a lo largo del proceso se registran muchos errores en la generación o registro de las actas, resoluciones e informes. Cada acta tiene un flujo de trabajo una depende de otra y también una alimenta a la otra. Al terminar todos los registros de los documentos, el departamento de jurídico revisa para poder dar el aval de que no presentan ningún error para un litigio futuro en contra de la institución.

En este filtro se descubre errores que afectan al proceso y a la institución, aquí se recomienda hacer las correcciones necesarias por parte del departamento de jurídico, el realizar estas correcciones consumen recursos tangibles, intangibles, humanos, entre otros.

El objetivo del estudio es diseñar un sistema orientado a la web para la gestión de documentos generados en la etapa de contratación pública en el Ecuador.

## DESARROLLO

Los procesos que se realizan actualmente fueron establecidos de manera que deben un cumplir un orden secuencial. Al momento de realizar la gestión de cada uno de los documentos se identificó que, al momento de registrar la información en Word, se la realiza de una manera incorrecta y genera muchos inconvenientes.

Estos problemas causan el consumo de recursos y pérdidas de tiempo que son indispensables en el departamento de servicios institucionales. Al darse un error la información no es real y no es confiable. Todo esto perjudica

al proceso de adjudicación, retardando y causando inconvenientes con los delegados y los proveedores que están concursando en el proceso.

La generación del informe de recomendación es la parte más importante, aquí es donde se realiza la recomendación, al mejor proveedor que cumple con los requisitos solicitados por la institución. Es importante que la información con la que se alimenta este informe sea verídica debido a que el director de la institución es el que abala este informe y toma la decisión final de dar el concurso al proveedor mejor calificado y que es recomendado en el informe.

Los procesos presentan inconvenientes al momento de realizar los registros de la información, se dan muchos descuidos y no se establece encargados de que estos documentos estén correctamente elaborados.

Garantizar y promover la gestión documental garantizaría que se cumplan con las fechas establecidas, permitiendo acelerar los procesos y lograr satisfacer las necesidades de los departamentos que solicitan la adquisición o compra de un bien o servicio. Esto permitirá atender otros procesos que son descartados por falta de tiempo del personal en el departamento.

Desde que se implementó el SERCOP la mayoría de las instituciones públicas manejan sus procesos de manera manual, debido a que hasta el momento no se han establecido procesos definitivos, con el paso del tiempo los procesos se establecieron de manera definida pero no se los ha migrado a una herramienta informática.

Con la experiencia del departamento que se encarga de las compras públicas, se definen los errores y procesos para poder automatizarlos con un sistema informático, actualmente existen herramientas de software libre y pagadas que permiten realizar la gestión de documentos.

Dentro de los Sistemas automatizados existentes relacionados con la gestión documental se tiene a Alfresco. Este sistema es de código abierto, permite la gestión documental de contenido empresarial. Es un sistema robusto y escalable, de gran nivel este sistema cuenta con las siguientes características:

Permite la gestión de documentos, gestión de contenido web, versionado a nivel de repositorio, superposición transparente, gestión de registros, gestión de imágenes entre otros elementos que hacen que este sistema cuente con gran variedad de herramientas.

Este software cumple con la gran mayoría de requisitos que se necesitan en la institución, no es escogida debido a su grado de robustez y de administración además que se necesita realizar cambios que se adapten a los procesos que se utilizan internamente que ya conocimos.

Por su parte, el sistema Quipux, es la plataforma de gestión documental oficial del Ecuador, fue creado para promover la eficiencia, transparencia y trazabilidad de la información. Gracias al sistemas muchos servicios cambiaron de manera presencial a en línea permitiendo: Consultar y recibir respuestas a documentos presentados físicamente en instituciones públicas que usen el sistema. Enviar y recibir documentos a varias instituciones públicas usuarias del sistema desde cualquier dispositivo electrónico con acceso a internet.

El software presento muchas mejoras a los procesos y servicios prestados por las instituciones públicas evitando aglomeraciones innecesarias y reduciendo el tiempo de respuesta a los clientes.

Este sistema es utilizado para el envío y recepción de documentos además esta enlazado con los sistemas públicos a nivel nacional.

Microsoft Share Point es un sistema de gestión de documentos basado en la suite de Microsoft. Esta herramienta es privativa, sin duda alguna tiene grandes ventajas debido a ser una suite de Microsoft tiene compatibilidad con las aplicaciones de Windows esta característica es sin duda alguna la mejor característica de SharePoint. En cuanto a la gestión documental presenta grandes opciones para la administración de los documentos.

Al revisar los aspectos metodológicos se decidió elegir la metodología tradicional debido a que se centra más en la planificación y énfasis en los procesos. Además, cuenta con una organización de trabajo más definida. Con el análisis realizado se optó por utilizar la metodología de desarrollo RUP.

La Metodología Proceso Unificado Racional RUP se caracteriza por 3 elementos fundamentales: es iterativa e incremental, es centrada en la arquitectura y es manejada a través de los casos de uso (Kruchten, 2003).

Está basada en la disciplina de asignar tareas y responsabilidades que permiten asegurar que la producción del software cuente con una buena calidad y que este dentro del presupuesto dispuesto y cumple con los plazos establecidos.

RUP nos permite realizar mediciones como: tiempo, nivel de avance y estimación de costos (Díaz, et al., 2015).

El Lenguaje Unificado de Modelado es un lenguaje de modelado. Un modelo es una simplificación de la realidad. El objetivo del modelado de un sistema es capturar las partes esenciales del sistema. Para facilitar este modelado, se realiza una abstracción y se plasma en una notación gráfica. Esto se conoce como modelado visual.

El Lenguaje Unificado de Modelado (UML) desempeña un rol importante no solo en el desarrollo de software, sino también en los sistemas que no tienen software en muchas

industrias, ya que es una forma de mostrar visualmente el comportamiento y la estructura de un sistema o proceso. el UML ayuda a mostrar errores potenciales en las estructuras de aplicaciones, el comportamiento del sistema y otros procesos empresariales (Microsoft, 2019).

*“Uno de los sistemas de gestión de bases de datos es MariaDB, este es una derivación de MySQL tiene una licencia GPL, aunque es una derivación posee más funcionalidades y un mejor rendimiento que MySQL, manteniendo todos sus comandos, interfaces, librerías y APIs. MariaDB es un reemplazo mejorado de MySQL, creado por su fundador principal Michael “Monty” Widenius y algunos desarrolladores de MySQL. Cuenta con nuevas características entre ellas mejoras en el rendimiento, mejores pruebas, corrección de errores, alta disponibilidad, entre otras.”* (Leal, 2014)

MariaDb es una de las bases de datos más utilizadas a lo largo del tiempo, esta adecuada para el manejo de grandes cantidades de información adaptándose a las necesidades del mercado actual. Su simplicidad de sintaxis es una de las características que más atraen debido a que se las pueda utilizar en bases de datos simples o complejas con una gran facilidad, además que existen un sin número de herramientas para la administración de la base de datos. Cuenta también con una gran comunidad de apoyo debido a que es una base de datos de software de código abierto, esto permite que se pueda realizar las mejoras necesarias de acuerdo a las necesidades de los clientes.

*“XAMPP, es un servidor de plataforma libre, es un software que integra en una sola aplicación, un servidor web Apache, intérpretes de lenguaje de scripts PHP, un servidor de base de datos MySQL, un servidor de FTP FileZilla, el popular administrador de base de datos escrito en PHP, MySQL, entre otros módulos. Te permite instalar de forma sencilla Apache en tu propio ordenador, sin importar tu sistema operativo (Linux, Windows, MAC o Solaris). Y lo mejor de todo es que su uso es gratuito.”* (Pérez et al., 2017)

Sublime Text es un editor de texto y editor de código fuente está escrito en C++ y Python para los plugins. Desarrollado originalmente como una extensión de Vim, con el tiempo fue creando una identidad propia, por esto aún conserva un modo de edición tipo vi llamado Vintage

Mode. Se puede descargar y evaluar de forma gratuita. Sin embargo, no es software libre o de código abierto y se debe obtener una licencia para su uso continuado, aunque la versión de evaluación es plenamente funcional y no tiene fecha de caducidad. Está disponible para OS X, Windows y Linux. Sublime Text utiliza un paquete de herramientas de interfaz de usuario personalizado, optimizado para la velocidad y la belleza, aprovechando la funcionalidad nativa de cada plataforma. Cuenta con una potente API de complementos basada en Python. Junto con la API, cuenta con una consola integrada en Python para experimentar interactivamente en tiempo real.

PHP, es un lenguaje de programación de código abierto, muy conocido y utilizado a nivel mundial, es un lenguaje que es apropiado para el desarrollo web debido a que puede ser mezclado con HTML.

Lo mejor de este lenguaje es que es fácil de poder programar mediante scripts que se mezclan con HTML, cuenta con grandes características avanzadas, además cuenta con una sincronización con el servidor y puede disponer dinámicamente toda la información actualizada de manera inmediata.

La forma de trabajar de php es muy simple, la detallaremos a continuación:

Para dar inicio comenzamos con el envío de una petición web del usuario hacia el servidor mediante una página web, es decir un usuario necesita realizar algún tipo de gestión y envía la petición web mediante el sistema, esto llega hasta el servidor web donde se procesa la solicitud realizada por el cliente, al momento de realizar todo el proceso necesario que haya solicitado el cliente, se presenta la respuesta mediante una respuesta desde el servidor hacia el cliente donde se envía la información solicitada, mediante un procesamiento de php y mezclado con HTML para que sea visible al usuario. Es de manera segura como se presenta la información al finalizar la petición de cada usuario gracias a los servicios de php que brindan estas solicitudes del Backend.

El Diagrama de casos de uso es una representación visual simple de las interacciones del sistema con el mundo exterior. Los actores que interactúan con el sistema son representados de usuarios, otros sistemas o cualquier otra entidad fuera del sistema que se comunica con él.

**Gestión total del sistema – Usuario administrador.**

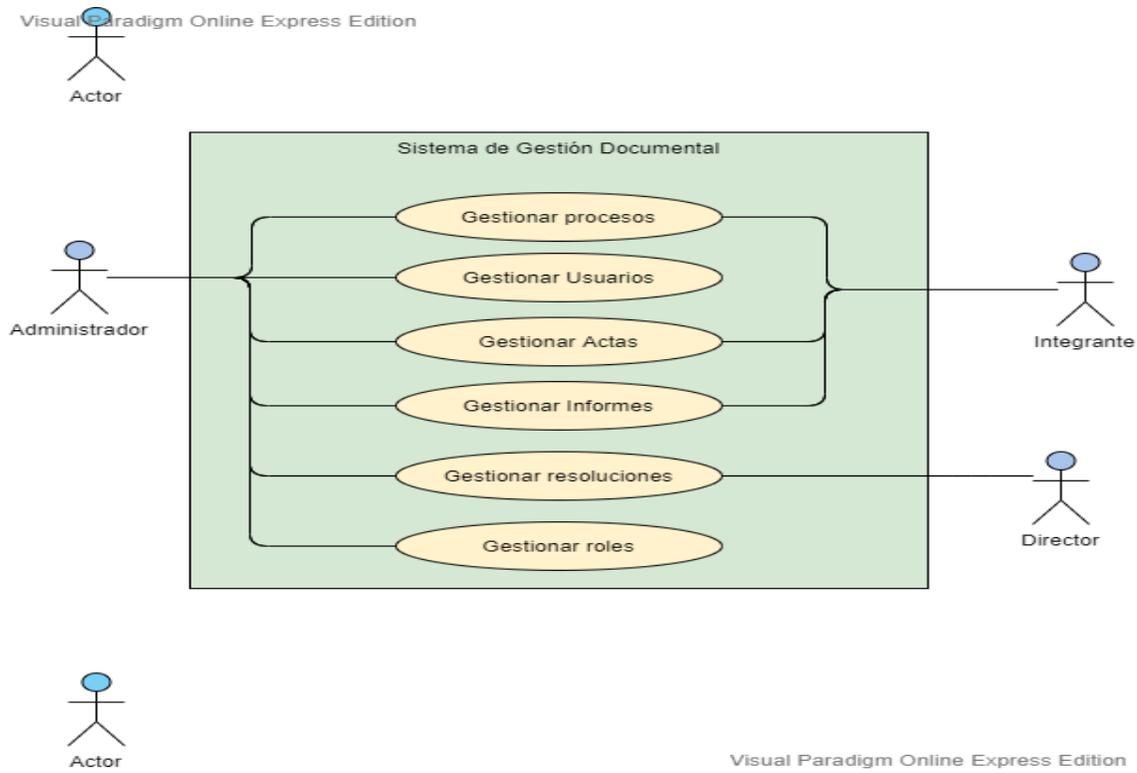


Figura 1. Gestión total módulos del sistema.

En la Figura 1 se muestra el sistema de gestión documenta diseñado para posteriormente será implementado.

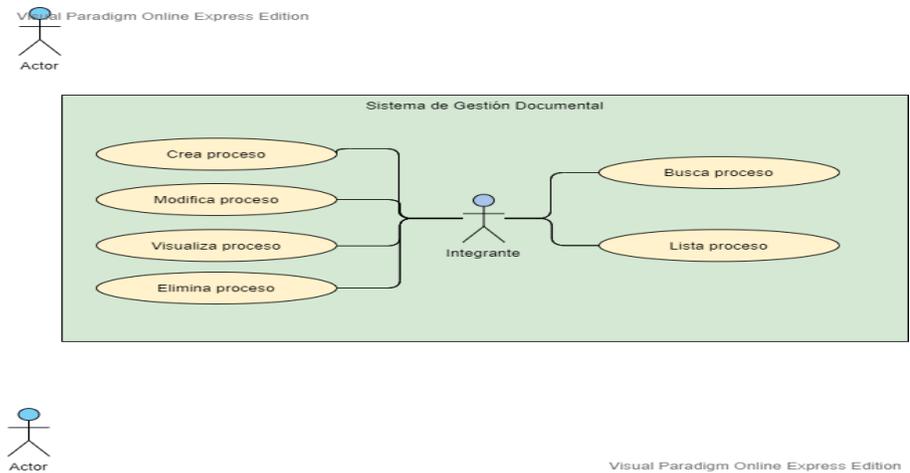


Figura 2 Gestión de actas.

En la Figura 2 se muestra cómo se generarán las gestiones de los distintos procesos el sistema de gestión documental.

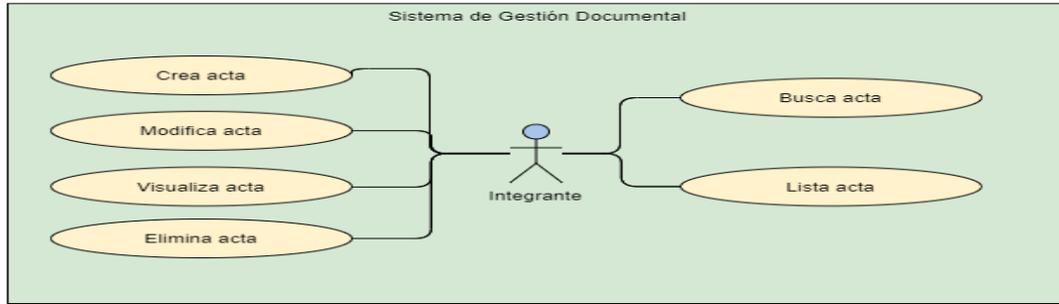


Figura 3 Gestión de resoluciones.

En la Figura 3 se muestra cómo se gestionarán las actas de adquisición pública dentro el sistema de gestión documental.

Según Berdote (2019), cliente servidor es una arquitectura o modelo de comunicación entre servidores que proveen uno o varios servicios. Así como clientes que solicitan unos o varios servicios.

Pongamos un ejemplo sobre un servicio muy común y que es muy usado por casi todo el mundo: el correo electrónico. El correo electrónico utiliza servidores tanto como para enviar los mensajes como para recibirlos.

El usuario accede a esos mensajes a través de un cliente de correo, el cual solicita el servicio de envío de mensajes a los servidores de correo. Así como el servicio para acceder a los mensajes almacenados en los servidores de correo.

En síntesis, el modelo cliente servidor nos permite ordenar comunicaciones entre dispositivos en internet y por ende poder proporcionar servicios a los usuarios (Figura 4) (García, 2014).

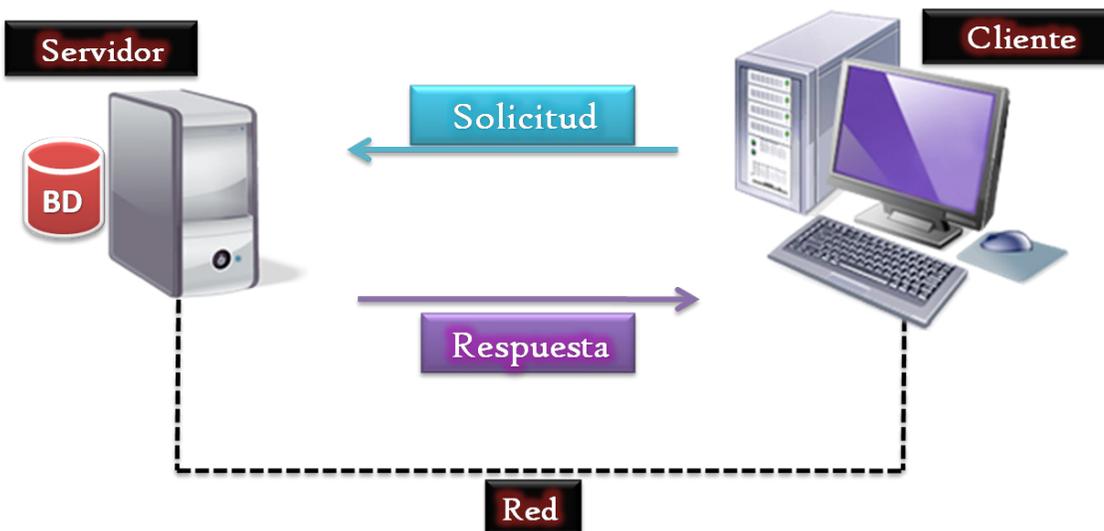


Figura 4. Infraestructura Cliente-Servidor.

La Arquitectura para el diseño del sistema (Modelo-Vistas-Controlador) es un patrón en el diseño de software comúnmente utilizado para implementar interfaces de usuario, datos y lógica de control. Enfatiza una separación entre la lógica de negocios y su visualización. Esta “separación de preocupaciones” proporciona una mejor división del trabajo y

una mejora de mantenimiento. Algunos otros patrones de diseño se basan en MVC, como MVVM (Modelo-Vista-modelo de vista), MVP (Modelo-Vista-Presentador) y MVW (Modelo-Vista-Whatever).

Las tres partes del patrón de diseño de software MVC se pueden describir de la siguiente manera:

- Modelo: Maneja datos y lógica de negocios.
- Vista: Se encarga del diseño y presentación.
- Controlador: Enruta comandos a los modelos y vistas (Mozilla Corporation, 2020).

El modelo define qué datos debe contener la aplicación. Si el estado de estos datos cambia, el modelo generalmente notificará a la vista (para que la pantalla pueda cambiar según sea necesario) y, a veces, el controlador (si se necesita una lógica diferente para controlar la vista actualizada).

Volviendo a nuestra aplicación de lista de compras, el modelo especificará qué datos deben contener los artículos de la lista (artículo, precio, etc.) y qué artículos de la lista ya están presentes (Mozilla Corporation, 2020).

La vista define cómo se deben mostrar los datos de la aplicación. En nuestra aplicación de lista de compras, la vista definiría cómo se presenta la lista al usuario y recibiría los datos para mostrar desde el modelo (Mozilla Corporation, 2020). El controlador contiene una lógica que actualiza el modelo y/o vista en respuesta a las entradas de los usuarios de la aplicación.

Entonces, por ejemplo, nuestra lista de compras podría tener formularios de entrada y botones que nos permitan agregar o eliminar artículos. Estas acciones requieren que se actualice el modelo, por lo que la entrada se envía al controlador, que luego manipula el modelo según corresponda, que luego envía datos actualizados a la vista.

Sin embargo, es posible que también se desee actualizar la vista para mostrar los datos en un formato diferente, por ejemplo, cambiar el orden de los artículos de menor a mayor precio o en orden alfabético. En este caso, el controlador podría manejar esto directamente sin necesidad de actualizar el modelo (Mozilla Corporation, 2020).

El diagrama de clases es uno de los catorce diagramas que posee UML para la modelación de un sistema. Se trata de un diagrama estructural, que muestra cada una de las clases con sus métodos y atributos, así como sus relaciones con otras clases (Figuras 5 y 6) (Pérez Lovelle et al., 2014).

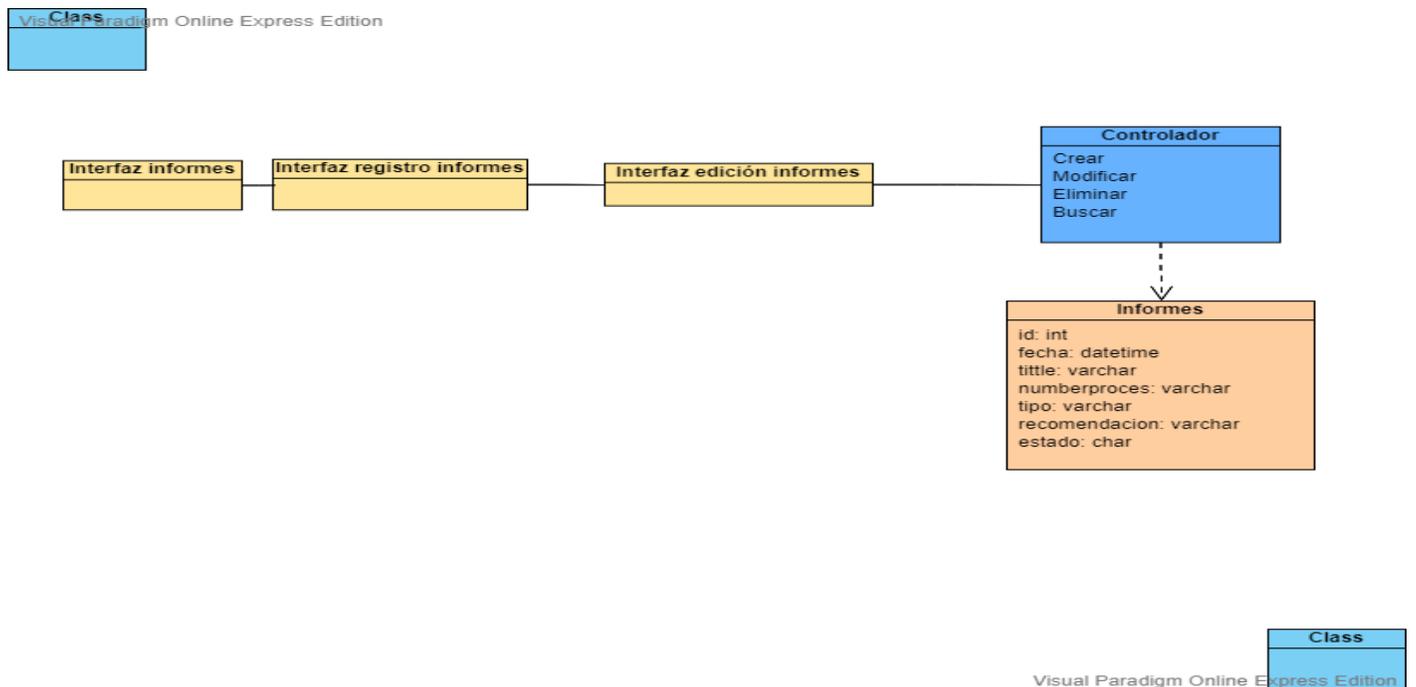


Figura 5. Diagrama de clases informes de recomendación.

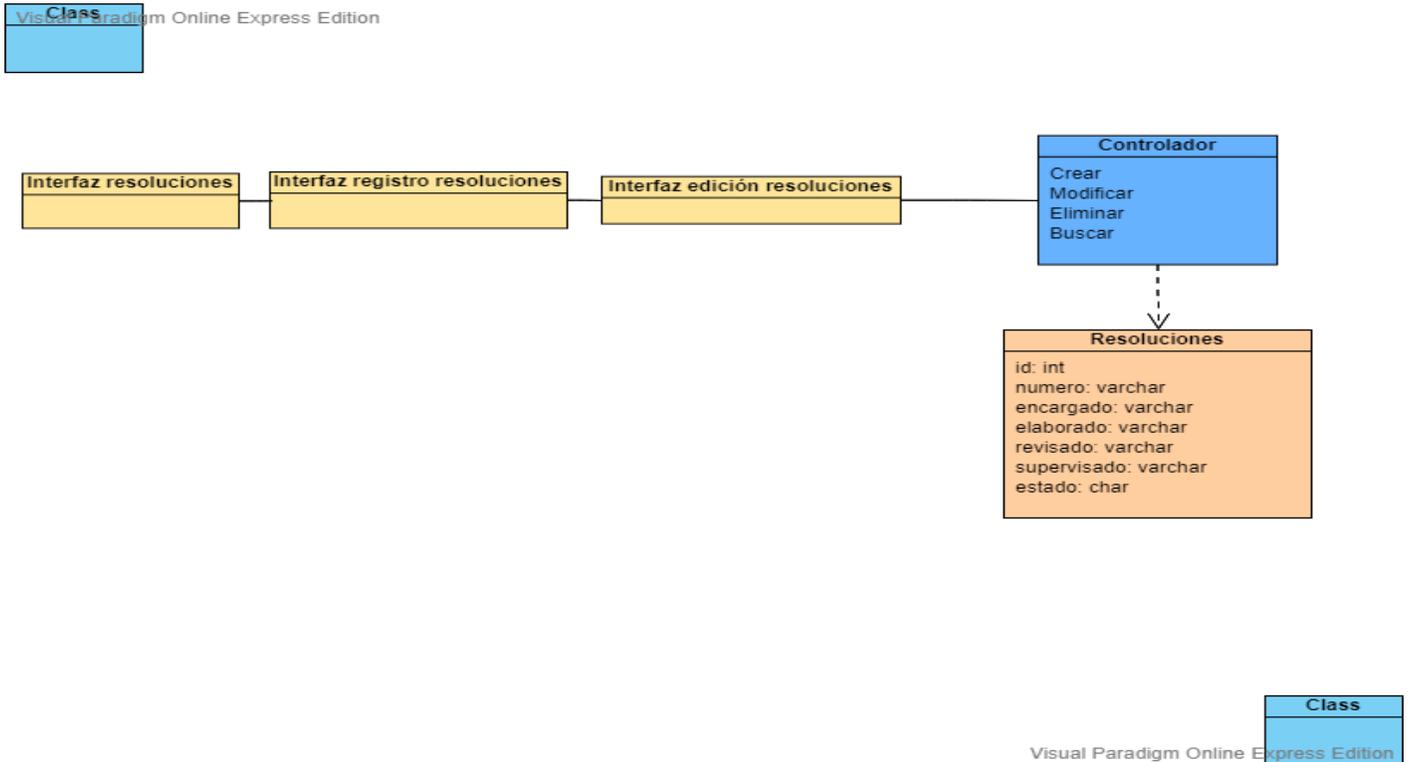


Figura 6. Diagrama de resoluciones.

En los diagramas de secuencia puede tener objetos e instancias de actor, junto con mensajes que describan cómo interactúan. El diagrama describe lo que ocurre en los objetos participantes, en términos de activaciones, y cómo se comunican los objetos enviando mensajes entre sí. Puede realizar un diagrama de secuencia para cada variante del flujo de sucesos del caso de uso. A continuación se ejemplifica cómo fueron utilizados:

### Diagrama de secuencia gestionar actas

Se inicia desde la vista home, el usuario ingresa a la interfaz de autenticación o login, aquí se llena las credenciales que irán hasta el controlador, aquí se verificara mediante un proceso si los datos ingresados corresponden a un usuario en la base de datos o no, al realizar esta petición se recibirá dos respuestas la primera será, si el usuario no se encuentra registrado en a base de datos o las credenciales no son compatibles, arrojara un error de autenticación, la otra respuesta será que el usuario logro con éxito autenticarse. Posterior se visualizará un menú tipo lista, aquí ingresaremos a las actas, en esta interfaz visualizaremos las actas que existen en el sistema, además de la visualización de los botones de mantenimiento de las actas como son: crear, eliminar y modificar (Figura 7).

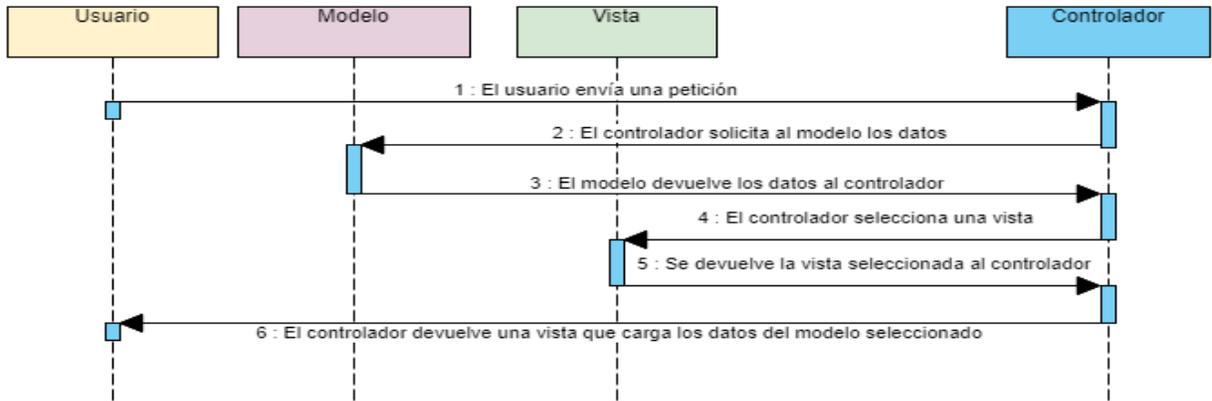


Figura 7. Diagrama de secuencia gestionar actas.

### Diagrama de secuencia gestionar informes

Al momento de seleccionar en el menú tipo lista los informes, visualizaremos una lista de los informes existentes, se visualizará los botones de mantenimiento de los informes como son: crear, modificar y eliminar (Figura 8).

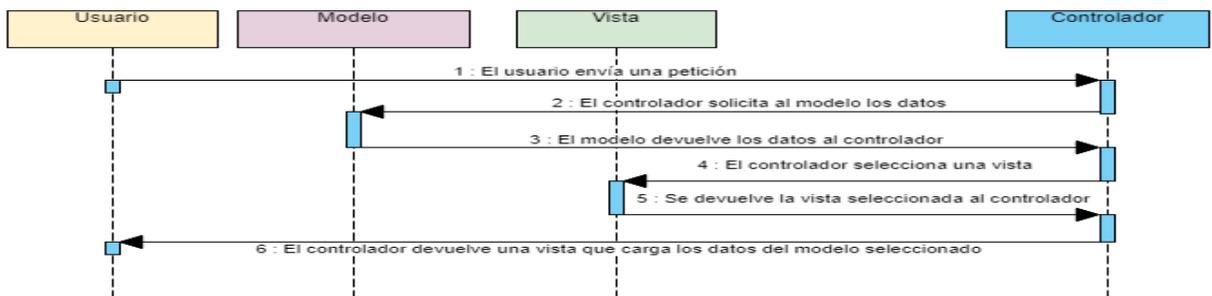


Figura 8. Diagrama de secuencia gestionar informes.

En la lista de menú visualizaremos las resoluciones en este módulo se visualizará las resoluciones del sistema, además de los botones de mantenimiento como son: crear, modificar y eliminar (Figura 9).

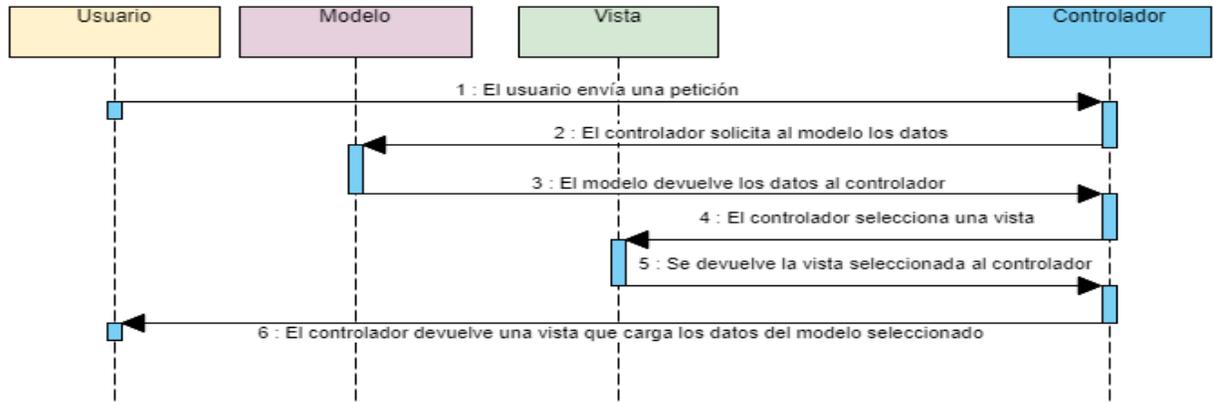


Figura 9. Diagrama de secuencia gestionar resoluciones.

Para verificar los errores se empleó Laravel permite visualizar los errores presentados en el sistema mediante debug una configuración en el archivo config/app.php, esto determinará cuanto se visualizará del error presentado al momento. Por defecto esta opción viene establecida para trabajar con APP\_DEBUG, esto es almacenado en el archivo env.

Es importante que esta configuración sea cambiada cuando el sistema esté en producción, debido a que se pueden mostrar datos sensibles que el usuario puede aprovechar para vulnerar el sistema o poder robar información (Figura 10).

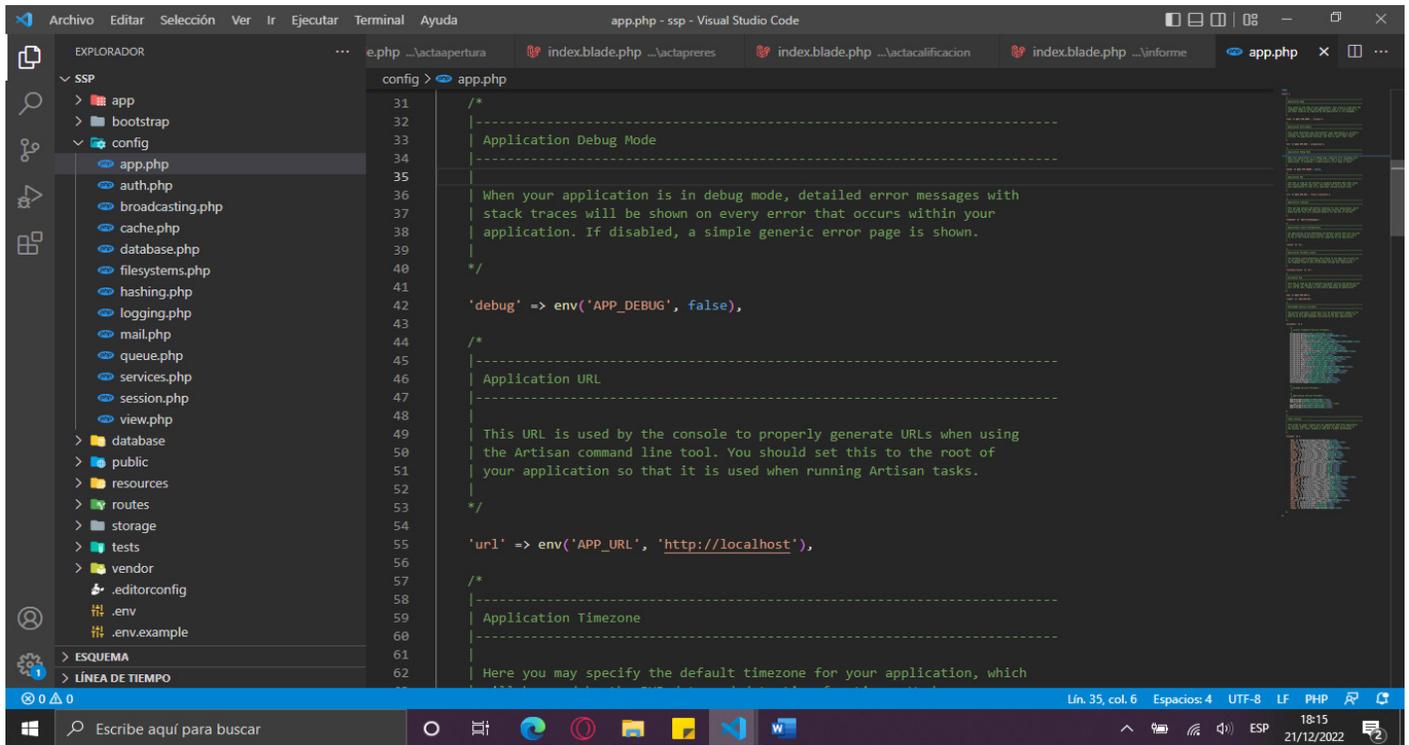


Figura 10. Ejemplo captura control errores.

## CONCLUSIONES

La presente Investigación logró comprobar que el Departamento de Servicios Institucionales del Ministerio de Gobierno se verá beneficiado por el sistema implementar ya que permitirá mejorar y optimizar los tiempos para cada proceso de compras públicas.

Se evaluaron los sistemas de gestión documental que existen en la actualidad, para verificar si son viables para las necesidades planteadas y que cumplan con los requisitos, sin embargo, no lograron cumplir con los requisitos necesarios para ser tomados en consideración y poder utilizarlos.

Con el análisis y diseño del sistema se logra establecer las mejores vías para lograr establecer los requisitos funcionales necesarios para su posterior implementación, permitiendo modelos adecuados que ayudarán a la mejora sustantiva de los procesos de compras públicas.

## REFERENCIAS BIBLIOGRÁFICAS

- Berdote Jiménez, A. (2019). Arquitectura Cliente / Servidor, el mercado de la comunicación. <https://techriders.tajamar.es/arquitectura-cliente-servidor-2/>
- Díaz-Domínguez, L., Castillo-Sequera, J. L., & Navarro, M. A. (2015). Gestión de la cartera de proyectos de TIC. Universidad Alcalá de Henares.
- García, C. (2014). Modelo de Aplicación Cliente – Servidor. <http://charliedaw2236.blogspot.com/p/arquitectura-cliente-servidor.html>
- Kruchten, P. K. (2003). The rational unified process made easy. Grady Booch.
- Leal, V. (2014). MariaDB. <https://www.clubensayos.com/Tecnolog%C3%ADa/Articulo-Maria-DB/1555009.html>
- Microsoft. (2019). La guía sencilla para la diagramación de UML y el modelado de la base de datos. Microsoft 365 Team. <https://www.microsoft.com/es-ww/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling#:~:text=El%20Lenguaje%20Unificado%20de%20Modelado,de%20un%20sistema%20o%20proceso>
- Mozilla Corporation. (2020). Glosario de MDN Web Docs: Definiciones de términos relacionados con la Web. <https://developer.mozilla.org/es/docs/Glossary/MVC>
- Pérez Cornejo, J., González Plascencia, J., Martínez Martínez, L. (2017). XAMPP. <http://desarrollaaplicacioneswebcecytej29.blogspot.com/2017/05/que-es-xampp-xampp-un-servidor-de.html>

Pérez Lovelle, S., Cue Galindo, J. C., Hernández Perenzuela, A., Arredondo López, A., Recio Nápoles, L. R., & Carnero González, F. (2014). Una propuesta para incorporar más semántica de los modelos al código generado. *Polibits*, 49, 47-50.

Ramírez, H., & Herrera, J. (2013). Un viaje a través de bases de datos espaciales NoSQL. *Redes de Ingeniería*, 4(2), 57-69.