Índice

[**Capítulo 1** **Introducción** 5](#_Toc176513728)

[**Capítulo 2** **Fundamentos teóricos sobre los sistemas web de información geográfica** 9](#_Toc176513729)

[**2.1** **Conceptos para el dominio del problema** 9](#_Toc176513730)

[**2.2** **Sistemas de información Geográficos** 13](#_Toc176513731)

[**2.2.1** **Historia y evolución de los GIS** 14](#_Toc176513732)

[**Ventajas que permiten las aplicaciones GIS** 16](#_Toc176513733)

[**Campos que utilizan los sistemas de información geográfica** 17](#_Toc176513734)

[**Ejemplos de sistemas GIS** 18](#_Toc176513735)

[**Gis en la nube** 19](#_Toc176513736)

[**Web** 21](#_Toc176513737)

[**Beneficios de las aplicaciones web** 22](#_Toc176513738)

[**Navegadores web más utilizados** 23](#_Toc176513739)

[**Selección tecnológica** 25](#_Toc176513740)

[**Lenguaje de programación** 25](#_Toc176513741)

[**Bases de datos** 28](#_Toc176513742)

[**Framework para el desarrollo del servidor** 30](#_Toc176513743)

[**Conexión a la base de datos:** 31](#_Toc176513744)

[**Sistema para el desarrollo de interfaces visuales** 33](#_Toc176513745)

[**Controlador de versiones** 34](#_Toc176513746)

[**Repositorio de código** 34](#_Toc176513747)

[**Entorno de desarrollo integrado (IDE) / Editor de código** 34](#_Toc176513748)

[**Solución propuesta** 35](#_Toc176513749)

[**Introducción** 35](#_Toc176513750)

[**Descripción del negocio** 35](#_Toc176513751)

[**Descripción del sistema** 36](#_Toc176513752)

[**Modelo de dominio** 36](#_Toc176513753)

[**Reglas de negocio** 37](#_Toc176513754)

[**Requisitos funcionales** 38](#_Toc176513755)

[**Requisitos no funcionales** 39](#_Toc176513756)

[**Requisitos de apariencia** 39](#_Toc176513757)

[**Requisitos de usabilidad** 39](#_Toc176513758)

[**Requisitos de rendimiento** 39](#_Toc176513759)

[**Requisitos de soporte** 40](#_Toc176513760)

[**Requisitos de portabilidad** 40](#_Toc176513761)

[**Requisitos de seguridad** 40](#_Toc176513762)

[**Requisitos de confiabilidad** 40](#_Toc176513763)

[**Requisitos de software** 40](#_Toc176513764)

[**Requisitos de hardware** 40](#_Toc176513765)

[**Diagrama de caso de uso del sistema** 41](#_Toc176513766)

[**Diagrama de clases** 43](#_Toc176513767)

[**Microservicio Login** 43](#_Toc176513768)

[**Microservicio Gestión** 44](#_Toc176513769)

[**Patrones Arquitectónicos** 46](#_Toc176513770)

[**Patrón N-Capas** 46](#_Toc176513771)

[**Patrón Cliente-Servidor** 47](#_Toc176513772)

[**Patrón de filtro de tubería** 48](#_Toc176513773)

[**Patrón de modelo-vista-controlador** 48](#_Toc176513774)

[**Modelo físico de la base de datos** 49](#_Toc176513775)

[**Microservicio Login** 50](#_Toc176513776)

[**Microservicio Gestion** 51](#_Toc176513777)

[**Seguridad** 51](#_Toc176513778)

[**Autenticación y Autorización** 51](#_Toc176513779)

[**Spring Security** 53](#_Toc176513780)

[**Autenticación Sin Estado (Stateless)** 53](#_Toc176513781)

[**JSON Web Token** 54](#_Toc176513782)

[**Configuración del uso compartido de recursos entre orígenes (CORS)** 55](#_Toc176513783)

[**Protección contra Inyecciones SQL** 56](#_Toc176513784)

[**Monitoreo y registro de actividades de seguridad** 58](#_Toc176513785)

[**Conclusión de la seguridad** 58](#_Toc176513786)

[**Diagrama de despliegue** 58](#_Toc176513787)

[**Validación de la solución propuesta** 60](#_Toc176513788)

[**Referencia Bibliográfica** 60](#_Toc176513789)

Índice de ilustraciones:

[Figura 1: Aplicaciones de un Sistemas de información geográfica (SIG)[5] 10](#_Toc176509786)

[Figura 2: Environmental Systems Research Institute(ESRI)[14] 15](#_Toc176509787)

[Figura 3: Modelo de dominio 37](#_Toc176509788)

[Figura 4: Diagrama de caso de uso del sistema 41](#_Toc176509789)

[Figura 5: Diagrama de clases del microservicio de Login 44](#_Toc176509790)

[Figura 6: Diagrama de clases del microservicio de Gestión 45](#_Toc176509791)

[Figura 7: Modelado del patrón N-Capas 47](#_Toc176509792)

[Figura 8: Modelado del patrón Cliente - Servidor 47](#_Toc176509793)

[Figura 9: Patrón de tubería y filtros 48](#_Toc176509794)

[Figura 10: Modelado del patrón Modelo Vista Controlador (MVC) 49](#_Toc176509795)

[Figura 11: Modelo físico de la base de datos del microservicio de Login 50](#_Toc176509796)

[Figura 12: Modelo físico de la base de datos del microservicio de Gestión 51](#_Toc176509797)

[Figura 13: Modelado del diagrama de despliegue 59](#_Toc176509798)

Índice de tablas:

[Tabla 1: Reglas de negocio 38](#_Toc176459060)

[Tabla 2: Caso de uso Autentificación 42](#_Toc176459061)

[Tabla 3: Caso de uso Gestionar información del sistema 42](#_Toc176459062)

[Tabla 4:Caso de estudio Gestionar información de los usuarios 42](#_Toc176459063)

[Tabla 5: Caso de estudio Visualizar registros del sistema 42](#_Toc176459064)

[Tabla 6: Caso de uso Gestionar entidades 42](#_Toc176459065)

[Tabla 7: Caso de estudio Gestionar empleados 42](#_Toc176459066)

[Tabla 8: Caso de estudio Importar Excel 43](#_Toc176459067)

[Tabla 9:Caso de estudio Visualizar datos geográficos 43](#_Toc176459068)

# **Introducción**

El proceso de informatización de la sociedad cubana iniciado en 2018, en disposición del decreto ley No.370/2018, indica:

La informatización de la sociedad en Cuba desempeña un papel significativo en el desarrollo político, económico y social del país y constituye un medio efectivo para la consolidación de las conquistas del Socialismo y el bienestar de la población. [1]

Para llevar a cabo el cumplimiento de dicha obligación, se ha orientado la actualización y mejoramiento de los obsoletos sistemas informáticos presentes en el país.

El Estado promueve el desarrollo y utilización de las Tecnologías de la Información y la Comunicación, con el objetivo de que constituyan una fuerza política, científica y económica, que contribuya y propicie la integración y conducción de los procesos asociados a la informatización de la sociedad. [1]

Mediante esta premisa se llevó a cabo un proceso de informatización que comprendió el desarrollo, actualización y modificación de la infraestructura y sistemas informáticos, obsoletos en el país, permitiendo así una mejora en los servicios brindados por diversas instituciones y mejores análisis de datos para la toma de decisiones cruciales para el país.

El Ministerio de Transporte (Mitrans) es un organismo perteneciente a la Administración Central del Estado y subordinado al Consejo de Ministros, que tiene como objetivo fundamental ejercer control sobre todas las entidades de transporte que operen en la República de Cuba: transporte urbano, transporte turístico, entre otros.

El bloqueo financiero impuesto por los Estados a Cuba, desde 1960 impone restricciones a la isla e impide el correcto funcionamiento de las instituciones gubernamentales que operan en la región. Esto implica una escasez de recursos agravada cada vez más con nuevas medidas que buscan hundir la economía. A la parte la crisis económica global a raíz de la pandemia del Covid-19 del año 2020, ha puesto al mundo en su situación inflacionaria nunca vista con grandes economías en descensos, una subida de precios a causa de la inflación y un mercado que aún siente los efectos de la pandemia.

Con todos estos factores, Cuba necesita siendo un país pequeño, salir adelante con los recursos disponibles y un factor decisivo en dicha tarea son las tecnologías de la comunicación, que pueden aportar desde sistema de ayuda a la ciudadanía como soluciones empresariales que ayuden en la toma de decisiones.

La escasez del combustible es habitual en Cuba, por las razones anteriormente mencionadas, pero el MITRANS como rector del transporte, debe intentar maximizar la utilidad del combustible y minimizar su gasto, teniendo en cuenta la movilidad poblacional desde sus hogares hasta sus centros de trabajo, pero se encuentran con diversos obstáculos.

Actualmente los métodos de almacenamiento de la entidad son archivos de Excel, donde se almacenan tanto los trabajadores como las distintas entidades a las cuales tienen que debe dirigirse. Los sistemas de almacenamiento basados en ficheros Excel, han ido cayendo en desuso a medida que los sistemas han necesitado ir aumentando su tamaño y complejidad. Las hojas de cálculo a pesar de ser herramientas útiles se encuentras con problemáticas que han sido resueltas por las bases de datos como la flexibilidad y centralización de los datos permitiendo un acceso simultáneo desde múltiples dispositivos, logrando acceder a la misma información en tiempo real.

Otro punto negativo de las hojas de cálculo es la propia especialización del sistema ya que cada uno de los usuarios debe tener un amplio conocimiento de la herramienta y sus diversas funciones. Un Excel es muy fácil de corromper a través del propio fallo humano: un simple cambio en una función y no saber cómo revertir el cambio puede ser fatal, ya que corrompería el sistema haciendo que algunas o todas las funciones emitiesen resultados inseguros o erróneos.

Para poder acceder de una manera detallada a la información necesitan hacer uso de tecnología geoespacial para poder analizar los datos de manera que los datos tengan almacenados datos como las coordenadas cartesianas con el uso de la georreferencia.

Los sistemas de información geográfica (SIG) son sistemas informáticos que producen visualizaciones conectadas de datos geoespaciales, es decir, datos referenciados espacialmente a la Tierra. Además de crear visualizaciones, los SIG son capaces de capturar, almacenar, analizar y gestionar datos geoespaciales. [2]

Con los SIG, los usuarios pueden crear consultas interactivas, analizar información espacial, editar datos, integrar mapas y presentar los resultados de estas tareas. Los SIG forman parte de la ciencia de la información geográfica, que es el campo que abarca todos los aspectos de los SIG: hardware y software, lenguajes de programación, datos geoespaciales y su funcionamiento conjunto. [2]

La georreferenciación es la ubicación de un punto concreto sobre la superficie terrestre. Es una técnica de procesamiento espacial que utiliza las coordenadas de mapa para determinar una localización geográfica única en el espacio y, así, poder facilitar esta información a las diferentes entidades cartográficas de forma digitalizada. [3]

Para resolver todos estos problemas y poder actualizar el sistema del MITRANS en cuanto a gestión de movilidad del personal se desarrolla un Sistema de Gestión Geográfica.

El presente trabajo de diploma tiene como objetivo general desarrollar un sistema de información geográfica, que permita la migración de un fichero Excel a una base de datos, eliminando el antiguo método de almacenamiento de datos de la entidad, por uno moderno que resuelva las necesidades del cliente y acelere los procesos dentro de la empresa.

Para ello se han dispuesto un grupo de objetivos, que tienen la finalidad de dar cumplimiento a este ejercicio:

1. Desarrollar sistema web que dé respuesta a la problemática planteada
2. Documentar la aplicación permitiendo que se legible para futuros desarrolladores
3. Escribir ayuda para los usuarios, permitiendo que la aplicación sea accesible para todos

Para dar solución al presente trabajo de diploma, el documento se va a estructura de la siguiente manera:

**Capítulo 1. Fundamentos Teóricos**: Se describen los principales problemas que componen la investigación: conceptos fundamentales, historia, análisis de soluciones informáticas similares, selección tecnológica y análisis de la solución propuesta.

**Capítulo 2. Análisis y diseño de la solución**: Se describen los principales artefactos que componen todo el proceso del negocio. Análisis de las principales funcionalidades del sistema, permitiendo determinar la solución final a desarrollar.

**Capítulo 3. Desarrollo de la solución**: Se abordan los principales aspectos de la solución propuesta, reflejando los componentes y diagramas utilizados.

**Capítulo 4. Pruebas**: Se verifican los resultados de las distintas pruebas realizadas al sistema y las condiciones del entorno de ejecución que propiciaron dichos resultados

# **Fundamentos teóricos sobre los sistemas web de información geográfica**

En la actualidad los sistemas de información geográfica (SIG) han alcanzado un desarrollo significativo, permitiendo acceder a información geográfica en tiempo real sin necesidad de consultar antiguos métodos de almacenamiento geográfico como enciclopedias, libros y atlas. Los sistemas web actualmente son ampliamente utilizados en el mundo, ya que estos pueden tener mayor velocidad y menor consumo de recursos, ser multiplataformas y utilizarse a través del Internet permitiendo que cualquiera pueda acceder a ellas y utilizar sus diversas funcionalidades. En el transcurso del presente capítulo se tiene como objetivos: abordar conceptos y definiciones asociadas a la investigación en curso, breve resumen histórico sobre los sistemas de información geográficos y los sistemas web, análisis de estudios relacionados con la investigación desarrollados por otros colegas del medio en universidades tanto residentes en Cuba como foráneas y sus aportes a la misma y análisis de las tecnologías empleadas para la realización de dicho sistema.

## **Conceptos para el dominio del problema**

En la actual sección del trabajo, se procederá a describir conceptos necesarios para el mejor entendimiento de la actual investigación.

**Sistemas de información geográfica (SIG) o Geographic Information System(GIS)**: Son sistemas informáticos que producen visualizaciones conectadas de datos geoespaciales, es decir, datos referenciados espacialmente a la Tierra. Además de crear visualizaciones, los SIG son capaces de capturar, analizar y gestionar datos geoespaciales. Con los SIG, los usuarios pueden crear consultas interactivas, analizar información espacial, editar datos, integrar mapas y presentar los resultados de estas tareas. Los SIG forman parte de la ciencia de la información geográfica, que es el campo que abarca todos los aspectos de los SIG: hardware y software, lenguajes de programación, datos geoespaciales y su funcionamiento conjunto.[4]



Figura 1: Aplicaciones de un Sistemas de información geográfica (SIG)[5]

**Datos Geoespaciales**: Describen objetos, acontecimientos u otros accidentes geográficos con una ubicación en la superficie terrestre o cerca de ella. Los datos geoespaciales combinan datos de localización (como las coordenadas) y datos de atributos (las características del objeto, evento u otras características geográficas) con datos temporales (el tiempo o periodo de vida en el que existen la localización y los atributos). Los datos geoespaciales contienen grandes conjuntos de datos procedentes de diversas fuentes, como datos censales y demográficos, imágenes por satélite (incluidos datos de teledetección), datos inmobiliarios, datos meteorológicos, datos de telefonía móvil, imágenes dibujadas y datos de redes sociales.[4]

**Datos ráster**: Los datos ráster consisten en cuadrículas o celdas de píxeles con información espacial asociada a cada celda, como la elevación, la temperatura o incluso el uso del suelo. Los datos ráster se utilizan para crear imágenes complejas de alta resolución, como fotografías e imágenes por satélite. Por ejemplo, una imagen de satélite representada por una matriz de datos que contiene la información meteorológica de una ciudad permite a los ciudadanos comprobar si llueve.[4]

**Datos vectoriales**: Los datos vectoriales son la representación de un elemento geoespacial a través de sus coordenadas x e y. La forma más básica de los datos vectoriales es un punto. Dos o más puntos forman una línea, y tres o más líneas forman un polígono. Por ejemplo, Google Maps, un mapa web común y una representación visual que utiliza datos vectoriales, define la ubicación de una ciudad mediante puntos; carreteras con líneas; y edificios o límites mediante polígonos.[4]

**Georreferenciación**: Es el uso de coordenadas de mapa para asignar una ubicación espacial a entidades cartográficas. Todos los elementos de una capa de mapa tienen una ubicación geográfica y una extensión específicas que permiten situarlos en la superficie de la Tierra o cerca de ella. La capacidad de localizar de manera precisa las entidades geográficas es fundamental tanto en la representación cartográfica como en SIG.[6]

**Latitud y longitud**:  Estas son mediciones de los ángulos (en grados) desde el centro de la Tierra hasta un punto en su superficie. Este tipo de sistema de referencia de coordenadas generalmente se denomina sistema de coordenadas geográficas.[6]

**Sistema de coordenadas cartesianas**: Utiliza dos ejes: uno horizontal (x), que representa el este y el oeste, y otro vertical (y), que representa el norte y el sur. El punto de intersección de los ejes se denomina el origen. Las ubicaciones de los objetos geográficos se definen en relación al origen, utilizando la notación (x,y), donde x se refiere a la distancia del eje horizontal, e y se refiere a la distancia del eje vertical. El origen se define como (0,0).[6]

**Sistema de coordenadas proyectadas**: Es cualquier sistema de coordenadas diseñado para una superficie llana, como un mapa impreso o una pantalla de ordenador.[6]

**World Wide Web**: Comúnmente conocida como WWW, W3, o la Web— es un sistema interconectado de páginas web públicas accesibles a través de Internet. La Web no es lo mismo que el Internet: la Web es una de las muchas aplicaciones construidas sobre Internet.[7]

**HTTP**: El protocolo de transferencia de hipertexto o HTTP (HyperText Transfer Protocol) es el protocolo de red que permite la transferencia de documentos de hipermedia en la red, generalmente entre un navegador y un servidor, para que los humanos puedan leerlos.[8]

**Aplicación web**: Es un software que se ejecuta en el navegador web. Las empresas tienen que intercambiar información y proporcionar servicios de forma remota. Utilizan aplicaciones web para comunicarse con los clientes cuando lo necesiten y de una forma segura. Las funciones más comunes de los sitios web, como los carros de compra, la búsqueda y el filtrado de productos, la mensajería instantánea y los canales de noticias de las redes sociales, tienen el mismo diseño que las aplicaciones web. Le permiten acceder a funcionalidades complejas sin la necesidad de instalar o configurar un software.[9]

**Arquitectura cliente-servidor**: La arquitectura cliente-servidor es una estructura de aplicación distribuida que separa las tareas y las cargas de trabajo entre servidores y clientes. Los servidores proporcionan el recurso o servicio, mientras que los clientes lo solicitan. Tanto el cliente como el servidor son programas independientes que se comunican a través de una red. Un navegador web y un servidor web son un ejemplo de arquitectura cliente-servidor. Se trata de una arquitectura comúnmente utilizada en la computación distribuida.[10]

**Arquitectura del lado del cliente**: El script del lado del cliente se encarga de la funcionalidad de la interfaz de usuario, como los botones y los cuadros con menús desplegables. Cuando el usuario final hace clic en el enlace de la aplicación web, el navegador web carga el script del lado del cliente y renderiza los elementos gráficos y el texto para la interacción del usuario. Por ejemplo, el usuario puede leer contenidos, ver videos o cumplimentar la información de un formulario de contacto. Las acciones como hacer clic en el botón de enviar se dirigen al servidor como una solicitud del cliente.[9]

**Arquitectura del lado del servidor**: El script del lado del servidor se encarga del procesamiento de datos. El servidor de la aplicación web procesa las solicitudes del cliente y envía una respuesta de vuelta. Las solicitudes suelen ser obtener más datos, editar datos o guardar nuevos datos. Por ejemplo, si el usuario hace clic en el botón Leer más, el servidor de la aplicación web enviará contenido al usuario. Si el usuario hace clic en el botón Enviar, el servidor de la aplicación guardará los datos del usuario en la base de datos. En algunos casos, el servidor completa la solicitud de datos y envía la página HTML completa al cliente. Esto se llama renderizado del lado del servidor. [9]

**API**: Significa “interfaz de programación de aplicaciones”. En el contexto de las API, la palabra aplicación se refiere a cualquier software con una función distinta. La interfaz puede considerarse como un contrato de servicio entre dos aplicaciones. Este contrato define cómo se comunican entre sí mediante solicitudes y respuestas. La documentación de su API contiene información sobre cómo los desarrolladores deben estructurar esas solicitudes y respuestas. Las API son mecanismos que permiten a dos componentes de software comunicarse entre sí mediante un conjunto de definiciones y protocolos.[11]

**API de REST**: Estas son las API más populares y flexibles que se encuentran en la web actualmente. El cliente envía las solicitudes al servidor como datos. El servidor utiliza esta entrada del cliente para iniciar funciones internas y devuelve los datos de salida al cliente. Veamos las API de REST con más detalle a continuación.[11]

## **Sistemas de información Geográficos**

Un sistema de información geográfica (GIS) es un entorno para recopilar, gestionar y analizar datos. Arraigado en la ciencia de la geografía, el GIS integra muchos tipos de datos. Analiza la ubicación espacial y organiza capas de información en visualizaciones usando mapas y escenas 3D. Con esta capacidad única, los GIS revelan conocimientos más profundos de los datos, como patrones, relaciones y situaciones, ayudando a los usuarios a tomar decisiones más inteligentes.[12]

Cientos de miles de organizaciones nacionales e internacionales de prácticamente todos los campos están utilizando el GIS para hacer mapas que se comunican, realizan análisis, comparten información y resuelven problemas complejos. Esto está cambiando la forma en que funciona el mundo.[12]

### **Historia y evolución de los GIS**

Uno de los primeros casos de análisis espacial se produjo en 1854, cuando el médico británico John Snow cartografió las localizaciones de los brotes de cólera de Londres y otros datos geográficos. Descubrió que los casos de cólera se producían a lo largo de las líneas de agua.[4]

El campo de los sistemas de información geográfica (GIS) se inició en los años sesenta, cuando surgieron los ordenadores y los primeros conceptos de geografía cuantitativa y computacional. Los primeros trabajos de SIG incluían investigaciones importantes de la comunidad académica. Más tarde, el National Center for Geographic Information and Analysis (NCGIA), dirigido por Michael Goodchild, formalizó la investigación sobre temas clave relacionados con la ciencia de información geográfica, como el análisis y la visualización espaciales. Este trabajo impulsó una revolución cuantitativa en el mundo de la ciencia geográfica y sentó las bases para los SIG.[13]

La labor innovadora de Roger Tomlinson de iniciar, planear y desarrollar el Sistema Canadiense de Información Geográfica dio como resultado el primer SIG informatizado del mundo en 1963. El gobierno de Canadá encargó a Tomlinson la creación de un inventario manejable de sus recursos naturales. La intención era utilizar los ordenadores para unir los datos sobre los recursos naturales de todas las provincias. Tomlinson diseñó un sistema informático automatizado para almacenar y procesar grandes cantidades de datos, lo que permitió a Canadá empezar su programa nacional de administración del uso del suelo. También dio nombre a los SIG.[13]

Mientras estaba en la Universidad Northwestern en 1964, Howard Fisher creó uno de los primeros programas de software de representación cartográfica por ordenador, conocido como SYMAP. En 1965, creó el Harvard Laboratory for Computer Graphics. En este laboratorio se crearon y perfeccionaron algunos de los primeros programas de software, y se convirtió en un centro de investigación sobre el análisis y la visualización espaciales. Muchos de los conceptos iniciales sobre SIG y sus aplicaciones los concibieron en el laboratorio un grupo de talentosos geógrafos, planificadores, informáticos y otros expertos de varios ámbitos.[13]

En 1969, Jack Dangermond, miembro del laboratorio de Harvard, y su mujer Laura fundaron Environmental Systems Research Institute, Inc. (Esri). La consultoría aplicó la representación cartográfica por ordenador y el análisis espacial para ayudar a planificadores de uso del suelo y administradores de recursos del suelo a tomar decisiones informadas. Los trabajos iniciales de la empresa demostraron el valor de los SIG para resolver problemas. Esri continuó desarrollando muchos de los métodos de representación cartográfica SIG y análisis espacial que se utilizan hoy en día como ArcGis. Estos resultados generaron un mayor interés en las herramientas de software y los flujos de trabajo de la empresa que ahora son estándares en los SIG.[13]



Figura 2: Environmental Systems Research Institute(ESRI)[14]

En la década de 1970, unos ordenadores más rápidos, baratos y avanzados permitieron la comercialización de softwares SIG. Esto, junto con el auge de los satélites y la tecnología de teledetección, animó a gobiernos, empresas e instituciones académicas a adoptar los SIG.[4]

Con el aumento del potencial de la informática, Esri mejoró sus herramientas de software. Trabajar en proyectos que resolvían problemas reales llevó a la empresa a innovar y desarrollar herramientas de SIG sólidas y enfoques que se pudieran utilizar ampliamente. El trabajo de Esri se ganó el reconocimiento de la comunidad académica como una nueva forma de llevar a cabo análisis espaciales y planificaciones. Esri necesitaba analizar cada vez más proyectos con mayor efectividad, por lo que desarrolló ARC/INFO, el primer producto comercial de SIG. Esta tecnología se lanzó en 1981 y empezó la evolución de Esri hasta convertirse una empresa de software.[13]

Hoy en día, los SIG son omnipresentes. Se puede acceder fácilmente a datos SIG de código abierto del Servicio Geológico de Estados Unidos (USGS) y otros organismos, y a aplicaciones SIG (como QGIS) y sus tutoriales. Fuera de los ámbitos gubernamental y académico, los SIG se utilizan para rastrear paquetes, trazar rutas y compartir viajes. [4]

Los SIG ofrecen a la gente un modo de crear sus propias capas de mapas digitales para ayudar a resolver problemas reales. También han evolucionado hasta convertirse en un medio para intercambiar datos y colaborar, lo que ha alentado una visión que se está convirtiendo rápidamente en realidad: una base de datos de SIG continua, coincidente e interoperable del mundo, sobre prácticamente todos los temas. Hoy en día, cientos de miles de organizaciones comparten su trabajo y crean miles de millones de mapas cada día para contar historias y revelar patrones, tendencias y relaciones sobre todo.[13]

### **Ventajas que permiten las aplicaciones GIS**

* **Identificar problemas**: Usar el GIS para resolver problemas cuyo origen es un aspecto geográfico.[12]
* **Monitorizar el cambio**: Monitorizar de manera visibles cambios que pueden ocurrir en el aspecto geográfico de una determinada región. Por ejemplo: un mapa puede revelar de manera concisa la extensión del retroceso de los glaciares en el hemisferio sur. [12]
* **Administrar y responder a eventos**: El GIS proporciona un conocimiento de la situación en tiempo real. Por ejemplo: un mapa huracanes y ciclones puede mostrar el acto impacto que puede tener para las personas de una determinada región y poder darles seguimientos a los fenómenos en tiempo real para llevar a cabo una toma de decisiones que permita la menor cantidad de daños en la zona afectada. [12]
* **Realizar pronósticos**: Utilizar el Gis para poder realizar pronósticos y prever situaciones que conlleven a un mal funcionamiento de los renglones de la sociedad. Por ejemplo: Prever el tráfico de una avenida en fechas o días importantes, donde se prevé que generalmente el tráfico empeore en la zona.[12]
* **Establecer prioridades**: El GIS ayuda a establecer prioridades basadas en el análisis espacial. Analizando los patrones de los delitos, las fuerzas de seguridad pueden identificar las áreas objetivo y asignar oficiales en esas áreas.[12]

La tecnología de los GIS aplica la ciencia geográfica con herramientas para la comprensión y la colaboración. Ayuda a la gente a alcanzar un objetivo común: obtener inteligencia procesable a partir de todo tipo de datos.[12]

### **Campos que utilizan los sistemas de información geográfica**

* **Cartografía o mapeo**: Refinamiento de la cobertura de los datos de intervención, clasificación de la información del terreno, sistema de planificación de una ciudad, comunidad o edificios.[21]
* **Infraestructura**: Brindar una gran optimización y mantenimiento de sistemas de gas, agua, alcantarillado y teléfonos. También ayuda en la planificación de la red y otros propósitos.[15]
* **Recursos minerales**: Analizar elementos específicos y apoyar la interpretación de fallas y estructuras geológicas.[15]
* **Medio ambiente**: Evaluar el impacto de los proyectos de riego, reforestación, ecosistemas y agricultura.[15]
* **Geomarketing**: Determinar áreas de alcance de clientes potenciales, análisis de mercado estratégico y otros usos.[15]
* **Información de rutas**: Obtener modelos de tráfico y carreteras, planifique sistemas de tráfico, etc.[15]
* **Bienes inmobiliarios**: Determinar el precio del terreno del bloque; análisis de impacto del transporte y determinar el mejor uso.[15]
* **Atención sanitaria**: Enlistar los servicios, epidemiología y análisis y estudio de las necesidades de la comunidad. [15]

### **Ejemplos de sistemas GIS**

* **GRASS GIS**: Es un SIG que proporciona potentes capacidades ráster y vectorial, así como un motor de procesamiento geoespacial en una única suite integrada. Incluye herramientas para el modelado de terrenos y ecosistemas, **hidrología**, visualización de datos ráster y vectoriales, gestión y análisis de datos geoespaciales y procesamiento de imágenes satelitales y aéreas. Viene con un marco temporal para el procesamiento avanzado de series de tiempo y una **API de Python** para una programación geoespacial rápida. En las últimas versiones, GRASS GIS se ha optimizado para ofrecer un mayor rendimiento y análisis de grandes volúmenes de datos geoespaciales.[16]
* **ArcGIS de Esri:** ArcGIS es un software de mapeo GIS que proporciona una plataforma de análisis y mapeo para los datos geográficos capturados. ArcGIS amplía parte de la potencialidad única junto con licencias flexibles para aplicar análisis basados ​​en la ubicación a cualquier práctica comercial. Proporciona una perspectiva para visualizar y analizar sus datos y compartir los datos en forma de mapas, tableros, informes, etc. ArcGIS se puede usar como una aplicación independiente y combinarse con otras para respaldar el mapeo de ubicaciones. Es útil para trabajar con un conjunto de datos analíticos y algoritmos espaciales. El software ArcGIS es útil para monitorear la ubicación de cualquier tipo de sensor o dispositivo.[17]
* **MAPBOX**: Un mapa es una representación gráfica del área. El software de mapeo Mapbox GIS proporciona las herramientas de diseño de mapas y las bibliotecas de mapas necesarias para crear mapas dinámicos, de alto rendimiento y personalizados que se adapten a los requisitos. Los mapas generados por las herramientas de Mapbox son completos y precisos en todo el mundo. Sus API de mapas admiten más de 5 mil millones de solicitudes por día. MapBox hace uso de las tecnologías OpenGL para la representación en el dispositivo, lo que da como resultado mapas de alto rendimiento con la máxima flexibilidad, lo que permite aplicar estilos dinámicos y optimizar el mapa en función de sus datos personalizados.[17]
* **QGIS**: Es un software de sistema de información geográfica de código abierto, gratuito y fácil de usar que le permite crear, editar, visualizar, analizar y publicar información geoespacial en cualquier sistema operativo como Windows, Mac, sistemas Linux y admite vectores, raster, y formatos y funcionalidades de bases de datos. QGIS es un producto de la Open Source Geospatial Foundation (OSGeo). La publicación de proyectos de QGIS en la web se facilita con el beneficio de las potentes funciones de simbología, etiquetado y combinación para una mejor comprensión de los mapas.[17]

### **Gis en la nube**

El concepto “GIS en la nube o GIS Cloud” hace referencia a la utilización de servicios y aplicaciones GIS en Internet donde el usuario/cliente puede acceder mediante un navegador web sin necesidad de instalar ningún tipo de software en su ordenador. De este modo podemos utilizar una serie de aplicaciones GIS en la nube para publicar nuestros mapas y datos en la web.[18]

En los últimos años, las aplicaciones GIS en la nube se han consolidado y han mejorado de manera destacada sus funcionalidades de análisis, edición y visualización de datos. La mayoría de estos servicios permiten almacenar, analizar, publicar y visualizar datos geográficos en la nube, lo que está convirtiendo a estas aplicaciones en un complemento ideal y en una alternativa a los GIS tradicionales de escritorio[18]

La nube ha revolucionado la forma tradicional de trabajar de los GIS haciendo posible que, hoy en día, un gran número de usuarios sin necesidad de ser expertos sean capaces de realizar complejos análisis de datos para publicar y compartir con todo el mundo sus mapas online.[18]

La creciente demanda por parte de las empresas de realizar análisis de datos y geolocalizar la información ha ayudado de manera destacada al desarrollo de aplicaciones GIS en la nube.[18]

Con este nuevo escenario las principales empresas y organizaciones del sector ya están ofreciendo, en mayor o menor medida, parte de sus servicios en la nube ante las grandes posibilidades que ofrecen estos servicios.[18]

Ejemplos de aplicaciones Gis en la nube:

* **ArcGIS Online**: Es una plataforma de ESRI basada en la nube, utilizada para crear y compartir mapas, aplicaciones, contenido y datos geográficos. A través de su interfaz puedes acceder a la galería de mapas base y a una gran variedad de herramientas y funcionalidades que permiten trabajar, editar o analizar tu propia información para crear mapas web. ArcGIS Online dispone de potentes herramientas de análisis que permiten enriquecer la calidad de información del mapa, geocodificar y localizar cualquier lugar o realizar complejos análisis espaciales. Ofrece una plataforma geoespacial bastante completa donde los desarrolladores pueden crear aplicaciones personalizadas usando las API y SDK de ArcGIS para crear aplicaciones móviles y web. ArcGIS API for JavaScript, ArcGIS Runtime SDK for Android, ArcGIS Runtime SDK for Java, etc.
* [**GeoWE**](http://www.geowe.org/): Es un GIS en la web que tiene como principal objetivo trasladar toda la potencialidad de la **edición de datos geográficos** de los GIS de escritorio a la Web. Nace como un proyecto software Open Source basado en el framework Google Web Toolkit y orientado a la creación y edición de información geográfica, cuya culminación toma la forma de una aplicación cliente accesible desde diversos dispositivos.
* **GIS Cloud**: Es una aplicación con funcionalidades GIS en la nube que permite alojar, gestionar y acceder a datos geoespaciales en la nube para crear y compartir mapas en la web a través de la herramienta **Map Editor.** **Map Editor**permite que cualquier usuario, de forma fácil y sencilla, pueda editar, analizar y visualizar información geográfica. Uno de sus puntos fuertes es la posibilidad de trabajar con bases de datos espaciales **PostGIS** y utilizar su potencial para realizar diversas y complejas consultas espaciales..
* **Google My Maps**: Es un servicio de Google que permite crear mapas personalizados sobre los mapas base de Google Maps. El único requisito para poder utilizar este servicio es disponer de una cuenta de Google. Esta diseñado para que los usuarios puedan crear y aplicar estilos a los mapas a partir de datos geográficos, de forma rápida y sencilla, para compartirlos y publicarlos en Internet. Es una aplicación escalable, fácil de usar y dispone de aplicación móvil. Una de sus principales características es la posibilidad de realizar geo codificación con Google Maps Geocoding API.

Con esta información, se puede concluir que es factible desarrollar un sistema de Información Geográfica (GIS) basado en la web o en la nube, que proporcione servicios accesibles a los usuarios mediante una conexión a Internet y un navegador web.

## **Web**

WEB (World Wide Web, o www), es un conjunto de documentos (webs) interconectados por enlaces de hipertexto, disponibles en Internet que se pueden comunicar a través de la tecnología digital. Se entiende por “hipertexto” la mezcla de textos, gráficos y archivos de todo tipo, en un mismo documento.[19]

Web no son sinónimo de Internet; Internet es la red de redes donde reside toda la información, siendo un entorno de aprendizaje abierto, más allá de las instituciones educativas formales. La web es un subconjunto de Internet que contiene información a la que se puede acceder usando un navegador. Tanto el correo electrónico, como facebook, twiter, wikis, blogs, juegos, etc. son parte de Internet, pero no la web.[19]

Estos sistemas son diseñados para ser independientes de plataformas específicas o sistemas operativos, lo que los hace altamente accesibles y flexibles. Al estar alojados en servidores remotos, los usuarios pueden acceder a ellos desde cualquier dispositivo con conexión a internet, lo que facilita la colaboración y el trabajo remoto.[20]

### **Beneficios de las aplicaciones web**

Las aplicaciones web tienen muchos beneficios, y casi todas las empresas grandes las utiliza como parte de sus ofertas para usuarios. A continuación, se muestran alguno de los beneficios comunes asociados a las aplicaciones web.[9]

* **Accesibilidad**: Las aplicaciones web son accesibles desde todos los navegadores web y desde diferentes dispositivos personales y empresariales. Equipos de diferentes ubicaciones pueden acceder a documentos compartidos, sistemas de administración de contenidos y otros servicios empresariales a través de aplicaciones web basadas en suscripciones.[9]
* **Desarrollo eficiente**: El proceso de desarrollo para aplicaciones web es relativamente sencillo y rentable para las empresas. Los equipos pequeños pueden lograr ciclos de desarrollo cortos, lo que hace que las aplicaciones web sean una manera eficiente y asequible de desarrollar programas de computación. Además, dado que la misma versión funciona en todos los navegadores y dispositivos modernos, no tendrá que crear un número elevado de iteraciones diferentes para varias plataformas.[9]
* **Simplicidad para el usuario**: Los usuarios no tienen que descargar las aplicaciones web, lo que hace que sean fáciles de acceder a la vez que se prescinde de mantenimiento y capacidad en el disco duro por parte del disco duro. Las aplicaciones web reciben actualizaciones de software y seguridad de manera automática, lo que significa que siempre están actualizadas y presentan menor riesgo de sufrir brechas de seguridad.[9]
* **Escalabilidad**: Las empresas que utilizan aplicaciones web pueden agregar usuarios cuando sea necesario, sin necesidad de infraestructura adicional o hardware costoso. Además, la mayor parte de los datos de las aplicaciones web se almacena en la nube, lo que significa que su empresa no tendrá que invertir en capacidad de almacenamiento adicional para ejecutar aplicaciones web.[9]
* **Actualizaciones automáticas**: Las actualizaciones de las aplicaciones web suelen implementarse automáticamente, lo que garantiza que siempre tengas acceso a la última versión del software.
* **Datos en la nube:** la mayor parte de las aplicaciones web guardan la información en la nube, lo que permite acceder a tus datos desde cualquier lugar y reduce el riesgo de perderlos.

### **Navegadores web más utilizados**

La funcionalidad de los navegadores web va más allá de la mera búsqueda de información; son claves en la conectividad global. Permiten el acceso a redes sociales, plataformas educativas, portales de noticias, tiendas en línea… eliminando las barreras físicas y temporales para la comunicación y el comercio. Esta conectividad no solo enriquece nuestro conocimiento y perspectivas, sino que también abre puertas a nuevas oportunidades de aprendizaje, trabajo y entretenimiento. La elección del navegador web afecta nuestra velocidad y modo de acceso a recursos online, subrayando la importancia de elegir uno que se ajuste a nuestras necesidades y gustos.[21]

Los datos presentados a continuación fueron extraídos de la página We Are Social. Listado de navegadores más utilizados (Top 3 navegadores):

1. **Google Chrome (64,70 %)**: Se posiciona como el navegador web más popular y utilizado en el mundo, con una impresionante cuota de mercado del 64,70 %. Desarrollado por el gigante tecnológico Google, este navegador ha conquistado la confianza de millones de personas gracias a su rapidez, seguridad y sencillez de manejo. Su diseño intuitivo y su integración con servicios de Google, como Gmail y Google Drive, hacen de Chrome una herramienta clave para una navegación eficaz y personalizada. Además, su enfoque constante en innovación y seguridad protege contra amenazas digitales online. Indiscutiblemente, Google Chrome no solo lidera el mercado por su extenso uso, sino también por ser una plataforma sólida y de confianza para explorar internet.[21]
2. **Safari (18,59 %)**: El navegador preferido en dispositivos Apple, ocupa el segundo lugar en popularidad con un 18,59 % de cuota de mercado. Su profunda integración con el ecosistema Apple garantiza una navegación fluida y coherente en todos los dispositivos, desde MacBooks hasta iPhones. Safari destaca por su enfoque en la privacidad y la eficiencia energética, mejorando el rendimiento y extendiendo la vida útil de la batería. Funciones como el bloqueo inteligente de rastreadores y una interfaz clara protegen los datos personales y facilitan una navegación ágil. Safari es ideal para usuarios de Apple que buscan seguridad, privacidad y una experiencia web integrada.[21]
3. **Microsoft Edge (4,96 %)**: El sucesor del legendario Internet Explorer, ha logrado posicionarse firmemente como una opción moderna y competitiva en el mundo de los navegadores web, ostentando un 4,96 % de cuota de mercado. Completamente rediseñado, Edge opera con el mismo motor que Google Chrome (Chromium), facilitando así una navegación rápida y compatible con una extensa variedad de sitios web. Su característica distintiva es la profunda integración con el ecosistema de Microsoft, ofreciendo funcionalidades exclusivas como la incorporación de Cortana, el asistente digital de Microsoft, y la posibilidad de realizar anotaciones directamente en las páginas web. Además, su compromiso con la privacidad y seguridad, mediante herramientas como el bloqueo de rastreadores y una navegación protegida, garantiza una experiencia en la web notablemente segura. Edge representa una opción sobresaliente para quienes priorizan la productividad y desean una integración fluida con los servicios de Microsoft, brindando eficiencia en la navegación y un conjunto de herramientas innovadoras.[21]

Con la presentación de dicha estadística se evidencia que el principal navegador en el mercado es Google Chrome que comprende el 64.70% del mercado, pero si tenemos en cuenta los 3 primeros puestos se evidencia una cuota de mercado de: 88.25% de cuota de mercado. Dicha conclusión, añade un nuevo requisito: el sistema ha desarrollar, debe funcionar en la mayoría de los navegadores del mercado, siempre teniendo en cuenta que algunas tecnologías son incompatibles con algunos navegadores.

## **Selección tecnológica**

### **Lenguaje de programación**

#### **Java**

Java es un lenguaje de programación de alto nivel y orientado a objetos que se utiliza ampliamente en la creación de aplicaciones de software y en el desarrollo de aplicaciones web. Fue desarrollado originalmente por Sun Microsystems (ahora propiedad de Oracle Corporation) en la década de 1990.[22]

#### **Ventajas de Java**

1. Es un lenguaje de programación multiplataforma: Un lenguaje de programación es multiplataforma cuando puede utilizarse para desarrollar aplicaciones que se ejecutan en diferentes plataformas o sistemas operativos, como Windows, Mac, Linux, iOS, Android, etc. Esto se debe a que el código fuente de las aplicaciones escritas en un lenguaje multiplataforma puede ser compilado o interpretado de manera similar en diferentes sistemas, lo que significa que no es necesario escribir distintas versiones de la aplicación para cada plataforma específica. Permite a los desarrolladores y desarrolladoras escribir una sola vez el código fuente de una aplicación y luego compilarlo o interpretarlo en diferentes plataformas. Esto ahorra tiempo y esfuerzo, ya que los desarrolladores no tienen que mantener diferentes versiones del código para cada plataforma. Además, esto también significa que las aplicaciones escritas en un lenguaje multiplataforma pueden distribuirse fácilmente en diferentes plataformas sin necesidad de realizar modificaciones significativas en el código.[22]
2. Ofrece una amplia gama de librerías y herramientas: Las librerías de Java son colecciones de código predefinido que proporcionan funcionalidades comunes y útiles para los desarrolladores. Estas bibliotecas permiten a los programadores reutilizar código ya escrito y probado en lugar de tener que escribir todo el código desde cero, lo que ahorra tiempo y reduce errores. En este sentido, las bibliotecas de Java están estrechamente relacionadas con el concepto de DRY (Don’t Repeat Yourself), que es un principio de programación que sugiere que el código debe escribirse una sola vez y que no debe haber duplicación innecesaria del mismo código en diferentes lugares del programa. Esto mejora la legibilidad, el mantenimiento y la escalabilidad del código, ya que cualquier cambio realizado en un lugar se refleja en las demás instancias del mismo código.[22]
3. Cuenta con un sistema de seguridad incorporado: El sistema de seguridad de Java es un conjunto de mecanismos que ayudan a proteger las aplicaciones Java de posibles amenazas de seguridad, como virus, malware, ataques de hackers y otros tipos de intrusos. Estos mecanismos están diseñados para proporcionar una capa adicional de protección a los programas Java y a sus usuarios.[22]
4. Es un lenguaje de programación orientado a objetos: El enfoque orientado a objetos de Java permite a los programadores reutilizar el código existente. Por ejemplo, una clase definida para un objeto puede ser utilizada en otro programa Java, lo que ahorra tiempo y esfuerzo en la creación de un nuevo código. Las clases en Java están diseñadas para ser modulares y fácilmente integrables en diferentes partes de un programa. Esto ayuda a organizar el código en bloques lógicos y simplifica la creación, mantenimiento y actualización de programas Java. La encapsulación es una técnica utilizada en Java para ocultar los detalles de implementación de un objeto. Esto significa que los datos y métodos de un objeto están protegidos y solo se pueden acceder a ellos mediante métodos definidos en la misma clase, lo que garantiza la seguridad y la integridad de los datos. La abstracción es un concepto clave en la programación orientada a objetos que permite a los programadores definir un objeto en términos generales, sin tener que preocuparse por los detalles de su implementación. El polimorfismo es otra técnica utilizada en la programación orientada a objetos que permite a un objeto ser tratado de diferentes maneras.[22]
5. Es un lenguaje de programación de alto nivel: Al ser un lenguaje de programación de alto nivel, Java es fácilmente entendido y utilizado por humanos. Está diseñado para ser independiente de la arquitectura de la computadora y del sistema operativo subyacente, lo que significa que el programador no necesita preocuparse por los detalles de bajo nivel de la computadora.[22]

#### **Desventajas de Java**

1. Tiene un rendimiento más lento: Java tiene un rendimiento más lento en comparación con otros lenguajes de programación, como C y C++. El código fuente se compila a bytecode, que es interpretado por la Máquina Virtual de Java (JVM) durante la ejecución del programa. Además, Java utiliza un compilador JIT (Just-In-Time), que compila el bytecode en código de máquina en tiempo de ejecución. Este proceso de compilación y ejecución puede ralentizar el rendimiento del programa en comparación con lenguajes que se compilan directamente en código de máquina. La administración de memoria es gestionada automáticamente por la JVM, lo que significa que el programador no tiene que preocuparse por la asignación y liberación de memoria. Sin embargo, esto puede ralentizar el rendimiento del programa debido a la sobrecarga de la administración de memoria.[22]
2. Puede requerir más memoria que otros lenguajes: Una de las características clave de Java es su administración automática de memoria. En Java, la administración de memoria es gestionada automáticamente por la Máquina Virtual de Java (JVM). Esto significa que la JVM se encarga de asignar y liberar memoria según sea necesario para el programa en tiempo de ejecución. Aunque esto ofrece muchas ventajas, como la eliminación de problemas comunes de memoria como fugas de memoria y referencias no válidas, también puede aumentar el uso de memoria del programa, ya que la JVM necesita tener una cantidad considerable de memoria disponible para realizar su trabajo. Además, Java utiliza objetos para representar datos, y cada objeto en Java requiere memoria para su creación y almacenamiento. Esto significa que si un programa en Java utiliza una gran cantidad de objetos, puede requerir más memoria que un programa equivalente escrito en un lenguaje que no utiliza objetos, como C. Java también requiere memoria adicional para cargar sus bibliotecas y clases estándar. Aunque un programa en Java puede ser relativamente pequeño, la JVM y sus bibliotecas pueden precisar una cantidad significativa de memoria para funcionar correctamente.[22]

### **Bases de datos**

#### **PostgreSQL**

PostgreSQL también conocido simplemente como Postgre, es un gestor de bases de datos relacionales gratuito que cuenta con gran aceptación hoy en día. Es una alternativa real y eficiente a MySQL, que ofrece una gran cantidad de opciones avanzadas y la posibilidad de manejar grandes volúmenes de información de forma eficiente.[23]

#### **Ventajas de PostgreSQL**

* **Instalación ilimitada y gratuita**: Podemos instalarlo en todos los equipos que queramos. Independientemente de la plataforma y la arquitectura que usemos, PostgreSQL está disponible para los diferentes SO, Unix, Linux y Windows. Esto hace de PostgreSQL un sistema multiplataforma y también hace que sea más rentable con instalaciones a gran escala.[24]
* **Gran escalabilidad**: Nos permite configurar PostgreSQL en cada equipo según el hardware. Por lo que es capaz de ajustarse al número de CPU y a la cantidad de memoria disponible de forma óptima. Con ello logramos una mayor cantidad de peticiones simultáneas a la base de datos de forma correcta.
* **Estabilidad y confiabilidad**: Tiene más de 20 años de desarrollo activo y en constante mejora. No se han presentado nunca caídas de la base de datos. Ésto es debido a su capacidad de establecer un entorno de Alta disponibilidad y gracias a Hot-Standby, que permite que los clientes puedan realizar consultas de solo lectura mientras que los servidores están en modo de recuperación o espera. Así podemos hacer tareas de mantenimiento o recuperación sin bloquear completamente el sistema.
* **pgAdmin**: Se trata de una herramienta gráfica con la que podemos administrar nuestras bases de datos de forma fácil e intuitiva. Podemos ejecutar sentencias SQL, e incluso crear copias de seguridad o realizar tareas de mantenimiento.
* **Estándar SQL**: Implementa casi todas las funcionalidades del estándar ISO/IEC 9075:2011, así pues, resulta sencillo realizar consultas e incluir scripts de otros Motores de Bases de Datos.
* **Potencia y Robustez**: PostgreSQL cumple en su totalidad con la característica ACID Compliant. ACID es un acrónimo de Atomicity, Consistency, Isolation y Durability (Atomicidad, Consistencia, Aislamiento y Durabilidad en español). Por ello permite que las transacciones no interfieran unas con otras. Con ello se garantiza la información de las Bases de Datos y que los datos perduren en el sistema.
* **Extensibilidad**: Tenemos a nuestra disponibilidad una gran variedad de extensiones distribuidas por el grupo de desarrolladores de PostgreSQL. También por terceros o incluso nosotros mismos podemos crear nuestras propias extensiones. Estas extensiones pueden ser lenguajes de programación, tales como, Perl, Java, Python, C++ y muchos más.

#### **Desventajas de PostgreSQL**

* **Velocidad**: Es relativamente lento en inserciones y actualizaciones en bases de datos pequeñas, PostgreSQL está diseñado para ambientes de alto volumen. Ésto hace que la velocidad de respuesta pueda parecer lenta en comparación con bases de datos de pequeño tamaño.
* **Soporte oficial**: No cuenta con un soporte en línea o telefónico. PostgreSQL cuenta con foros oficiales donde los usuarios pueden exponer sus dudas que responden otros usuarios de la comunidad. Cabe resaltar que la comunidad de usuarios PostgreSQL es una de las más activas en el mercado.

#### **PostGIS**

Es un software compatible con Open Geospatial Consortium (OGC) utilizado como una extensión para PostgreSQL, que es una forma de base de datos objeto-relacional. Aunque PostGIS es libre y de código abierto, se utiliza tanto en software comercial (por ejemplo, ArcGIS) como en software de código abierto (por ejemplo, QGIS). PostGIS amplía las capacidades de PostgreSQL a fin de aumentar sus capacidades de gestión mediante la adición de tipos y funciones geoespaciales para mejorar los datos espaciales manejados dentro de una estructura de base de datos relacional.[25]

### **Framework para el desarrollo del servidor**

#### **Spring Framework**

Spring es un framework de código abierto para la creación de aplicaciones empresariales Java, con soporte para Groovy y Kotlin. Tiene una estructura modular y una gran flexibilidad para implementar diferentes tipos de arquitectura según las necesidades de la aplicación.[26]

#### **Ventajas de Spring Framework**

1. **Facilidad de Integración**: Una de las principales ventajas de Spring Framework es su facilidad de integración con otras tecnologías y frameworks. Spring se integra sin problemas con Hibernate, Struts, JPA y muchas otras tecnologías populares, lo que facilita la construcción de aplicaciones complejas.[27]
2. **Inversión de Control (IoC)**: Spring Framework utiliza el principio de Inversión de Control (IoC), lo que significa que gestiona la creación y el ciclo de vida de los objetos en la aplicación. Esto simplifica el desarrollo y facilita la prueba de las aplicaciones.[27]
3. **Modularidad**: Spring se basa en un enfoque modular, lo que significa que puedes utilizar solo los módulos que necesitas en tu aplicación. Esto ayuda a mantener el código limpio y facilita la escalabilidad de la aplicación.[27]
4. **Soporte Activo**: La comunidad de Spring es muy activa y ofrece un amplio soporte a los desarrolladores. Puedes encontrar una gran cantidad de recursos, documentación y tutoriales en línea que te ayudarán a aprovechar al máximo Spring Framework.[27]

#### **Desventajas de Spring Framework**

1. **Curva de Aprendizaje**: Debido a su amplia gama de funcionalidades, Spring Framework puede tener una curva de aprendizaje pronunciada para los desarrolladores principiantes. Es importante dedicar tiempo a comprender los conceptos fundamentales de Spring para utilizarlo de manera efectiva.[27]
2. **Configuración Compleja**: En ocasiones, la configuración de Spring puede resultar compleja y tediosa, especialmente en proyectos de gran escala. Es importante planificar y organizar la configuración de manera eficiente para evitar posibles problemas en el futuro.[27]
3. **Dependencia de Spring**: Al utilizar Spring Framework, tu aplicación puede volverse altamente dependiente de Spring y sus módulos. Esto puede limitar la portabilidad de la aplicación y dificultar la migración a otras plataformas en el futuro.[27]

### **Conexión a la base de datos:**

#### **JPA**

JPA es la propuesta estándar que ofrece Java para implementar un Framework Object Relational Mapping (ORM), que permite interactuar con la base de datos por medio de objetos, de esta forma, JPA es el encargado de convertir los objetos Java en instrucciones para el Manejador de Base de Datos (MDB).[28]

#### **Hibernate**

Hibernate es una herramienta de mapeo objeto-relacional (ORM) bajo licencia GNU LGPL para Java, que facilita el mapeo de atributos en una base de datos tradicional, y el modelo de objetos de una aplicación mediante archivos declarativos o anotaciones en los beans de las entidades que permiten establecer estas relaciones. Todo lo dicho, que suena a vendedor de seguros, se resume en que agiliza la relación entre la aplicación y nuestra base de datos SQL, de un modo que optimiza nuestro flujo de trabajo evitando caer en código repetitivo.[29]

Imagina un programa sencillo. Necesitas un método que permita dar de alta, baja, o modificar los datos de usuarios. Estos datos se almacenan en una base de datos, y cada objeto tiene diferentes clases. Para cada objeto debemos crear una clase que al menos permita crear, insertar, eliminar, consultar o modificar la información contenida en sus atributos. Esto, con excepción de alguna consulta un poco especial, es siempre lo mismo (SELECT, UPDATE, CREATE, FROM, WHERE, GROUP BY, ORDER BY y etc). Y no sé tú, pero no me gusta desperdiciar mi café en esas tareas cuando debo preocuparme de las consultas especiales que es donde suele estar el tomate.[29]

#### **Spring Data JPA**

Esta biblioteca es parte de Spring Framework, uno de los marcos de Java más populares en la actualidad. El objetivo de Spring Data es reducir la cantidad de repeticiones necesarias para implementar capas de acceso a datos para varias bases de datos.[30]

Spring Data JPA es una biblioteca que agrega una capa adicional de abstracción además de la implementación de ORM JPA. De forma predeterminada, Spring Data JPA utiliza Hibernate como proveedor de ORM (para ejecutar consultas). Esto, por cierto, se puede cambiar usando la configuración de Spring. Aunque no recomendaría hacer esto a usuarios inexpertos.[30]

#### **Ventajas de usar JPA**

* No es necesario escribir consultas DDL/DML, en su lugar podemos mapear usando XML/anotaciones.[31]
* Se usa JPQL y, dado que es independiente de la plataforma, no necesitamos depender de ninguna tabla SQL nativa. Las expresiones complejas y las expresiones de filtrado se manejan solo a través de JPQL.[31]
* La entidad puede almacenarse parcialmente en una base de datos como MySQL y el resto puede estar en el sistema de gestión de bases de datos Graph.[31]
* Es posible la generación dinámica de consultas.[31]
* La integración con Spring Framework es más fácil con un espacio de nombres personalizado.[31]

### **Sistema para el desarrollo de interfaces visuales**

#### **React**

React es una de las bibliotecas JavaScript más populares entre los desarrolladores Front End, programadores y testers de software. Te permite crear interfaces de usuario interactivas, ahorrando tiempo y reduciendo los costos de desarrollo. Sigue leyendo para descubrir los principios de React.[32]

#### **Ventajas de React**

1. Los componentes de React agilizan la creación de una interfaz sensible a cualquier cambio en un sitio web o una aplicación de cualquier complejidad.
2. Gracias al DOM virtual, la biblioteca ahorra recursos y tráfico.
3. El código de React tiene una lógica clara, es fácil de leer, entender y depurar, lo que ayuda a reducir errores.
4. Las interfaces interactivas creadas con React garantizan una mejor experiencia de usuario.
5. React es fácil de aprender, tiene una documentación accesible y muchos recursos gratuitos online.
6. Dominar React es una de las habilidades más demandadas para conseguir el trabajo de desarrollo Front End.

#### **Desventajas de React**

1. Necesitas un conocimiento sólido de HTML y JavaScript para aprender la sintaxis de JXS.
2. La biblioteca puede aumentar el tamaño de tu aplicación.
3. React solo visualiza la interfaz, pero para crear un proyecto completo, necesitas una pila de tecnología.

### **Controlador de versiones**

#### **Git**

Git es un sistema de control de versiones distribuido, lo que significa que un clon local del proyecto es un repositorio de control de versiones completo. Estos repositorios locales plenamente funcionales permiten trabajar sin conexión o de forma remota con facilidad. Los desarrolladores confirman su trabajo localmente y, a continuación, sincronizan la copia del repositorio con la del servidor. Este paradigma es distinto del control de versiones centralizado, donde los clientes deben sincronizar el código con un servidor antes de crear nuevas versiones.[33]

### **Repositorio de código**

#### **GitHub**

Github es un portal creado para alojar el código de las aplicaciones de cualquier desarrollador, y que fue comprada por Microsoft en junio del 2018. La plataforma está creada para que los desarrolladores suban el código de sus aplicaciones y herramientas, y que como usuario no sólo puedas descargarte la aplicación, sino también entrar a su perfil para leer sobre ella o colaborar con su desarrollo.[34]

### **Entorno de desarrollo integrado (IDE) / Editor de código**

#### **Kit de herramientas JetBrains**

Paquetería de software de desarrollo compuesta por 12 IDE entre los cuales se pueden encontrar: Intellij Idea Ultimate, Web Storm, Rider, Data Spell; 3 extensiones y 2 perfiladores[35]

#### **Visual Studio Code**

Editor de código fuente independiente que se ejecuta en Windows, macOS y Linux. La elección principal para desarrolladores web y JavaScript, con extensiones para admitir casi cualquier lenguaje de programación.[36]

# **Solución propuesta**

## **Introducción**

Para comprender el contexto en el que se desarrolla un proceso específico, es fundamental identificar las entidades gestionadas dentro del negocio, los actores que las manipulan, sus responsabilidades y qué información resulta relevante. En este capítulo, se presenta una descripción del funcionamiento actual del sistema, el modelo de dominio correspondiente, así como el diagrama de casos de uso que representa las interacciones principales. Además, se define la descripción detallada de los casos de uso más relevantes, con énfasis en la funcionalidad del sistema y sus requisitos funcionales y no funcionales. También se ahondará en la seguridad del sistema y como varios mecanismos en influyen de manera que la solución propuesta pueda enfrentarse a varias de las ciber amenazas que abundan en el mercado actual.

## **Descripción del negocio**

El sistema en desarrollo es un sistema de gestión geográfico que pretende mejorar la manera en que se analizan, procesar y visualizan datos geoespaciales, permitiendo al usuario final extraer de manera más sencilla los datos almacenados en la aplicación. El sistema está orientado para facilitar la migración de datos almacenados en fuentes externas: archivos Excel, con el objetivo de proporcionar una plataforma visual interactiva que permita a los usuarios analizar los datos y visualizar reportes.

La solución permite a los usuarios cargar, procesar y gestionar grandes volúmenes de datos geográficos, optimizando así la toma de decisiones basadas en la ubicación. Esto se logra a través de una interfaz web intuitiva, accesible desde cualquier dispositivo, que garantiza una experiencia fluida y segura.

El negocio comienza cuando el usuario, se autentifica en el sistema para comprobar su identidad. Si el proceso anteriormente explicado es satisfactorio, el usuario puede importar desde su almacenamiento local, un fichero Excel (extensión xls o xlsx) para que sistema extraiga toda la información almacenada y la inserte en un base de datos. Desde este punto, cualquier usuario de la aplicación puede analizar y visualizar los datos en un mapa de coordenada cartesianas.

## **Descripción del sistema**

El sistema está diseñado para extraer datos almacenados en archivos Excel, analizar su contenido, y filtrar aquellos que cumplan con los requisitos establecidos. Los datos que pasen las validaciones serán ingresados automáticamente en la base de datos del sistema. Además, el sistema se integra con un servicio externo para procesar y extraer información geográfica de las direcciones almacenadas, permitiendo enriquecer los datos con coordenadas y otros atributos geoespaciales.

Una vez procesados, el sistema permite la visualización de estos datos geográficos en un mapa interactivo, proporcionando una representación clara y dinámica de la información. También ofrece la funcionalidad de generar e imprimir reportes sobre diversos aspectos de los datos almacenados, facilitando así la toma de decisiones y el análisis de información.

## **Modelo de dominio**

El modelo de dominio es un artefacto de la disciplina de análisis, construido con las reglas de Lenguaje Unificado de Modelado (UML, por sus siglas en inglés) durante la fase de concepción. Este no contiene conceptos propios de un sistema de software, sino de la propia realidad física. La utilidad del mismo radica en ser una forma de “inspiración” para el diseño de los objetos de software, y es la entrada para muchos de los artefactos que se construyen en un proceso software

Un modelo de dominio se centra en las abstracciones relevantes, vocabulario del dominio e información del dominio. Este permite describir las entidades que conforman el dominio, sus relaciones y atributos. En él se representan los conceptos del dominio, mostrando aspectos estáticos. Además, es el artefacto clave del análisis orientado a objetos

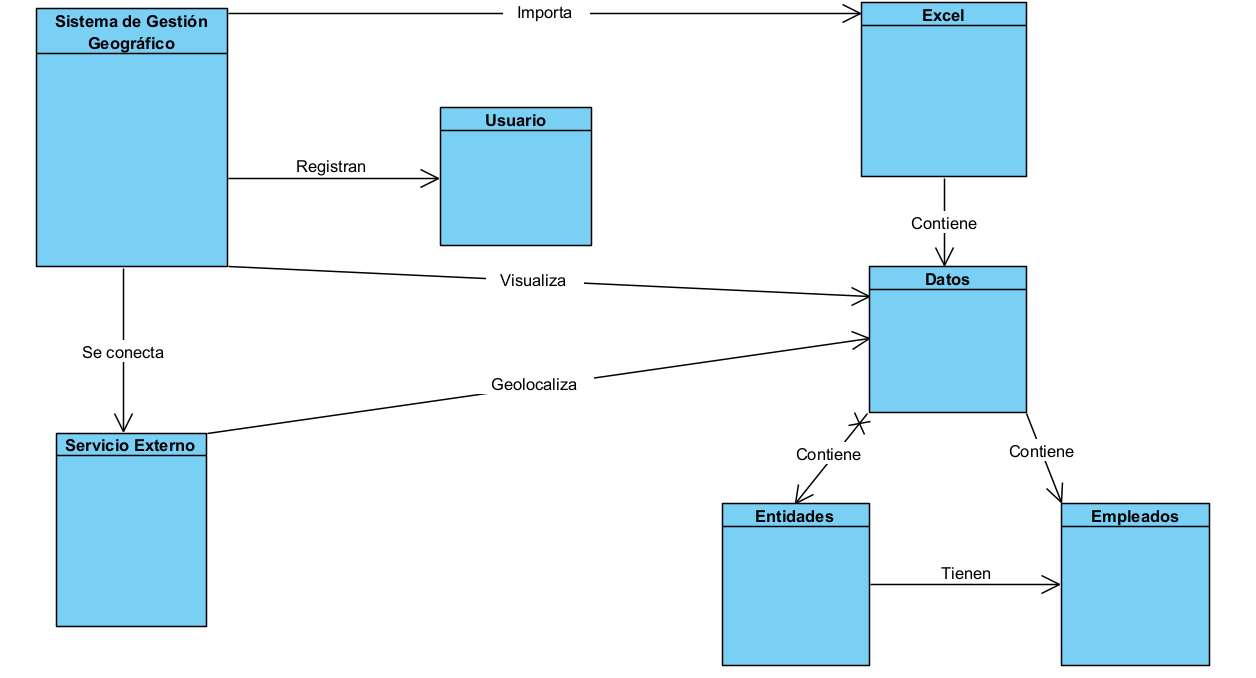


Figura 3: Modelo de dominio

## **Reglas de negocio**

Las reglas de negocio guían la toma de decisiones diarias dentro de las empresas trazando las relaciones entre los objetos, como los nombres de cliente y sus pedidos correspondientes. Esta conversión de las actividades de negocio de una organización en lógica empresarial concreta permite a los ingenieros de software y analistas de negocio aplicar estas reglas en herramientas de flujos de trabajo u otras aplicaciones para habilitar la automatización de procesos. Sin ellas, los procesos de actualización pueden volverse más pesados y lentos, y los documentos están más expuestos a error humano e inconsistencias. Una empresa que implementa reglas de negocio puede ahorrar tiempo y dinero gracias a la optimización del trabajo y un menor abandono.

|  |  |
| --- | --- |
| Patrón | Regla |
| Validación | Cada entidad o empleado insertado en el sistema debe tener una dirección afiliada |
| Estímulo/Respuesta | Si cuando se procesa un Excel todos los datos son correctos, se le notifica al usuario, en caso contrario se debe retornar que datos fueron erróneos y cuál fue su causa |
| Autorización | Solamente los usuarios con rol Administrador o superior puede autorizar la creación de nuevos usuarios |
| Cumplimiento | Implementación de una seguridad cumpla con las normativas de seguridad cumpliendo con el artículo 19 del reglamento de seguridad para las tecnologías de la información |
| Contingencia | El sistema a través de la API Gateway y el balaceo permite, en caso de distribuirse, se puedan redirigir todas las conexiones a otros servidores |
| Precondición | Para procesar un Excel debe tener como extensión xls o xlsx |

Tabla 1: Reglas de negocio

## **Requisitos funcionales**

* El software debe permitir exporta e importar datos.
* El software debe permitir almacenar datos en una base de datos
* El software debe permitir en proceso de importación de los datos, tratar errores de manera que el proceso no se detenga, pero debe informarle al usuario los errores encontrados.
* El sistema debe tener un sistema de seguridad que mantenga la integridad de los datos.
* El software no debe encargarse de la gestión de la información, pero debe encargarse de su análisis, pero debería tener módulos que puedan gestionar de manera manual los datos
* El software debe realizar una matriz destino-origen para almacenar la información a través de los filtrados
* El software debe permitir señalar los datos en un mapa
* El sistema debe permitir el filtrado de datos en su presentación.
* El software deberá ampliar sus funcionalidades en harás de permitir mayor usabilidad.

## **Requisitos no funcionales**

### **Requisitos de apariencia**

* Profesional o del tipo ejecutivo: El software será utilizado en un ambiente profesional como empresas.
* Muy legible: El usuario de un vistazo debe ser capaz de encontrar o reconocer las distintas funcionales, sin perderse. Para eso apoyarse en el buen uso de los iconos y en los nombre cortos y concisos en la interfaz que resuman las ideas.
* Simple de usar: Cualquier usuario sea tanto del ámbito profesional o debe poder ser capaz de entender la interfaz del programa.

### **Requisitos de usabilidad**

* Porciento de aceptación de los usuarios: Conocer la satisfacción de los usuarios con el software desarrollado, buscando puntos para mejorar, puntos positivos y negativos.
* Facilidad de uso por personas que hablen otros idiomas distintos al del país donde el producto fue creado: Buscando internacionalidad el software o poder presentarlo para personas de muchos países.
* Documentación de usuario, material de entrenamiento: Material de guía sobre el uso del sistema.
* Facilidad de uso por personas sin experiencia previa con las computadoras: Comprender que el usuario final de la aplicación puede ser un novato en la utilización de las computadoras.

### **Requisitos de rendimiento**

* Velocidad de procesamiento: Velocidad en el momento de geo codificar los datos y presentarlos al cliente.
* Disponibilidad: El sistema debe estar disponible todo el tiempo, para así poder garantizar su uso en todo momento.

### **Requisitos de soporte**

* Prueba: Se debe probar todo el sistema permitiendo depurar errores en los distintos momentos del sistema y permitiendo saber si tiene un correcto funcionamiento en todo momento.
* Instalación: Se deben instalar todos los programas necesarios para la ejecución del programa.

### **Requisitos de portabilidad**

* El sistema debe utilizarse en Windows o Linux. En caso de una futura versión web en cualquier dispositivo.

### **Requisitos de seguridad**

* Integridad: La información debe ser cuidada de manera exhaustiva al ser una información potencialmente delicada para el país.

### **Requisitos de confiabilidad**

* Protección contra fallos: Los distintos fallos que puede tener el sistema, han sido tratados con mucho cuidado

### **Requisitos de software**

* Máquina virtual de Java como mínimo 17
* EL Windows necesario mínimo es Windows 7
* La base de datos PostgresSQL 12 y con PgAdmin 4

### **Requisitos de hardware**

* Una computadora de escritorio o Laptop como mínimo 2GB de Ram.
* Una conexión por Modem a un servidor.
* Un servidor con 100 GB de almacenamiento

## **Diagrama de caso de uso del sistema**

El diagrama de casos uso es un tipo de diagrama que permite mediante una representación gráfica, reflejar la relación existente entre los diferentes roles o interacciones de personas o cosas dentro un sistema[37]

El diagrama de casos de uso es una representación gráfica que permite visualizar diferentes roles en un sistema y las interacciones para la ejecución de funciones o actividades, sean estos desempeñados por personas o cosas. La utilidad de este sistema se basa en representar las interacciones en la ejecución de diferentes roles dentro de un mismo sistema, lo que facilita la ejemplificación desde el punto de vista de los ejecutores *“actores”*el funcionamiento de sus principales funciones.[37]

Este diagrama es útil en las organizaciones para guiar a los equipos de trabajo en el desempeño de cada función asignada y cómo cada miembro debe interactuar con el resto del equipo, maquinarias u objetos para el desempeño de sus actividades mediante esta representación gráfica.[37]

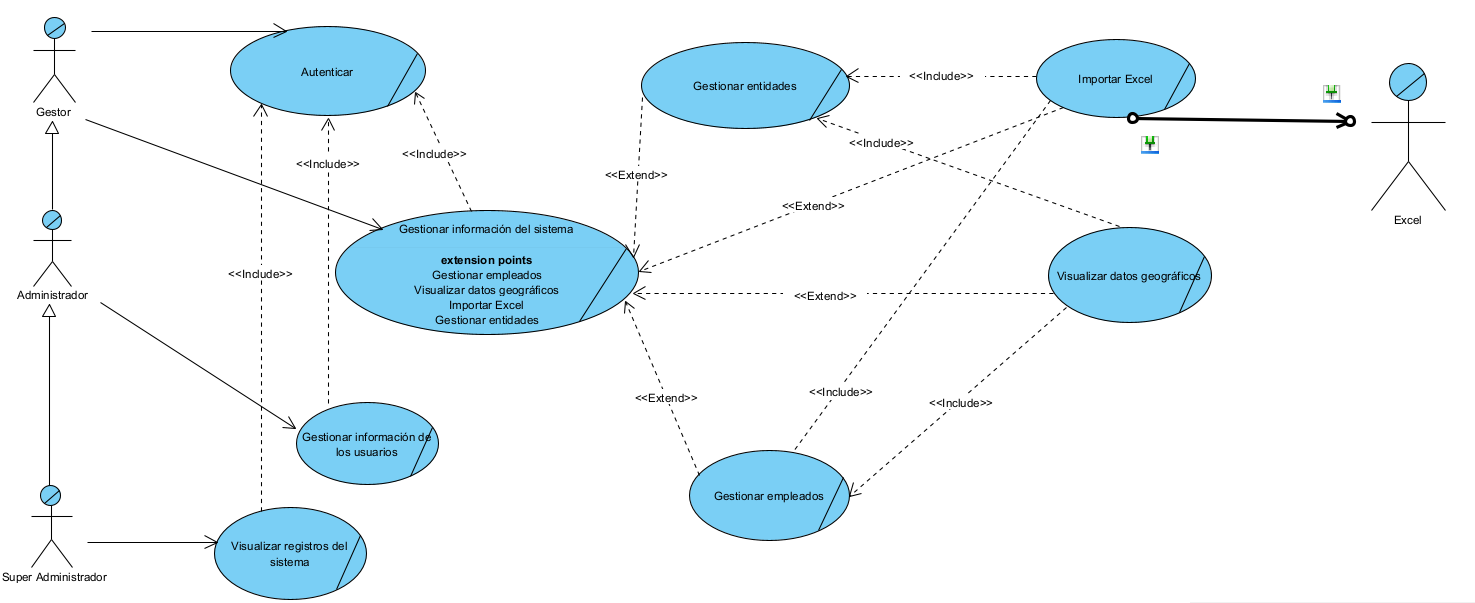


Figura 4: Diagrama de caso de uso del sistema

|  |  |
| --- | --- |
| Nombre del caso de estudio | Autentificación |
| Actor | Todos los actores, menos el Excel |
| Resumen | Permite el inicio de sesión por parte de los distintos usuarios. Consiste en ingresar las credenciales |
| Precondición | Cuenta creada con anterioridad |
| Postcondición |  |

Tabla 2: Caso de uso Autentificación

|  |  |
| --- | --- |
| Nombre del caso de estudio | Gestionar información del sistema |
| Actor | Todos los actores, menos el Excel |
| Resumen | Permite la gestión de la información que almacena el proyecto tanto de las entidades y los trabajadores |
| Precondición | El usuario debe estar autenticado correctamente |
| Postcondición |  |

Tabla 3: Caso de uso Gestionar información del sistema

|  |  |
| --- | --- |
| Nombre del caso de estudio | Gestionar información de los usuarios |
| Actor | Administrador y Super Administrador |
| Resumen | Permite la gestión de todo lo relacionado con los usuarios |
| Precondición | El usuario debe estar autenticado correctamente |
| Postcondición |  |

Tabla 4:Caso de estudio Gestionar información de los usuarios

|  |  |
| --- | --- |
| Nombre del caso de estudio | Visualizar registros del sistema |
| Actor | Super Administrador |
| Resumen | Permite visualizar todos los registros del sistema |
| Precondición | El usuario debe estar autenticado correctamente |
| Postcondición |  |

Tabla 5: Caso de estudio Visualizar registros del sistema

|  |  |
| --- | --- |
| Nombre del caso de estudio | Gestionar entidades |
| Actor | Todos los actores, menos Excel |
| Resumen | Permite la gestión de todo lo relaciones con las entidades, ya sea la inserción, modificación y eliminación de los datos |
| Precondición | El usuario debe estar autenticado correctamente |
| Postcondición |  |

Tabla 6: Caso de uso Gestionar entidades

|  |  |
| --- | --- |
| Nombre del caso de estudio | Gestionar empleados |
| Actor | Todos los actores, menos Excel |
| Resumen | Permite la gestión de todo lo relaciones con los empleados, ya sea la inserción, modificación y eliminación de los datos |
| Precondición | El usuario debe estar autenticado correctamente |
| Postcondición |  |

Tabla 7: Caso de estudio Gestionar empleados

|  |  |
| --- | --- |
| Nombre del caso de estudio | Importar Excel |
| Actor | Todos los actores, menos Excel |
| Resumen | Permite la extracción de los datos del Excel, filtrar por los datos correctos y almacener dichos datos en la base de datos |
| Precondición | El usuario debe estar autenticado correctamente |
| Postcondición |  |

Tabla 8: Caso de estudio Importar Excel

|  |  |
| --- | --- |
| Nombre del caso de estudio | Visualizar datos geográficos |
| Actor | Todos los actores, menos Excel |
| Resumen | Permite la visualización de los datos en reportes, mapas y tablas |
| Precondición | El usuario debe estar autenticado correctamente |
| Postcondición |  |

Tabla 9:Caso de estudio Visualizar datos geográficos

## **Diagrama de clases**

Como en el sistema se ha aplicado una arquitectura de microservicios, se ha decidido plantear el diagrama de clases según su microservicio.

### **Microservicio Login**

Explicación de las distintas clases expresadas en el modelo:

* Sexo: Representa el sexo que puede tener asignado una persona, ya sea masculino o femenino
* Rol: Representa el rol que ocupa el usuario en el sistema
* Usuario: Representa al usuario que se autentica en el sistema. Esta almacena datos como el nombre usuario, la contraseña, el correo del usuario y demás datos.
* Estado: Representa el estado de una petición: aceptada o rechazada y que determina si la tarea se completó o no con éxito
* Registro: Representa los registros de todas las actividades que ocurren en el sistema



Figura 5: Diagrama de clases del microservicio de Login

### **Microservicio Gestión**

Explicación de algunas clases expresadas en el modelo:

* Provincia: Es la representación de las provincias en el territorio nacional, inclusiva con el almacenamiento de datos geográficos
* Municipio: Es la representación de los municipios en el territorio nacional, inclusiva con el almacenamiento de datos geográficos
* Entidad: Es la representación de los distintos centros laborales que maneja el programa junto con sus datos geográficos
* Empleado: Es la representación de los empleados en el sistema, incluyendo sus datos geográficos



Figura 6: Diagrama de clases del microservicio de Gestión

## **Patrones Arquitectónicos**

Un patrón arquitectónico es una solución general y reutilizable a un problema común en la arquitectura de software dentro de un contexto dado. Los patrones arquitectónicos son similares al patrón de diseño de software, pero tienen un alcance más amplio.[38]

### **Patrón N-Capas**

Este patrón se puede utilizar para estructurar programas que se pueden descomponer en grupos de subtareas, cada una de las cuales se encuentra en un nivel particular de abstracción. Cada capa proporciona servicios a la siguiente capa superior.[38]

* Capa de presentación (también conocida como capa UI): Capa donde se tiene todos los valores de las interfaces de usuario
* Capa de aplicación (también conocida como capa de servicio): Capa donde se tienen acceso a los servicios de la aplicación y que se encarga del proceso de la información
* Capa de lógica de negocios (también conocida como capa de dominio): Capa donde se tiene todo el modelo de datos y sus relaciones que influyen en la aplicación
* Capa de acceso a datos (también conocida como capa de persistencia): Capa donde se tiene todo lo que concierte al trabajo con la persistencia de datos con el ORM

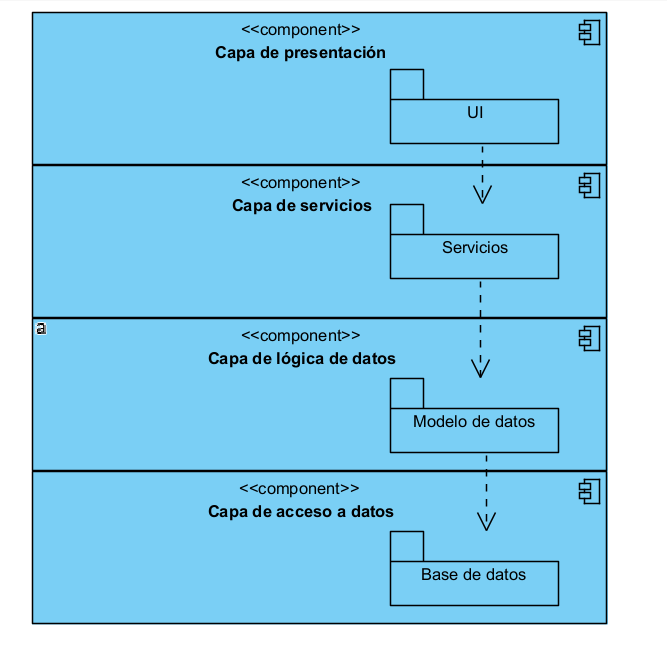


Figura 7: Modelado del patrón N-Capas

### **Patrón Cliente-Servidor**

Este patrón consiste en dos partes; un servidor y múltiples clientes. El componente del servidor proporcionará servicios a múltiples componentes del cliente. Los clientes solicitan servicios del servidor y el servidor proporciona servicios relevantes a esos clientes. Además, el servidor sigue escuchando las solicitudes de los clientes.

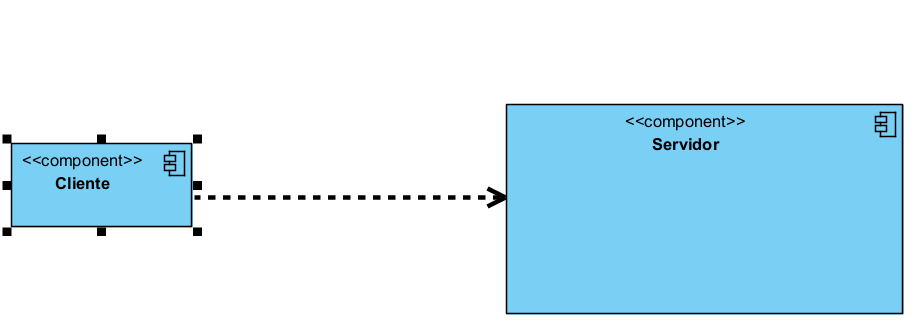


Figura 8: Modelado del patrón Cliente - Servidor

### **Patrón de filtro de tubería**

Este patrón se puede usar para estructurar sistemas que producen y procesan una secuencia de datos. Cada paso de procesamiento se incluye dentro de un componente de filtro. Los datos que se procesarán se pasan a través de las tuberías. Estas tuberías se pueden utilizar para el almacenamiento en búfer o con fines de sincronización.

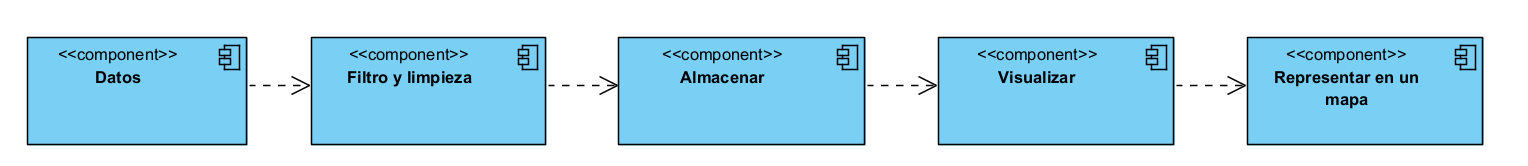


Figura 9: Patrón de tubería y filtros

### **Patrón de modelo-vista-controlador**

Este patrón, también conocido como patrón MVC, divide una aplicación interactiva en 3 partes, como

1. **Modelo**: contiene la funcionalidad y los datos básicos
2. **Vista**: muestra la información al usuario (se puede definir más de una vista)
3. **Controlador**: maneja la entrada del usuario

Esto se hace para separar las representaciones internas de información de las formas en que se presenta y acepta la información del usuario. Desacopla los componentes y permite la reutilización eficiente del código.[38]

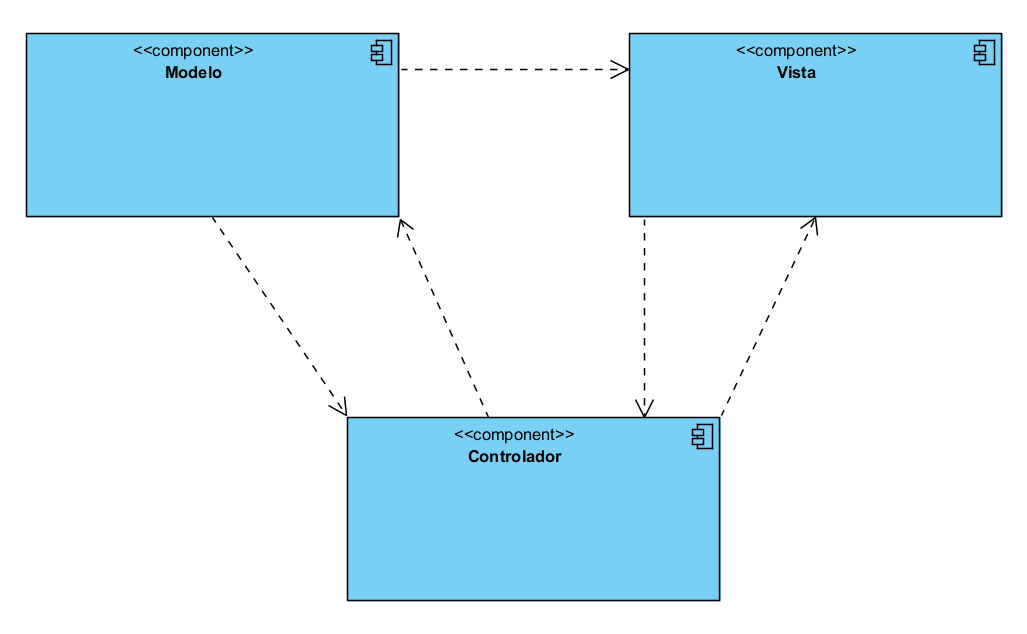


Figura 10: Modelado del patrón Modelo Vista Controlador (MVC)

## **Modelo físico de la base de datos**

El modelado de [datos](https://geekflare.com/es/database-modeling-and-design-tools/) es el proceso de generar una representación racionalizada de una aplicación de software y de los componentes de datos que incluye, utilizando texto y símbolos para representar la información y su flujo. Es una técnica para esbozar y visualizar todos los diferentes puntos en los que un programa o aplicación de software guarda datos y cómo estas diferentes fuentes de información se integrarán y fluirán entre sí. El modelado de datos es un aspecto importante de la gestión de datos. Ayuda a identificar los requisitos de información para los flujos de trabajo al ofrecer una representación visual de los puntos de datos y su patrón de comportamiento.[39]

Un modelo de datos físicos es un modelo específico de bases de datos que representa objetos de datos relacionales (por ejemplo, tablas, columnas, claves principales y claves externas) y sus relaciones. Un modelo de datos físicos se puede utilizar para generar sentencias DDL que, después, se pueden desplegar en un servidor de base de datos.[40]

### **Microservicio Login**

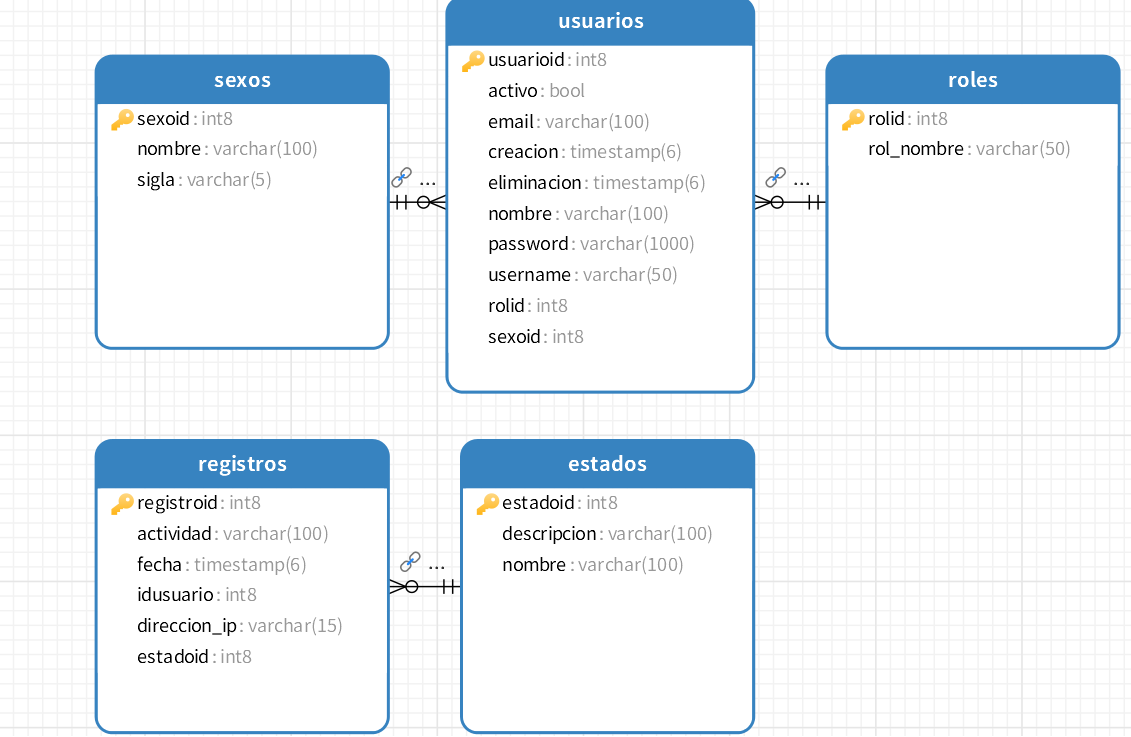


Figura 11: Modelo físico de la base de datos del microservicio de Login

### **Microservicio Gestion**

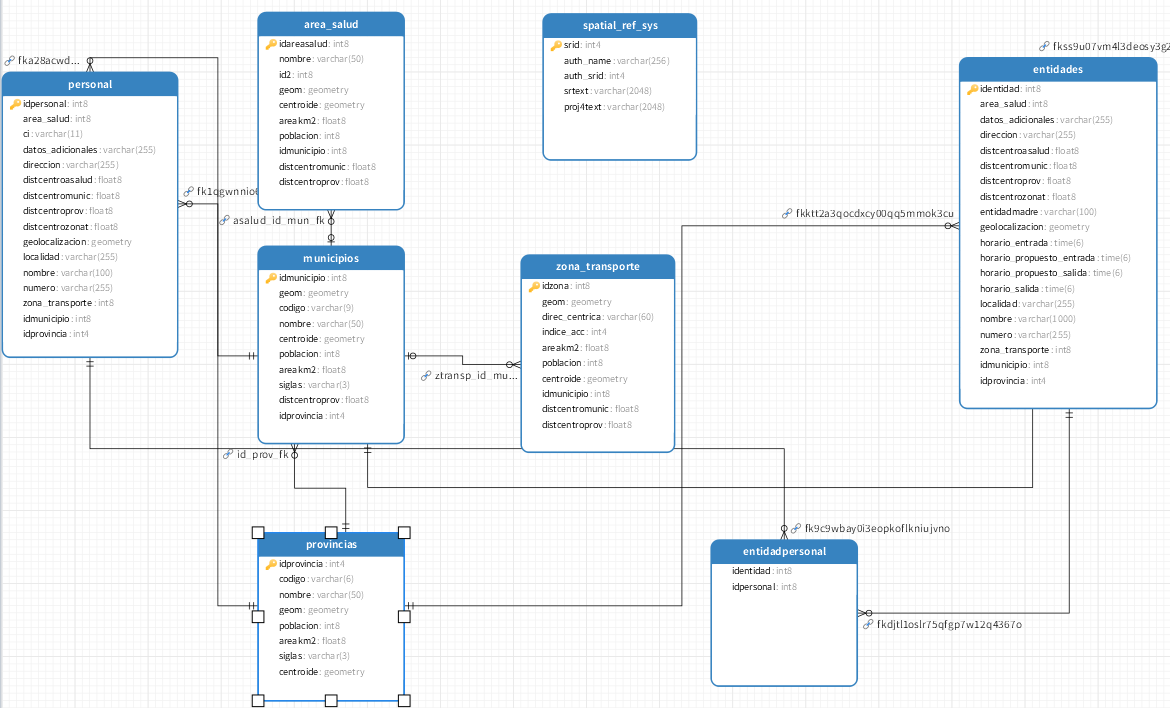


Figura 12: Modelo físico de la base de datos del microservicio de Gestión

## **Seguridad**

La seguridad web o seguridad de sitios web es la práctica de salvaguardar las redes, las comunicaciones en línea, el hardware y el software para que no sean manipulados o utilizados con fines maliciosos. En esta era creciente de ciber amenazas y vulnerabilidades, los principales objetivos son los sitios web[41]

### **Autenticación y Autorización**

En las aplicaciones web, es imprescindible garantizar que solamente los usuarios autorizados y autentificados puedan acceder a las funciones y datos críticos del sistema, evitando así que visitantes no autorizados pueda acceder y hacer un mal uso de dicha información.

La autenticación es el proceso de identificar a los usuarios y garantizar que los mismos sean quienes dicen ser. Esto evita que cualquiera pueda entrar en un determinado sistema o iniciar sesión en alguna plataforma de forma indebida, sin que realmente sea el usuario legítimo que tiene el poder para hacerlo.[42]

La autorización es lo que define a qué recursos de sistema el usuario autenticado podrá acceder. Que haya logrado pasar la instancia de la autenticación, no significa que podrá utilizar el sistema por completo como super administrador. De acuerdo a una serie de reglas, normas y regulaciones propias de cada red interna, se determina que el usuario A tendrá acceso a los recursos X e Y. Sin embargo, el usuario B sólo podrá acceder al recurso Z.[42]

Ejemplos de métodos de autenticación:

* **Sin contraseña o Passwordless.** Este es uno de los métodos modernos más prácticos. Un ejemplo de aplicación es el uso de un enlace mágico (*magic link*). Consiste en que, cada vez que quieras iniciar sesión a un recurso o servicio, se enviará a tu correo electrónico un enlace que te permitirá acceder sin necesidad de contraseña. Este es un método recomendable, ya que se necesita del acceso al correo electrónico y, por ende, hay más garantías de asegurar que es el propio usuario quien está accediendo.[43]
* **Por redes sociales.**Sin duda, ya habrás utilizado este método. Varias aplicaciones y servicios te dan como opción iniciar sesión directamente con alguna cuenta social. La ventaja principal es que no hace falta crear una cuenta aparte de forma manual, directamente los datos de esa cuenta social hacen ese paso al iniciar la sesión. Las plataformas sociales más utilizadas son Facebook, Twitter y la cuenta Google. De esta forma podremos iniciar sesión en programas o páginas de forma más rápida, sin tener que registrarnos.[43]
* **Autenticación API.** Este es el proceso de certificar la identidad de un usuario que quiera acceder a recursos y/o servicios en el servidor. Para tener en cuenta.[43]
* **Autenticación Biométrica.**Se vale de las huellas dactilares para validar la identidad del usuario. El caso de uso más popular es en los lugares de trabajo, en donde tanto para registrar la entrada como salida, se posa el dedo para validar la huella dactilar. Esa huella es validada mediante un previo registro de la misma que se almacena en la base de datos. Es cada vez más utilizado este método también en dispositivos móviles, para iniciar sesión, realizar un pago, etc.[43]

### **Spring Security**

La seguridad de la aplicación se gestiona con Spring Security, un marco integral que proporciona un conjunto de herramientas para implementar autenticación, autorización, y protección de acceso a aplicaciones basadas en Java. Spring Security permite configurar políticas de seguridad a nivel de aplicación de manera flexible, adaptándose a diferentes necesidades y escenarios.

1. **Configuración General**: Spring Security intercepta todas las solicitudes HTTP entrantes y aplica reglas de autenticación y autorización. Permite definir qué rutas de la aplicación requieren autenticación, qué tipos de autenticación se utilizan, y qué acciones están permitidas para cada tipo de usuario.
2. **Filtros de Seguridad**: Spring Security proporciona una serie de filtros de seguridad que se ejecutan en cada solicitud para autenticar y autorizar usuarios. Estos filtros pueden ser personalizados para adaptarse a necesidades específicas, como la validación de tokens JWT.

### **Autenticación Sin Estado (Stateless)**

Una de las principales características de una API REST es que el servicio no tiene estado (*stateless*), lo que implica que cada vez que se recurra a él será necesario recordarle nuestros datos, ya sean credenciales de usuario o cualquier otro tipo de información. Lo que por una parte podría parecer una desventaja, al implicar la tediosa tarea de repetir datos, es en realidad una de sus fortalezas: al no memorizarlos, permite una mayor escalabilidad. No harán falta servidores tan potentes, capaces de almacenar todos los estados de sus clientes.[44]

Este factor es especialmente relevante para cualquier API bancaria, como por ejemplo las de BBVA. Si el *fintech* aspira a atraer a todos los clientes de instituciones bancarias y lograr que los desarrolladores introduzcan en sus aplicaciones herramientas financieras, la escalabilidad de sus API es primordial.[44]

### **JSON Web Token**

El token web JSON (JWT) es un estándar abierto (RFC 7519) que define un método compacto y autocontenido para la transmisión segura de información entre partes codificadas como un objeto JSON. Esta información puede verificarse y ser fiable porque está firmada digitalmente. Los JWT pueden firmarse utilizando un secreto (con el algoritmo HMAC) o un par de claves públicas/privadas utilizando RSA.[45]

Estos son algunas situaciones en las que los tokens web JSON son útiles:

* **Autenticación**: Esta es la situación típica para el uso de JWT, una vez que el usuario ha iniciado sesión, cada solicitud posterior incluirá el JWT, permitiendo al usuario acceder a las rutas, servicios y recursos que están permitidos con ese token. El inicio de sesión único es una característica que utiliza ampliamente JWT hoy en día, debido a su pequeña sobrecarga y su capacidad para ser utilizado fácilmente entre los sistemas de diferentes dominios.[45]
* **Intercambio de información**: Los JWT son una buena forma de transmitir información de forma segura entre las partes, porque como se pueden firmar, por ejemplo, utilizando un par de claves públicas/privadas, se puede estar seguro de que el remitente es quien dice ser. Además, como la firma se calcula utilizando el encabezado y la carga útil, también puedes verificar que el contenido no ha cambiado.[45]

Spring Security proporciona un marco de trabajo robusto para aplicaciones web, que, junto con el uso de JSON Web Tokens (JWT) y la autenticación sin estado (Stateless Authentication), permite implementar de manera segura y eficiente los procesos de autenticación y autorización en la aplicación. Estas herramientas combinadas garantizan que solo los usuarios autorizados accedan a los recursos protegidos, cumpliendo con los más altos estándares de seguridad.

### **Configuración del uso compartido de recursos entre orígenes (CORS)**

El intercambio de recursos de origen cruzado (CORS, por sus siglas en inglés), es un mecanismo basado en cabeceras HTTP que permite a un servidor indicar cualquier dominio, esquema o puerto con un origen distinto del suyo desde el que un navegador debería permitir la carga de recursos. CORS también se basa en un mecanismo por el cual los navegadores realizan una solicitud de "verificación previa" al servidor que aloja el recurso de origen cruzado, con el fin de comprobar que el servidor permitirá la solicitud real. En esa comprobación previa, el navegador envía cabeceras que indican el método HTTP y las cabeceras que se utilizarán en la solicitud real.[46]

Por razones de seguridad, los navegadores restringen las peticiones HTTP de origen cruzado iniciadas desde scripts. Por ejemplo, XMLHttpRequest y la API Fetch siguen la Política Same-origin. Esto significa que una aplicación web que utilice esas API solo puede solicitar recursos del mismo origen desde el que se cargó la aplicación, a menos que la respuesta de otros orígenes incluya las cabeceras CORS adecuadas.[46]

El mecanismo CORS soporta peticiones seguras de origen cruzado y trasferencias de datos entre navegadores y servidores. Los navegadores modernos utilizan CORS en API como XMLHttpRequest o Fetch para mitigar los riesgos de las peticiones HTTP de origen cruzado.[46]

En aplicaciones desarrolladas con Spring Boot y Spring Security, CORS se configura para definir qué dominios están autorizados a acceder a los recursos y qué métodos HTTP son permitidos. Esta configuración garantiza que solo las solicitudes autorizadas sean aceptadas, contribuyendo a una mayor seguridad.

1. **Configuración de Orígenes Permitidos**: Define qué dominios pueden acceder a la API, limitando el acceso a solicitudes provenientes de orígenes no confiables.
2. **Preflight Requests**: Responde a las solicitudes de pre-verificación (OPTIONS) que envía el navegador para validar que el servidor permita las solicitudes CORS.

### **Protección contra Inyecciones SQL**

Uno de los peligros, más usuales de los sistemas web, son las inyecciones SQL. La inyección de SQL es un tipo de ciberataque encubierto en el cual un hacker inserta código propio en un sitio web con el fin de quebrantar las medidas de seguridad y acceder a datos protegidos. Una vez dentro, puede controlar la base de datos del sitio web y secuestrar la información de los usuarios.[47]

Los ataques de inyección de SQL únicamente son viables cuando un sitio web carece de un *saneamiento de entrada* adecuado: el proceso que vela por que la información que introducen los usuarios finales no pueda colarse por ningún resquicio y funcionar como código ejecutable en el servidor. Esto requiere más trabajo por parte del desarrollador, pero, en última instancia, protege frente a la inyección de SQL, las secuencias de comandos en sitios cruzados y otras clases de ataques a sitios web.[47]

Los hackers recurren a los ataques de inyección de SQL con el fin de introducirse en la base de datos de un sitio web. A veces solo quieren eliminar datos para provocar el caos y, en otras ocasiones, lo que buscan es editar la base de datos, especialmente en el caso de sitios web financieros. En el momento en que el hacker ha logrado el control de la base de datos, ya es fácil interferir en los saldos de las cuentas de los clientes y mandarse dinero a su propia cuenta.[47]

Sin embargo, a menudo lo que elciberdelincuente quiere son los datos de usuario guardados en el sitio web, como las credenciales de inicio de sesión. Estos datos de inicio de sesión robados puede emplearlos para realizar acciones en nombre de los usuarios afectados o reunirlos en una gran lista que luego venderá a otros ciberdelincuentes en la red oscura. Las personas que compran información robada lo hacen, frecuentemente, con la finalidad de robar identidades y cometer fraudes.[47]

Para prevenir dichos ataques, se ha utilizado el control que establece Spring Data JPA y la validación de la entrada de datos a la aplicación y el sistema, que no resulten en una consultas o palabras reservadas de SQL que resulten perjudiciales para el sistema.

Solución de Spring Data JPA contra las inyecciones SQL:

* **Uso de JPQL (Java Persistence Query Language):** JPQL se utiliza para escribir consultas en Spring Data JPA y está orientado a objetos. A diferencia de SQL, que trabaja directamente con tablas y columnas, JPQL opera sobre entidades y atributos. Esto reduce el riesgo de inyección SQL, ya que las consultas no interactúan directamente con el esquema de la base de datos.
* **Consultas Basadas en Métodos Derivados:** Métodos Derivados en Spring Data JPA permiten generar consultas automáticamente basadas en el nombre del método en el repositorio. Por ejemplo, un método como findByUsername(String username) genera una consulta segura basada en el nombre del método y los parámetros proporcionados, sin exponer la base de datos a inyecciones SQL.
* **Uso Seguro de Parámetros:** Cuando se utilizan parámetros en consultas JPQL, Spring Data JPA maneja estos parámetros de manera segura. Los parámetros se envían al motor de consultas de forma que están adecuadamente escapados y no pueden modificar la consulta original, lo que previene la inyección SQL.
* **Protección en Consultas Nativas:** Si necesitas ejecutar consultas SQL nativas en Spring Data JPA, es crucial utilizar parámetros nombrados o posicionados en lugar de concatenar directamente cadenas SQL. El uso de @Query con parámetros seguros garantiza que las entradas del usuario no puedan modificar la consulta de manera maliciosa.

### **Monitoreo y registro de actividades de seguridad**

Se tiene un sistema de registro que permite almacenar cada actividad que el usuario hay realizado y no es eliminable a nivel de código ya que está protegido, contra eliminaciones o modificaciones tanto a nivel de la API como directamente en la base de datos.

### **Conclusión de la seguridad**

La implementación de estas medidas seguridad es un buen comienzo para la protección de un sistema, pero en un terreno donde las amenazas van evolucionando diariamente, es siempre mantener a los sistemas actualizados en el aspecto de seguridad.

## **Diagrama de despliegue**

En la figura 13, que se puede observar a continuación se ofrece una vista del diagrama de despliegue que emplea el sistema web, en cuestión. Consta de dos elementos un cliente y un servidor. Ambos dispositivos se encuentran interconectado a través del protocolo HTTP.

En el servidor podemos encontrar diversos componentes:

* API Gateway: Es un gestor que acepta todas las llamadas API y luego actúa como un proxy inverso, recuperando recursos de las aplicaciones de back-end en nombre de la aplicación del cliente; maneja tareas relacionadas con los servicios de API, como autenticación de usuarios, limitación de velocidad y monitoreo.[48]
* Eureka Server: Es un servidor para el registro y localización de microservicios, balanceo de carga y tolerancia a fallos. La función de Eureka es registrar las diferentes instancias de microservicios existentes, su localización, estado, metadatos...
* Microservicio Gestión: Es el microservicio encargado de toda la gestión de las entidades y sus diversos empleados.
* Microservicio Login: Es el microservicio encargado de la gestión los usuarios, sus roles y su registro de actividades.
* Microservicio Geográfico: Es el microservicio encargado de la conexión con los servicios geográficos empleados.
* Microservicio Automatizado: Es el microservicio encargado de automatizar tareas independientemente del usuario

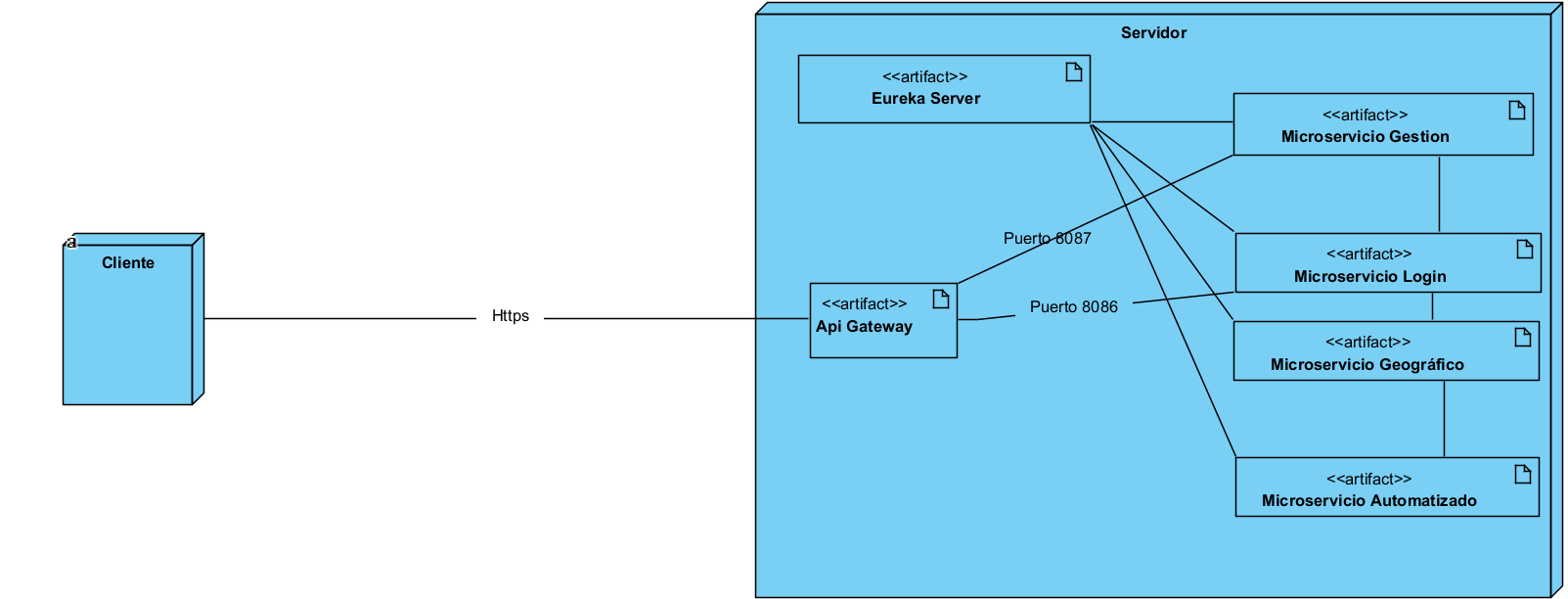


Figura 13: Modelado del diagrama de despliegue

El servidor empleado que resultó eficiente para las tareas tenía las siguientes propiedades de hardware:

* Sistema Operativo: Windows 11 Home Single Language 64-bit (10.0, Build 22631) (22621.ni\_release.220506-1250)
* Procesador: 11th Gen Intel(R) Core(TM) i3-1115G4 @ 3.00GHz (4 CPUs), ~3.0GHz
* Memoria Ram: 8192MB RAM
* Memoria Ram disponible: 7976MB RAM
* Almacenamiento: 250 GB
* Almacenamiento disponible: 160 GB
* Conector de red Wifi o Ethernet

# **Validación de la solución propuesta**

# **Referencia Bibliográfica**

1. Ministerio de la Justicia de la República de Cuba. (s. f.). *Gaceta Oficial No. 45 Ordinaria de 4 de julio de 2019*.

2. *¿Qué es un sistema de información geográfica (SIG)?* (2024, abril 17). https://www.ibm.com/es-es/topics/geographic-information-system

3. Georreferenciación: Qué es y para qué se utiliza. (2018, abril 10). *Certicalia | Soluciona todos tus trámites y gestiones*. https://www.certicalia.com/blog/georreferenciacion-que-es-y-para-que-se-utiliza

4. *¿Qué es un sistema de información geográfica (SIG)? | IBM*. (2023, octubre 30). https://www.ibm.com/es-es/topics/geographic-information-system

5. What is GIS(Geographic Information System)? (s. f.). *Biplap KC*. Recuperado 4 de julio de 2024, de https://talktechnologynepal.blogspot.com/2022/08/what-is-gisgeographic-information-system.html

6. *Georreferenciación y sistemas de coordenadas | ArcGIS Resource Center*. (s. f.). Recuperado 4 de julio de 2024, de https://resources.arcgis.com/es/help/getting-started/articles/026n0000000s000000.htm

7. *World Wide Web - Glosario de MDN Web Docs: Definiciones de términos relacionados con la Web | MDN*. (2023, noviembre 13). https://developer.mozilla.org/es/docs/Glossary/World\_Wide\_Web

8. *HTTP - Glosario de MDN Web Docs: Definiciones de términos relacionados con la Web | MDN*. (2023, noviembre 13). https://developer.mozilla.org/es/docs/Glossary/HTTP

9. *¿Qué es una aplicación web? - Explicación de las aplicaciones web - AWS*. (s. f.). Amazon Web Services, Inc. Recuperado 2 de julio de 2024, de https://aws.amazon.com/es/what-is/web-application/

10. *¿Qué son los diagramas de arquitectura? - Explicación de los diagramas de arquitectura de software y sistemas - AWS*. (s. f.). Amazon Web Services, Inc. Recuperado 2 de julio de 2024, de https://aws.amazon.com/es/what-is/architecture-diagramming/

11. *¿Qué es una API? - Explicación de interfaz de programación de aplicaciones - AWS*. (s. f.). Amazon Web Services, Inc. Recuperado 2 de julio de 2024, de https://aws.amazon.com/es/what-is/api/

12. *¿Qué es el GIS? | Sistema de Información Geográfica Tecnología de Cartografía*. (s. f.). Recuperado 8 de julio de 2024, de https://www.esri.es/es-es/descubre-los-gis/qu-es-sig/que-es-sig

13. *Historia de los SIG | Línea temporal de los inicios de los SIG y su futuro*. (s. f.). Recuperado 1 de julio de 2024, de https://www.esri.com/es-es/what-is-gis/history-of-gis

14. Bruin, R.-J. de. (s. f.). GIS Software. *Unigis.Net*. Recuperado 4 de julio de 2024, de https://unigis.net/gis-software/

15. ClasificaciónDe. (2021, octubre 11). *5 ejemplos de sistemas de información geográfica—¿Cómo se clasifican?* https://www.clasificacionde.org/ejemplos-de-sistemas-de-informacion-geografica/

16. Morales, A. (2022, julio 20). Las 30 aplicaciones GIS open source que debes conocer. *MappingGIS*. https://mappinggis.com/2022/07/aplicaciones-gis-open-source/

17. *10 Mejores Softwares de información geográfica*. (2022, julio 8). https://mejor-software.com/informacion-geografica/

18. Climent, P. V. (2022, septiembre 27). 10 aplicaciones GIS en la nube para publicar mapas. *MappingGIS*. https://mappinggis.com/2022/09/10-aplicaciones-gis-en-la-nube-para-publicar-mapas-en-la-web/

19. Marino Latorre. (2018). *HISTORIA DE LAS WEB, 1.0, 2.0, 3.0 y 4.0*.

20. Alarcon, C. (2024, mayo 17). Sistema web ¿Qué es? *Data Trust*. https://www.datatrust.pe/web/sistema-web/

21. JuliánMarquina. (2024, marzo 13). Los 7 navegadores web más utilizados en el mundo [2024]. *Julián Marquina | Bibliotecas, libros y tecnología*. https://www.julianmarquina.es/los-navegadores-web-mas-utilizados-en-el-mundo/

22. *Ventajas y desventajas de Java que debes conocer*. (s. f.). Recuperado 12 de julio de 2024, de https://blog.hubspot.es/website/ventajas-desventajas-java

23. PostgreSQL: Ventajas y desventajas | Blog | Hosting Plus Mexico. (2021, diciembre 31). *Hosting Plus*. https://www.hostingplus.mx/blog/postgresql-ventajas-y-desventajas/

24. *Ventajas y Desventajas de PostgreSQL - TodoPostgreSQL*. (2018, agosto 30). https://www.todopostgresql.com/ventajas-y-desventajas-de-postgresql/

25. franzpc. (2017, octubre 4). *¿Qué es PostGIS?* El blog de franz. https://acolita.com/que-es-postgis/

26. *Líderes en formación tecnológica, reskilling y upskilling | OpenWebinars*. (s. f.). OpenWebinars.net. Recuperado 12 de julio de 2024, de https://openwebinars.net/blog/que-es-spring-framework/

27. programacionpro.com. (2024, junio 17). *Ventajas y Desventajas de Spring Framework*. ProgramaciónPro. https://programacionpro.com/ventajas-y-desventajas-de-spring-framework/

28. Java Persistence API (JPA). (s. f.). *Oscar Blancarte - Software Architecture*. Recuperado 12 de julio de 2024, de https://www.oscarblancarteblog.com/tutoriales/java-persistence-api-jpa/

29. *¿Qué es Java Hibernate? ¿Por qué usarlo? | ifGeekThenNTTDATA*. (s. f.). Recuperado 12 de julio de 2024, de https://ifgeekthen.nttdata.com/s/post/que-es-java-hibernate-por-que-usarlo-MC5FU56AIPGBGIHNJ677RBIXUHOI?language=es

30. *La diferencia entre JDBC, JPA, Hibernate y Spring Data JPA | HackerNoon*. (s. f.). Recuperado 12 de julio de 2024, de https://hackernoon.com/the-difference-between-jdbc-jpa-hibernate-and-spring-data-jpa

31. *¿Qué es Spring Data JPA? – Barcelona Geeks*. (s. f.). Recuperado 12 de julio de 2024, de https://barcelonageeks.com/que-es-spring-data-jpa/

32. Saavedra, J. A. (2023, julio 17). *Qué es React y para qué sirve*. Ebac. https://ebac.mx/blog/que-es-react

33. mijacobs. (2023, octubre 5). *¿Qué es Git? - Azure DevOps*. https://learn.microsoft.com/es-es/devops/develop/git/what-is-git

34. Fernández, Y. (2019, octubre 30). *Qué es Github y qué es lo que le ofrece a los desarrolladores*. Xataka. https://www.xataka.com/basics/que-github-que-que-le-ofrece-a-desarrolladores

35. *All Products Pack: Acceda a todas las herramientas de escritorio de JetBrains, incluidos IDEs, extensiones y perfiladores.* (s. f.). JetBrains. Recuperado 12 de julio de 2024, de https://www.jetbrains.com/es-es/all/

36. *Visual Studio: IDE y Editor de código para desarrolladores de software y Teams*. (s. f.). Visual Studio. Recuperado 12 de julio de 2024, de https://visualstudio.microsoft.com/es/

37. *4 Ejemplos de Diagramas de Casos de Uso—Web y Empresas*. (2022, enero 4). https://www.webyempresas.com/ejemplos-de-diagramas-de-casos-de-uso/

38. huaman, W. C. (2018, septiembre 7). Los 10 patrones comunes de arquitectura de software. *Medium*. https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b

39. Jenifa, A. (2022, diciembre 5). *Modelo físico de datos en DBMS: Lo que necesita saber | Geekflare*. Geekflare Spain. https://geekflare.com/es/physical-data-model/

40. *IBM Data Studio 4.1.1*. (2021, marzo 17). https://www.ibm.com/docs/es/data-studio/4.1.1?topic=modeling-physical-data-models

41. *¿Qué es la seguridad web? Buenas prácticas de seguridad de sitios web*. (s. f.). Recuperado 4 de septiembre de 2024, de https://powerdmarc.com/es/web-security-website-security-explained/

42. *Autenticación vs autorización: Diferencias y ejemplos de cómo funcionan*. (s. f.). RedesZone. Recuperado 4 de septiembre de 2024, de https://www.redeszone.net/tutoriales/seguridad/diferencias-autenticacion-autorizacion/

43. *Autenticación vs autorización: Diferencias y ejemplos de cómo funcionan*. (s. f.). RedesZone. Recuperado 4 de septiembre de 2024, de https://www.redeszone.net/tutoriales/seguridad/diferencias-autenticacion-autorizacion/

44. Cinco claves para desarrollar con API REST. (s. f.). *BBVA API\_Market*. Recuperado 4 de septiembre de 2024, de https://www.bbvaapimarket.com/es/mundo-api/cinco-claves-para-desarrollar-con-api-rest/

45. *Introducción a los tokens web JSON*. (s. f.). Auth0. Recuperado 4 de septiembre de 2024, de https://auth0.com/es/learn/json-web-tokens#!

46. *Intercambio de recursos de origen cruzado (CORS)—HTTP | MDN*. (2024, julio 28). https://developer.mozilla.org/es/docs/Web/HTTP/CORS

47. *¿Qué es la inyección de SQL y cómo funciona?* (s. f.). ¿Qué es la inyección de SQL y cómo funciona? Recuperado 4 de septiembre de 2024, de https://www.avast.com/es-es/c-sql-injection

48. *Qué es una API Gateway, cómo funciona y las mejores opciones*. (s. f.). Recuperado 6 de septiembre de 2024, de https://blog.hubspot.es/website/que-es-api-gateway