MEMORIA ESCRITA DEL PROYECTO

CFGS Desarrollo de Aplicaciones Web

**Sistema de Administración de Inventario (SAI)**

**Autor:** Jonathan Jimenez Díaz

**Tutor:** Álvaro Alonso

**Fecha de entrega:** 17/06/2024

**Convocatoria:** \_S\_\_\_\_ (semestre de la convocatoria)

**Documentos del proyecto:** Enlace a la carpeta del Drive

Índice de contenidos

[**1. Introducción (3-4 Páginas 3**](#_heading=h.1fob9te)

[**1.1. Motivación 3**](#_heading=h.3znysh7)

[**1.2. Abstract 3**](#_heading=h.2et92p0)

[**1.3. Objetivos propuestos (generales y específicos) 3**](#_heading=h.tyjcwt)

[**2. Estado del Arte (4-5 páginas) 4**](#_heading=h.3dy6vkm)

[**3. Metodología usada (2-4 páginas) 5**](#_heading=h.1t3h5sf)

[**4. Tecnologías y herramientas utilizadas en el proyecto  
(2-3 páginas) 6**](#_heading=h.4d34og8)

[**5. Planificación, Diagnóstico y Contexto Laboral (1-2 páginas) 7**](#_heading=h.2s8eyo1)

[**6. Análisis del proyecto (8-16 páginas) 8**](#_heading=h.17dp8vu)

[**7. Diseño del proyecto (6-14 páginas) 9**](#_heading=h.3rdcrjn)

[**8. Despliegue y pruebas (4-10 páginas) 10**](#_heading=h.26in1rg)

[**9. Conclusiones (1-2 páginas) 11**](#_heading=h.lnxbz9)

[**10. Vías futuras (1-2 páginas) 12**](#_heading=h.35nkun2)

[**11. Bibliografía/Webgrafía (1-2 páginas) 13**](#_heading=h.1ksv4uv)

[**12. Anexos 14**](#_heading=h.44sinio)

[**1.1. Manual de Usuario 14**](#_heading=h.2jxsxqh)

***En la normativa de proyectos vigente encontrarás una breve descripción de cada uno de estos apartados para saber qué información debes incluir en ellos***

# Introducción (3-4 Páginas)

## Motivación

Actualmente el sector de la logística ha subido exponencialmente en las últimas décadas y más, que se ha ido globalizando el mercado e incorporando avances tecnológicos en ella. En los últimos 6-7 años de experiencia trabajando en el sector de la logística he probado diversos softwares de administración de almacenes, dado que mi enfoque académico ha sido el desarrollo de aplicaciones he querido desarrollar el mío propio dado que es una herramienta muy útil y que facilitara el trabajo a muchas empresas, ya sea pequeñas o grandes, se puede adaptar fácilmente a muchos tipos de sectores que requieran una gestión de inventario, y con la posibilidad de incluir nuevas funcionalidades y mejoras en un desarrollo futuro o actualización hasta convertirse en un Sistema de Gestión de Almacenes (SGA) muy completo.

Actualmente el desarrollo y enfoque de esta aplicación, el cual la llamaremos **SAI** (Sistema de Administración de Inventario), es principalmente el poder administrar los usuarios y su acceso a SAI, poder administrar los artículos para que puedan ser dados de entrada, administrar las ubicaciones las cuales vamos a tener para poder asignar los artículos en ellas y, como se ha comentado anteriormente, poder dar de entrada el articulo y asignarle una ubicación donde se encontrará.



## Abstract

Currently the logistics sector has grown exponentially in the last decades and more, as the market has been globalising and incorporating technological advances in it. In the last 6-7 years of experience working in the logistics sector I have tried various warehouse management software, since my academic focus has been the development of applications I wanted to develop my own because it is a very useful tool that will facilitate the work of many companies, whether small or large, it can be easily adapted to many types of sectors that require inventory management, and with the possibility of including new features and improvements in a future development or upgrade to become a Warehouse Management System (WMS) very complete.

Currently the development and focus of this application, which we will call SAI (Inventory Management System), is mainly to manage users and their access to SAI, to manage the items so that they can be given entry, manage the locations which we will have to be able to assign the items in them and, as mentioned above, to give entry to the item and assign a location where it will be located.

## Objetivos propuestos (generales y específicos)

Este Proyecto tiene el objetivo general de poder facilitar la gestión de un negocio el cual requiera un constante control de los productos, tanto de entrada de mercancía como su salida, como tambien, la facilidad de poder adaptarse a cualquier entorno o infraestructura por la creación de espacios de ubicación, y todos estos puntos junto a una interfaz fácil de entender, vistosa y accesible.

Los objetivos específicos se centran en ofrecer una administración total del producto y el servicio que negocio este ofreciendo. Al final, lo que se quiere establecer es un estándar en SAI que sea adaptativo y fiable.

Objetivos generales:

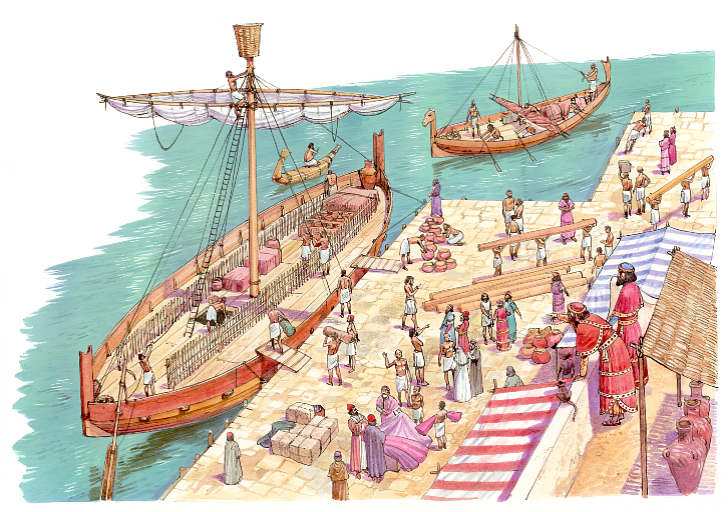
* **Desarrollar una aplicación web** de sistemas de administración de inventario fácil de usar e intuitiva el cual permita a los usuarios cumplir con sus funciones de logística.
* Tener una **administración constante** en lo que se refiere al inventario proporcionando funciones para llevarlo a cabo
* **Facilitar la gestión de un negocio**
* **Contribuye al medio ambiente**, dado que disminuye el consumo y producción de recursos naturales como el papel al eliminar los registros físicos

Objetivos específicos:

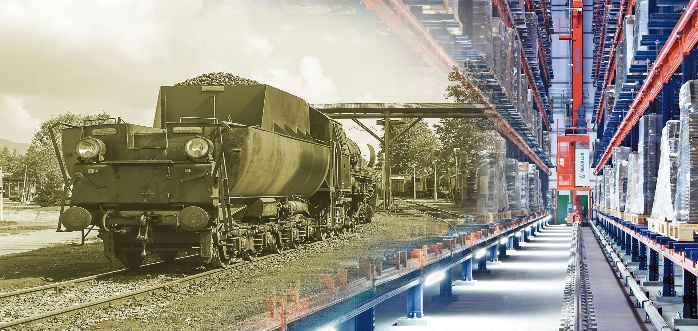
* **Implementar un sistema de autentificación** de usuarios el cual pueda identificar al administrador o el usuario estándar.
* **El administrador** tendrá funciones de gestión de usuarios, es decir, dar de alta o baja a usuarios de la aplicación, como tambien la creación, edición y eliminar los productos y las ubicaciones que el administrador crea conveniente.
* **La creación de artículos** el cual indicaremos el nombre, el EAN el cual será único para cada articulo ya que nos facilitara a la hora de hacer transacciones como la de darle de entrada y ubicarlo y por ultimo, una imagen del articulo
* **La creación de una ubicación** requerirá la información del nombre de su ubicación, que se recomienda que sea claro y conciso para los usuarios como tambien el pasillo en el que se encuentra la ubicación, de todas maneras, la aplicación contará con parámetros para que a la hora de crear una ubicación se haga en el formato adecuado.
* Los usuarios estándar tendrán las funciones de ubicar los productos los cuales están dados de alta en la base de datos.

# Estado del Arte (4-5 páginas)

La logística ha estado presente desde los primeros inicios del comercio, el origen como tal se puede remontar desde las civilizaciones mas antiguas como la Mesopotámica, China, Egipcia…etc. Ya que ejercían sistemas de transporte y almacenamiento para abastecer su sociedad.



En el siglo XVIII se produjo un punto de inflexión ya que se desarrollo maquinaria, el ferrocarril y los barcos a vapor haciendo un flujo más rápido de las mercancías. Ya en el siglo XX la logística fue más un área de estudio e investigación formal el cual, ayudo la introducción de tecnologías como la radio, el teléfono y ya posteriormente el internet y la introducción de métodos como el jus-in-time que tenia como objetivos tener la cantidad necesaria de productos o materias primas en el espacio justo y momento preciso.



Mas adelante, con la era digital, la logística sufrió un cambio radical debido a los SGA y sistemas de transporte, los códigos de barras y RFID; acrónimo de radio-frequency identificación, el cual es una tecnología que tiene como objetivo la identificación y el registro de datos. Todo ello impulsaba entregas más rápidas y personalizadas, mayor sostenibilidad y una mayor capacidad de adaptación.

Durante la década de 1970 en adelante fueron desarrollando los primeros sistemas informáticos para la gestión de almacenes. **Manhattan Associates Warehouse Management System** era un SGA y uno de los lenguajes para el desarrollo de esos sistemas fue COBOL dado que era muy utilizado en ese momento y podía manejar bastantes cantidades de datos y procesarlo, pero actualmente, hay infinidades de SGA o WMS en inglés, como **NetSuit de Oracle**, el cual ofrece una visibilidad del inventario con seguimiento real de los artículos que entran y salen, **Esfera WMS** el cual es un sistema que esta basado en la nube lo que le permite realizar una implementación rápida de pedidos y el cual tiene un app movil àra acceder de forma remota, **SAP** que es una muy conocida organización internacional con amplia experiencia en la gestion de la cadena de suministros, **Webstock**…etc.



En estos últimos años y en especial en la crisis sanitaria provocada por la COVID-19 ha disparado el crecimiento de la logística y con ello el e-commerce, dado que anteriormente los consumidores iban a los establecimientos para adquirir los productos, ahora y cada vez va mas en aumento, compramos los productos desde casa a base de un clic o un toque en la pantalla del dispositivo móvil y esperamos a que nos traigan los productos a casa e incluso servicios que son capaces de realizar entregas en el mismo día. Lo que conlleva a que software como los SGA sea más populares y demandados en el sector logístico.



Los SGA es uno de los pilares en la **optimización** de las operaciones logísticas en diversas empresas enfocadas en dicha industria. La **integración de nuevas tecnológicas** en los SGA como, por ejemplo, internet, dispositivos lectores de códigos de barras, IA...etc. permiten un seguimiento **mas preciso** a tiempo real de los inventarios como también la **movilidad y accesibilidad** ya que podemos realizar la **gestión de manera remota** a través de dispositivos móviles como, portátiles PDAs o un simple móvil. Es importante añadir que los SGA están evolucionando para ser mas escalables y adaptables a las necesidades que requiera la empresa, los **servicios de la nube** como Azure, AWS Amazon o Google Cloud Platform facilita la capacidad de ser mas escalable y el poder **integrarse con otros sistemas empresariales** como también, es importante recalcar en estos sistemas la importancia de su seguridad e integridad de los datos por lo que se está fortaleciendo en ese aspecto.

Es decir, este tipo de aplicaciones continúan evolucionando y cumpliendo satisfactoriamente la demanda que hay.

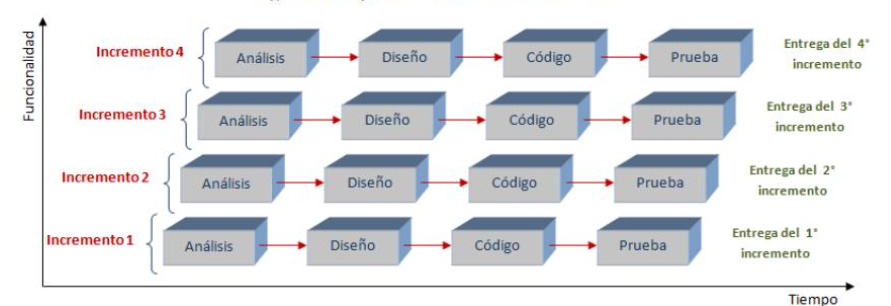
# Metodología usada (2-4 páginas)

La metodología en el desarrollo del proyecto escogida es la incremental, ya que tiene un modelo el cual es evolutivo dado que no todos los aspectos y funcionalidades de la aplicación están definitivamente definidas, porque como anteriormente se indicaba es una aplicación en una fase inicial en la que buscamos en un futuro agregar nuevas funciones. Hay ventajas al haber escogido esta metodología, como el tiempo de desarrollo inicial por las funcionalidades parciales de la aplicación, disminuyen riegos dado que se basan en retroalimentación sobre los avances y facilita aplicar los cambios al acotar el tamaño de los incrementos. Hay que objetar que esta metodología requiere al final metas claras, tener mucha planificación previa para conocer el resultado o la meta final del proyecto.

A continuación, podemos definir algunas funcionalidades las cuales podremos implementar mas adelante usando esta metodología:

* Seguimiento y monitorizacion del stock, creando reportes o avisos cuando un articulo llega a un stock minimo para poder solicitar una reposicion
* Integracacion con sistemas de envio gracias a la conexión de APIs de empresa de logisitica
* Registros de proveedores
* Sistema de preparación de pedidos e-commerce
* Monitorización de producción en los empleados, ya sea en preparacion de pedidos o ubicación de artículos
* Registro en el log de usuarios para comprobar en que momento iniciaron sesión y en qué momento la finalizan

Como se puede ver, es un proyecto que estará constantemente creciendo si así lo desea el cliente a medida que pasa el tiempo. Haciendo un breve resumen, la metodología incremental es idónea dado que habrá entregas rápidas, una gestion de riesgos efectiva, muy buena capacidad en adaptarse a cambios que se soliciten y facilitara la organización a la hora de desarrollo dado que seria capaz de trabajar varios equipos de desarrolladores en distintos ciclos o incrementos de la aplicación en paralelo.



# Tecnologías y herramientas utilizadas en el proyecto (2-3 páginas).

SAI está compuesto por un conjunto de tecnologías las cuales son muy utilizadas a día de hoy, son bastante fiables y aún siguen actualizándose, por lo que es idóneo para el entorno en el que se esta desarrollando el proyecto. Las tecnologías escogidas son las siguientes:

* **HTML5**: Lenguaje de marcas el cual es utilizado para estructurar una página web.
* **CSS3**: Es un tipo de lenguaje que permite definir y crear una visualización a un documento HTML ya estructurado el cual genera un diseño visual de paginas web e interfaces de usuario. Cabe destacar que es un lenguaje el cual se sigue actualizando e implementando nuevas funcionalidades.
* **JavaScript**: Es un lenguaje de programación ligero, interpretado y orientado a objetos, es utilizado principalmente para el desarrollo web pero puede abarcar más entornos fuera de un navegador. Es ideal para agregar dinamismo a las paginas web, por lo que, principalmente se utilizará transacciones de una pagina a otra y para verificar los datos que se están dando de entrada en el momento ya que es un lenguaje compilado en línea o también conocido como just-in-time.
* **Servidor** **Apache**: es uno de los servidores web mas importantes del mundo. Utilizado por grandes empresas como Netflix, Microsoft, PayPal…etc. Es, además un servidor web HTTP de código abierto para plataformas Unix y sigue mantenido y desarrollado por Apache Software Foundation. A continuación, voy a indicar las ventajas por las cuales ha sido escogido:
  + Código abierto y gratuito, incluso para uso comercial.
  + Software confiable y estable
  + Parches de seguridad regulares y actualizados con frecuencia
  + Fácil de configurar, ideal para principiantes
  + Multiplataforma
  + Tiene una gran comunidad y soporte lo que facilita la solución de cualquier duda.
* **PHP:** Lenguaje de programación del lado del servidor el cual se encargara de establecer la conexión entre el usuario y la base de datos.
* **Visual Studio Code:** Es un editor de código fuente desarrollado por Microsoft. Es software libre y multiplataforma, está disponible para Windows, GNU/Linux y macOS. VS Code tiene una buena integración con Git, cuenta con soporte para depuración de código, y dispone de un sinnúmero de extensiones, que básicamente te da la posibilidad de escribir y ejecutar código en cualquier lenguaje de programación.
* **Base de datos MySQL:** Es un sistema open source de gestión de base de datos relacional, desarrollado y soportado por Oracle. La elección de haber elegido este sistema de base de datos son las siguientes:
  + **Popularidad** **y** **comunidad** **activa**: MySQL es uno de los sistemas de gestión de bases de datos mas utilizados en el mundo., por lo que, muchos usuarios, desarrolladores e incluso expertos pueden ayudar en caso de alguna incidencia.
  + **Mucha** **documentación**: MySQL cuenta con una gran documentación y bastante extensa que abarca todas sus características al detalle e incluso en Castellano, aquí dejo un enlace para echarle un vistazo al [manual](https://downloads.mysql.com/docs/refman-5.0-es.pdf).
  + **Coste**: Como antes hemos comentado es de código abierto y esta disponible de manera gratuita bajo una licencia publica GNU(GPL). Lo que se traduce a un ahorro significativo en costes
  + **Seguridad**: Tiene características solidas en lo que se refiere a la seguridad lo que ayuda a proteger la integridad y confidencialidad de los datos que se almacenan
  + **Escalabilidad**: MySQL es capaz de manejar una gran cantidad de datos y hacer transacciones simultaneas.
  + **Integración** **con** **PHP**: PHP proporciona una API la cual puede interactuar con las bases de datos de MySQL de manera nativa, es fácil de implementar la relación entre ambas tecnologías y son muy eficientes en términos de rendimiento dado que PHP fue diseñado para trabajar con MySQL.
* **XAMPP**: Utilizaremos esta herramienta de desarrollo ya que nos permitirá probar la aplicación web en nuestro propio equipo y también sin tener necesidad de tener acceso a internet. En él incluyen diferentes softwares libres como como el servido Apache, MySQL/Mariadb el cual encontraremos phpMyAdmin que será otro administrador de base de datos.

Dicho lo cual, el conjunto de estas tecnologías crea una buena base para un proyecto el cual tiene la capacidad de seguir creciendo en muchos aspectos para poder mejorar la eficiencia y producción de nuestros usuarios.

* **Driagrams.net**: Es una aplicación web gratuita que nos permite crear diagramas, que incluyen diagramas de flujos, diagramas de entidad relacion entre otros tipos de graficos.

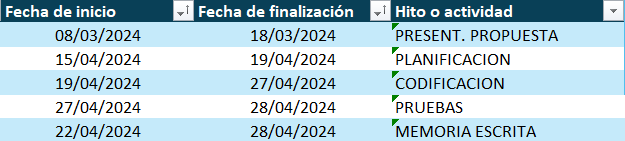
# Planificación, Diagnóstico y Contexto Laboral (1-2 páginas)

Para la planificación he utilizado el diagrama de Gantt, a través de una plantilla ofrecida por Excel. He ido apuntando el registro de cada día que he avanzado en el proyecto y en lo que me he enfocado.

No he tenido suficiente tiempo ya que he tenido un viaje personal y no pude avanzar como me hubiera gustado, pero se ha logrado avances importantes en este pequeño periodo de tiempo que he podido estar disponible, y crear una base para poder continuar en el en un futuro.

Como se puede ver, la presentación de la propuesta y planificación ha sido algo larga debido a que, quería abarcar tanto al principio para un SGA que no era posible, también en un principio estaba enfocado en usar Python como lenguaje de programación, pero se opto por los lenguajes que había utilizado en este grado y sin utilizar ningún framework. Es decir, este proyecto se ha elaborado con lo que hemos visto en las clases.





La fase de codificación y pruebas realmente iban de la mano, ya que cuando iba acabando alguna funcionalidad la ponía a prueba e iba puliendo los errores.

Para hacer un diagnostico del proyecto se ha realizado un análisis DAFO, es una estupenda herramienta para identificar las debilidades, amenazas, fortalezas y oportunidades de negocio.

DEBILIDADES:

Despliegue de la aplicación muy basico, ya que ha sido enfocado en que el proyecto resida en un equipo de manera local

Poco conocimiento de desarrollo en aplicaciones web el cual faltan muchos conceptos para elaborar un proyecto competente

Escasas funcionalidades

AMENAZAS

Fuerte competencia frente a otros productos que están en el mercado desde hace tiempo

FORTALEZAS

Experiencia en lo que un usuario de un SGA quiere para ser mas productivo y eficiente a la hora de ejercer su trabajo ya que he estado en logística este tiempo

Al tener esta aplicación web un modelo iterativo incremental permite a seguir adaptándose a las necesidades que el cliente requiera

Puede adaptarse a cualquier sector comercial que requiera una gestión de inventario

OPORTUNIDADES

Como antes se ha comentado, el crecimiento de lo que es logística y por ende sus sistemas para organizar, monitorizar, producir… seguirá en aumento y buscando siempre nuevas ideas para que aumente la eficiencia de los conceptos antes mencionados.

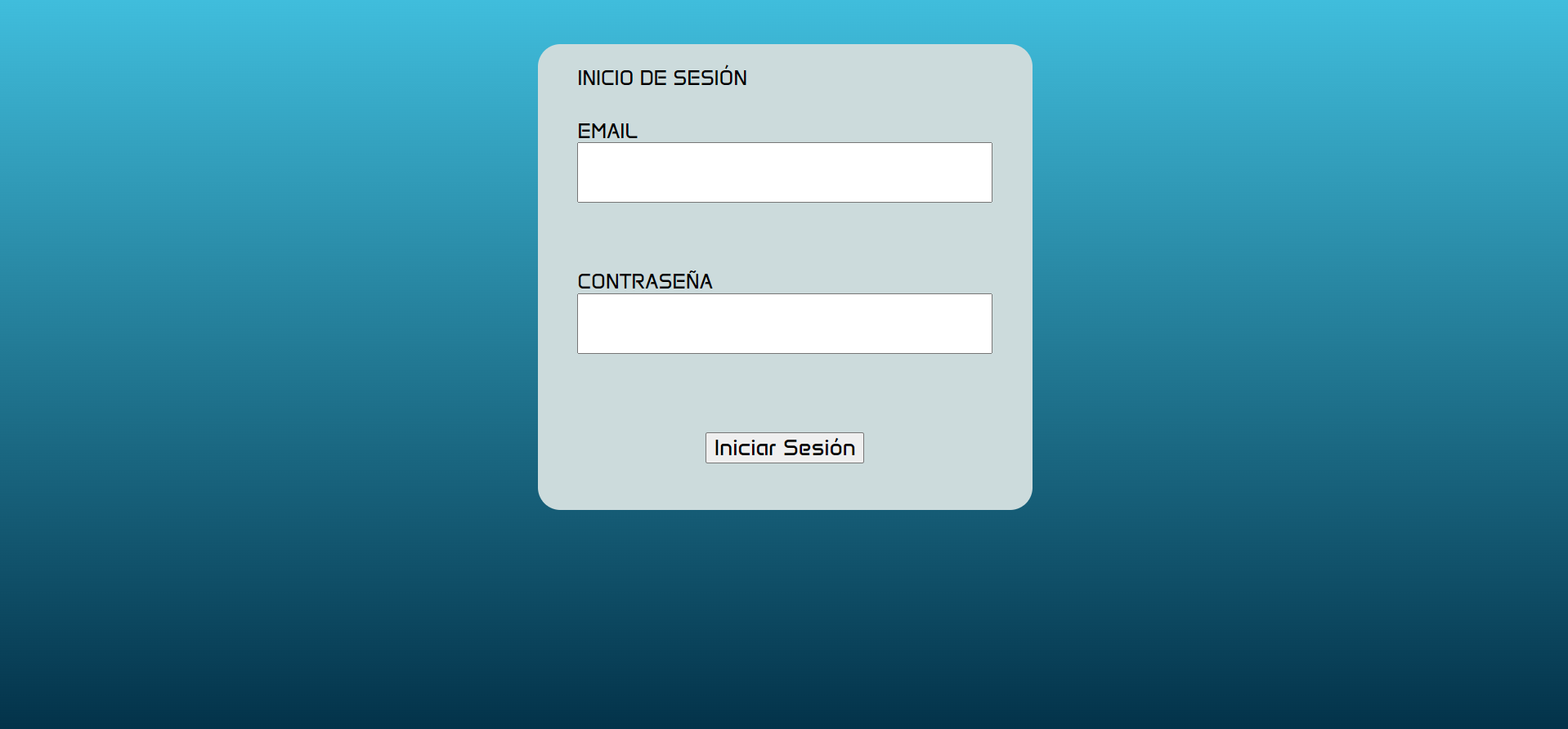
SAI en contexto laboral puede ejercer en una infinidad de procesos en los que se basa un bloque logístico, actualmente cabe recalcar que no puede abarcar mucho debido al desarrollo escaso que se ha tenido, pero si a la gestion de un inventario. En un futuro puede ayudar a la eficiencia operativa, a un seguimiento y trazabilidad de los suministros más exacto, obtener información detallada del almacén; cuantos artículos hará falta, producción de empleados y optimización del recorrido a la hora de preparar pedidos. Es decir, es una herramienta muy importante el cual mejora la rentabilidad de las operaciones logísticas de la empresa.

# Análisis del proyecto (8-16 páginas)

### Requisitos Funcionales

ACCESO

El acceso en la aplicación solo será posible si están registrados, si no lo estuvieran no tendrían acceso, lo que ayuda a denegar el acceso indebido de agentes externos y solo podrá registrar usuario el *super-admin.*

**

En caso de que el login sea correcto habrá 3 casos dependiendo de que rol tenga el usuario:

1. Si el usuario es super-admin: accederá directamente a la página de adm.usuarios.
2. Si el usuario es admin: accederá directamente a la página de adm.articulos.
3. Si el usuario es estándar: accederá directamente a la página de entradas

ROLES

* **Super-admin**: Este usuario puede acceder a cualquier funcionalidad, es decir, puede crear usuarios, eliminar y modificarlos al igual con las ubicaciones y artículos, como también, puede dar de entrada en caso que haga falta.
* **Admin**: Este usuario tendrá el mismo acceso que el super-admin excepto, el poder crear, modificar y eliminar usuarios en la página de adm.usuarios.
* **Standar**: Este ultimo usuario solo tendrá permiso para dar de entrada a los artículos y ubicarlos.

NAVEGACION

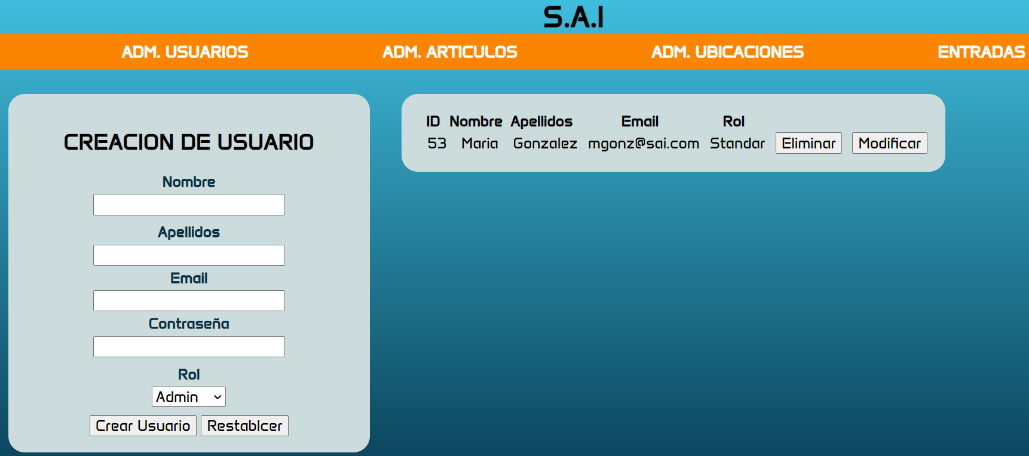
* ADM. USUARIOS

Los usuarios deberán de ser registrados siempre por el super-admin a través de la aplicación en adm.usuarios el cual se deberá introducir:

* Nombre
* Apellidos
* Email
* Contraseña,
* Rol, solo habrá disponibles dos opciones; Admin y Stándar.

Es importante recalcar que en caso de que exista un usuario existente, el cual se identificara si lo es por el email, no dejara crearlo. De todas maneras, tendremos una tabla al lado del formulario el cual aparecerá los usuarios creados y sus datos como, id, nombre, apellidos, email y su rol, junto con dos botones; eliminar y modificar, que como su propio nombre indica podremos eliminar y modificar dichos usuarios.

Gracias a la función que implementa PHP de cifrar las contraseñas sera mas difícil para los agentes externos no autorizados a poder acceder a la aplicación dado que cogerá la contraseña de entrada y la cifrará utilizando un algoritmo de cifrado seguro. Posteriormente utilizaremos otra función de PHP para comparar la contraseña introducida con la contraseña cifrada en la base de datos a la hora de hacer login.



**IMPORTANTE**

Nunca aparecerá el super-admin para evitar posibles incidencias, si se desea modificar algún dato de él, habrá que realizarlo directamente en la base de datos.

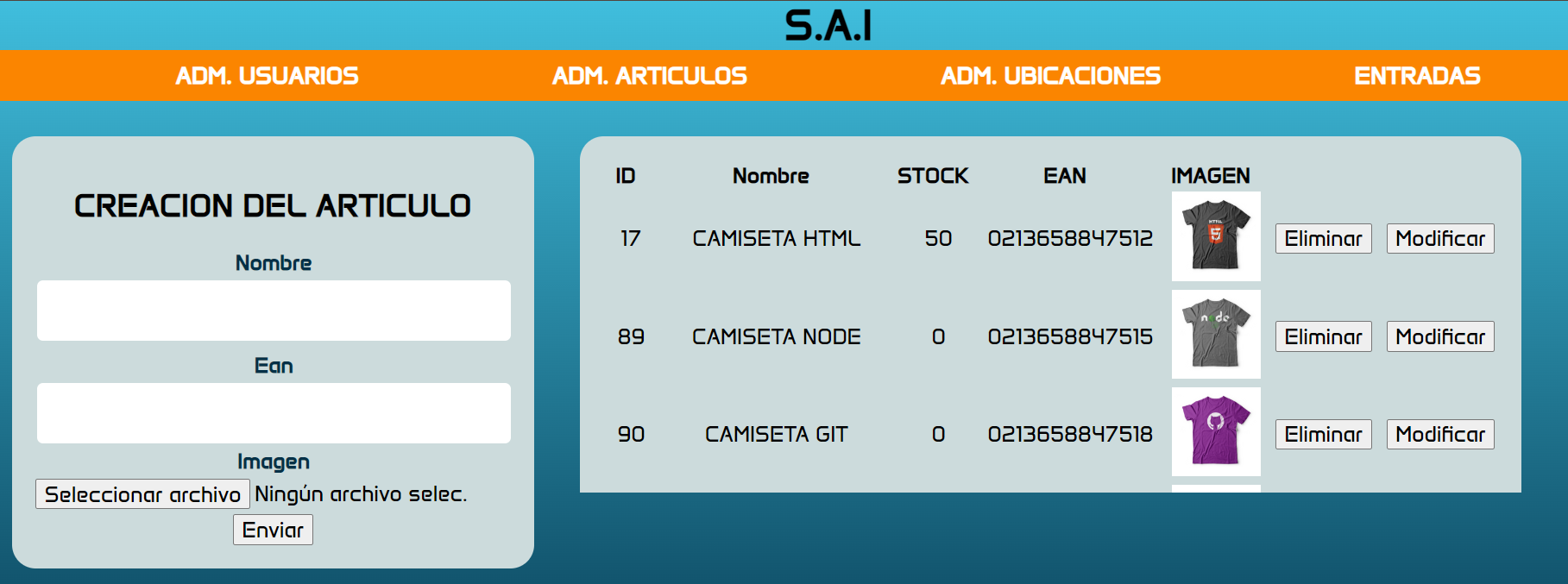
* ADM.ARTICULOS

Aquí podremos crear los artículos que vayamos a tener en nuestro inventario para poder gestionar con ellos posteriormente, los datos que nos solicitaran será el nombre, el EAN del articulo y si deseamos una imagen. Los campos de nombre y EAN son requeridos por lo que no podemos dejarlos en blanco.

Es importante saber que el EAN se compone de 13 cifras por lo que en caso de excedernos nos arrojara un error y no podremos continuar.

He elegido el almacenar las imágenes en nuestra raiz del proyecto ya que si almacenamos las imágenes directamente en nuestra base de datos puede ocasionar muchos problemas como, que aumente considerablemente el tamaño de la base de datos, disminuya el rendimiento… etc. Por lo tanto, podremos escoger una imagen de nuestro equipo local y la aplicación creara un carpeta en caso de que no exista en la raíz y almacenara ahí las imágenes, y lo que se guardara realmente en la base de datos de cada articulo será su el nombre de la imagen que será única ya que si almacenamos el mismo nombre sobrescribirá la imagen así evitamos errores.

Como en todas las paginas tendremos una tabla al lado del formulario el cual contendrán los datos de los artículos, cuando sobrepase cierta dimensión la tabla podremos hacer scroll en ella para ver los artículos faltantes.



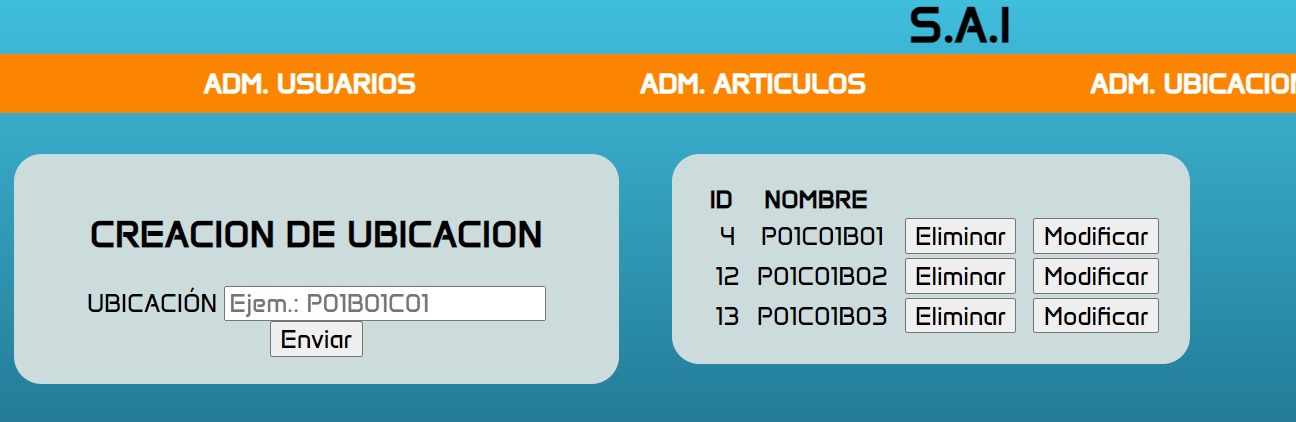
**IMPORTANTE**

A la hora de querer eliminar un articulo el cual tenga stock y por ende tenga una ubicación asignada no lo permitirá la aplicación, en caso de que queramos eliminar el articulo tendremos que previamente eliminar las entradas de ese artículo.

* ADM.UBICACIONES

Aquí podremos crear las ubicaciones nuestro sitio, he optado por especificar un formato en el nombre de la ubicación que sera PnºnºCnºnºBnºnº, haciendo referencia a P = pasillo, C = columna, B = balda, un ejemplo es P02C05B07. Para garantizar que el formato que introduzcamos sea el que queremos he optado por dos moderadores, uno de ellos en el propio formulario el cual sea lo que introduzcamos en el nombre de la ubicación será en mayúscula siempre y la otra, es un script de JavaScript el cual, sino cumple el formato, nos saltará un pop-up o una ventana emergente el cual nos avisara que el formato introducido no es correcto y anulará el envío del formulario.

En adm.ubicaciones tendremos un formulario y una tabla de igual manera a las demás paginas mencionadas.



**IMPORTANTE**

A la hora de querer eliminar una ubicación el cual tenga artículos anclados en ella, no lo permitirá la aplicación, en caso de que queramos eliminar el articulo tendremos que previamente eliminar las entradas de ese artículo.

* ENTRADAS

En entradas podremos ubicar los artículos con una cantidad, en el formulario deberemos de introducir el ean, la posición y la cantidad que vayamos a dar de entrada a nuestro inventario.

Cuando vamos a dar de entrada un articulo es importante que previamente se haya dado de alta el articulo y también la ubicación, sino nos arrojara la aplicación un error y no se creara la entrada.

Aquí también, tendremos una tabla la cual nos proporcionara los datos de cada entrada que hayamos hecho con su boton de accion eliminar.



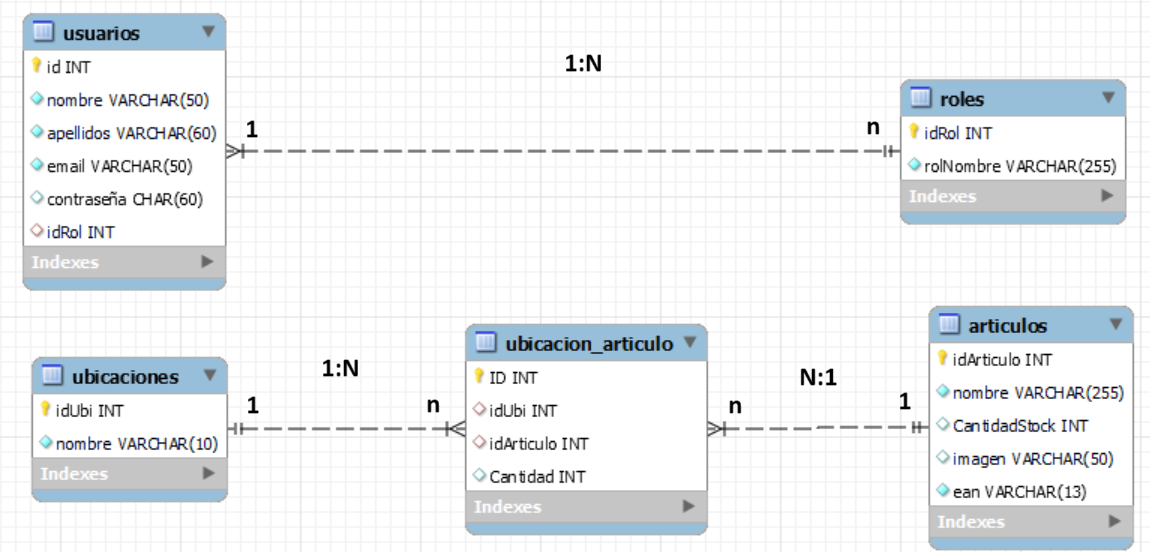
Tanto cuando se da de entrada, como cuando se elimina una entrada, se actualizará el stock del articulo mediante unos triggers que he aplicado en la base de datos, lo cual se activaran cuando se inserte o se elimina un registro en la tabla de ubicaciones\_articulo y el stock esta actualizado constantemente.

### Diagrama Entidad – Relación (ER)

* Roles
  + idRol (PK)
  + rolNombre
* Usuarios
  + idRol (FK)
  + id
  + nombre
  + apellidos
  + email
  + contraseña
* Articulos
  + idArticulo (PK)
  + nombre
  + CantidadStock
  + Imagen
  + Ean
* Ubicación\_articulo
  + idArticulo (FK)
  + idUbi (FK)
  + ID
  + Cantidad
* Ubicaciones
  + idUbi (PK)
  + nombre

Relaciones

* **Rol** tiene una relación de 1 a muchos **Usuarios**
* **Artículos** puede tener muchas **ubicaciones** como **ubicaciones** puede tener muchos **artículos**, por ende, se crea la tabla **artituclos**\_**ubicaciones**
* **Artículos** tiene una relación de 1 a muchos con **ubicaciones**\_**articulos**, ya que un **articulo** puede estar en distintas **ubicaciones**
* **Ubicaciones** tiene una relación de 1 a muchos con **ubicaciones**\_**articulos**, ya que una **ubicación** puede tener varios registros en **ubicaciones**



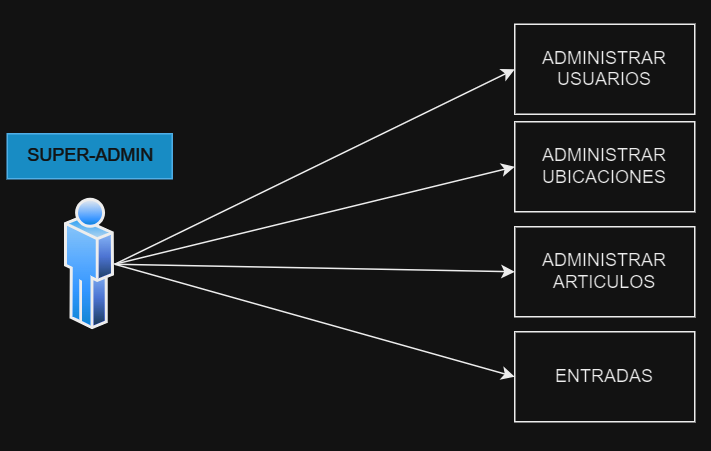
ACTORES

* Super-admin
* Admin
* Estándar

Casos de uso en Super-Admin

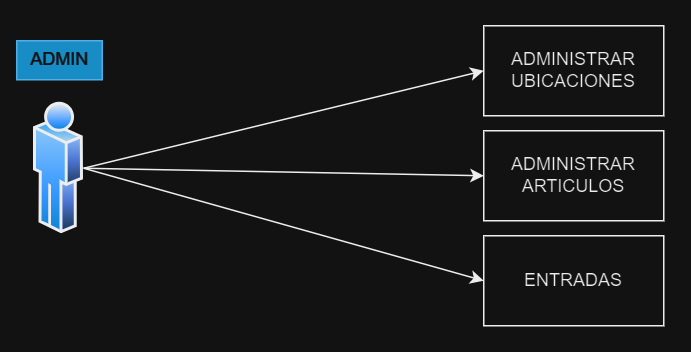
1. Registrar, modificar o eliminar usuarios
2. Registrar, modificar o eliminar ubicaciones
3. Registrar, modificar o eliminar artículos
4. Dar entradas a los artículos

El super-admin tiene acceso a todas funciones de la aplicación



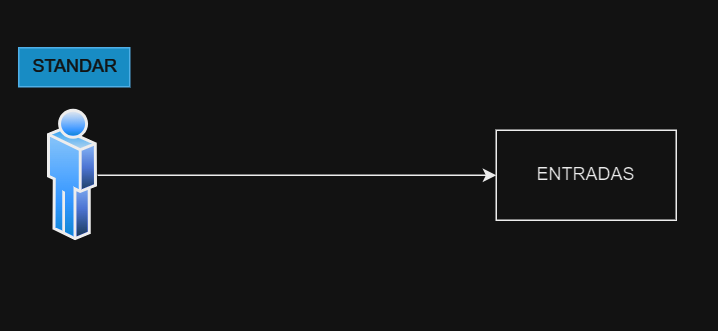
Casos de uso en Admin

1. Registrar, modificar o eliminar ubicaciones
2. Registrar, modificar o eliminar artículos
3. Dar entradas a los artículos



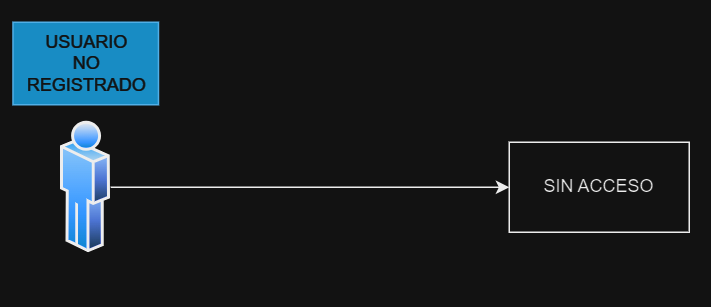
Casos de uso Standar

1. Dar entradas a los artículos



Caso de uso siendo un usuario no registrado

1. Sin acceso a la aplicacion



# Diseño del proyecto (6-14 páginas)

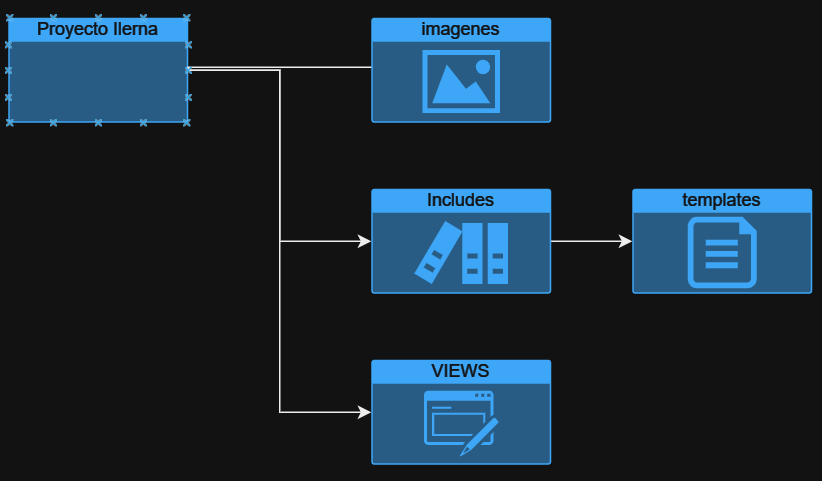
### Base de datos

Lo primero que se planteó en la aplicación fue como elaborar la base de datos, ya que tiene mucho más peso que el apartado visual. La base de datos que se utiliza fue realizada con MySQL Workbench.

Como las funciones principales era la administración de usuarios y en funciones de stock, lo que hicimos fue crear las tablas usuarios y roles ya que seria primordiales a la hora del acceso a la aplicación y el control de acceso a las funciones, dependiendo del rol de cada usuario.

Por otra parte, creamos las tablas de artículos y ubicaciones para dar funcionalidad a la aplicación, pero como la relación entre ambas tablas era de N:M se tenia que crear una tabla auxiliar el cual ayudaría a la regla de que un articulo podía estar en varias ubicaciones y una ubicación podría contener varios artículos distintos.

### Estructura



Principalmente nos encontraremos con la carpeta raiz **Proyecto Ilerna**, dentro de ella nos encontraremos con la carpeta de **imágenes** en la que almacena las imágenes que subimos a través del formulario al crear un articulo si asi lo deseamos.

En la carpeta **includes** se encontrarán todas las funciones y consultas que realizara la aplicación con la base de datos, en ella encontraremos los siguientes archivos:

* db.php: Realiza la conexión de la base de datos.
* funciones\_articulos.php: Realiza las consultas y funciones que usaremos en la vista de adm\_articulos.php y mod\_articulo.php.
* funciones\_entrada.php: Realiza las consultas y funciones que usaremos en la vista de entrada\_articulos.php
* funciones\_ubicaciones.php: Realiza las consultas y funciones que usaremos en la vista de adm\_ubicaciones.php
* funciones\_usuarios.php: Realiza las consultas y funciones que usaremos en la vista de entrada\_articulos.php y mod\_usuario.php
* logout.php: Es un script PHP el cual su función es la de cerrar la sesión del usuario y redirigirlo a la pagina de inicio de sesión, es decir, a login.php.

Dentro de la carpeta includes, encontraremos otra carpeta llamada templates el cual se encontrará los siguientes archivos:

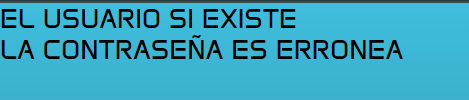
* estilo.css: archivo que da formato y estilo a nuestro html.
* header.php: archivo PHP el cual utilizara como plantilla para todas las vistas de la aplicación, asi no tendremos que repetir constantemente el mismo codigo en cada uno de los archivos de vista como adm\_ubicaciones.php, adm\_articulos.php, adm\_usuarios.php, etc.
* normalize.css: Es un fichero CSS el cual ayuda a mantener los estilos similares en los distintos navegadores, ya que podría presentarse alteraciones en cada uno de ellos porque aplican algunos sus propios estilos por defecto.

Y, por último, nos encontramos con la carpeta VIEWS, que en ella contendrá todas las los ficheros los cuales contendrán lo visual de la aplicación, estos son los ficheros:

* adm\_articulos.php
* adm\_ubicaciones.php
* adm\_usuarios.php
* entrada\_articulos.php
* login.php
* mod\_articulo.php
* mod\_usuario.php

Comenzaremos una breve introducción al diseño y funcionamiento con el super-admin para poder explicar todas las vistas y procesos de cada apartado ya que es el único usuario que tiene acceso a todas ellas.

Accederemos a login.php para poder iniciar sesion, e introduciremos nuestras credenciales de super.admin en el formulario que consultará si los datos introducidos son correctos a traves de la funcion inicioSesion, esta funcion primero accederá a la base de datos, hara una consulta si el email se encuentra en la tabla de usuarios y si existe, seguidamente se usara una funcion que integra PHP llamada password\_verify el cual compara la contraseña introducida con la contraseña cifrada, si es correcto, iniciará la sesion y registrará su idRol para saber que tipo de usuario es e identificar en cuales vistas tiene acceso y en cuales no. En caso contrario, si el usuario introducido existe pero la contraseña es erronea, nos indica dicho error o en caso de que el email introducido no existe nos indicara que no existe.

Como accederemos con el rol de super-admin, nos redirigirá automáticamente a la vista de adm\_usuarios, la cual solo tendra acceso el super-admin. En la pantalla veremos el nombre de la aplicación S.A.I seguidamente y en la parte izquierda el enlace de cerrar sesion el cual esta asociado a un script de PHP que se encuentra en el archivo logout.php el cual cerrara completamente la sesion actual y nos redirige a la pagina de inicio de sesión, continuando la la explicación del diseño, seguido del enlace de cerrar sesión nos encontramos con un navegador del menú, el cual se encuentra todas las vistas de la aplicación web. Todo lo detallado anteriormente será usado como una plantilla, la cual se encuentra en el archivo header.php.



# Despliegue y pruebas (4-10 páginas)

Despliegue.

a. Configuración del servidor web:

- Utilizare Apache como servidor web en el torno de producción para alojar la aplicación, para mi aplicación escogí la XAMMP.

b. Base de datos MySQL:

- Instalación y configuración de Mysql para el almacenamiento de datos, en mi caso he escogido MySQL Workbench.

Despliegue de la Aplicación:

a. Migración de Código y base de datos:

- Transferir el código fuente y la base de datos desde el entorno de pruebas al entorno de producción.

Verificación post-despliegue:

a. Comprobación de servicios.

- Se realizará pruebas para asegurar de que todos los servicios, incluyendo el servidor web y la base de datos, estén operativos.

## Pruebas Funcionales

* Pruebas en login.php

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Funcionalidad a probar | Descripción | Entradas | Resultados Esperados |
| 1 | Mostrar formulario de inicio de sesión | Verificar que el formulario de inicio de sesión se muestre correctamente. | Acceso a la página | Formulario de inicio de sesión visible. |
| 2 | Inicio de sesión con credenciales válidas | Verificar que el usuario puede iniciar sesión con credenciales correctas. | Email y contraseña correctos | Usuario redirigido según el rol. |
| 3 | Inicio de sesión con credenciales inválidas | Verificar que el usuario no puede iniciar sesión con credenciales incorrectas. | Email y/o contraseña incorrectos | No redirección y posiblemente mensaje de error. |
| 4 | Redirección administrador | Verificar que un usuario con rol de administrador (idRol=1) sea redirigido a adm\_usuarios.php. | Email y contraseña de administrador | Redirección a adm\_usuarios.php. |
| 5 | Redirección usuario estándar | Verificar que un usuario con rol de usuario estándar (idRol=2) sea redirigido a adm\_ubicaciones.php. | Email y contraseña de usuario estándar | Redirección a adm\_ubicaciones.php. |
| 6 | Redirección usuario de inventario | Verificar que un usuario con rol de inventario (idRol=3) sea redirigido a entrada\_articulos.php. | Email y contraseña de inventario | Redirección a entrada\_articulos.php. |
| 7 | Campo email obligatorio | Verificar que el campo de email es obligatorio. | Dejar campo de email vacío | Mensaje de error y no se permite enviar el formulario. |
| 8 | Campo contraseña obligatorio | Verificar que el campo de contraseña es obligatorio. | Dejar campo de contraseña vacío | Mensaje de error y no se permite enviar el formulario. |
| 9 | Validación de formato de email | Verificar que el campo de email tiene un formato válido. | Email con formato incorrecto | Mensaje de error y no se permite enviar el formulario. |
| 10 | CSS y estilo | Verificar que los estilos CSS se cargan correctamente y el formulario tiene el diseño esperado. | Acceso a la página | Formulario de inicio de sesión con diseño aplicado correctamente. |

* Pruebas en adm\_articulos.php

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Funcionalidad a probar | Descripción | Entradas | Resultados Esperados |
| 1 | Redirección según el rol | Verificar que los usuarios no autorizados sean redirigidos. | Rol de usuario no autorizado (ej. 3) | Redirección a la página principal. |
| 2 | Acceso autorizado | Verificar que los usuarios autorizados puedan acceder. | Rol de usuario autorizado (1 o 2) | Acceso permitido a la página. |
| 3 | Mostrar formulario de creación de artículo | Verificar que el formulario de creación de artículo se muestre correctamente. | Acceso a la página | Formulario de creación de artículo visible. |
| 4 | Crear artículo | Verificar que un artículo se puede crear correctamente. | Datos de artículo (nombre, ean, imagen) | Mensaje de éxito "creado" y redirección a la página de administración. |
| 5 | Crear artículo duplicado | Verificar que no se puedan crear artículos duplicados. | Datos de artículo duplicado | Mensaje de error "ya existe el articulo". |
| 6 | Visualización de artículos | Verificar que los artículos existentes se muestren en la tabla. | N/A | Lista de artículos en la tabla con detalles correctos. |
| 7 | Eliminar artículo | Verificar que un artículo se puede eliminar correctamente. | ID del artículo a eliminar | Artículo eliminado de la tabla y redirección a la página de administración. |
| 8 | Modificar artículo | Verificar que se redirige a la página de modificación con el ID correcto. | ID del artículo a modificar | Redirección a mod\_articulo.php con el ID del artículo correcto. |
| 9 | Subida de imagen | Verificar que solo se permiten imágenes en formato JPEG y PNG. | Subida de imagen en formato permitido | Imagen cargada correctamente. |
| 10 | Subida de imagen incorrecta | Verificar que no se permitan otros formatos de imagen. | Subida de imagen en formato no permitido | Mensaje de error y no se carga la imagen. |
| 11 | Campo nombre | Verificar que el campo nombre se convierte a mayúsculas al escribir. | Texto en minúsculas | Texto convertido a mayúsculas. |
| 12 | Campos obligatorios | Verificar que los campos nombre y ean son obligatorios. | Formulario sin completar | Mensaje de error y no se permite enviar el formulario |

* Pruebas en adm\_ubicaciones.php

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Funcionalidad a probar | Descripción | Entradas | Resultados Esperados |
| 1 | Redirección según el rol | Verificar que los usuarios no autorizados sean redirigidos. | Rol de usuario no autorizado (ej. 3) | Redirección a la página principal. |
| 2 | Acceso autorizado | Verificar que los usuarios autorizados puedan acceder. | Rol de usuario autorizado (1 o 2) | Acceso permitido a la página. |
| 3 | Mostrar formulario de creación de ubicación | Verificar que el formulario de creación de ubicación se muestre correctamente. | Acceso a la página | Formulario de creación de ubicación visible. |
| 4 | Crear ubicación | Verificar que una ubicación se puede crear correctamente. | Nombre de la ubicación | Mensaje de éxito "creado" y redirección a la página de administración. |
| 5 | Crear ubicación duplicada | Verificar que no se puedan crear ubicaciones duplicadas. | Nombre de la ubicación duplicada | Mensaje de error "la ubicacion ya existe". |
| 6 | Visualización de ubicaciones | Verificar que las ubicaciones existentes se muestren en la tabla. | N/A | Lista de ubicaciones en la tabla con detalles correctos. |
| 7 | Eliminar ubicación | Verificar que una ubicación se puede eliminar correctamente. | ID de la ubicación a eliminar | Ubicación eliminada de la tabla y redirección a la página de administración. |
| 8 | Modificar ubicación | Verificar que se redirige a la página de modificación con el ID correcto. | ID de la ubicación a modificar | Redirección a mod\_ubicacion.php con el ID de la ubicación correcto. |
| 9 | Validación de formato de nombre | Verificar que el nombre de la ubicación cumple con el formato especificado. | Nombre en formato correcto | Formulario enviado correctamente. |
| 10 | Validación de formato incorrecto | Verificar que el nombre de la ubicación no se acepta si no cumple con el formato especificado. | Nombre en formato incorrecto | Mensaje de error y no se permite enviar el formulario. |
| 11 | CSS y estilo | Verificar que los estilos CSS se cargan correctamente y el formulario tiene el diseño esperado. | Acceso a la página | Formulario de creación de ubicación con diseño aplicado correctamente. |
| 12 | Mensajes de sesión | Verificar que los mensajes de sesión se muestran y se limpian correctamente. | Creación de ubicación | Mensaje de éxito o error y variable de sesión limpiada. |

* Prueba en adm\_usuarios.php

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Funcionalidad a probar | Descripción | Entradas | Resultados Esperados |
| 1 | Redirección según el rol | Verificar que los usuarios no autorizados sean redirigidos. | Rol de usuario no autorizado (ej. 2 o 3) | Redirección a la página /dasdsa. |
| 2 | Acceso autorizado | Verificar que los usuarios autorizados puedan acceder. | Rol de usuario autorizado (1) | Acceso permitido a la página. |
| 3 | Mostrar formulario de creación de usuario | Verificar que el formulario de creación de usuario se muestre correctamente. | Acceso a la página | Formulario de creación de usuario visible. |
| 4 | Crear usuario | Verificar que un usuario se puede crear correctamente. | Nombre, Apellidos, Email, Contraseña, Rol | Mensaje de éxito "creado" y redirección a la página de administración. |
| 5 | Crear usuario duplicado | Verificar que no se puedan crear usuarios duplicados. | Email duplicado | Mensaje de error "ya existe el usuario". |
| 6 | Visualización de usuarios | Verificar que los usuarios existentes se muestren en la tabla. | N/A | Lista de usuarios en la tabla con detalles correctos. |
| 7 | Eliminar usuario | Verificar que un usuario se puede eliminar correctamente. | ID del usuario a eliminar | Usuario eliminado de la tabla y redirección a la página de administración. |
| 8 | Modificar usuario | Verificar que se redirige a la página de modificación con el ID correcto. | ID del usuario a modificar | Redirección a mod\_usuario.php con el ID del usuario correcto. |
| 9 | Validación de campos obligatorios | Verificar que los campos obligatorios (Nombre, Apellidos, Email, Contraseña, Rol) están presentes y son requeridos. | Campos obligatorios | Formulario enviado correctamente si todos los campos están completos. |
| 10 | Restablecer formulario | Verificar que el formulario se puede restablecer correctamente. | N/A | Todos los campos del formulario vaciados al hacer clic en "Restablecer". |
| 11 | Mensajes de sesión | Verificar que los mensajes de sesión se muestran y se limpian correctamente. | Creación de usuario | Mensaje de éxito o error y variable de sesión limpiada. |

* Pruebas en entradas\_articulos.php

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Funcionalidad a probar | Descripción | Entradas | Resultados Esperados |
| 1 | Acceso no autorizado | Verificar que los usuarios no autenticados sean redirigidos correctamente. | Sesión no iniciada | Redirección a la página de inicio de sesión. |
| 2 | Acceso autorizado | Verificar que los usuarios con rol autorizado puedan acceder sin problemas. | Rol de usuario autorizado (cualquiera) | Acceso permitido a la página de entradas de artículos. |
| 3 | Formulario visible | Verificar que el formulario de entrada de artículos se muestre correctamente. | Página de entrada de artículos | Formulario visible con campos de EAN, Ubicación y Cantidad. |
| 4 | Registro de entrada de artículo | Verificar que se pueda registrar una entrada de artículo correctamente. | EAN, Ubicación, Cantidad | Entrada de artículo registrada correctamente en la base de datos. |
| 5 | Visualización de entradas de artículos | Verificar que las entradas de artículos registradas se muestren en la tabla. | N/A | Lista de entradas de artículos visible con detalles correctos. |
| 6 | Eliminación de entrada de artículo | Verificar que se pueda eliminar una entrada de artículo correctamente. | ID de la entrada a eliminar | Entrada de artículo eliminada de la tabla y confirmación visual. |
| 7 | Mensajes de confirmación | Verificar que se muestren mensajes de éxito o error después de realizar acciones (crear, eliminar). | Creación o eliminación de entrada | Mensaje de éxito o error visible según la acción realizada. |
| 8 | CSS y estilo | Verificar que los estilos CSS se carguen correctamente y la página tenga el diseño esperado. | Acceso a la página | Página con diseño aplicado correctamente. |

# Conclusiones (1-2 páginas)

Este es mi primer proyecto el cual he utilizado fuentes de información propias por la experiencia de uso en lo que laboralmente se refiere, lógicamente no he podido encontrar mucha información ya que son softwares que son muy privados e incluso diseñados especialmente para las empresas de logística, de todas maneras he intentado replicar con una base solida al respecto en la gestión de usuarios, ubicaciones, artículos y el uso de entradas aunque faltan mejoras de seguridad, estructura en el código, experiencia de usuario y diseño entre otros aspectos. De todas maneras, he intentado cubrir varios ámbitos como el manejo de errores y mensajes al usuario, interacciones con la base de datos, formularios y validaciones de datos, seguridad y gestión de sesiones…etc.

Pero he reflejado todo lo que he podido adquirir en el curso y aunque no sea una gran aplicación y peca en varios aspectos antes comentados, es una aplicación bastante funcional dentro de lo que cabe y con una gran oportunidad de mejoría una vez se adquieran mas conocimientos en el desarrollo web.

# Vías futuras (1-2 páginas)

* Mejoras en Seguridad
* Implementación de un sistema de preparación de pedidos
* Integración de APIs externas
* Mejores en la experiencia del usuario
* Mayor automatización
* Facilidad en el despliegue de la aplicación
* Añadir herramientas en análisis de datos para mayor eficiciencia
* Monitorización en la producción para el seguimiento de los empleados
* Soporte en otros idiomas

# Bibliografía/Webgrafía (1-2 páginas)

**PHP Manual**:

* Disponible en: <https://www.php.net/manual/es/>

**MySQL documentacion**:

* Disponible en: <https://dev.mysql.com/doc/>

**Google Fonts**:

* Disponible en: <https://fonts.google.com/>

**GitHub**:

* Disponible en: <https://github.com/>

**W3Schools Online Web Tutorials**:

* Disponible en: <https://www.w3schools.com/>

**Mozilla Developer Network (MDN) Web Docs**:

* Disponible en: <https://developer.mozilla.org/>

**UDEMY | Desarrollo Web Completo con HTML5, CSS3, JS AJAX PHP y MySQL**

* Disponible en: <https://www.udemy.com/course/desarrollo-web-completo-con-html5-css3-js-php-y-mysql/>

**ILERNA | Libros aportados en el curso**

# Anexos

# Manual de Usuario

