****

**IES AUGUSTO GONZÁLEZ DE LINARES**

**DEPARTAMENTO DE INFORMÁTICA**

**AGILIZANDO MENTES II**

PROYECTO FINAL 1º CURSO

GRADO SUPERIOR DE DESARROLLO DE APLICACIONES MULTIPLATAFORMA

2022/2023

Barrios Fernández, María Carmen

Díez de Paulino, Albano

Espinosa García, Daniel

Gutiérrez Valverde, Ramiro

Índice

[1 ANÁLISIS 3](#_Toc1781279401)

[1.1 Requisitos del proyecto 3](#_Toc1493304925)

[1.2 Planteamiento de la solución 4](#_Toc399838060)

[2 DISEÑO 8](#_Toc1942480978)

[2.1 Diseño lógico de la base de datos 8](#_Toc1142248045)

[2.2 Diseño de la interfaz grafica de java 10](#_Toc1878847232)

[2.3 Diseño web 13](#_Toc1272358406)

[2.4 Odoo 15](#_Toc377598900)

[3 DESARROLLO 20](#_Toc1689736803)

[3.1 Hardware 20](#_Toc299989759)

[3.2 Software 24](#_Toc53326565)

[3.2.1 Software Empleado 24](#_Toc91361736)

[3.2.2 Software Alternativo 27](#_Toc1208300546)

[3.3 Diseño físico de la base de datos 28](#_Toc1319147173)

[3.4 Código java 30](#_Toc315252823)

[3.5 Código HTML+ CSS+ RSS 36](#_Toc1183049109)

[4 PRUEBAS 38](#_Toc740499035)

[4.1 Pruebas unitarias sobre código java (JUnit5) 38](#_Toc1603680655)

[5 DESPLIEGUE 41](#_Toc174510268)

[5.1 Despliegue Web 41](#_Toc733946384)

[5.2 Despliegue app 41](#_Toc872034720)

[6 PLANTEAMIENTO FUTURO 41](#_Toc430776569)

[6.1 Ideas para continuar el desarrollo 41](#_Toc685247610)

[7 ANEXOS 42](#_Toc1754950294)

[7.1 Git 42](#_Toc868389219)

[7.2 Opinión personal de los integrantes del grupo 43](#_Toc1360880188)

[8 ADJUNTOS 43](#_Toc1236132385)

[8.1 Documentación Bases de datos](#_Toc325554173)

[8.2 Documentación Sistemas Informáticos](#_Toc350712539)

[8.3 Documentación FOL](#_Toc1441038054)

[Bibliografía 43](#_Toc621620000)

1. ANÁLISIS

# Requisitos del proyecto

Desde el departamento de informática del I.E.S Augusto González de Linares se nos plantea el siguiente problema a resolver.

El profesorado del departamento de Matemáticas ha solicitado al primer curso del ciclo formativo de grado superior Desarrollo de Aplicaciones Multiplataforma la realización de programas para la mejora del cálculo mental, que se utilizarán después en un concurso de rapidez mental entre los alumnos de 1º a 4º de la ESO.

Partiendo de la base del programa “Cálculo25” incluido en el videojuego “Brain Training” de Nintendo Switch, -del cual una posible primera aproximación se muestra en secciones posteriores, así como de un posible programa para realizar una calculadora-se deberán crear una serie de aplicaciones conforme a las siguientes características:

* Diferentes fases de complejidad, para primero, segundo, tercero y cuarto de la ESO, por ejemplo, teniendo la división como resultado una sola cifra en el caso de primero.
* Debe existir una aplicación que permita preguntar por el resto de la división entera. -Debe existir un programa que utilice el algoritmo de Euclides para hallar el máximo común divisor de dos números.
* El diseño de los programas debe garantizar una usabilidad adecuada para público al que se dirigen y debe incluir elementos que faciliten su accesibilidad para usuarios con posibles limitaciones físicas, visuales, auditivas o de otra índole.

**REQUISITOS TÉCNICOS**

* Cada equipo-empresa deberá elegir un nombre y generará un logo, que se utilizará luego durante toda su actividad corporativa.
* Cada miembro de la empresa deberá firmar un contrato/declaración responsable de obligado cumplimiento.
* Cada empresa deberá crear una página web corporativa en la que se alojará el manual de ayuda y las aplicaciones para su descarga.
* Se crearán pequeñas aplicaciones educativas, utilizando todo lo aprendido en cada módulo del ciclo formativo hasta el momento.
* Estas aplicaciones deberán tener un plan de pruebas y deben ser documentadas.
* Se creará una base de datos con los datos de los participantes en los distintos retos matemáticos. La base de datos debe tener como mínimo tres tablas para almacenar la información de participantes, retos existentes, y partidas. La base de datos deberá ser accesible desde los programas para poder registrar la información de cada participante en cada reto matemático con su nivel de dificultad una vez participe en el mismo, así como su puntuación obtenida. Cada participante deberá registrarse con un alias y un nivel al que pertenece que será único. Se deberán crear las vistas adecuadas que permitan obtener información estadística de las partidas jugadas, los retos matemáticos que más se hayan usado, los alias que más puntuación hayan obtenido en cada reto matemático, los retos con las puntuaciones más bajas...
* Se llevará a cabo una presentación por cada equipo en PowerPoint/Impress u otro gestor de Presentaciones explicando los programas realizados y la arquitectura/estructura sobre la que funciona el sistema completo (la base de datos, los programas, la web, el servicio web…)
* Se hará una presentación de la empresa -de diez minutos de duración- (El jueves de la segunda semana del reto se hará una presentación intermedia)-
* Al término de la presentación, se explicitará cuáles han sido las responsabilidades de cada alumno en la realización del proyecto.

# Planteamiento de la solución

Ante el problema que se nos plantea hemos tomado la solución de desarrollar una app sobre lenguaje “Java” con la librería “Swing” dentro del paquete “javax”, relacionado con el sistema gestor de bases de datos relacionadas “MySQL” y una página web.

Además, enfocamos la aplicación para un uso del profesorado en el análisis del conocimiento matemático, a la vez que el alumnado se divierte con juegos que mejoran su nivel matemático.

Todos los servicios necesarios serán montados sobre Docker en un Alpine Linux y a su vez sobre una máquina virtual, ya que no disponemos de un servidor físico. Además, esta tecnología nos permite una fácil escalabilidad ya que consume pocos recursos en comparación con otras tecnologías como un servidor sobre Ubuntu o Bitnami.

Las funciones que consideramos núcleo son:

* Interfaz gráfica que me permita seleccionar entre el juego Calculo25, una calculadora, juego de calcular el resto de una división al azar, juego de calcular el máximo común divisor de dos números.
* Los 4 programas anteriores en interfaz gráfica.
* Almacenar en una base de datos todos los resultados de las partidas
* Mostrar en la interfaz la clasificación de los juegos.
* Disponer de una web en local.
* Tener un sistema de organización empresarial en Odoo en local.
* Montar un sistema RSS.

Las funciones que consideramos principales son:

* Crear un sistema de “login” y de registro
* Implementar todas las interfaces graficas en un solo Jframe menos la calculadora, para poder ejecutar todas las calculadoras que quiera el usuario.
* Poner niveles según el curso del alumno al juego de Calcula25.
* Disponer de una web en un servidor de Docker.
* Disponer de un servidor de Odoo en Docker.
* Ayudas visuales y auditivas en la interfaz de java.

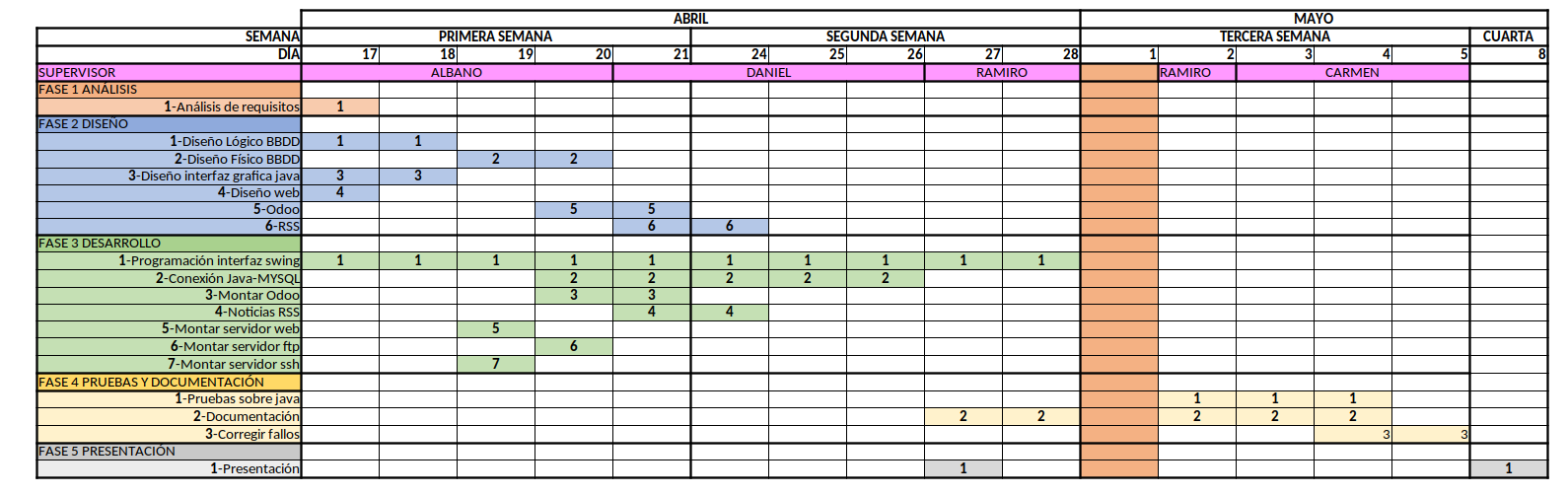
Las funciones que consideramos secundarias son:

* Dividir a los usuarios entre alumnos y profesores.
* Mostrar información diferente al profesor que al alumno (Información del profesor: Clasificación de todos los alumnos, Los datos personales de sus alumnos y acceso a la calculadora).
* Añadir una forma de cambiar los datos personales al usuario.
* Disponer de un servidor FTP, SSH para acceder al servidor web.
* Añadir el logo del I.E.S Augusto González de Linares a la interfaz web.
* Enlace a la web del Instituto desde el logo.

Las funciones que consideramos realizar en un futuro desarrollo:

* Añadir botón de cerrar sesión.
* Añadir más idiomas a la aplicación.
* Añadir un botón que envié a la página web de la empresa.
* Añadir múltiples resoluciones de pantalla.
* Crear una aplicación móvil.

Al solo disponer de 3 semanas de trabajo la organización del proyecto es la siguiente:

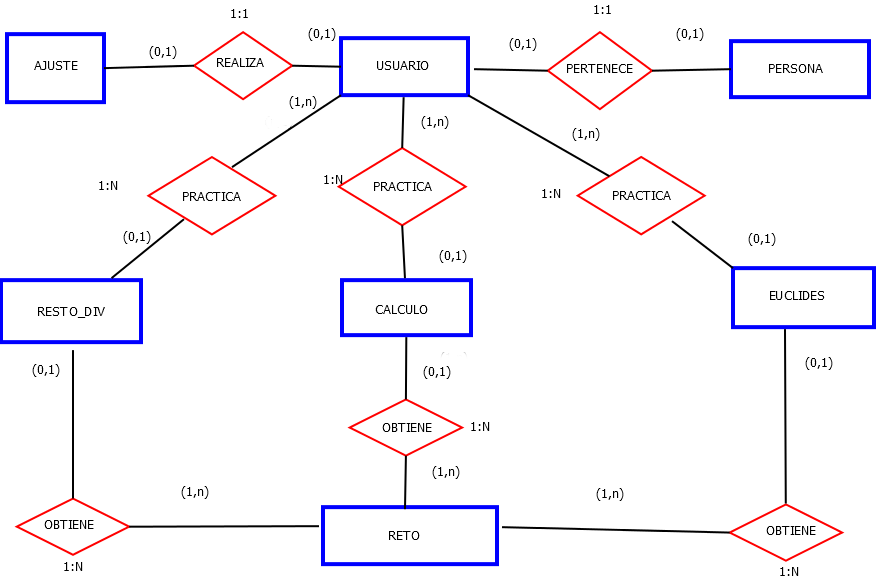


# DISEÑO

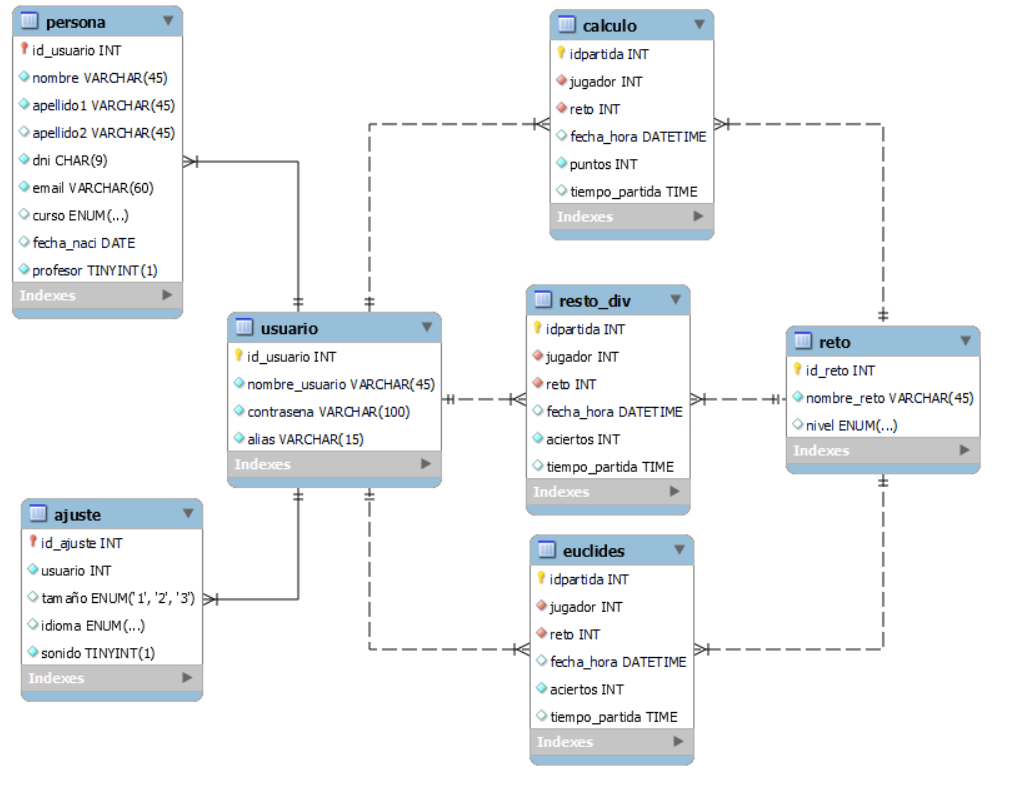
# Diseño lógico de la base de datos

El diseño lógico de la base de datos del proyecto lo hemos ido cambiando desde la idea original de 4 tablas (usuario, datos personales, reto, clasificaciones) que hicimos al inicio del reto, ya que al añadir más funcionalidades a la aplicación de java nos encontramos con la necesidad de almacenar nuevos datos que no estaban previstos al inicio, así que el diagrama Entidad-Relación final es el siguiente:

Teniendo el diagrama anterior, nos sale el siguiente diagrama entidad- relación:



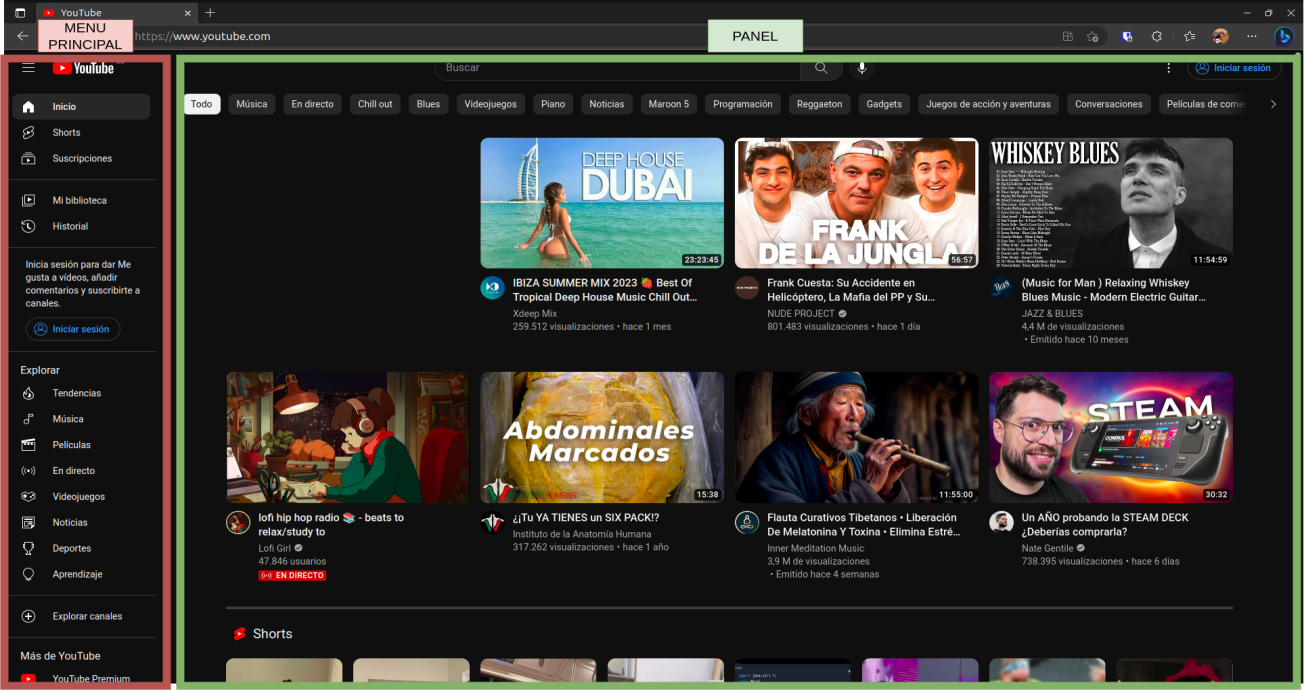
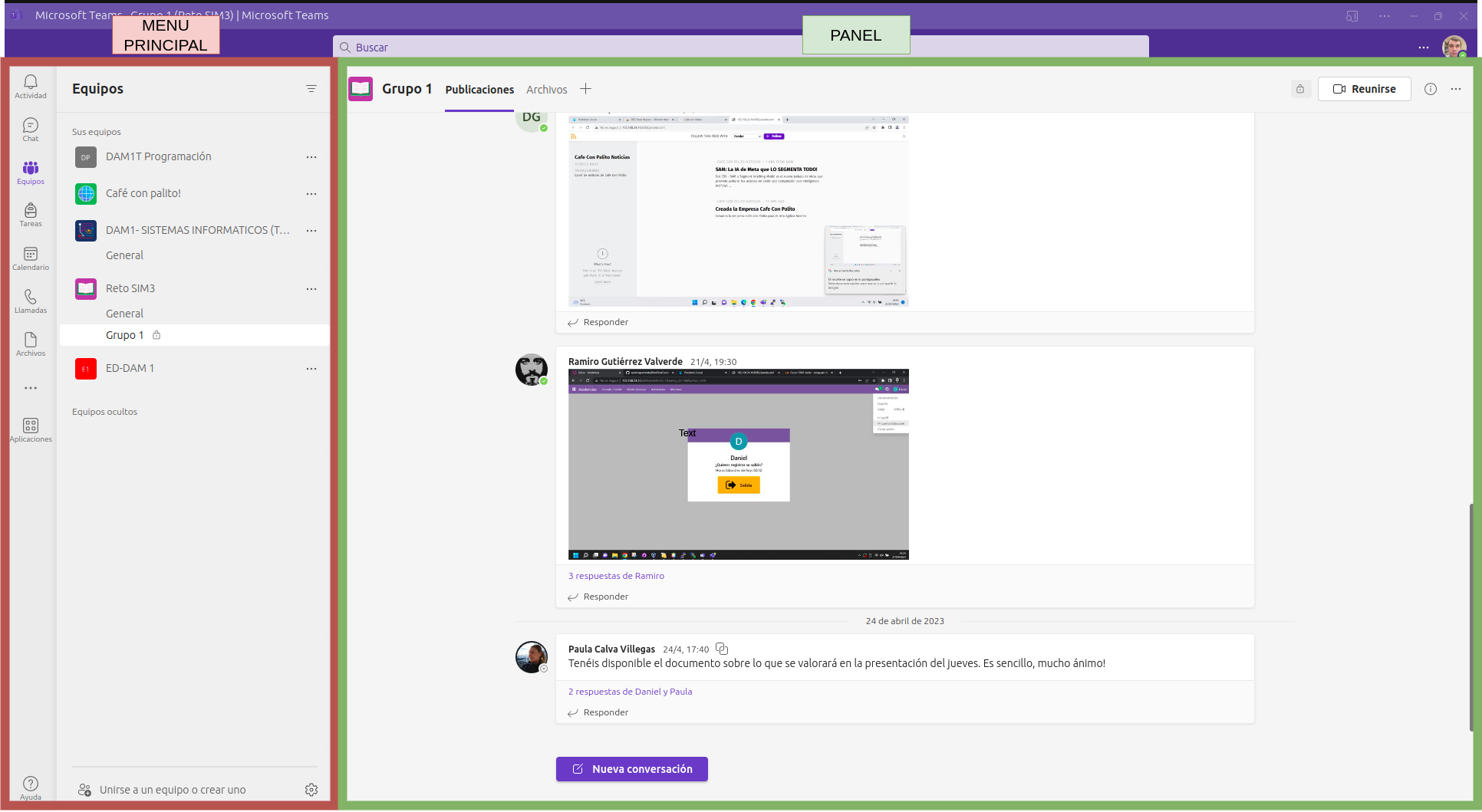
Partiendo del entidad- relación generamos el siguiente esquema relacional, para poder pasar al diseño físico de la base de datos.



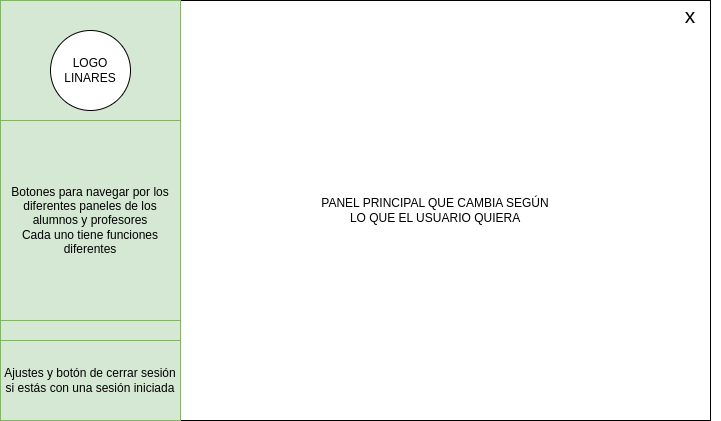
# Diseño de la interfaz gráfica de java

El diseño inicial que planteamos para la interfaz de java está basado en las aplicaciones o aplicaciones web modernas como por ejemplo Teams,

Dichas aplicaciones mantienen un estilo minimista, con una barra vertical a la izquierda que realiza la función de menú principal que nos permite interactuar con todas las funciones de la app y un panel que abarca el resto de la pantalla con las funciones que llamas desde el menú de la izquierda.



Con las referencias que buscamos para el diseño, nos quedamos con la siguiente base para la fase 3 (Desarrollo).

Las funciones que tendrán los alumnos son:

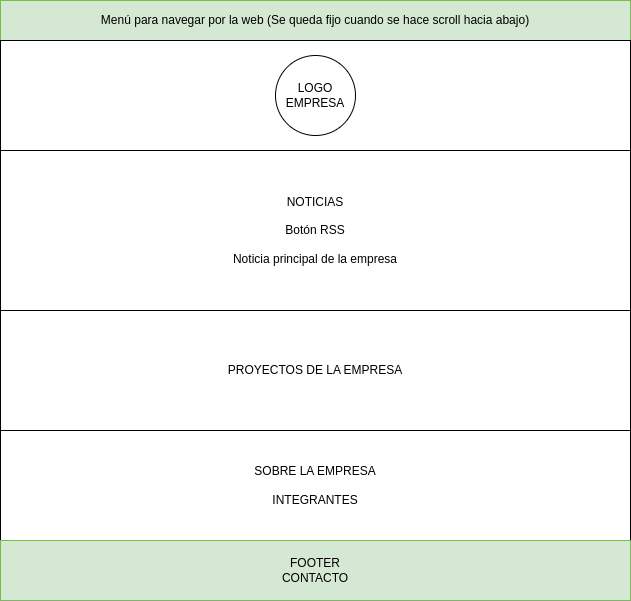
* Juego Calculo25
* Juego Calcular el Resto de una División de números enteros
* Juego para Calcular el algoritmo de Euclides (Máximo común Divisor).
* Calculadora (Que se ejecute en una nueva ventana y todas las que el usuario quiera).

Las funciones que tendrá el profesor son:

* Tabla con todas las clasificaciones de los alumnos y filtros para realizar una búsqueda más efectiva.
* Tabla con los datos personales de los alumnos para lo que considere el profesor oportuno y con filtros para la búsqueda.
* Calculadora igual que los alumnos.

# Diseño web

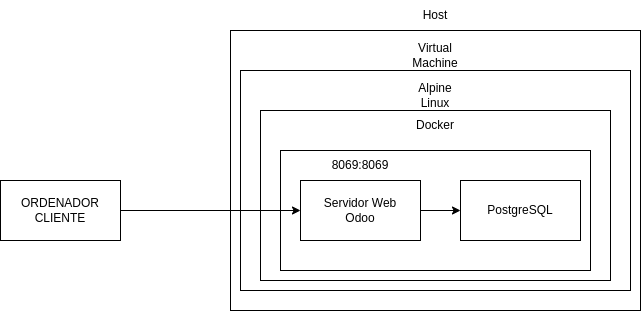
El diseño web será uno muy simple pero bonito con la misma paleta de colores que la aplicación web a continuación, se muestra un boceto inicial.



# Odoo

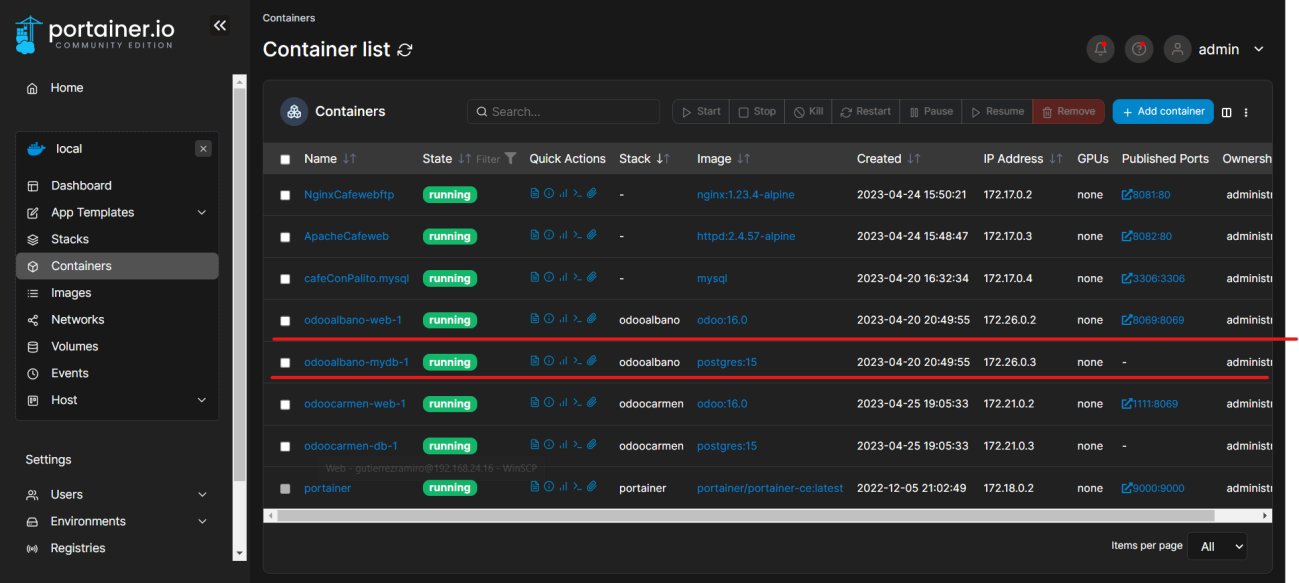
Para la gestión empresarial usaremos Odoo en un servidor Docker, para el correcto funcionamiento hay que instalar un contenedor PostgreSQL que contenga la base de datos de Odoo y un contenedor con un servidor web con Odoo relacionado con el anterior contenedor.

La estructura es la siguiente:

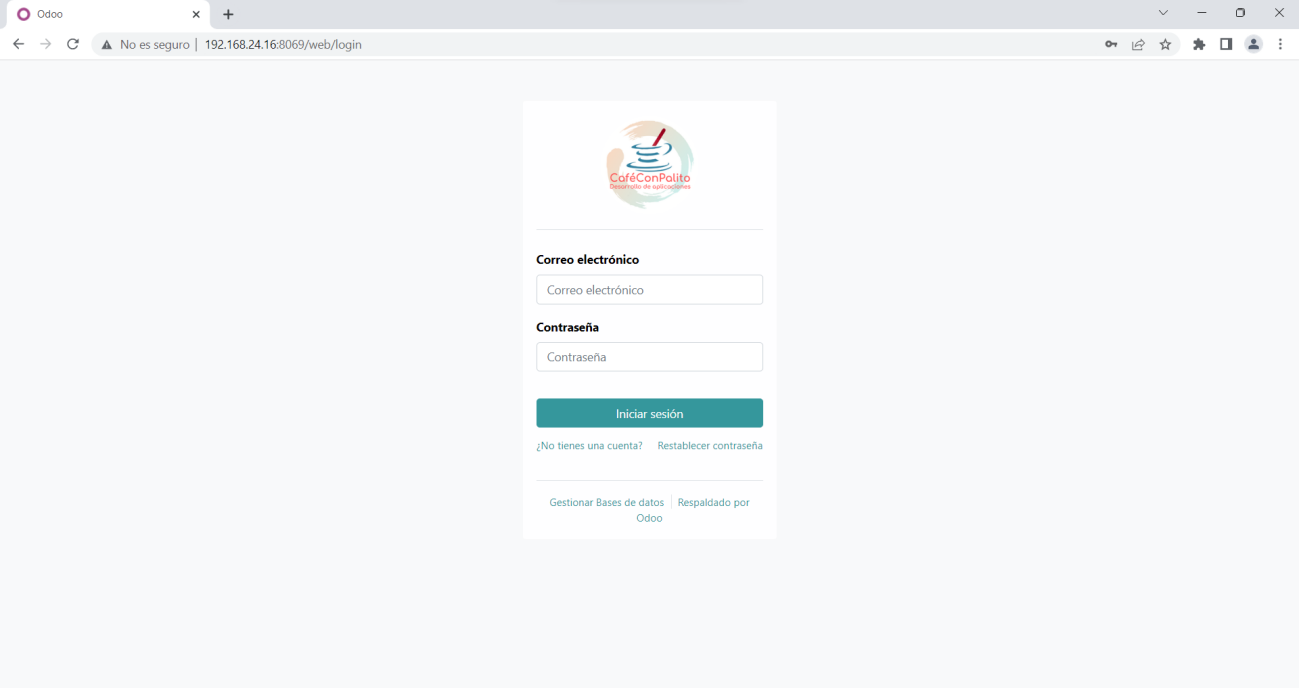
 Tras tener clara la estructura usamos el siguiente archivo docker-compose.yaml para lanzar los servicios desde Portainer para mayor comodidad.



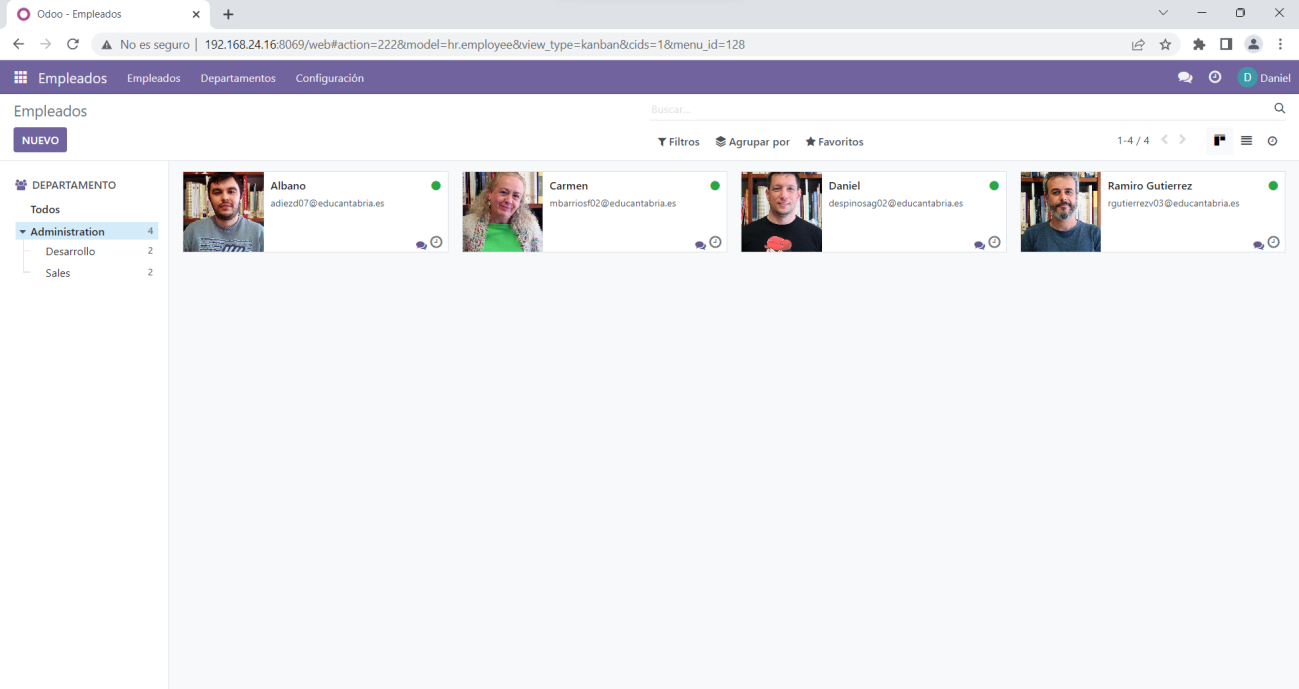
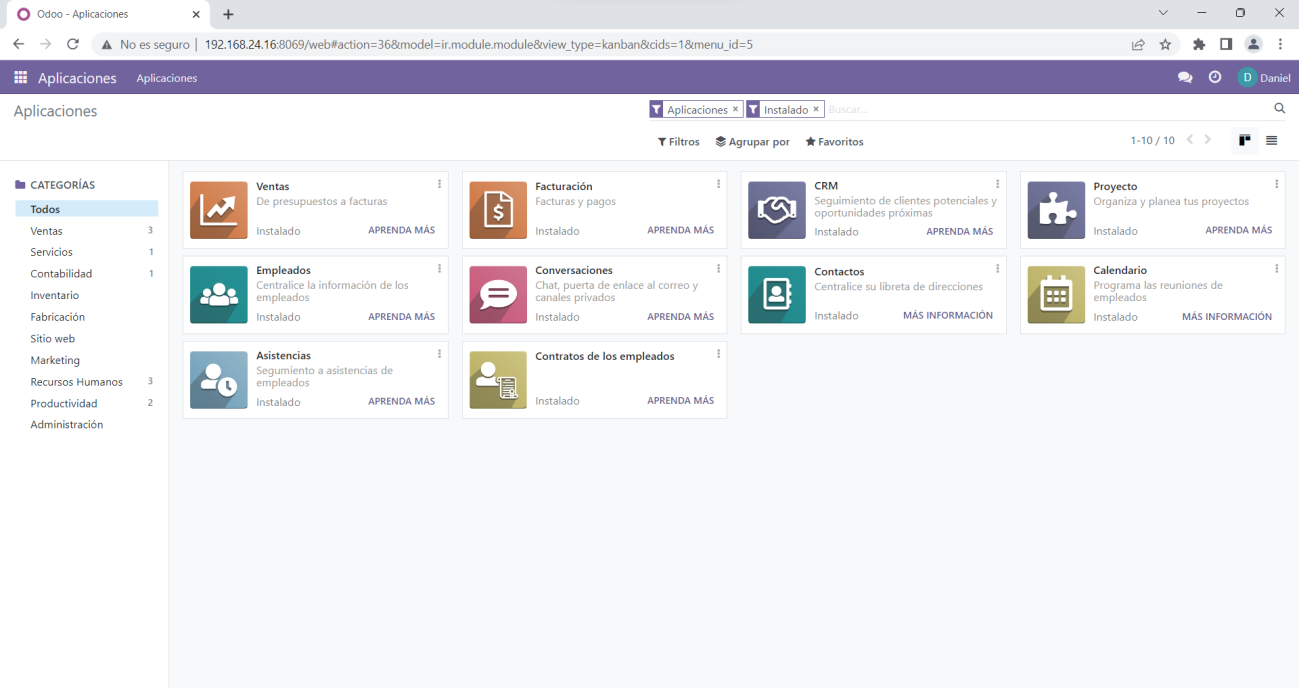
Si se ha ejecutado bien el docker-compose.yaml se puede observar en Portainer que los dos contenedores están ejecutados, sí se apaga la maquina no se corre el riesgo de perder los datos del contenedor PostgreSQL aunque estén montados sobre Docker, ya que este contenedor guarda la información sobre el SO anfitrión.

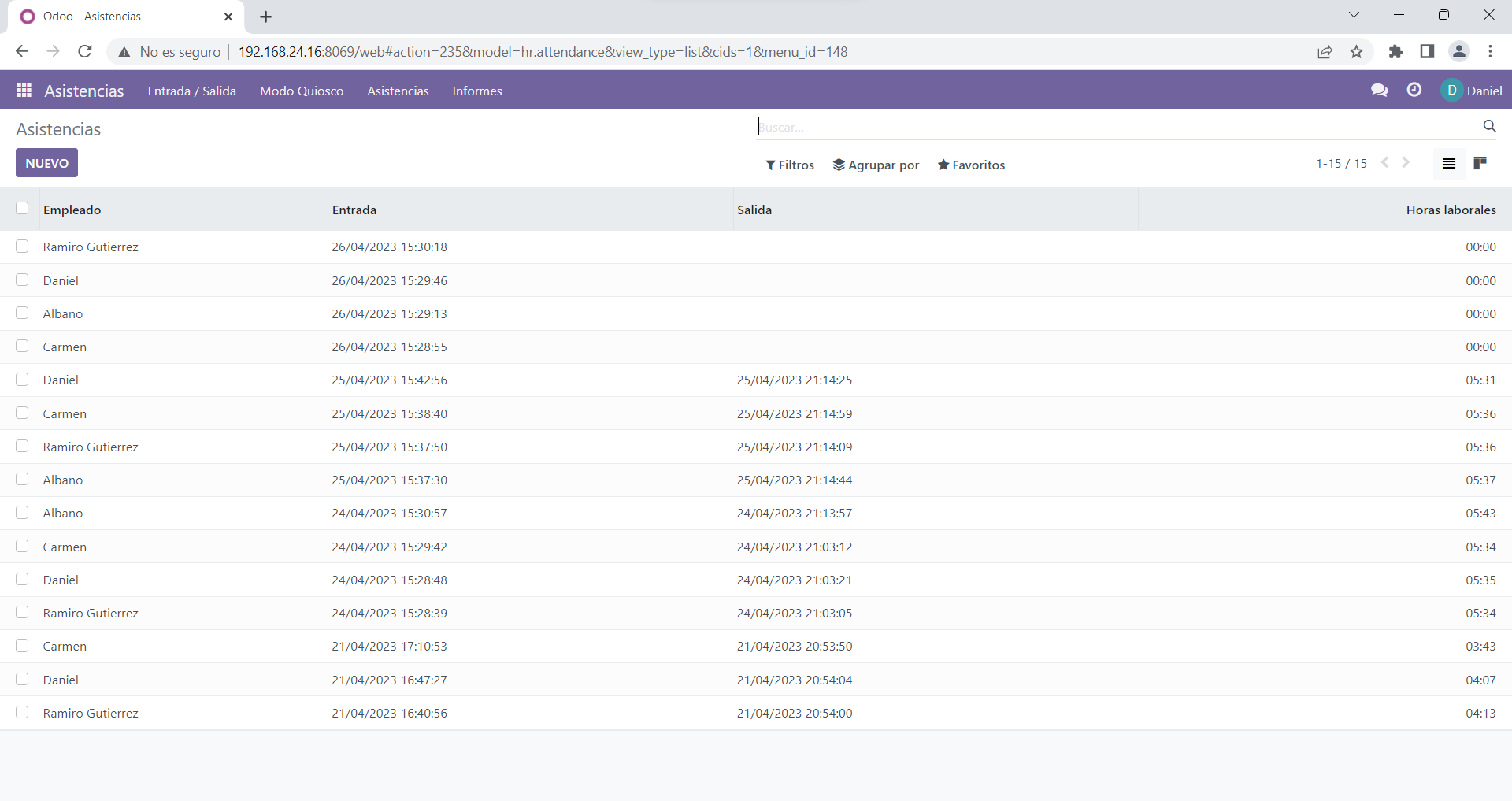


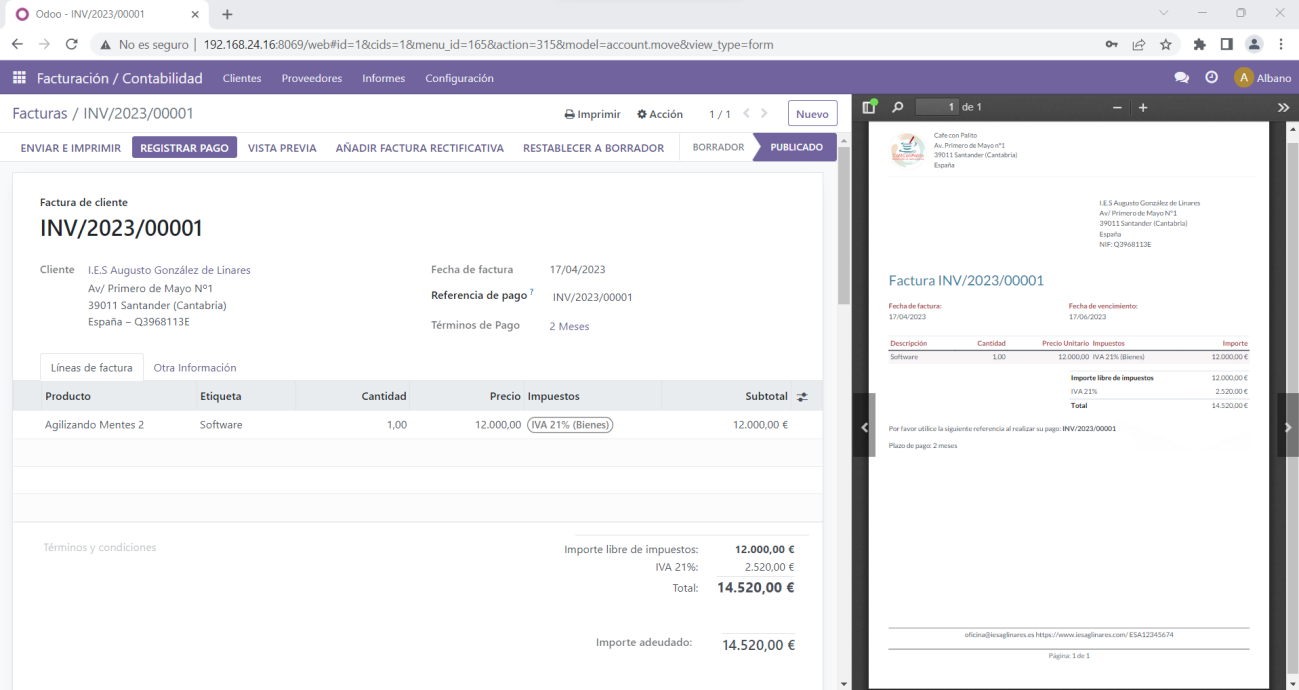
Solo queda configurar la base de datos de la empresa siguiendo los pasos que nos marca el servidor web de Odoo, si todos los pasos anteriores se han hecho correctamente nos aparece la pestaña de login al Odoo de la empresa.



Al ser una empresa pequeña hemos optado por la instalación de módulos básicos para gestión empresarial, a continuación, se muestra los módulos instalados y un par de ejemplos de los módulos que más hemos usado.







# DESARROLLO

# Hardware

Para el proyecto hemos usado dos tipos de ordenadores portátiles, el primero proporcionado por el departamento de informática del instituto I.E.S Augusto González de Linares y otro de uso personal. A continuación, se muestra las especificaciones de cada equipo.

* **Modelo**: Toshiba Dynabook Satellite Pro C50-J-11U
* **CPU:** Intel Core i5-1135G7
* **RAM:** 16 GB DDR4 3200 MHz
* **GRAFICA:** No
* **SSD:** 512 GB M.2 SATA
* **S.O:** Windows 11 Professional
* **Modelo:** **Lenovo IdeaPad Gaming 3 15ACH6**
* **CPU:** AMD Ryzen 7 5800H
* **RAM:** 16GB DDR4 3200 MHz
* **GRAFICA:** Nvidia RTX 3050Ti
* **SSD:** 512GB SSD M.2 2280 NVMe
* **S.O:** Ubuntu 23.04 Lunar Lobster

# Software

# Software Empleado

Para el proyecto hemos empleado software gratuito y a poder ser de software libre ya que ese tipo de licencias se amoldan a nuestra filosofía de empresa, a continuación, se muestra el listado del software:

* Programación:
  + **JDK (Java Development Kit)**: Versión 20 y Versión 11.
  + **IDE:** Apache Netbeans: Versión 17 y Versión 13.
* Bases de datos:
  + **Sistema gestor de bases de datos:** MySQL
  + **Entorno gráfico:** MySQL Workbench
* Sistemas Informáticos:
  + **Software de Virtualización:** Oracle VM Virtual Box
  + **Sistema de despliegue:** Docker
  + **Servidor Web:** Nginx
* Lenguaje de Marcas:
  + **Sistema de Gestión Empresarial:** Odoo
  + **Editor de código:** Visual Studio Code
* Entornos de desarrollo
  + **Control de Versiones:** Git /Github
  + **Entorno Grafico:** GitKraken/SourceTree

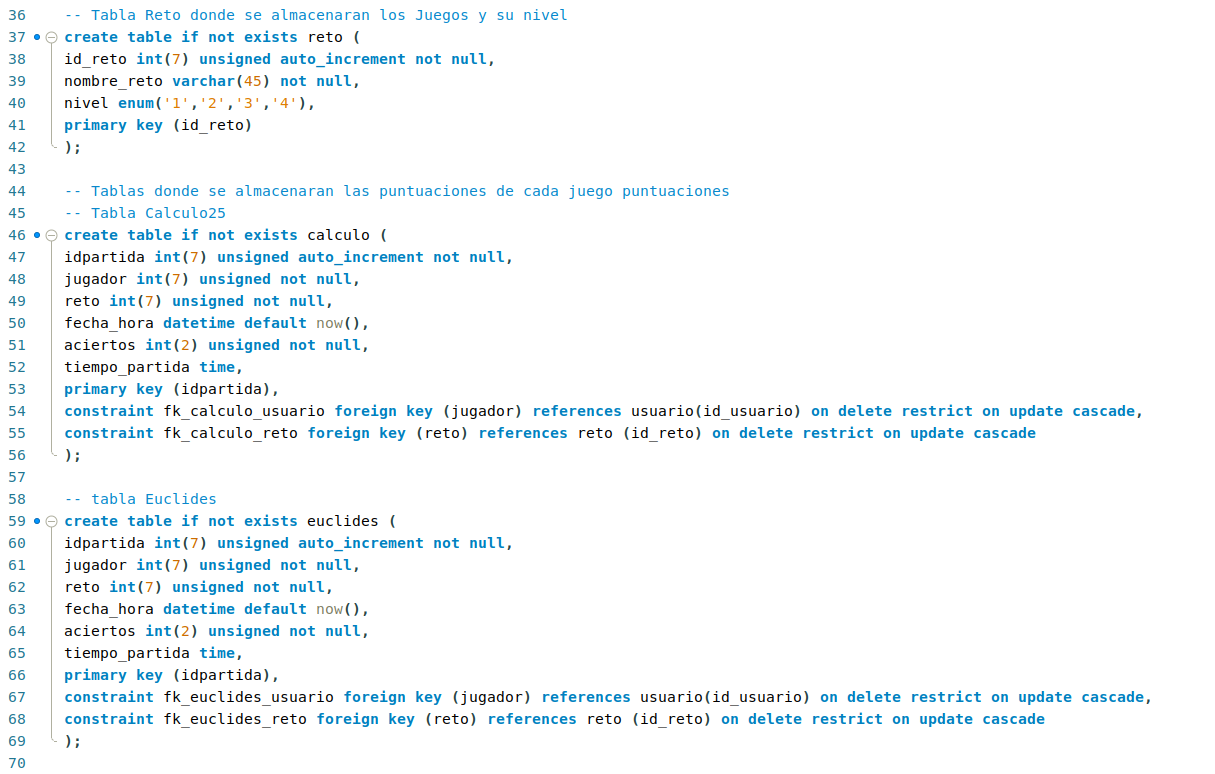
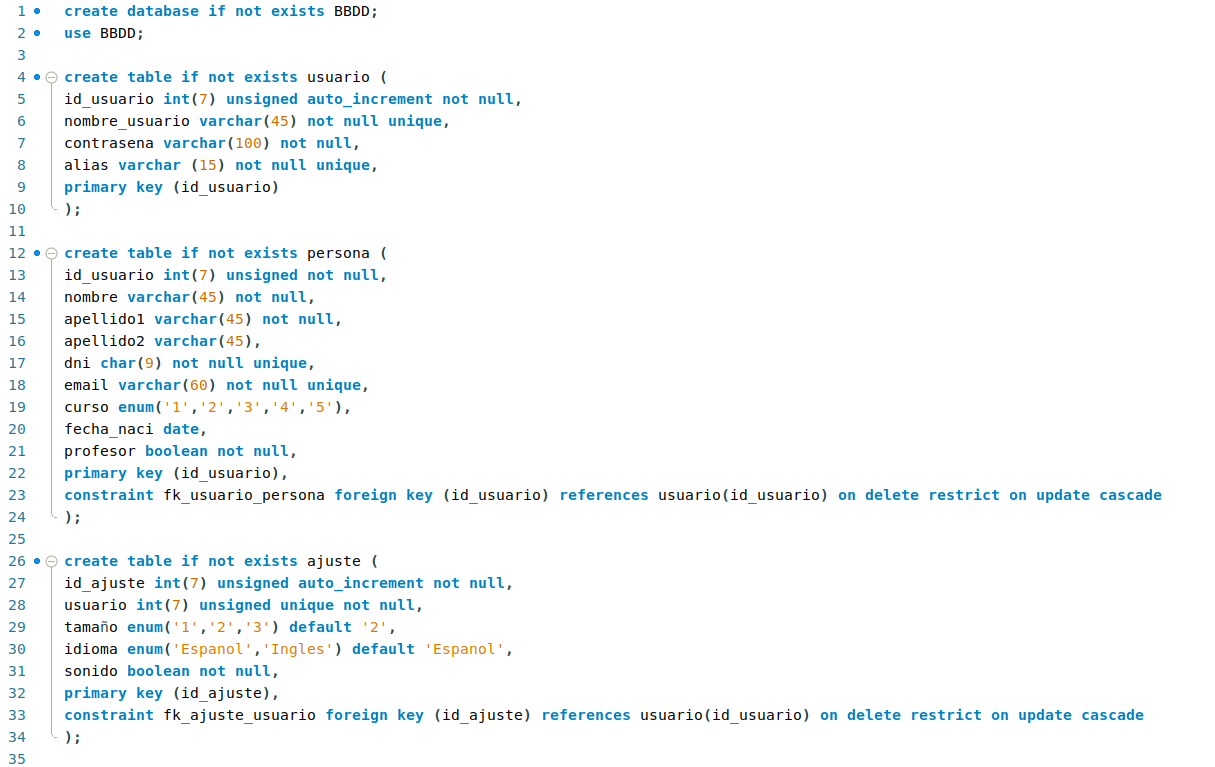
# Software Alternativo

Dejamos software alternativo para si nosotros u otra persona/empresa desea continuar el desarrollo pero con otro software pueda hacerlo.

* Programación:
  + **IDE:** Eclipse o IntelliJ Idea
* Bases de datos:
  + **Sistema gestor de bases de datos:** MariaDB
  + **Entorno gráfico:** PHPmyAdmin
* Sistemas Informáticos:
  + **Software de Virtualización:** Vmware
  + **Sistema de despliegue:** Kubernetes
  + **Servidor Web:** Apache
* Lenguaje de Marcas:
  + **Sistema de Gestión Empresarial:** Netsuite
* Entornos de desarrollo
  + **Control de Versiones:** Bit /Bitbucket

# Diseño físico de la base de datos

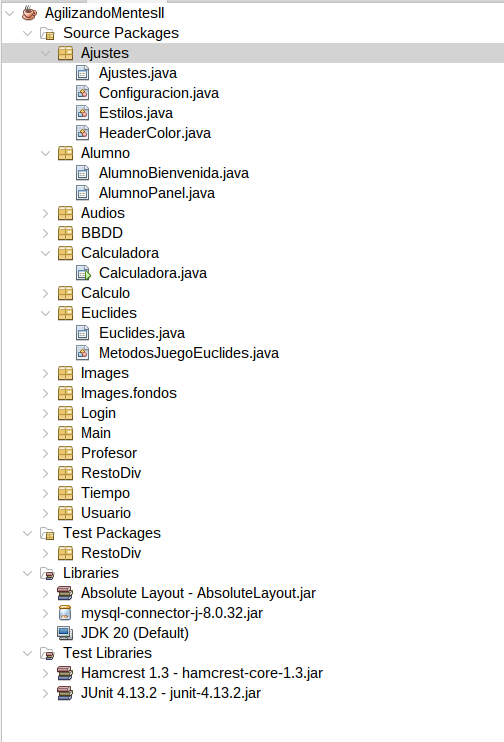
Partiendo del diseño lógico realizado en el apartado 2.1 (Diseño lógico de la base de datos) hemos realizado el siguiente diseño físico.





# Código java

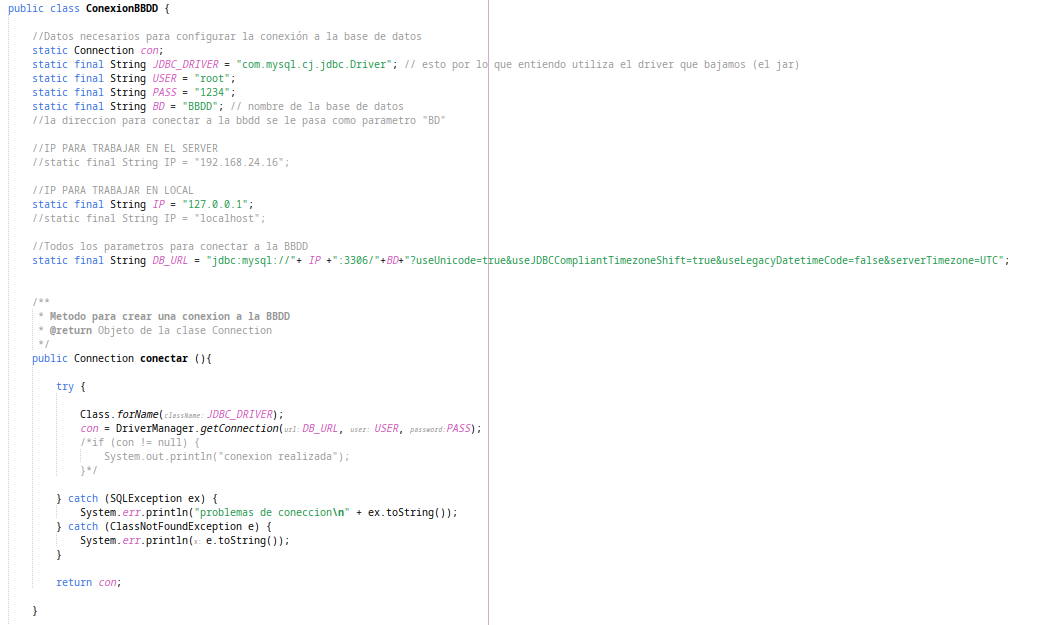
El código java se ha estructurado de forma modular en diferentes paquetes para aumentar la eficiencia y claridad del código y a su vez hemos desarrollado la parte visual de la app como el diseño original porque guarda la modularidad que hemos buscado en todos los apartados del proyecto.

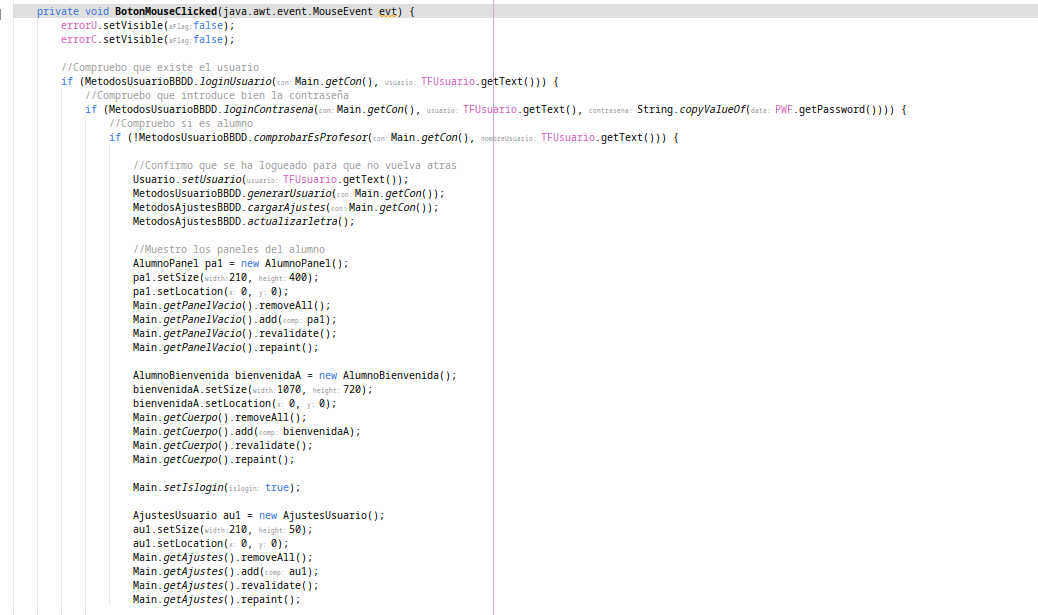


Para poder realizar la interfaz con diseño minimalista importamos la librería Absolute Layout, y para poder realizar la conexión a una base de datos MySQL importamos un conector.

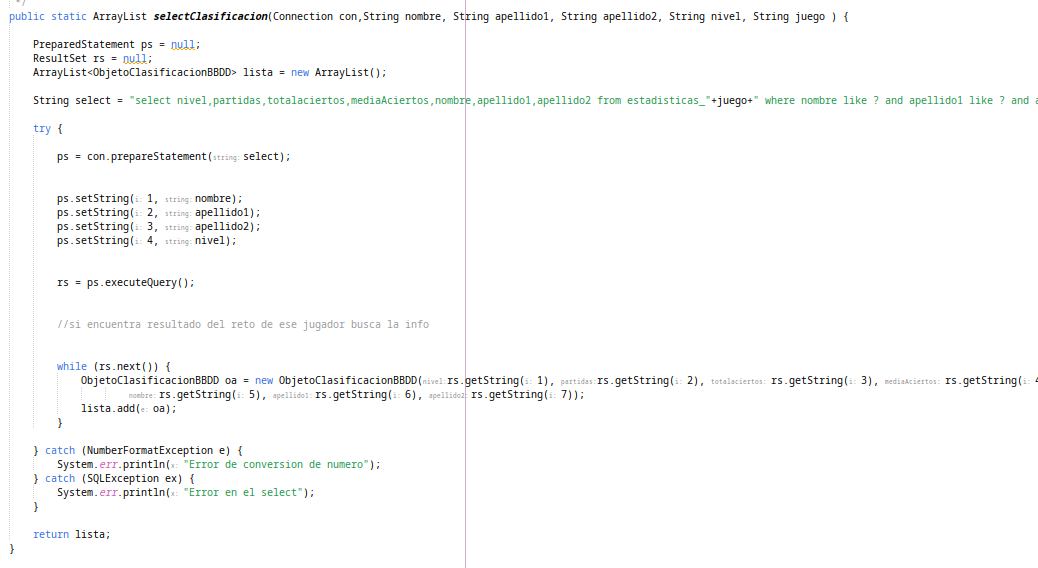
A continuación, se muestran unas capturas de pantalla de las partes más importantes del código.

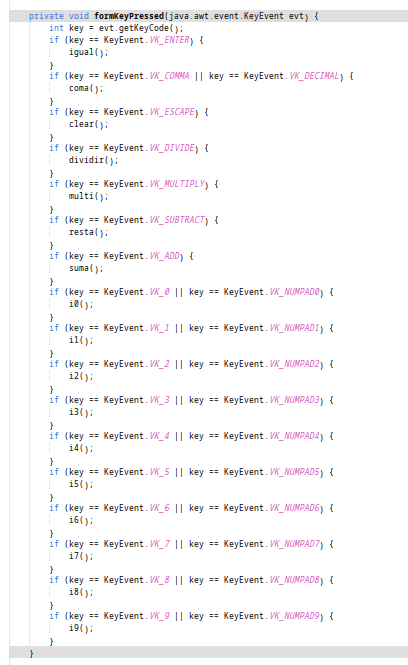
**Clase para la conexión a la Base de Datos**

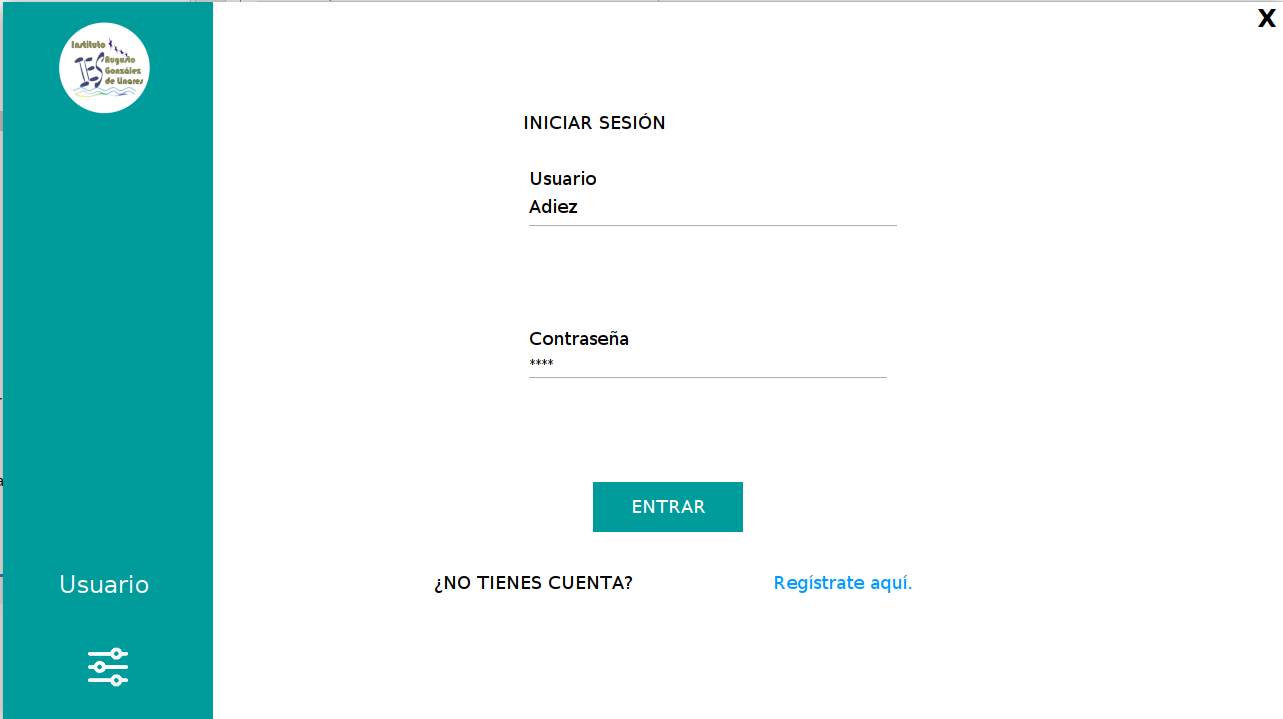
**F****unción para iniciar sesión si el usuario ha introducido bien sus datos**

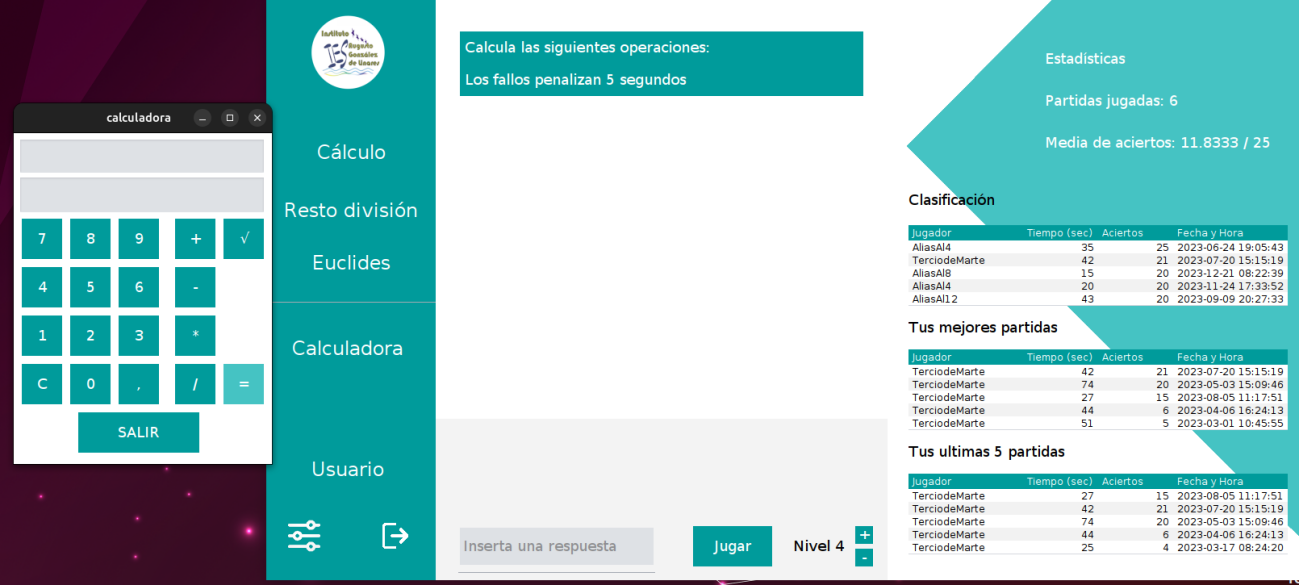


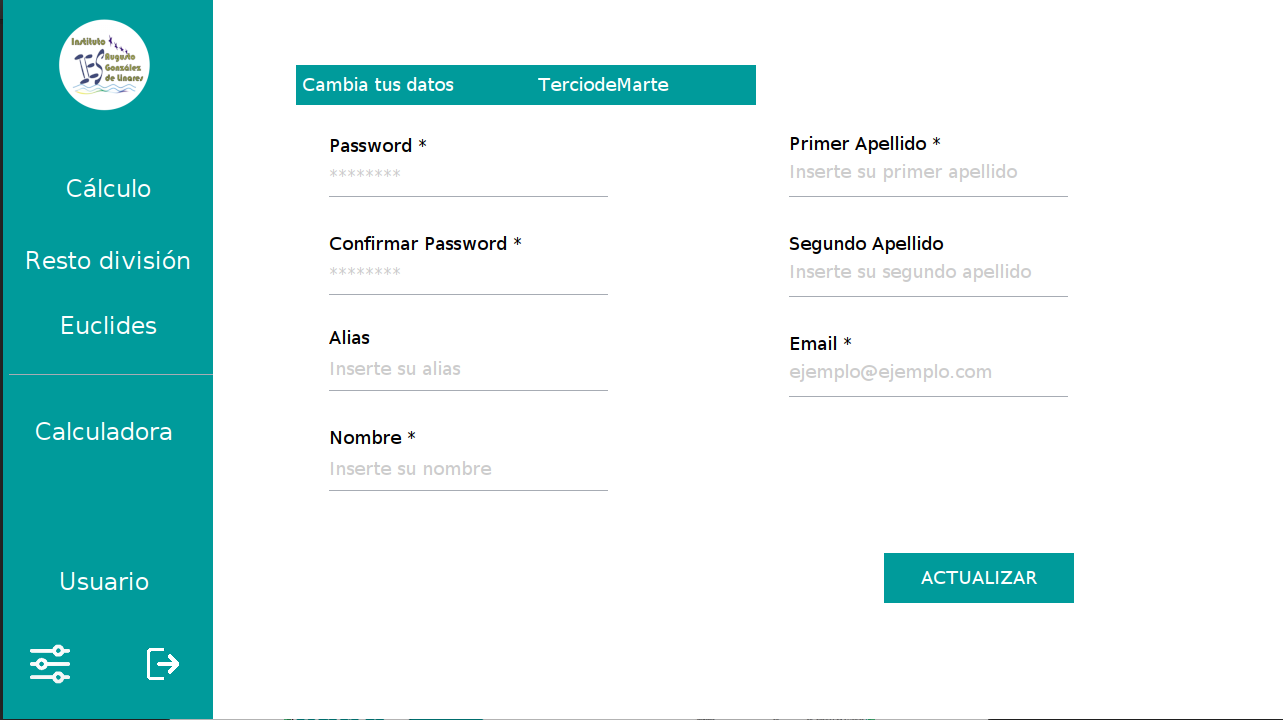
**Método para actualizar la tabla de clasificaciones de los juegos**

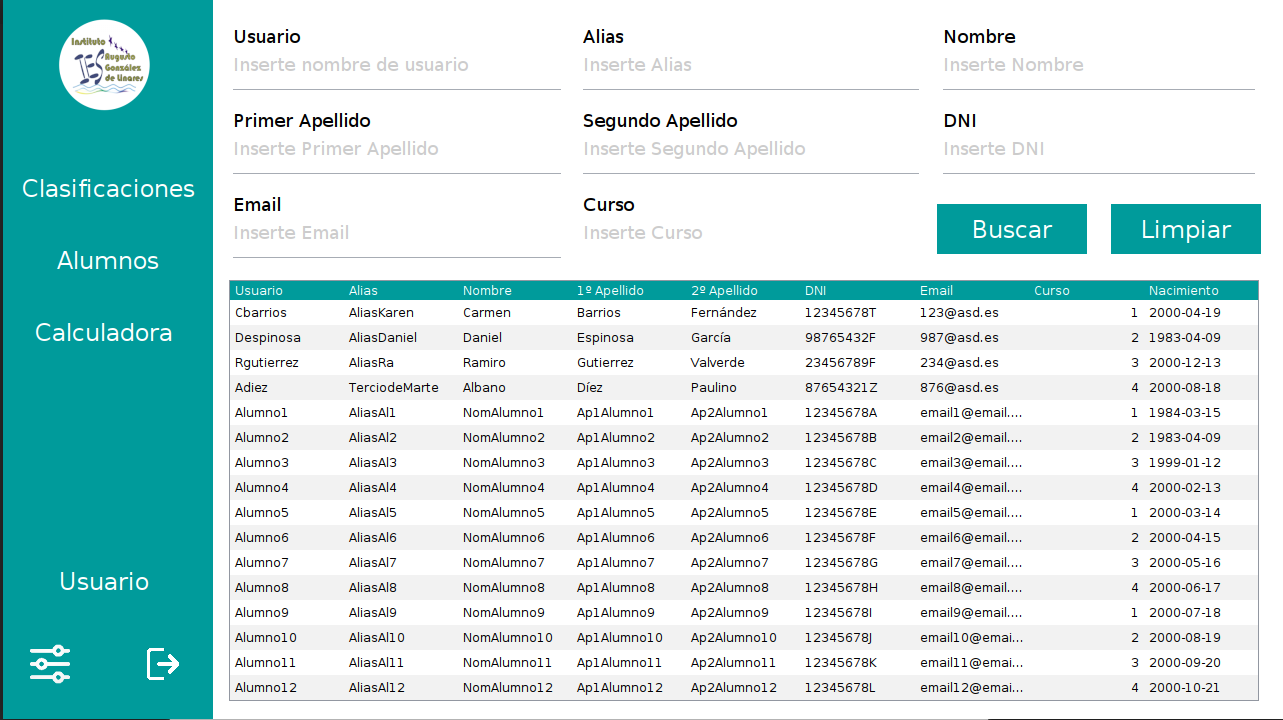
**Comprobar que se ha pulsado una tecla del teclado en la calculadora**



 Ahora se muestran unas pocas capturas para mostrar el diseño de la interfaz de usuario que hemos creado con la base del diseño que realizamos en el apartado 2.

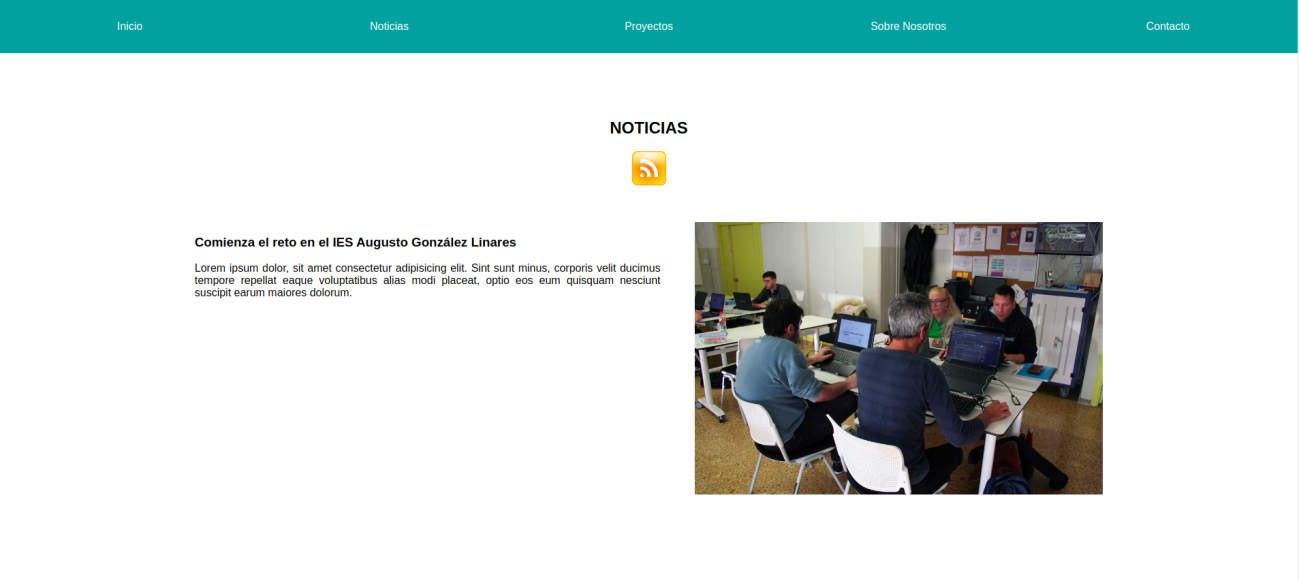


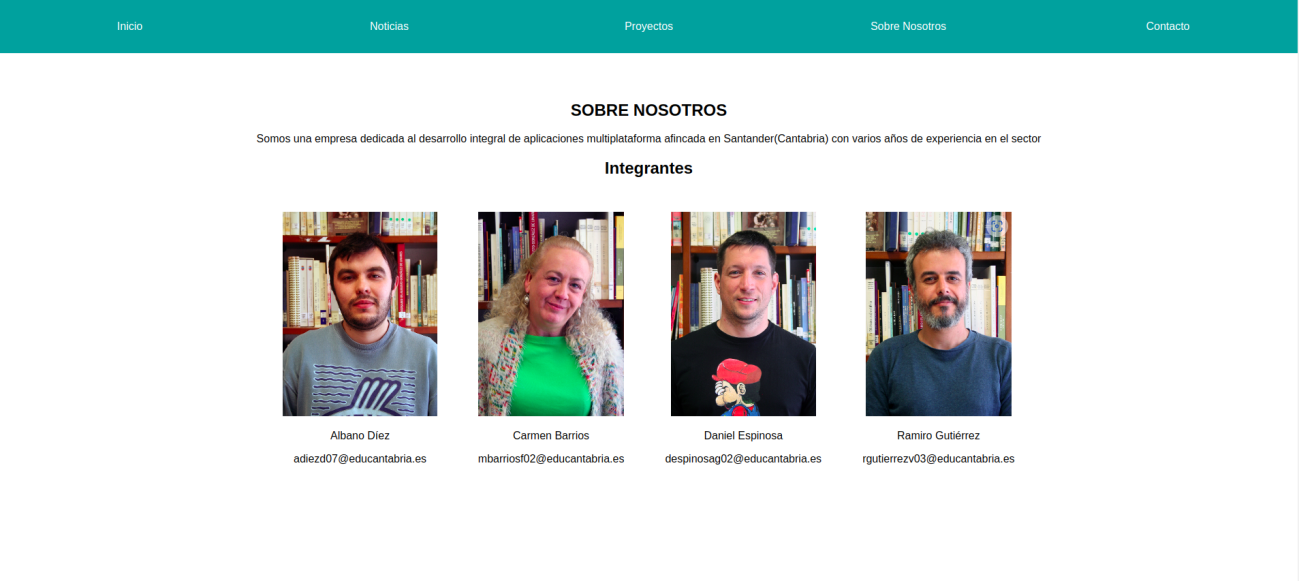




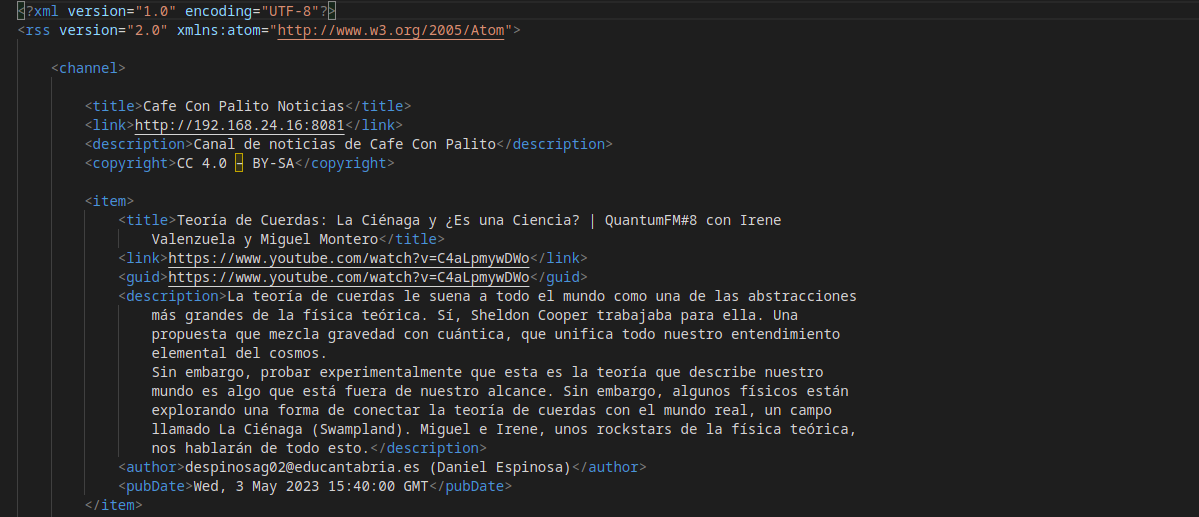
# Código HTML+ CSS+ RSS

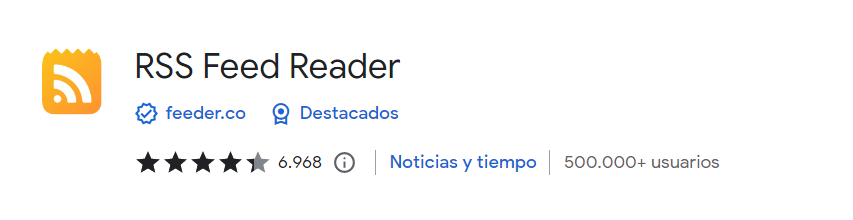
Ya que las especificaciones del reto se nos pidió crear una web de empresa para vender nuestro producto creamos un código HTML y CSS simple que en un futuro se podrá mejorar incluyendo funciones en JS o dar un estilo a página más visual y llamativo.



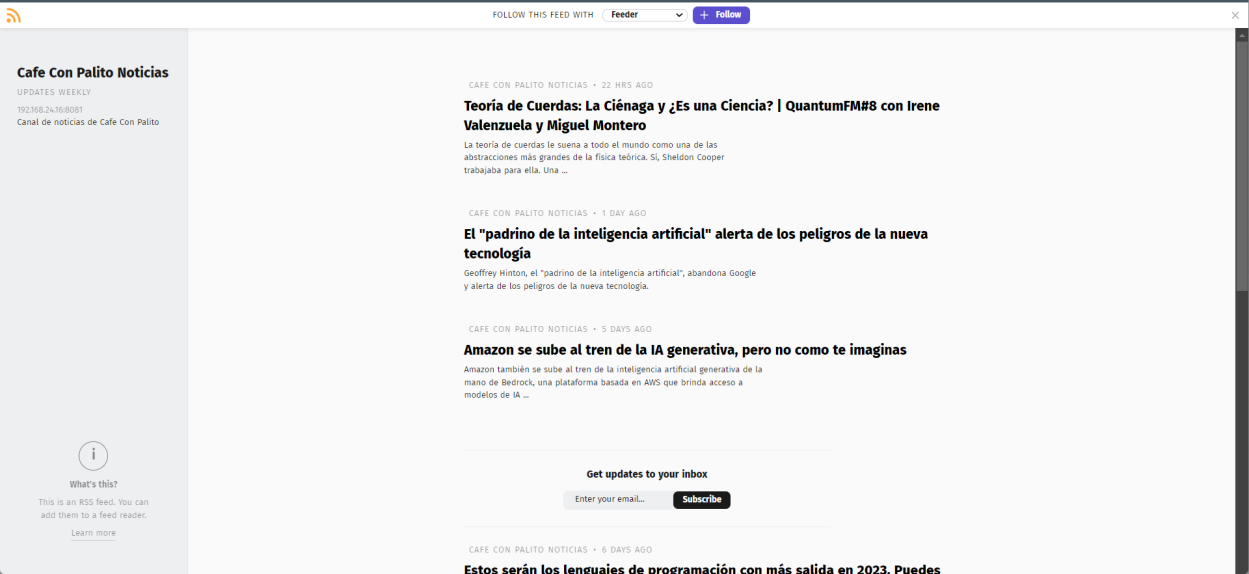


Para poder publicar una noticia en RSS hay que generar un fichero XML como el siguiente:

 Dicho fichero XML debe de ser incluido en el directorio de la página web para que cualquier usuario pueda acceder a él, pero se necesita un lector RSS en el navegador del usuario ya que si no solo le saldría el texto plano de la parte superior. Para navegadores basados en chromium (Chrome, edge, opera) recomendamos el siguiente lector



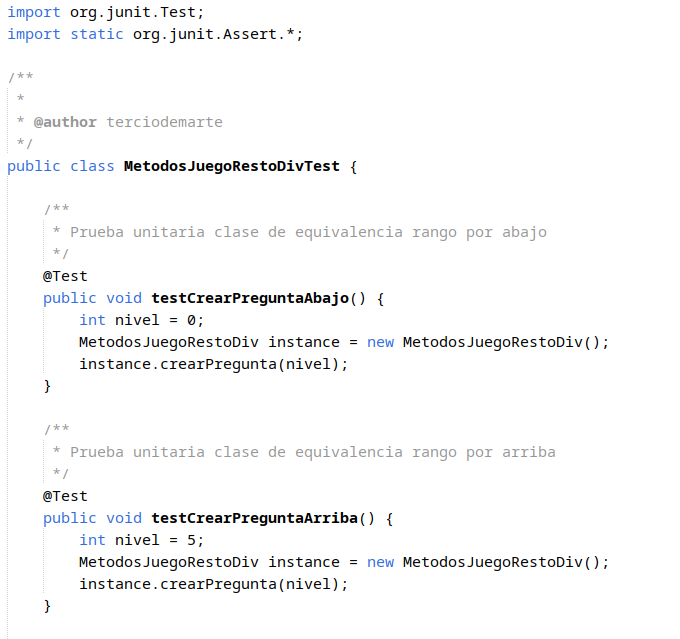
Si el usuario ya posee un lector de RSS, vera las noticias de la siguiente forma.



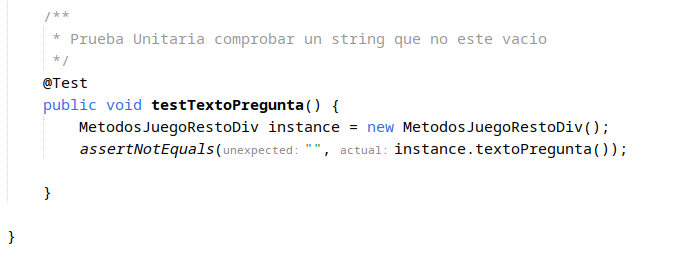
# PRUEBAS

# Pruebas unitarias sobre código java (JUnit5)

A lo largo del desarrollado hemos realizado pruebas unitarias sobre los métodos públicos de las clases del proyecto para comprobar el correcto funcionamiento de los métodos, a continuación, se muestra diferentes tipos de pruebas unitarias (Boolean, rango y null).







# DESPLIEGUE

# Despliegue Web

Como ya se ha dicho 1.2 (Planteamiento de la solución) la web se ha desplegado en un contenedor de Docker con el servidor web NGNIX, en la que va incluido el fichero XML del RSS, el ejecutable de Windows de la aplicación Agilizando Mentes II para que los usuarios se lo puedan descargar el manual de usuario de la aplicación, y el javadoc del proyecto para que otro/s desarrollador/es puedan continuar el proyecto y que sepan que función realizan las clases y sus métodos.

# Despliegue app

Hemos decidido que la app se portable ya que no tenemos ninguna dependencia externa a java, solo es necesario tener instalado el JRE, el que ya viene instalado en las distribuciones más populares de GNU/Linux para entornos de escritorio (Ubuntu, Manjaro, Linux Mint) y en las distribuciones de MacOs modernas.

Para el entorno Windows el usuario debe descargar el JRE porque no viene instalado, pero en el manual de usuario hemos advertido de dicho problema y hemos planteado una solución.

# PLANTEAMIENTO FUTURO

# Ideas para continuar el desarrollo

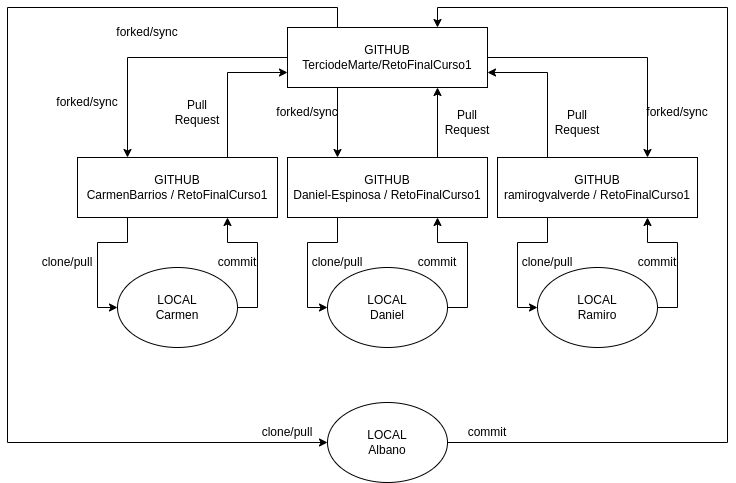
Aparte de las ideas a futuro que planteamos en el apartado 1.2 (Planteamiento de la solución) queremos dejar otra serie de ideas para mejorar los sistemas externos a la aplicación de java.

* Implementar el almacenado del proyecto en AWS
* Implementar la Base de datos en el servidor del instituto o en la nube.
* Dar un aspecto más visual a la web de la empresa.
* Continuar publicando noticias para dar visibilidad a la empresa.

# ANEXOS

# Git

A lo largo del proyecto hemos usado todos los miembros del equipo la herramienta de Git para el control de versiones y Github para poder tener un repositorio en la nube para una mejor sincronización de todos los repositorios de los integrantes del equipo. Para ello hemos seguido la siguiente estructura de repositorios.

Enlace al repositorio principal: <https://github.com/TerciodeMarte/RetoFinalCurso1>

# Opinión personal de los integrantes del grupo

# ADJUNTOS

# Documentación Bases de datos

# Documentación Sistemas Informáticos

# Documentación FOL

# Bibliografía