NEXO

CFGS DAW

CURSO 2022/2023

Raúl Pérez Esteban

COLEGIO CALASANZ SALAMANCA

Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

1. Introducción

Nexo es una aplicación web que permite la búsqueda de jugadores online para los principales juegos competitivos a través de la creación de grupos o uniéndose a grupos ya existentes.

Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

2. Descripción de la aplicación

La aplicación consiste en una plataforma web en la que jugadores de distintos títulos online pueden encontrar personas con las que jugar o participar en actividades junto a ellas.

En la página principal aparecerá una lista de títulos, cada uno con su portada. Haciendo click sobre estos nos redireccionará a la página en la cual se muestran los grupos existentes de ese juego y la opción de crear un grupo nuevo.

Cada usuario puede crear nuevos grupos en los que se indique el número de jugadores necesarios, el modo de juego o actividad, y otros requisitos que pueden variar dependiendo del juego en cuestión. Dentro de cada grupo aparecerá la lista de jugadores, que variará su número de participantes dependiendo del juego y las preferencias del líder de grupo.

Si un jugador está interesado en unirse al grupo simplemente tendrá que escribir su nombre de cuenta del juego en cuestión y opcionalmente su tag de Discord, para posteriormente añadirse mediante el botón “Unirse”.

Los jugadores podrán buscar fácilmente a los participantes del grupo dentro de cada juego mediante sus respectivos nombres de cuenta. También podrán utilizar la plataforma de Discord para comunicarse dentro de esta.

Una vez el grupo esté completo con el número de jugadores solicitado este se cierra e impide su modificación.

Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

3. Tecnologías escogidas y justificación

Tecnologías frontend

HTML, CSS, JavaScript: son las tecnologías básicas sobre las que funciona la web, sin las cuales sería imposible el desarrollo.

- HTML: define la estructura de la página, los elementos que se muestran en pantalla y la interfaz de usuario.
- CSS: se encarga de configurar el aspecto visual de la aplicación. Controla elementos básicos, cómo los colores, el tamaño de fuente, bordes, etc. Y otros apartados más avanzados como la disposición de los elementos en pantalla.
- JavaScript: se encarga de la programación de la funcionalidad de la web. Sin este no sería posible la utilización del framework Vue o la conexión con el backend.

Vue.js: El framework Vue permite ejecutar la aplicación dentro de una misma página, ofreciendo un funcionamiento fluido y sin necesidad de recargar las páginas de la web al acceder a diferentes apartados de la aplicación. Los recursos cargan de forma dinámica mejorando rendimiento y la experiencia de usuario.

También facilita la administración de recursos al separar en vistas las partes de la aplicación, en este caso, utilizando una vista por juego.

Otra ventaja que ofrece es la capacidad de instalar plugins para ampliar las funcionalidades de la página.

Por otra parte, gracias al sistema de vistas podemos reutilizar elementos de la página. La cabecera y el pie de página de la web son únicos y no es necesario añadirlos de nuevo al código HTML de cada página.

De cara a un mantenimiento y desarrollo futuro, por ejemplo, añadiendo nuevos títulos a la lista de juegos disponibles, podríamos crear una nueva vista a partir de una ya existente, configurando las características propias del nuevo título (nombre del juego, imagen de la cabecera, conexión con la API, etc.).

Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

Bootstrap: el framework frontend Bootstrap facilita el diseño de la aplicación ofreciendo multitud de herramientas y componentes predefinidos. Una de las mayores ventajas que ofrece es la utilización de un contenedor dentro del cual podemos alinear los elementos de la web en columnas.

Gracias a ello la aplicación se adapta de manera responsiva a las diferentes resoluciones, que pueden variar según el dispositivo desde el que se acceda a la web.

Otra ventaja significativa de Bootstrap son las clases CSS predefinidas para multitud de elementos HTML diferentes. Por ejemplo, podemos ajustar los márgenes de diferentes elementos, aplicarles un borde o una sombra, sin necesidad de utilizar estilos en un archivo CSS.

Tecnologías backend

Java: es el lenguaje de programación que sirve como base de todo el backend del proyecto.

Springboot: es un framework backend que permite facilitar el desarrollo de la aplicación.

Gracias a su sistema de dependencias podemos añadir las funcionalidades que necesitamos para el proyecto mediante el archivo “pom.xml”.

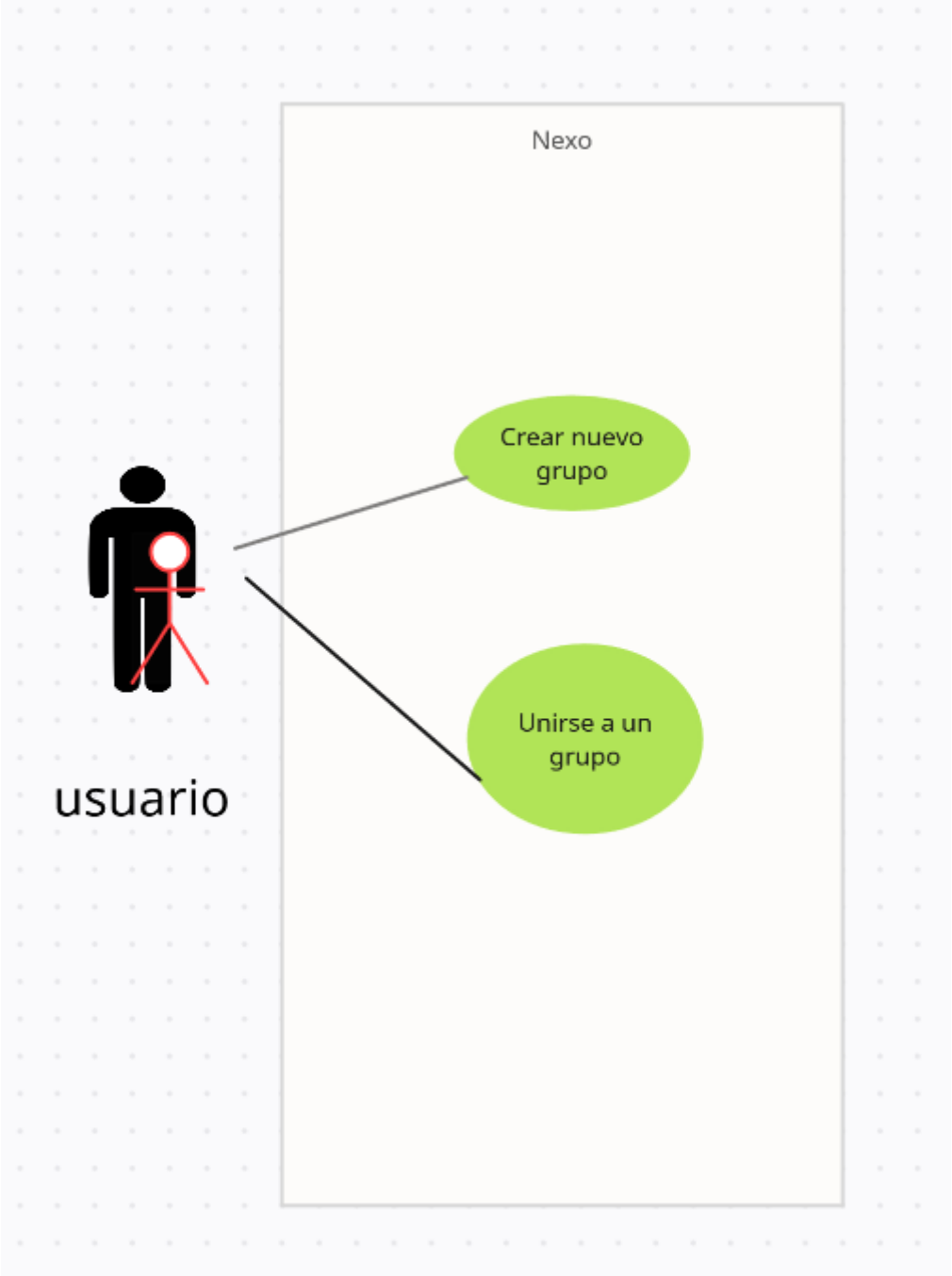
Tecnologías para la gestión de BBDD

MySQL: se encarga de la gestión de la base de datos que utiliza la aplicación. Almacena los datos de los grupos en tablas.

Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

4. Diseño de la aplicación

4.1. Diagramas y definición de casos de uso



Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

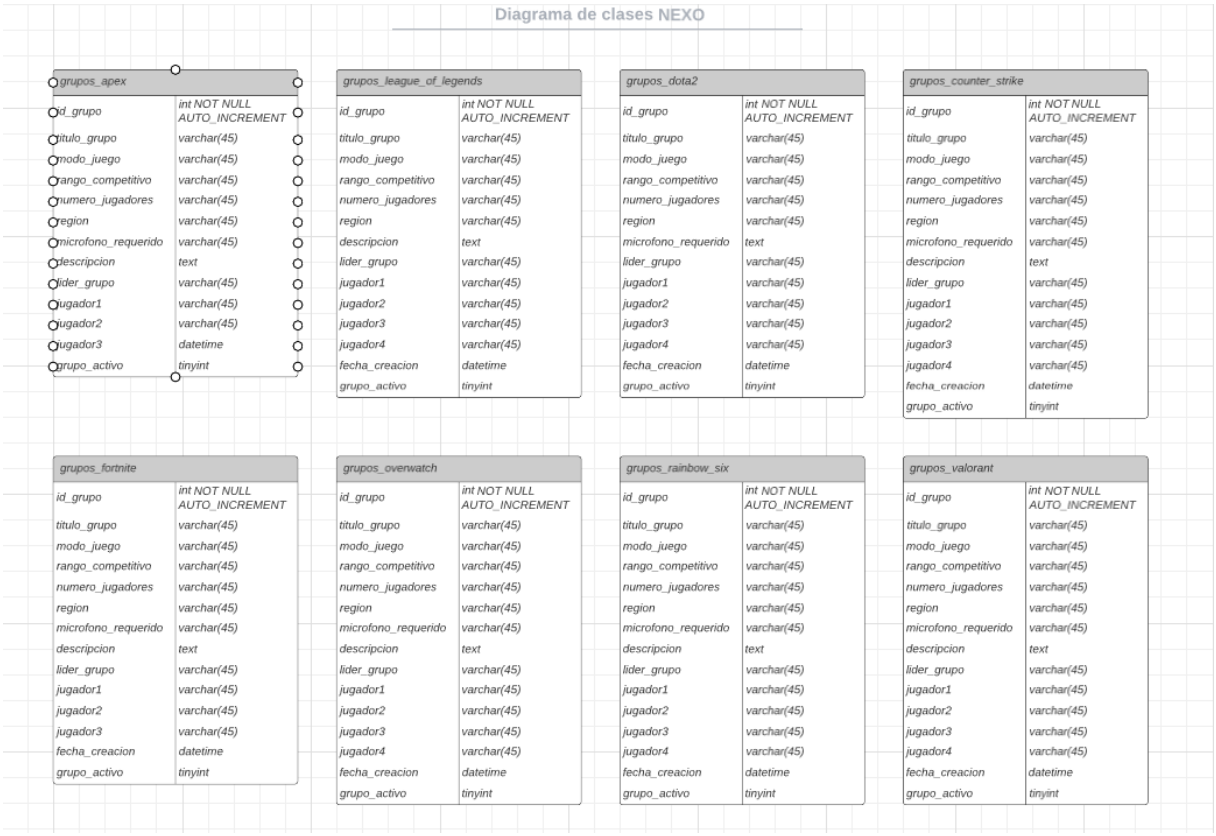
Definición de Casos de Uso:

1. **Caso de Uso: Crear Nuevo Grupo**
    - **Descripción:** El usuario crea un grupo de jugadores nuevo.
    - **Actores:** Usuario
    - **Flujo Principal:**
        1. El usuario introduce su nombre de cuenta del juego en cuestión.
        2. El usuario introduce su tag de Discord.
        3. El usuario introduce el título del grupo, el modo de juego, el número de jugadores y la región del servidor.
        4. El usuario añade una descripción al grupo.
        5. El usuario crea el grupo.
    - **Flujos Alternativos:**
        1. El usuario no ha introducido su nombre de cuenta de juego. Se muestra un mensaje de error e impide la creación del grupo.
        2. El usuario no ha introducido ningún campo obligatorio (título del grupo, el modo de juego, el número de jugadores, región del servidor). Se muestra un mensaje de error e impide crear el grupo.

2. **Caso de Uso: Unirse a un Grupo**
    - **Descripción:** El usuario se une a un grupo existente de la lista de grupos.
    - **Actores:** Usuario
    - **Flujo Principal:**
        1. El usuario comprueba que el grupo permite la incorporación de nuevos jugadores.
        2. El usuario introduce su nombre de cuenta del juego y opcionalmente su tag de Discord.
        3. El usuario hace clic en el botón "Unirse".
        4. El usuario se une al grupo y se muestra su nombre de cuenta de juego dentro de este.
    - **Flujos Alternativos:**
        1. El usuario no ha introducido su nombre de cuenta. Se muestra un mensaje de error e impide la incorporación.

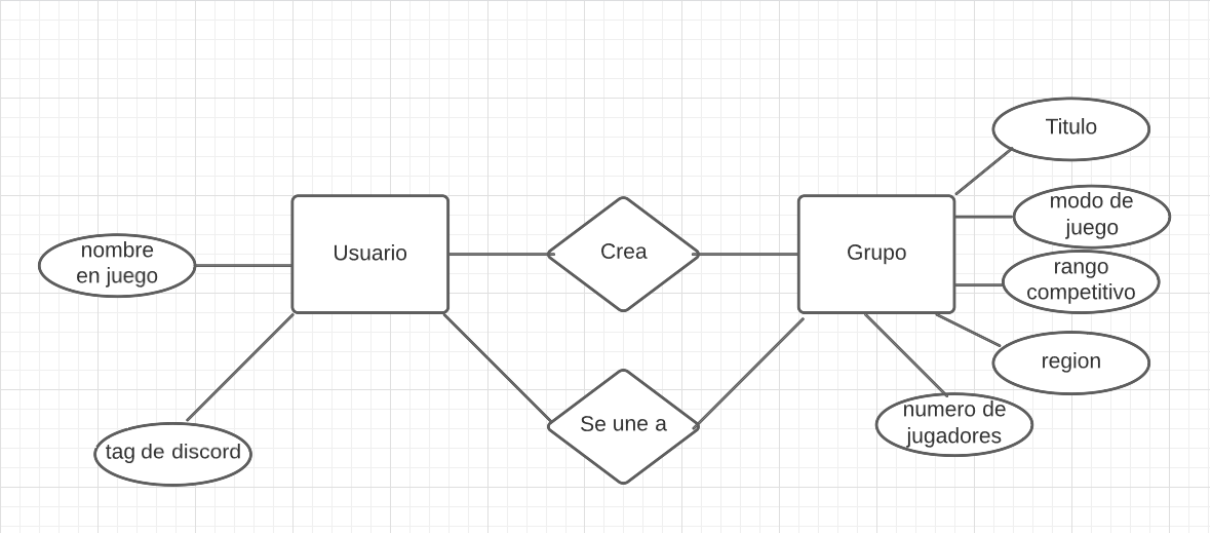
Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

4.2. Diagramas de clases



Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

4.3. Modelo entidad-relación



Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

5. Arquitectura de la aplicación

5.1. Estructura del proyecto

Carpeta bbdd: Aquí se encuentra el archivo principal de configuración y otros archivos auxiliares.

- Nexo.sql: archivo sql para crear la base de datos dentro del contenedor.

Carpeta NexoAPI: carpeta contenedora del backend de la aplicación

- Src/main/java/com/nexo/controllers: contiene los controladores con los endpoints correspondientes.
- Src/main/java/com/nexo/entities: contiene las entidades de cada tabla
- Src/main/java/com/nexo/repositories: contiene los repositorios que se encargan de realizar las consultas sql.
- Src/main/java/com/nexo/services: contiene los métodos para realizar las consultas solicitadas.
- Src/main/java/nexo/ NexoApiApplication.java: es el archivo de configuración de la aplicación Springboot.
- Pom.xml: define las dependencias utilizadas en el proyexto

Carpeta proyecto-final-daw: carpeta contenedora del frontend de la aplicación

- Public: contiene el archivo index.html y el logo de la página.
- Src/assets/css: contiene el css común para los contenedores de grupos
- Src/assets/img: contiene las carpetas con las imágenes utilizadas en la aplicación
- Src/router: contiene el archivo index.js en el cual se configuran las vistas de la aplicación
- Src/views: contiene las vistas de la aplicación.
- App.vue: es la vista principal contenedora de las demás. En ella se define el router-view
- Main.js: es el archivo que importa los plugins instalados

Paseo de Canalejas, 139-159
37001 Salamanca
Tel.: 923 26 79 61 | Fax: 923 27 04 54
www.calasanzsalamanca.es
CENTRO PRIVADO CONCERTADO

5.2. Librerías externas utilizadas

Frontend

Plugins de Vue.js

- Vue Router: gracias a este plugin es posible separar el sitio web en vistas definidas en el archivo index.js de la carpeta router.
- Bootstrap: permite la utilización del framework dentro de la aplicación de manera nativa. Así podemos utilizar las clases propias de este sistema sin tener que importarlo en cada página.
- Axios: posibilita la administración de solicitudes HTTP dentro de la aplicación. En este proyecto se utiliza para crear las conexiones con la base de datos a través de la API Springboot.

Backend

Dependencias de Springboot:

- spring-boot-starter-data-jpa: proporciona las bibliotecas necesarias para utilizar JPA y Hibernate, las cuales permiten la interacción con la BBDD.
- spring-boot-starter-web: agrega las bibliotecas necesarias para el desarrollo de aplicaciones web (controladores, servicios).
- spring-boot-devtools: añade algunas herramientas para el desarrollo, como la recarga automática de la aplicación al guardar cambios.
- mysql-connector-j: permite la comunicación y conexión con la base de datos MySQL.
- spring-boot-starter-test: añade las bibliotecas necesarias para la depuración y ejecución de test de la aplicación.

6. Manual de despliegue

Para desplegar la aplicación es necesario importar las siguientes imágenes:

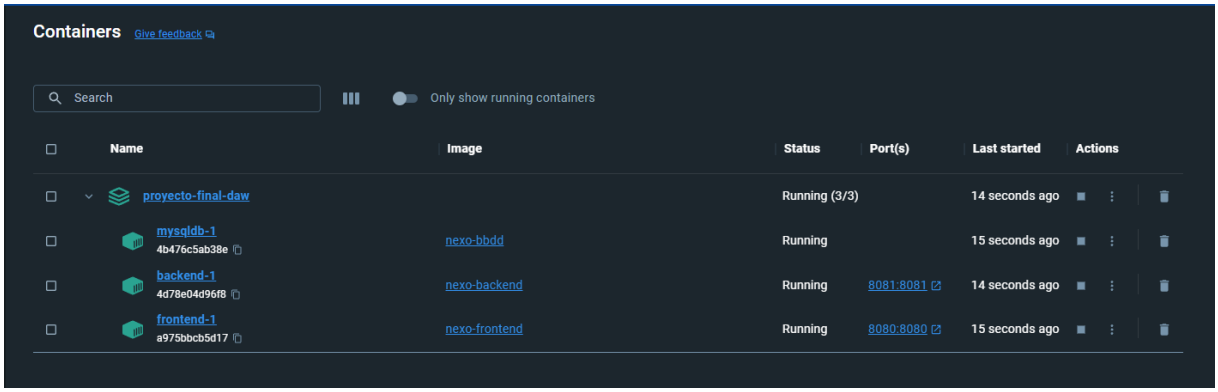
- BBDD: docker push raulpe/nexo-bbdd:latest
- Backend: docker push raulpe/nexo-backend:latest

- Frontend: docker push raulpe/nexo-frontend:latest

Una vez importadas las imágenes se ejecuta el archivo docker-compose.yml de la raíz del proyecto mediante el comando:

- docker-compose up

Si la ejecución es correcta se creará un contenedor ejecutando tres contenedores:



Para acceder a la aplicación se introduce en el navegador web la siguiente URL:

- http://localhost:8080/nexo/#/