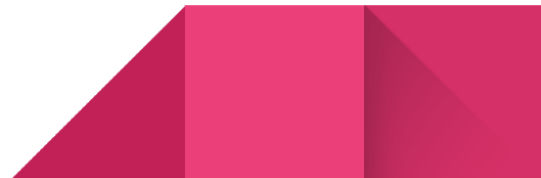


DOCUMENTACIÓN



Carlos Correa Falcón
Proyecto Integrado Ciclo Superior DAW
Departamento de Informática – IES Julio Verne



Índice

- 1. Introducción
- 2. Estudio de Viabilidad
 - 2.1. Alcance del proyecto
 - 2.2. Descripción del Sistema Actual
 - 2.3. Descripción del Sistema Nuevo
 - 2.4. Identificación de Requisitos del Sistema
 - 2.4.1. Requisitos de información
 - 2.4.2. Requisitos funcionales
 - 2.5. Descripción de la solución
 - 2.6. Planificación del proyecto
 - 2.6.1. Equipo de trabajo
 - 2.6.2. Planificación temporal
 - 2.7. Estudio del coste del proyecto
- 3. Análisis del Sistema de Información
 - 3.1. Identificación del entorno tecnológico
 - 3.2. Modelado de datos
 - 3.2.1. Modelo Entidad-Relación

3.1.2. Esquema de la base de datos

3.1.3. Datos de prueba

3.3. Identificación de los usuarios participantes y finales

3.4. Identificación de subsistemas de análisis

3.5. Establecimiento de requisitos

3.6. Diagramas de Análisis

3.7. Definición de interfaces de usuario

3.7.1. Especificación de principios generales de interfaz

3.7.2. Especificación de formatos individuales de la interfaz de pantalla

3.7.3. Identificación de perfiles de usuario

3.7.4. Especificación de formatos de impresión

3.7.5. Especificación de la navegabilidad entre pantallas

4. Construcción del Sistema

5. Glosario de términos

6. Bibliografía

1. Introducción

Se trata de una página web relacionada con la venta de entradas para los distintos conciertos que se realicen en tu ciudad o en alrededores. Un usuario registrado podrá buscar conciertos en la ciudad que haya introducido el usuario en el registro o en toda la página. Una vez encontrado el concierto al que quiere asistir y comprobado que queden entradas libres, el usuario realiza una compra.

Los distintos servicios que ofrece son los siguientes:

- Búsqueda de conciertos
- Compra de entradas
- Historial de entradas
- Búsqueda de bandas

La aplicación utilizará las siguientes tecnologías:

- Spring MVC
- MYSQL
- Spring Boot
- Thymeleaf
- Itext
- Bootstrap

2. Estudio de viabilidad

2.1 Alcance de proyecto

La página web será accesible por todos los navegadores web y tendrá un diseño responsive, en el cual dependiendo del tamaño de la página cambiará de tamaño o incluso visualmente.

Existirán dos tipos de usuarios:

- Usuario: Solo podrán buscar conciertos y comprar entradas
- Administrador: Podrá insertar nuevos conciertos o bandas

2.2 Descripción del sistema actual

Actualmente en muchas salas de conciertos la compra de tickets tiene que ser de manera física, es decir, que para poder comprar una entrada tiene que ir personalmente el cliente a la taquilla a comprarlo. Lo cual supone un costo de tiempo bastante amplio para el usuario, ya que se tiene que desplazar con antelación al lugar en cuestión.

Para poder ver los conciertos que hay en tu ciudad, debes de ir sala por sala para saberlo o bien leyéndolo en los múltiples carteles publicitarios, pudiendo perder la oportunidad de ir a algunos conciertos debido a que no se puede saberlo con la facilidad que te ofrece una página web.

Los tickets a entregarse en físico supondría un gran gasto de papel además de que el cliente podría perderlo con facilidad.

2.3 Descripción del sistema nuevo

Con esta aplicación se pretende informatizar todo y centralizar, ahorrando tiempo al usuario, que puede acceder a todos los conciertos con unos clicks.

2.4 Identificación de requisitos del sistema

2.4.1 Requisitos de información

IR- 0.1 USER	
Datos útiles	<ul style="list-style-type: none">- iduser- username- password- firstName- lastName- city- favoriteBand
Descripción	Usuario con el que se logea, solo puede un usuario con un nombre de usuario igual con el correo
Observaciones	Los valores de city y favoriteBand no son obligatorios, facilitan la búsqueda en la base de datos.

IR- 0.2 BAND	
Datos útiles	<ul style="list-style-type: none"> - idband - bandName - description - components - genre
Descripción	Solo puede existir una banda con un nombre
Observaciones	El género es obligatorio, se podría usar como campo búsqueda

IR- 0.3 CONCERT	
Datos útiles	<ul style="list-style-type: none"> - concertName - city - amountTickets - band_idband
Descripción	Solo puede existir un concierto con un nombre, una banda puede tener varios conciertos
Observaciones	La cantidad de tickets es una cantidad que muestra el total que se puede vender de un concierto

IR- 0.4 BUY

Datos útiles	<ul style="list-style-type: none">- idBuy- user_iduser
Descripción	Tabla intermedia donde sirve para almacenar la compra de los tickets
Observaciones	Se podrían realizar devoluciones borrando el campo, aunque no quedaría registro

IR- 0.4 TICKETS

Datos útiles	<ul style="list-style-type: none">- idtickets- price- concert_band_idband- buy_id
Descripción	Total de tickets de un concierto
Observaciones	La id compra puede ser nula, cuando compra se asigna un valor y se considera como entrada comprada

2.5 Descripción de la solución

Se utiliza un servidor Tomcat interno en la aplicación Spring, el cual está alojado en una máquina virtual (ubuntu server 16.04). La propia aplicación puede utilizarse en cualquier navegador además de en móvil debido a su diseño responsive y al Cross Browser.

2.6 Planificación del proyecto

2.6.1 Equipo de trabajo

El equipo de trabajo está compuesto por Carlos Correa Falcón. Este realiza la función de analista programador, es decir, lleva a cabo todo lo relacionado con el análisis del proyecto y la implementación del funcionamiento completo del proyecto.

2.6.2 Planificación temporal

PLAZO	A REALIZAR
22 de Marzo	Documentación de anteproyecto.
26 de Abril	Documentación del proyecto hasta el punto 3.6 inclusive (fase de Análisis). Igualmente deberán entregarse los datos de prueba de la aplicación, preferentemente generados con una herramienta de generación de datos aleatorios. En el caso de proyectos de investigación se deberá entregar lo que se lleve hecho hasta ese momento del tutorial de uso de la tecnología.
A partir del 18 de Junio	Defensas de proyecto. Con dos días de anterioridad a la defensa de su proyecto el alumno deberá proporcionar al profesorado todo el contenido de su proyecto.

2.7 Estudio del coste del proyecto

Debido a la corta duración del proyecto y su baja complejidad en lo que a grandes proyecto se refiere, el coste aproximado puede ser de 2000€ siendo casi el total dedicado al pago del analista-programador y en los recursos necesarios.

3. Análisis del Sistema de Información

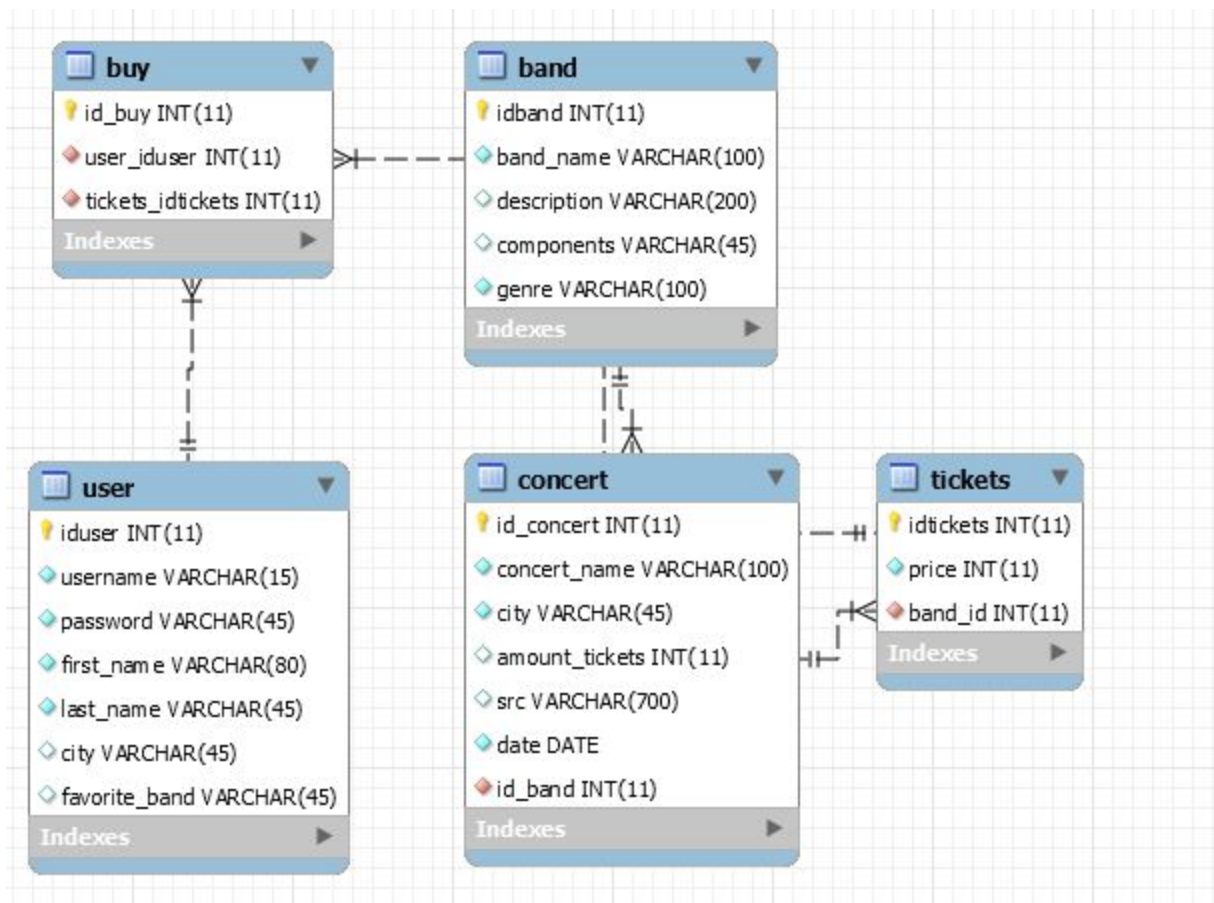
3.1.- Identificación del entorno tecnológico

En este apartado se muestra la infraestructura técnica usada durante la creación del proyecto.

- Hardware:
 - GPU GTX 750Ti
 - RAM 8GB
 - Intel Core i5-6400 2.60GHz
- Software:
 - Eclipse Spring
 - Windows 10 64 Bits
 - MYSQL 8
 - Distintos navegadores

3.2 Modelado de datos

3.2.1.- Modelo Entidad-Relación



3.1.2.- Esquema de la base de datos

Se usa una base de datos relacional, por lo que se adjunta un archivo sql con el esquema de la base de datos.

3.1.3.- Datos de prueba

Se adjunta una base de datos con unos usuarios ya creados además de unos conciertos y bandas ya creadas por defecto.

3.3 Identificación de los usuarios participantes y finales

Existen dos tipos de usuarios:

- Administrador: Es el usuario llamado "admin", quien podrá acceder al listado de usuarios además de crear los conciertos de las distintas bandas y eliminar usuarios.
- Usuario: Se encargará de la compra de los distintos tickets y la eliminación de estos para el concierto que desee, incluyendo la descarga de estos mismos.

3.4 Identificación de subsistemas de análisis

En la aplicación existen 4 subsistemas donde se encuentran el cómputo global de las funcionalidades del sistemas, las cuales son:

- **Gestión de usuarios:** Un usuario anónimo puede crearse un usuario o loguearse a la página web. Una vez logueado puede ver y modificar sus datos. El administrador puede ver todos los usuarios y borrar.
- **Gestión de tickets:** Una vez logueado, puede comprar el ticket del concierto que quiera, si quedan entradas.
- **Gestión de conciertos:** El administrador puede crear más conciertos si así lo desea de una banda previamente creada. El usuario solo podrá acceder a los datos del mismo.
- **Gestión de bandas:** Un usuario puede observar una banda y la información de esta. El administrador podrá añadir bandas.

3.5 Establecimiento de requisitos

Subsistema gestión de usuarios:

- Registrarse: los usuarios anónimos podrán registrarse en la aplicación
- Modificar datos: los usuarios registrados podrán modificar sus datos en cualquier momento.

Subsistema gestión ticket

- Comprar: Un usuario podrá comprar tickets si quedan entradas de este
- Devolución: Si un usuario ya no quiere asistir a un concierto, podrá devolver la entrada.
- Descargar entrada: Cuando se haya comprado una entrada, se puede descargar.

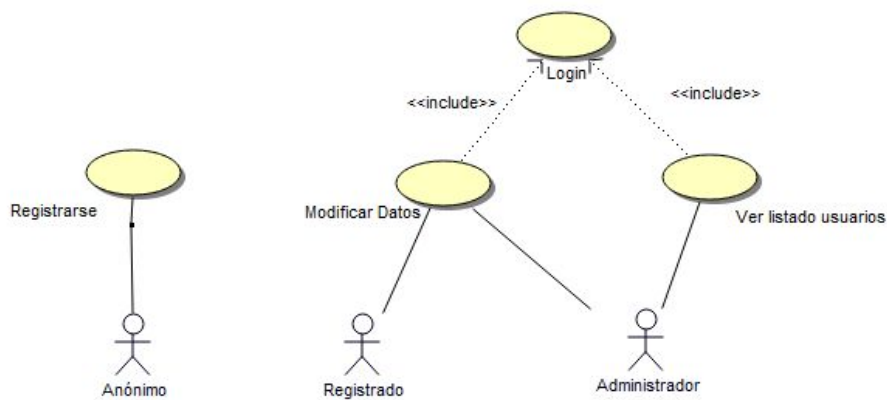
Subsistema gestión de conciertos

- Añadir: Un administrador podrá crear un concierto de una banda ya creada

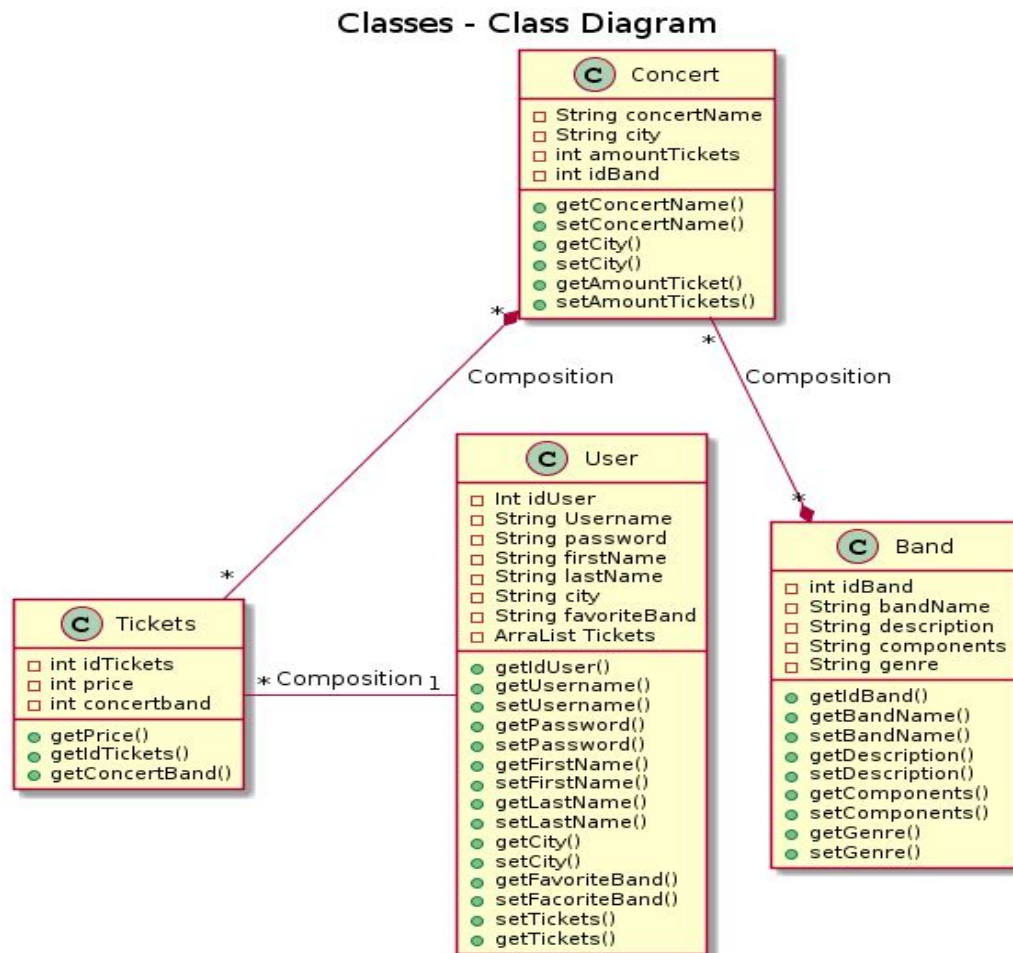
Subsistema gestión de bandas

- Añadir: Un administrador podrá crear un grupo.

3.5 Establecimiento de requisitos



3.6 Diagramas de Análisis



3.7. Definición de interfaces de usuario

3.7.1. Especificación de principios generales de interfaz.

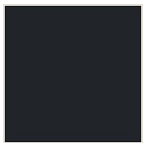
La página tiene una estructura similar en todas sus pantallas para facilitar en uso de la misma a los usuarios-

- **Colores**

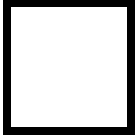
Los colores usado son:



#007bff Color principal -> Usado en la barra de navegación



#212529 Color secundario -> Para los títulos



#fff Usado para la navegación de la barra de navegación y las tablas



#343a40 Usado para el fondo de la cabecera de la tabla



#7abaff Usado para el fondo de la tabla y la compra de ticket

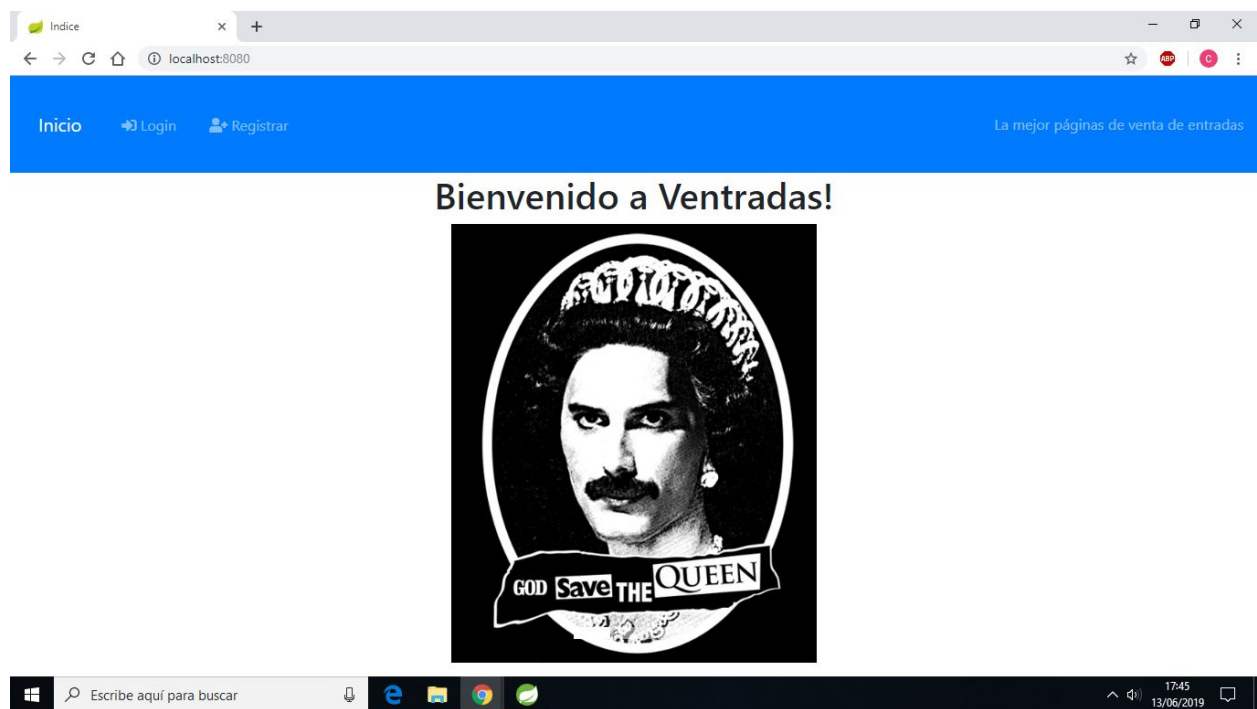
- **Tipografía:**

Todos los textos usan la tipografía por defecto de Bootstrap.

3.7.2 Especificación de formatos individuales de la interfaz de pantalla

- **Pantalla de inicio**

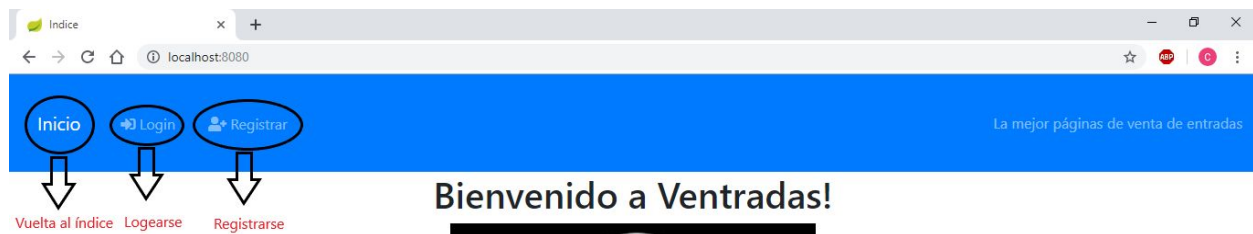
Al cargar la página por primera vez la página se muestra el índice con un carrousel de algunas imágenes de la página.



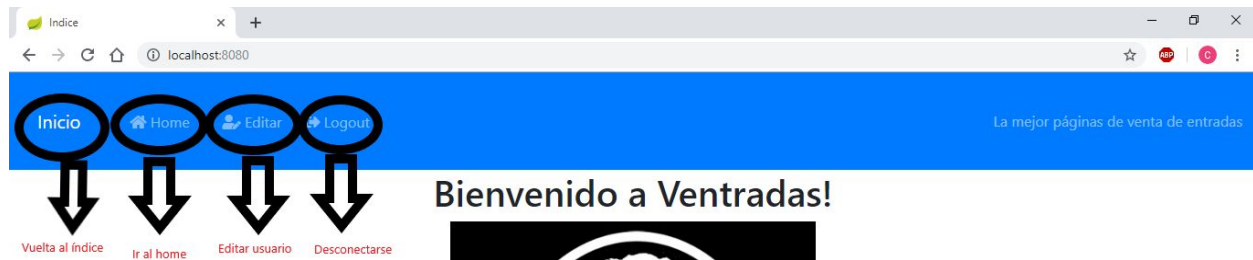
- **Barra de navegación.**

Todas las pantallas están compuesta de una barra de navegación la cual nos permite moverse entre las pantallas en cualquier momento. Dependiendo si estás logeado o no puede llegar a variar. También dependiendo en qué página te encuentres se colorea el enlace en la que estás.

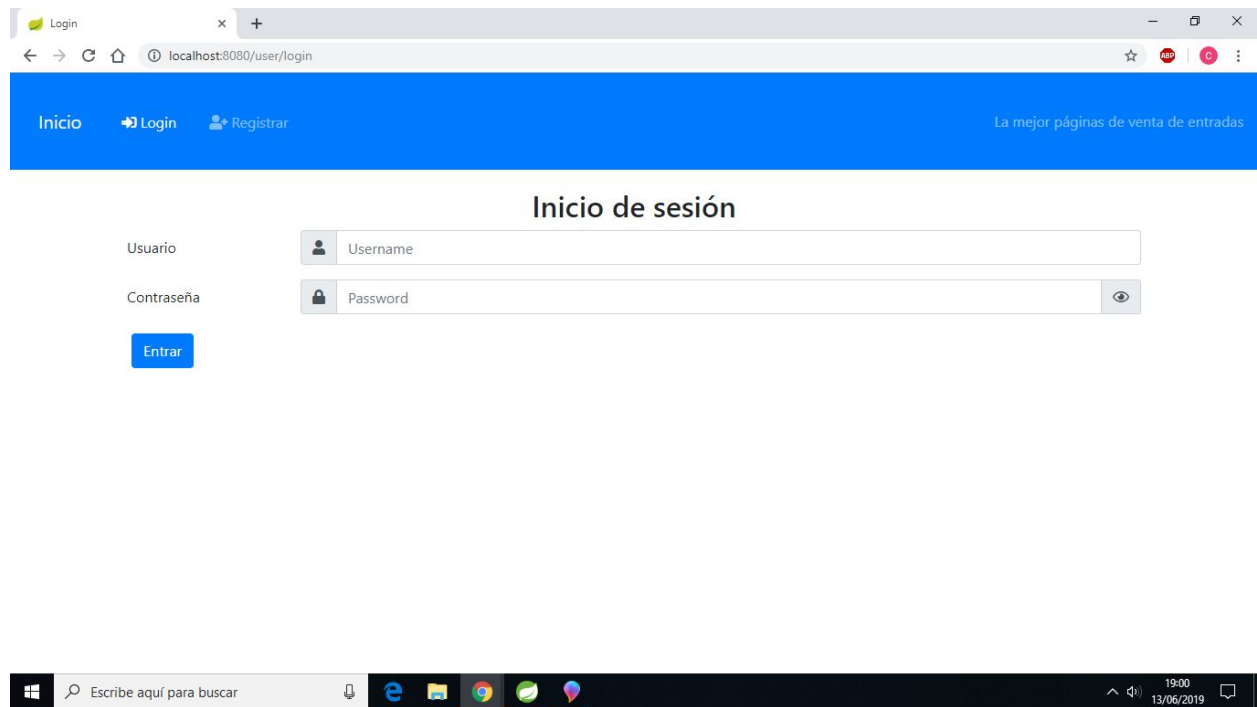
Barra sin logeo



Barra después de logearse



- Pantalla de Login



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/user/login'. The page has a blue header with navigation links: 'Inicio', 'Login', and 'Registrar'. A tagline on the right reads 'La mejor páginas de venta de entradas'. The main content area is titled 'Inicio de sesión' and contains a login form. The form has two input fields: 'Usuario' (Username) and 'Contraseña' (Password). The 'Contraseña' field has a toggle icon to show or hide the password. Below the fields is a blue 'Entrar' (Enter) button. The Windows taskbar at the bottom shows the search bar, taskbar icons for Edge, File Explorer, Chrome, and others, and the system clock indicating 19:00 on 13/06/2019.

Inicio de sesión

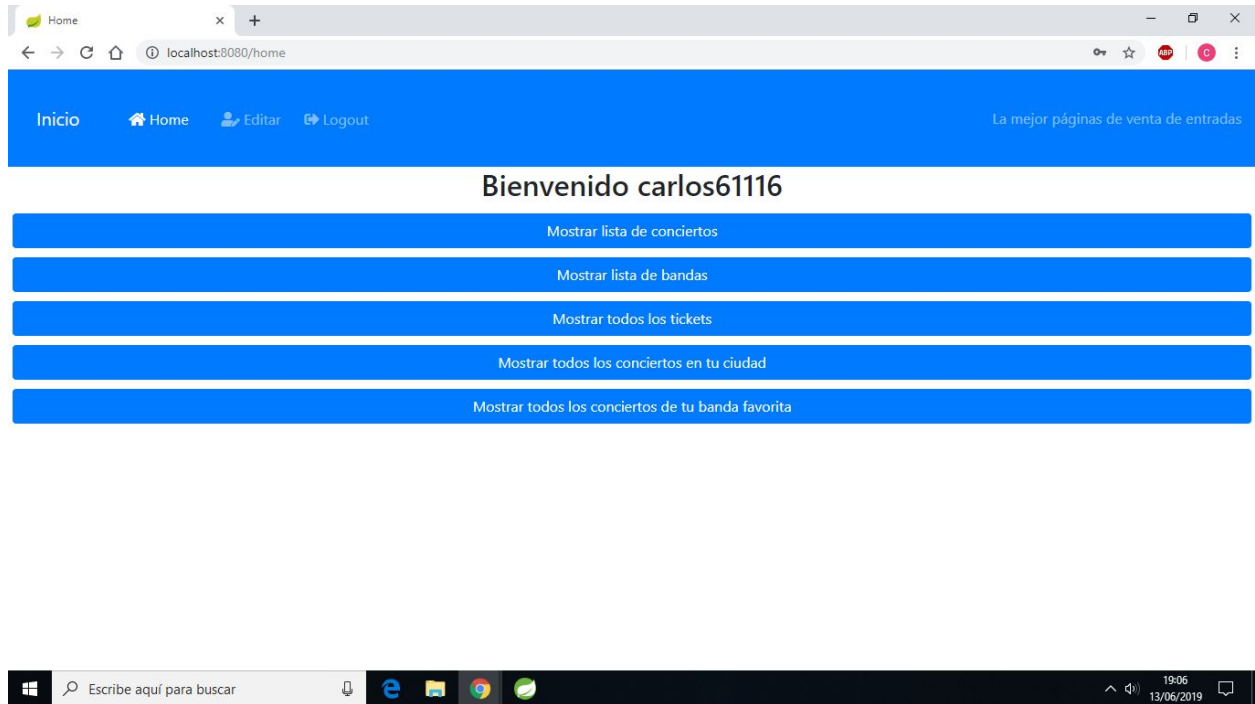
Usuario: Username

Contraseña: Password

Entrar

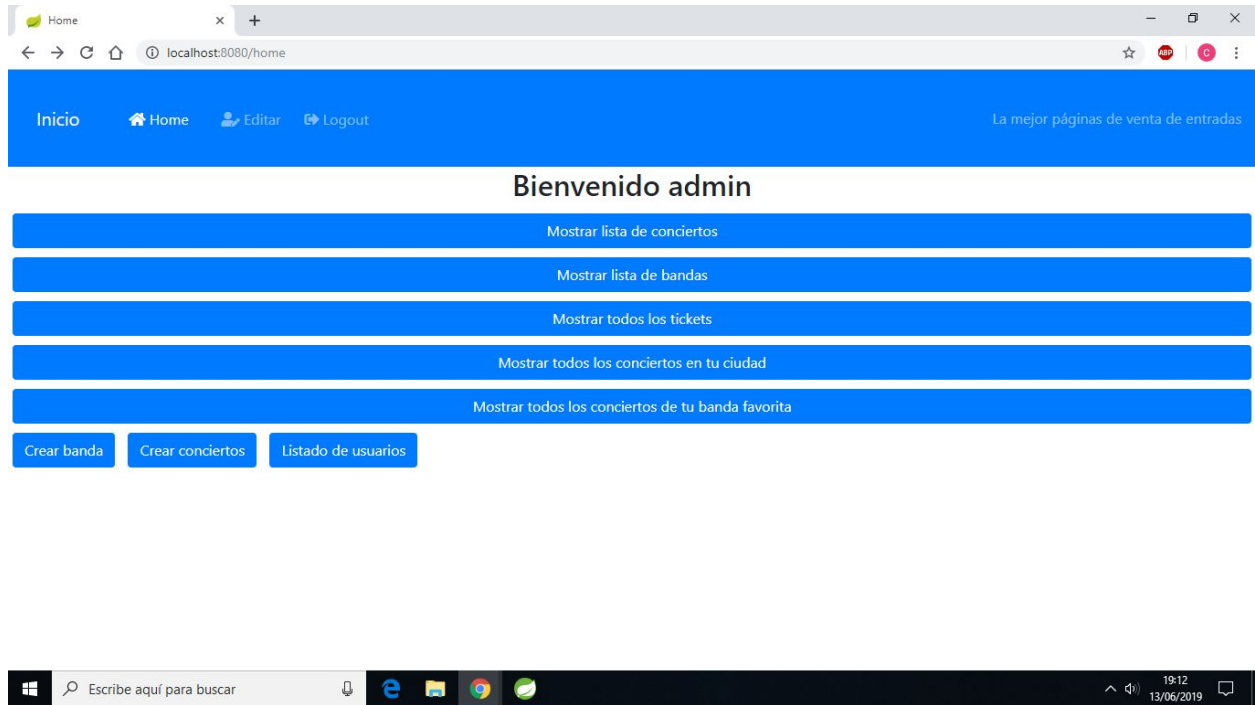
En esta pantalla se permite al usuario iniciar sesión. También tiene la posibilidad de registrarse o de volver al índice.

• Pantalla Home

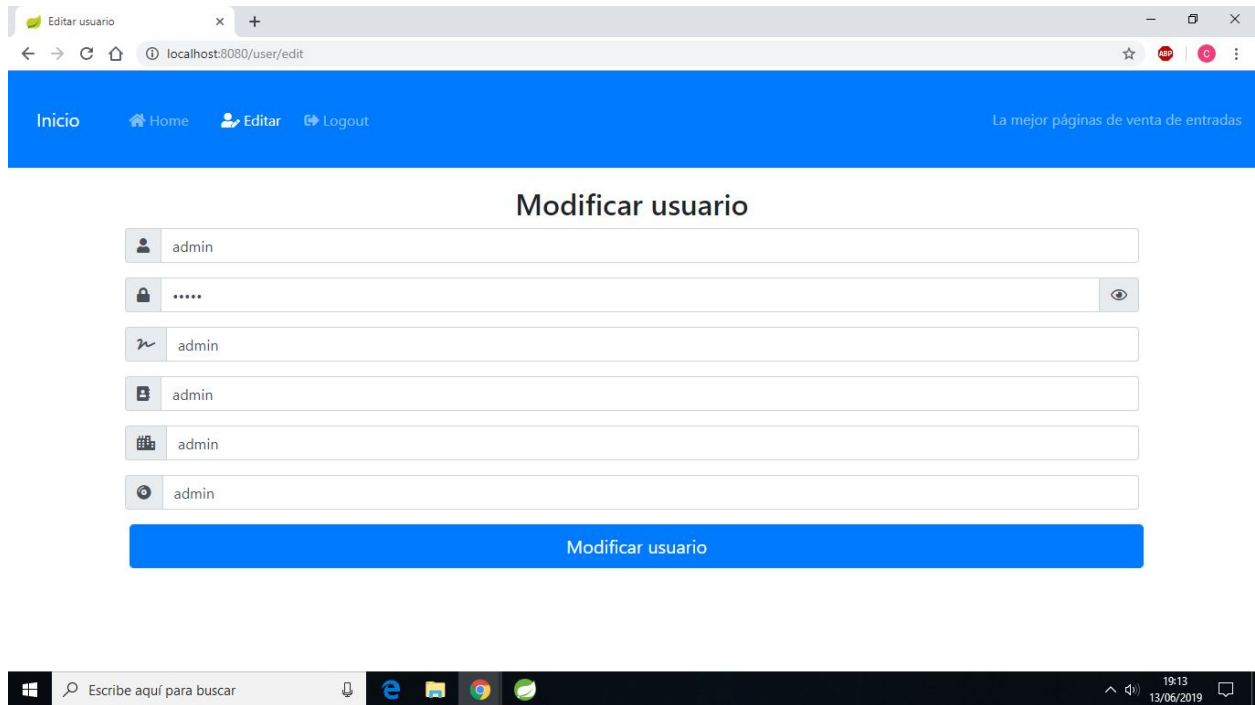


Una vez iniciado sesión accederemos a esta página, en la cual es la central de toda la aplicación. Desde ella podremos acceder a: comprar tickets, ver bandas, ver conciertos y dependiendo si tienes ciudad o banda favorita se te permite filtrar por alguno de los dos.

Esta pantalla también varía dependiendo si eres el administrador o no.

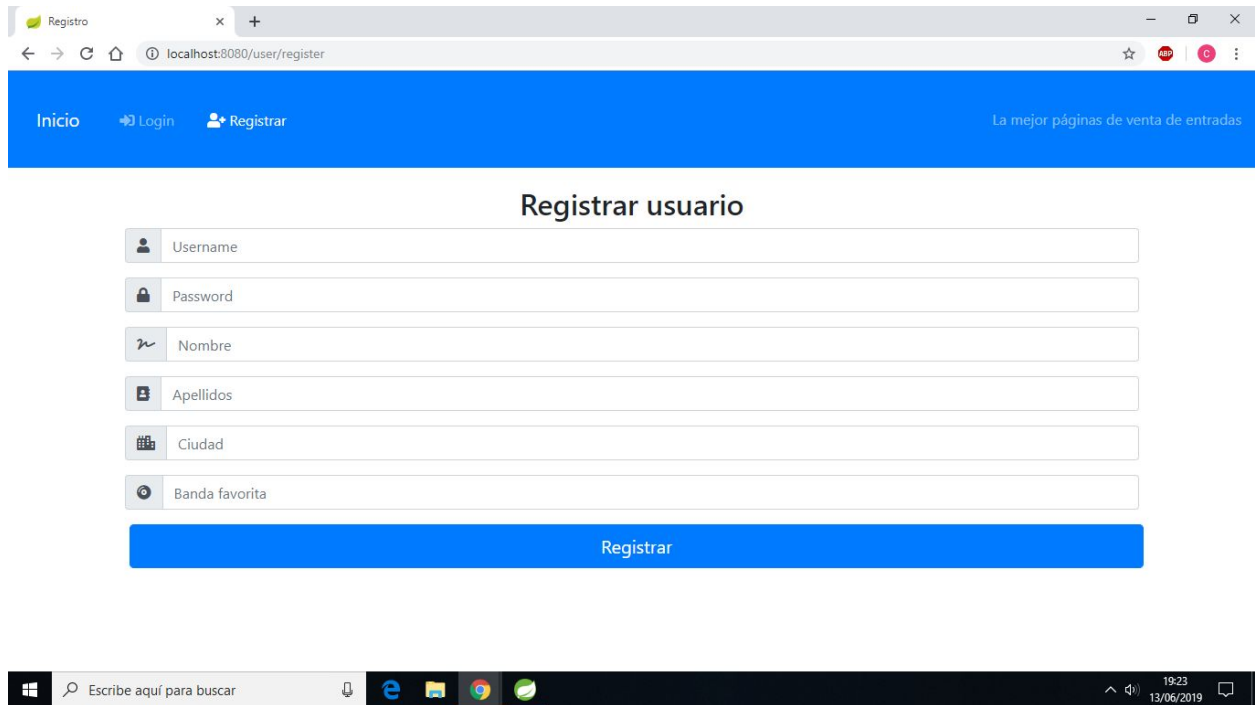


- **Modificar usuario**



La pantalla de modificar usuario es bastante parecida a la del registro, solo que los datos ya estarían cargados. Para identificar los campos al lado de cada campo tiene una imagen identificatoria.

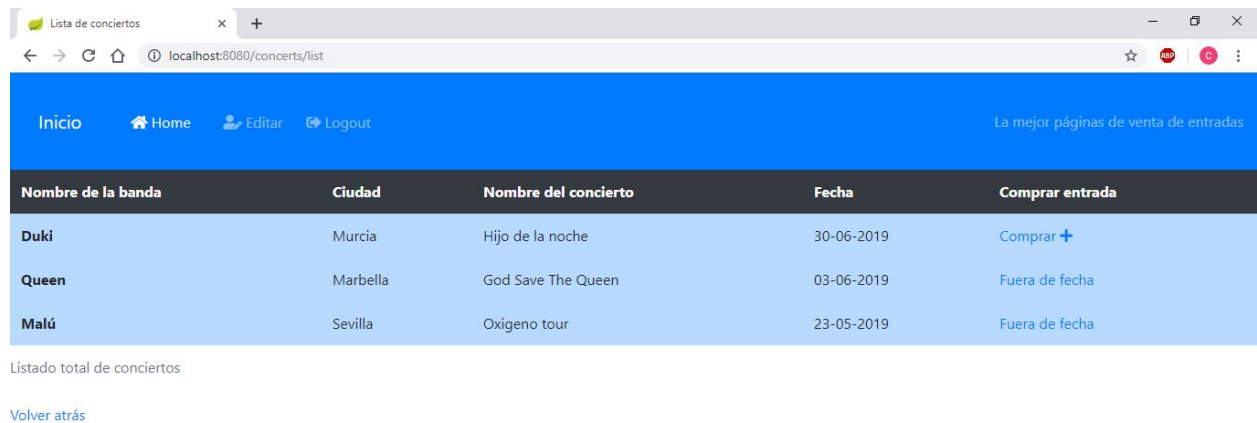
- Registro de usuario



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/user/register'. The page has a blue header with navigation links: 'Inicio', 'Login', and 'Registrar'. A tagline on the right reads 'La mejor páginas de venta de entradas'. The main content area is titled 'Registrar usuario' and contains a registration form with the following fields: 'Username', 'Password', 'Nombre', 'Apellidos', 'Ciudad', and 'Banda favorita'. Each field has a corresponding icon on the left. Below the fields is a large blue 'Registrar' button. The Windows taskbar at the bottom shows the search bar, taskbar icons for Edge, File Explorer, Chrome, and WhatsApp, and a system tray with the time '19:23' and date '13/06/2019'.

Es bastante parecida al de modificar.

- Listado de conciertos



Nombre de la banda	Ciudad	Nombre del concierto	Fecha	Comprar entrada
Duki	Murcia	Hijo de la noche	30-06-2019	Comprar +
Queen	Marbella	God Save The Queen	03-06-2019	Fuera de fecha
Malú	Sevilla	Oxígeno tour	23-05-2019	Fuera de fecha

Listado total de conciertos

[Volver atrás](#)

Una vez pulsemos para ver los conciertos nos saldrá la siguiente pantalla, la cual nos mostrará un listado de todos los conciertos que van a celebrarse o los que ya han sido. Dependiendo de la fecha podremos comprar tickets para ellos o no. También existen dos variables para esta pantalla, un filtro para ciudad y la banda.

Lista de conciertos

localhost:8080/concerts/list?city=Sevilla

Inicio Home Editar Logout La mejor páginas de venta de entradas

Nombre de la banda	Ciudad	Nombre del concierto	Fecha	Comprar entrada
Malú	Sevilla	Oxígeno tour	23-05-2019	Fuera de fecha

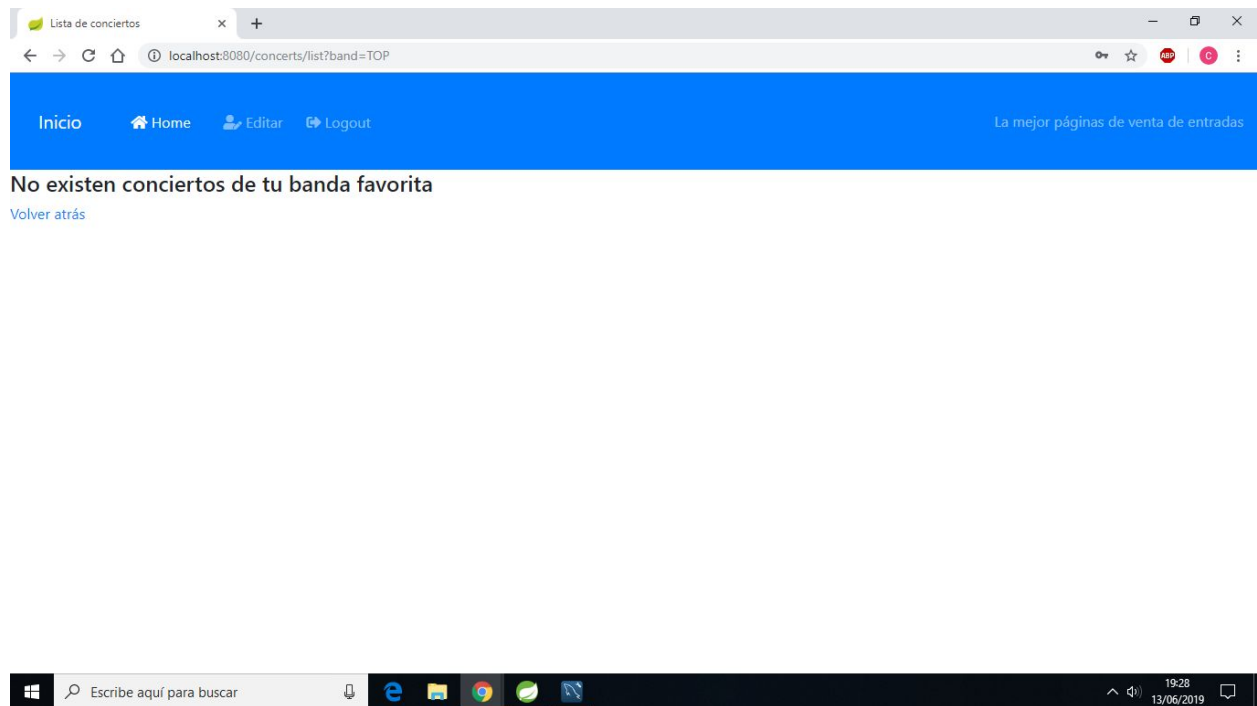
Listado total de conciertos

[Volver atrás](#)

Escribe aquí para buscar

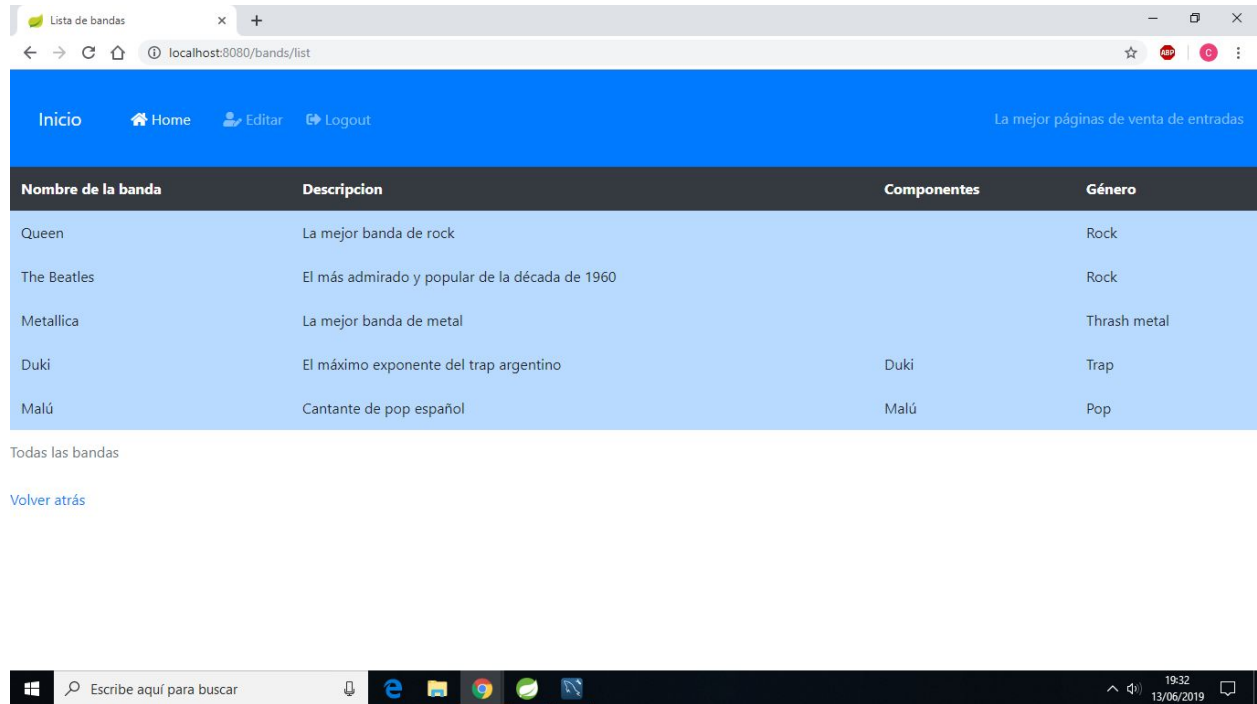
19:28 13/06/2019

Filtrado por ciudad



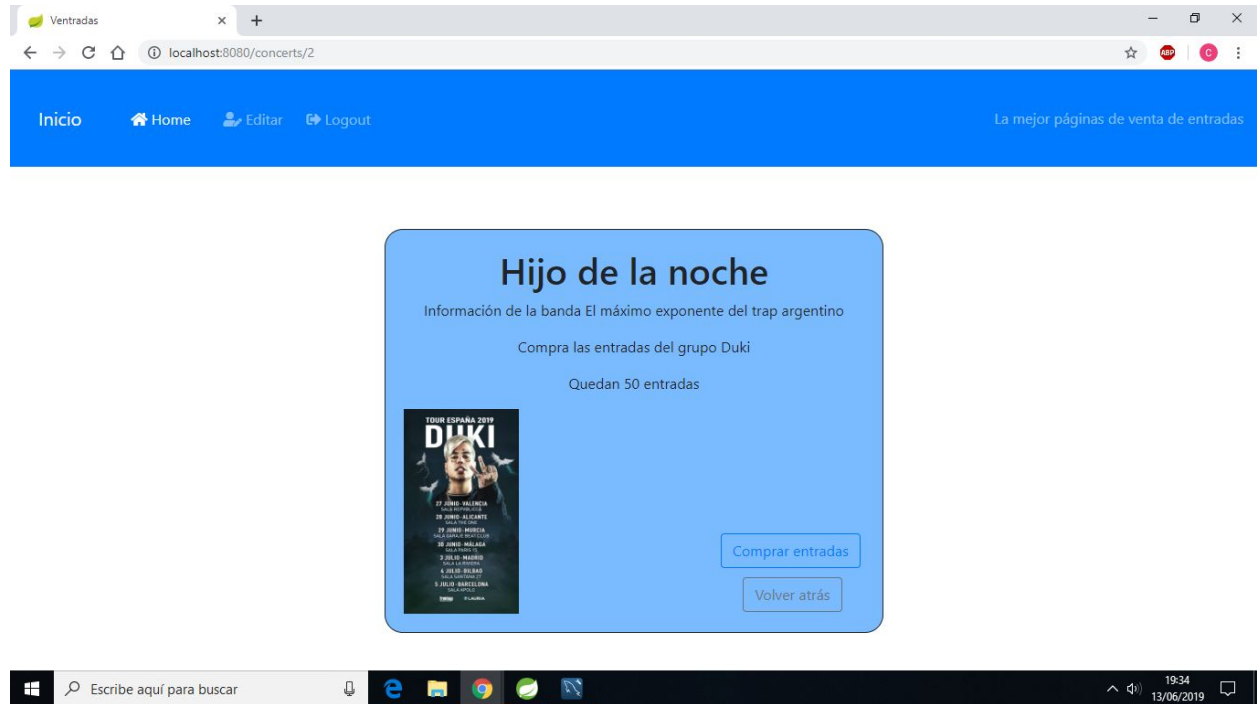
Filtrado por banda, si no existe banda favorita mostrará un mensaje de error.

- Listado de bandas.



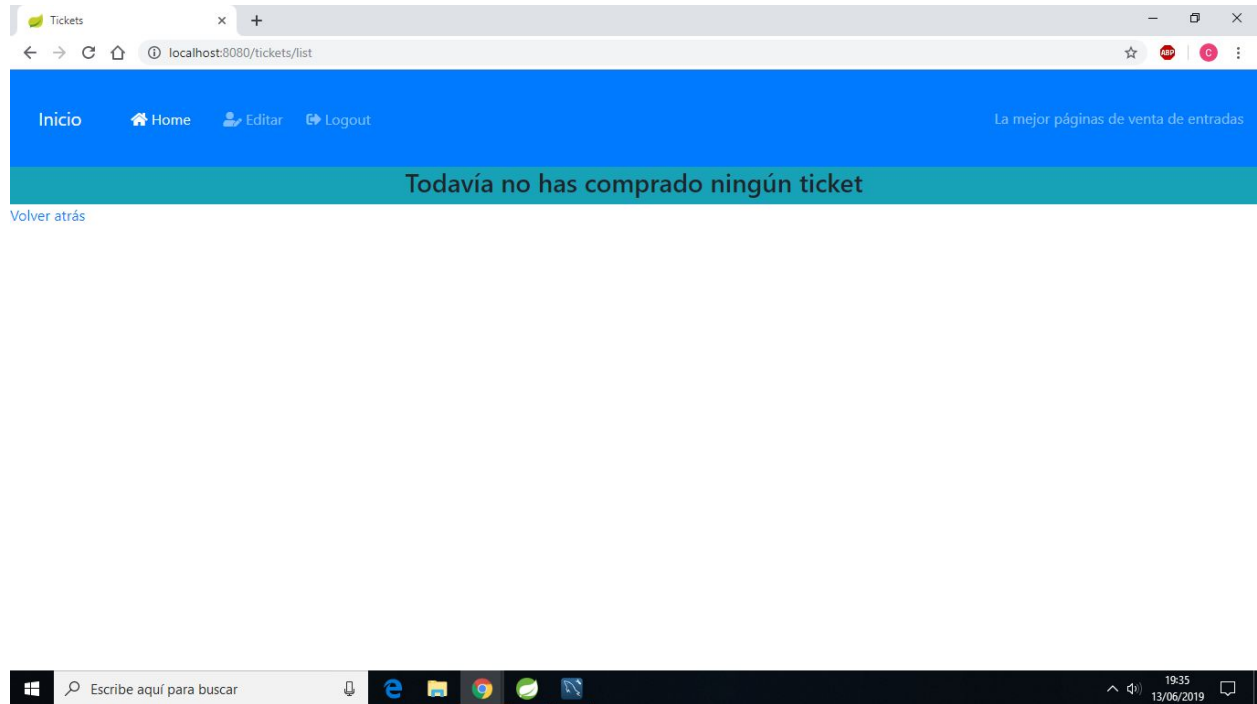
Aquí podremos acceder a todas las bandas registradas en nuestra web, independientemente de si dan concierto o no. En esta pantalla tendremos más información de cada banda.

- Compra de tickets

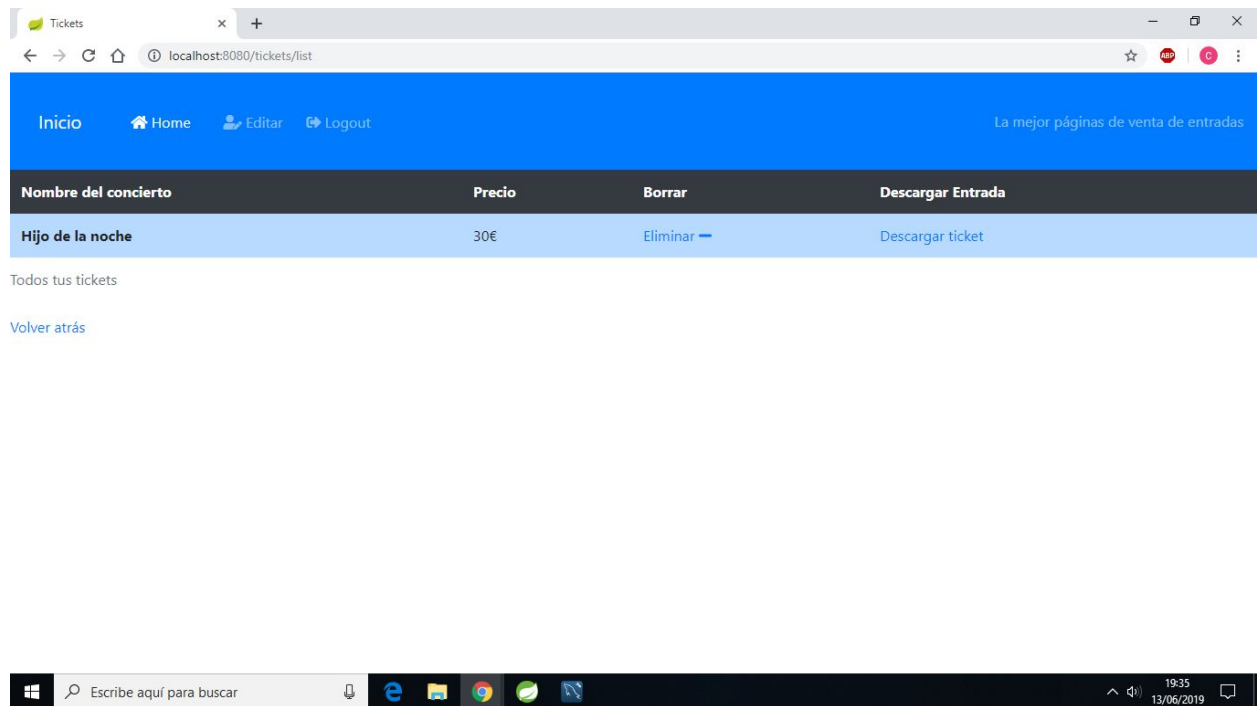


Una vez pulsemos para comprar ticket nos aparecerá la siguiente pantalla, en la cual nos mostrará más información del concierto, la cantidad de tickets restante y la posibilidad de volver atrás.

- Listado de tickets

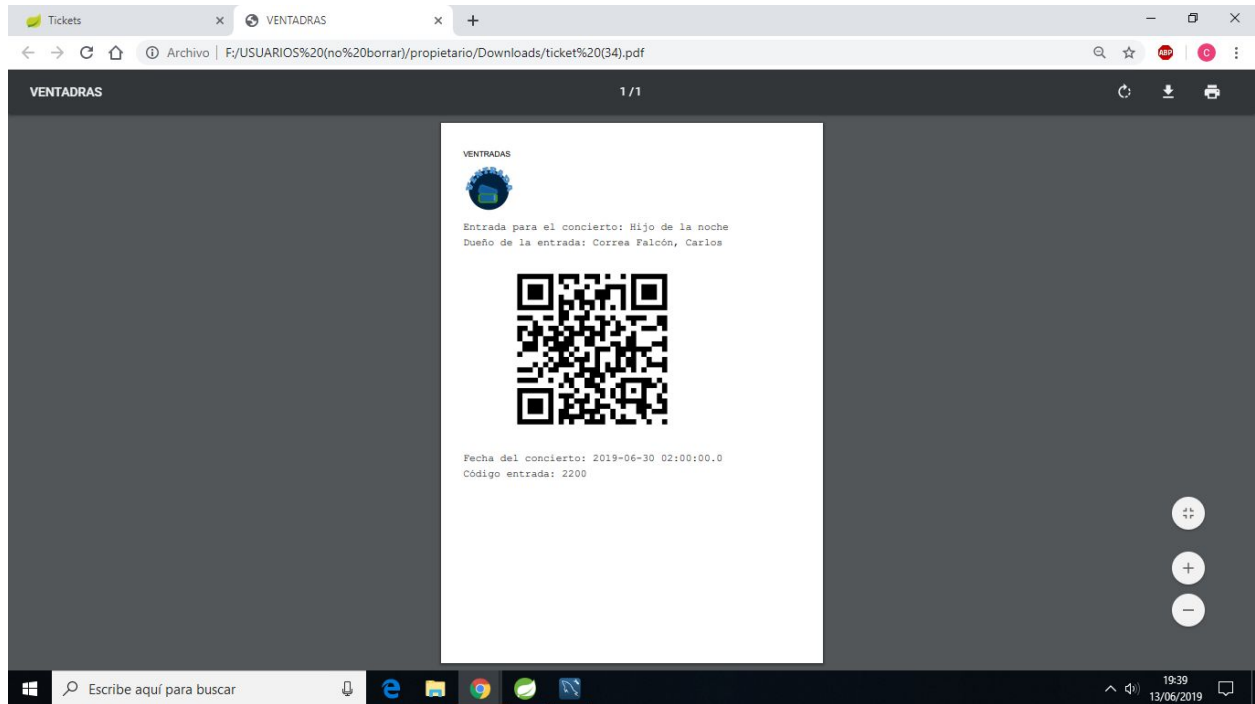


Una vez pulsemos en mostrar los tickets, nos mostrará todos los tickets que tenemos actualmente. Si no tenemos ninguno se mostrará la pantalla de arriba.



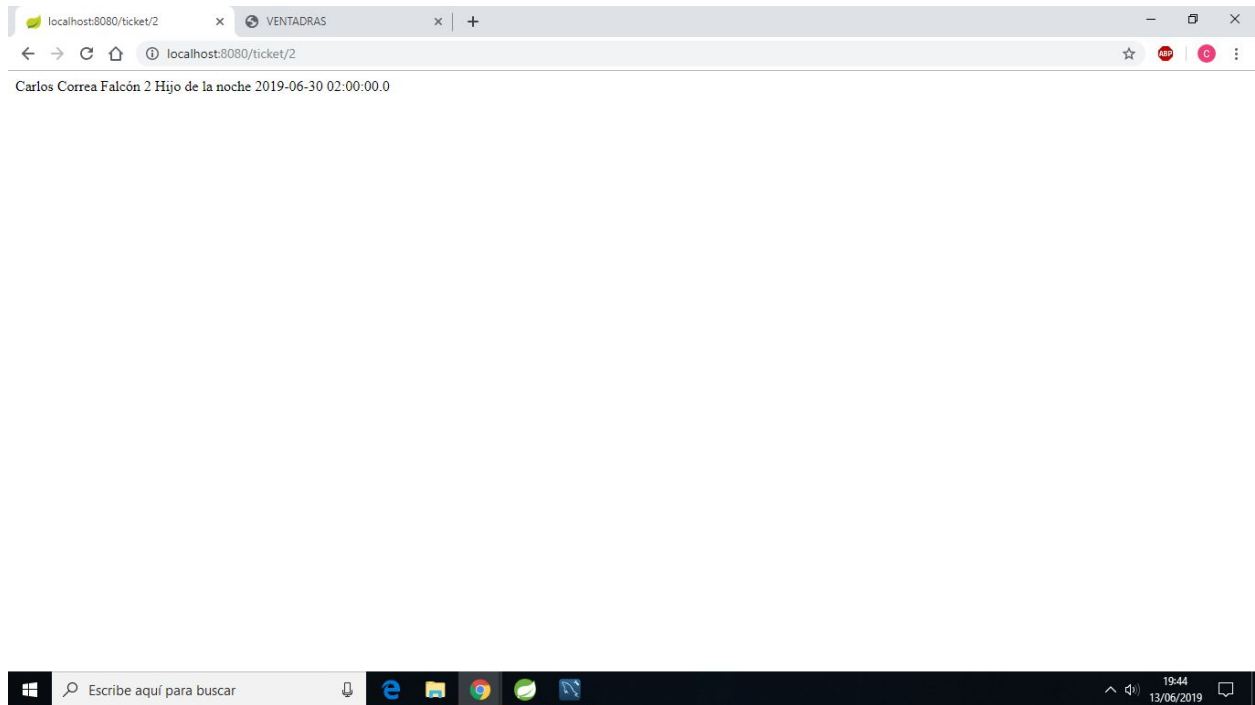
Una vez que tengamos tickets, nos mostrará la siguiente pantalla, la cual nos muestra todos los tickets en forma de tabla, una vez llegado podemos o devolver el ticket o descargarlo para que pueda ser usado.

- Ticket



Quando pulsemos en comprar ticket se nos mostrará la entrada para el concierto que queramos, con el título de nuestra página, el logo e información del comprador y del concierto. En la parte central se mostrará un código QR que contiene info sobre la entrada. Se muestra el código de la entrada, el cual tiene números aleatorios para seguridad.

- **Página del código QR**



Se muestra una mini página con info del concierto.

- Creación de conciertos

The screenshot shows a web browser window with the address bar displaying 'localhost:8080/admin/band'. The page has a blue header with navigation links: 'Inicio', 'Home', 'Editar', and 'Logout'. On the right side of the header, it says 'La mejor páginas de venta de entradas'. The main content area is titled 'Creación de bandas' and contains four form fields, each with a lock icon indicating they are required:

- Nombre *: Input field with placeholder text 'Nombre de la banda'.
- Componente *: Input field with placeholder text 'Componentes'.
- Descripción: Input field with placeholder text 'Descripción'.
- Género: Input field with placeholder text 'Género'.

Below the form fields is a blue button labeled 'Crear'. The Windows taskbar at the bottom shows the search bar with the text 'Escribe aquí para buscar', several application icons (Edge, File Explorer, Chrome, etc.), and the system clock showing '19:49' and '13/06/2019'.

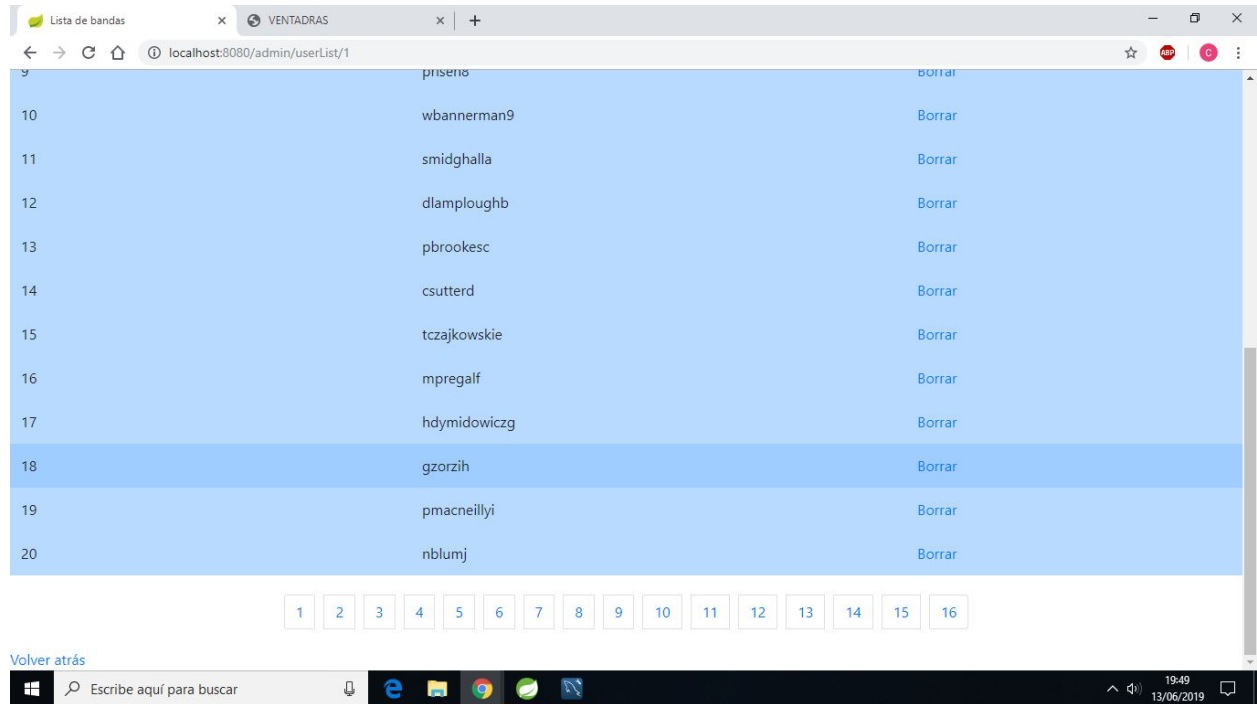
La siguiente pantalla se muestra solo al administrador, el cual puede crear bandas.

- Creación de conciertos

The screenshot shows a web browser window with two tabs: 'Bandas' and 'VENTADRAS'. The address bar shows 'localhost:8080/admin/concert'. The page has a blue header with navigation links: 'Inicio', 'Home', 'Editar', and 'Logout'. On the right side of the header, it says 'La mejor páginas de venta de entradas'. The main content area is titled 'Creación de conciertos'. It contains a form with the following fields: 'Nombre *' (with a person icon and placeholder 'Nombre del concierto'), 'Ciudad *' (with a location pin icon and placeholder 'Ciudad'), 'Enlace' (with a lock icon and placeholder 'Enlace imagen'), 'Fecha *' (with a lock icon and placeholder 'dd/mm/aaaa'), 'Tickets *' (with a person icon and placeholder 'Cantidad de tickets'), and 'Precio *' (with a lock icon and placeholder 'Precio del ticket'). Below these fields is a dropdown menu currently showing 'Queen' and a blue 'Crear' button. The Windows taskbar is visible at the bottom, showing the search bar and several application icons. The system clock indicates 19:49 on 13/06/2019.

Al igual que la anterior, solo se muestra al administrador y nos permite crear conciertos. Para asignarle una banda se hace mediante un select para evitar conflictos si introduces mal el nombre de la banda.

- **Control de usuarios.**



A esta pantalla también el administrador es el único que pueda acceder. En esta se muestran todos los usuarios registrados, y la capacidad de borrarlos. Si los usuarios tienen algún ticket comprado se les borran.

- **Versión móvil.**

Todas las capturas realizadas anteriormente han sido realizadas desde Chrome en versión PC, pero la página también permite una versión móvil y desde cualquier navegador sin sufrir ningún cambio que afecte a la página negativamente.

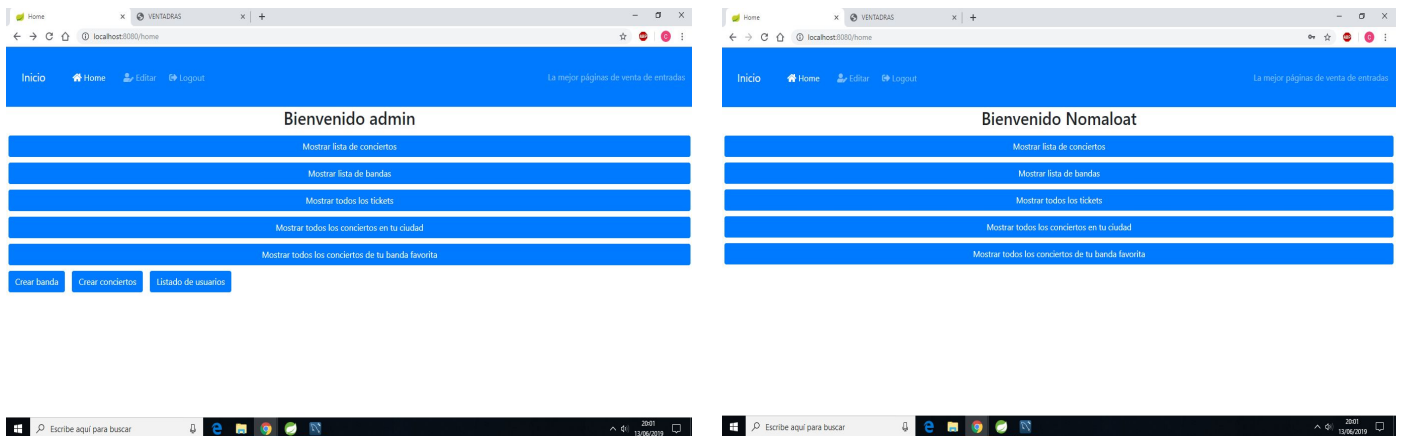
3.7.3 Especificación de la navegabilidad entre pantallas.

Todas las pantallas están unidas entre ellas por la barra de navegación principal.

3.7.3 identificación de perfiles de usuarios

La página tiene dos tipos de usuarios:

- **Usuario normal:** Tiene acceso al listado de conciertos, de bandas y de sus tickets
- **Administrador:** Tiene acceso a las mismas páginas que el usuario normal pero añadiendo la capacidad de crear bandas, conciertos y el listado de usuarios, permitiendo borrar uno



4. Análisis del Sistema de Información

Java

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos, que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo

- La principal característica de Java es que es independiente de la plataforma (multiplataforma).
- El lenguaje Java es orientado a objetos.
- En java no existen problemas con la liberación de memoria en el sistema.
- Librerías Estándar: Una de las características que más potencia aporta al lenguaje Java es que viene acompañado de una serie de librerías estándar para realizar multitud de operaciones comunes a la hora de programar. Es el llamado Java API.

Spring

Es el framework más popular para Java empresarial usado para crear código de alto rendimiento, liviano y reutilizable. Ya que su finalidad es estandarizar, agilizar, manejar y resolver los problemas que puedan ir surgiendo en el trayecto de la programación.

Spring, ofrece como elemento clave el soporte de infraestructura a nivel de aplicación, brindando un completo modelo tanto para la configuración como para la programación de aplicaciones empresariales desarrolladas bajo Java, sin discriminación en cuanto al despliegue de la plataforma.

Características:

- Acceso a datos: soporte DAO, JDBC, ORM, Marshalling XML.
- Gestión de transacciones.
- MVC (Modelo Vista Controlador).
- Seguridad.
- Frameworks web: Spring WebFlux y Spring MVC.
- Procesamiento de datos por lotes.
- Es un framework liviano debido a su implementación POJO (Plain Old Java Object), Spring Framework no obliga al programador a heredar ninguna clase ni a implementar ninguna interfaz.

- La Inyección de Dependencias (Dependency Injection): Al momento de escribir una aplicación Java compleja, las clases de la aplicación deben ser lo más independientes posible de otras clases Java, para aumentar la posibilidad de reutilizarlas y probarlas independientemente de otras clases, mientras se prueban las unidades. Básicamente la inyección de dependencias (DI) ayuda a unir estas clases y al mismo tiempo mantenerlas.

Ventajas:

- Spring está organizado de forma modular. A pesar de la cantidad de paquetes y clases que tiene, solo debemos ocuparnos de aquellos que necesitemos para nuestro desarrollo e ignorar el resto.
- Utiliza algunas de las tecnologías existentes, como varios frameworks ORM, JEE, temporizadores Quartz y JDK, frameworks de registro y otras tecnologías de visualización.
- El framework web de Spring es un framework MVC web bien diseñado, que proporciona una excelente alternativa a los frameworks web como Struts u otros frameworks web sobre diseñados o menos populares.
- Spring proporciona una API para traducir excepciones específicas de la tecnología (como por ejemplo las generadas por JDBC, Hibernate o JDO) en excepciones consistentes y no verificadas.
- Los contenedores de IoC (Inversion of Control) tienden a ser livianos, especialmente cuando se comparan con los Enterprise JavaBeans (EJB). Esto es ideal para desarrollar y desplegar aplicaciones en máquinas con memoria y recursos limitados.

- Cuenta con plantillas para diversas tecnologías entre la cuales podemos destacar las siguientes: JDBC, Hibernate y JPA, de forma tal que no hay necesidad de escribir un código extenso, ya que con estas plantillas simplifica el trabajo en cuanto a los pasos básicos a implementar de estas tecnologías.

Spring boot

Spring Boot es una herramienta que nace con la finalidad de simplificar aun más el desarrollo de aplicaciones basadas en el ya popular framework Spring Core. Spring Boot busca que el desarrollador solo si centre en el desarrollo de la solución, olvidándose por completo de la compleja configuración que actualmente tiene Spring Core para poder funcionar.

- Configuración: Spring Boot cuenta con un complejo módulo que autoconfigura todos los aspectos de nuestra aplicación para poder simplemente ejecutar la aplicación, sin tener que definir absolutamente nada.
- Resolución de dependencias: Con Spring Boot solo hay que determinar que tipo de proyecto estaremos utilizando y el se encarga de resolver todas las librerías/dependencias para que la aplicación funcione.
- Despliegue: Spring Boot se puede ejecutar como una aplicación Stand-alone, pero también es posible ejecutar aplicaciones web, ya que es posible desplegar las aplicaciones mediante un servidor web integrado, como es el caso de Tomcat, Jetty o Undertow.
- Métricas: Por defecto, Spring Boot cuenta con servicios que permite consultar el estado de salud de la aplicación, permitiendo saber si la aplicación está prendida o apagada, memoria utilizada y disponible, número y detalle de los Bean's creado por la aplicación, controles para el prendido y apagado, etc.

- Extensible: Spring Boot permite la creación de complementos, los cuales ayudan a que la comunidad de Software Libre cree nuevos módulos que faciliten aún más el desarrollo.

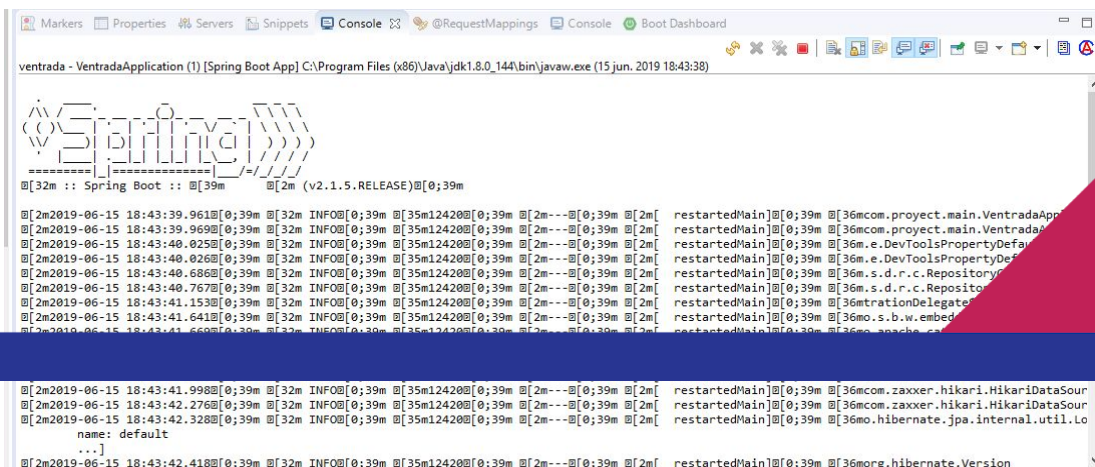
¿Como funciona? <https://start.spring.io/>

```

1 package com.proyect.main;
2
3
4 import org.springframework.boot.SpringApplication;
5
6
7 @SpringBootApplication
8 public class VentradaApplication extends SpringBootServletInitializer{
9
10     public static void main(String[] args) {
11         SpringApplication.run(VentradaApplication.class, args);
12     }
13
14     @Override
15     protected SpringApplicationBuilder configure(SpringApplicationBuilder application) {
16         return application.sources(VentradaApplication.class);
17     }
18
19     private static Class<VentradaApplication> applicationClass = VentradaApplication.class;
20 }

```

Tomcat Interno:



```

ventrada - VentradaApplication (1) [Spring Boot App] C:\Program Files (x86)\Java\jdk1.8.0_144\bin\javaw.exe (15 jun. 2019 18:43:38)

[32m :: Spring Boot :: [39m
[32m (v2.1.5.RELEASE)[0;39m

[2m2019-06-15 18:43:39.961][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36mcom.proyect.main.VentradaApp
[2m2019-06-15 18:43:39.969][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36mcom.proyect.main.VentradaApp
[2m2019-06-15 18:43:40.025][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36m.e.DevToolsPropertyDefau
[2m2019-06-15 18:43:40.026][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36m.e.DevToolsPropertyDefau
[2m2019-06-15 18:43:40.686][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36m.s.d.r.c.Repository
[2m2019-06-15 18:43:40.767][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36m.s.d.r.c.Repository
[2m2019-06-15 18:43:41.153][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36m.mtrationDelegat
[2m2019-06-15 18:43:41.641][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36m.s.b.w.embed
[2m2019-06-15 18:43:41.660][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36m.s.b.w.embed

[2m2019-06-15 18:43:41.998][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36mcom.zaxxer.hikari.HikariDataSour
[2m2019-06-15 18:43:42.276][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36mcom.zaxxer.hikari.HikariDataSour
[2m2019-06-15 18:43:42.328][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36m.hibernate.jpaw.internal.util.Lo
name: default
...
[2m2019-06-15 18:43:42.418][0;39m [32m INFO][0;39m [35m124200[0;39m [2m---[0;39m [2m restartedMain[0;39m [36morg.hibernate.Version

```

Propiedades de la aplicación

```
1 server.port=8080
2 server.error.whitelabel.enabled=false
3
4 spring.datasource.url=jdbc:mysql://localhost:3306/entradas?useSSL=false&useLegacyDatetimeCode=false&serv
5 spring.datasource.password=root
6 spring.datasource.username=root
7
8
9
10 spring.jpa.show-sql=true
11
12 spring.jpa.hibernate.ddl-auto = update
13
14 # Allows Hibernate to generate SQL optimized for a particular DBMS
15
16
17 spring.thymeleaf.suffix=.html
18
```

Thymeleaf

Thymeleaf es un motor de plantillas que permite realizar tareas que se conocen como natural templating. Es decir, como está basada en añadir atributos y etiquetas, sobre todo HTML, va a permitir que nuestras plantillas se puedan renderizar en local, y esa misma plantilla después utilizarla también para que sea procesada dentro del motor de plantillas. Por lo cual las tareas de diseño y programación se pueden llevar conjuntamente.

- Expresiones variables: Son quizás las más utilizadas, como por ejemplo `${...}`
- Expresiones de selección: Son expresiones que nos permiten reducir la longitud de la expresión si prefijamos un objeto mediante una expresión variable, como por ejemplo `*{...}`
- Expresiones de mensaje: Que nos permiten, a partir de ficheros properties o ficheros de texto, cargar los mensajes e incluso realizar la internalización de nuestras aplicaciones, como por ejemplo `#{...}`
- Expresiones de enlace: Nos permiten crear URL que pueden tener parámetros o variables, como por ejemplo `@{...}`
- Expresiones de fragmentos: Nos van a permitir dividir nuestras plantillas en plantillas más pequeñas e ir cargándolas según las vayamos necesitando, como por ejemplo `~{...}`

Itext

Itext es una herramienta que nos permite editar, crear ficheros pdf a partir de la información que nosotros le aportemos. Es de código abierto, a menos que se use en proyecto cerrados. Su implementación es sencilla en proyectos Maven, con la siguiente dependencia ya estaría disponible para su uso.

```
<dependency>
  <groupId>com.itextpdf</groupId>
  <artifactId>itextpdf</artifactId>
  <version>5.5.10</version>
</dependency>
```

Debido a ser bastante sencillo de implementar es muy usado hoy en día, un ejemplo:

```
Document document = new Document();
PdfWriter.getInstance(document, new FileOutputStream("ticket.pdf"))
    .setInitialLeading(20);

String ipAddress = getIP();

document.open();

Font font = FontFactory.getFont(FontFactory.COURIER, 16, BaseColor.BLACK);

BarcodeQRCode my_code = new BarcodeQRCode("http://" + ipAddress + ":8080/ticket/" + compraActual.getIdBuy(),
    400, 300, null);

Image qr_image = my_code.getImage();

Image logo = Image.getInstance("logo.png");
logo.scalePercent(40);
document.addHeader("ORIGEN", "VENTRADAS");
document.addTitle("VENTRADAS");
document.add(new Paragraph("VENTRADAS"));
document.add(logo);
document.add(
    new Paragraph("Entrada para el concierto: " + ticketActual.getConcert().getConcertName(), font));
document.add(new Paragraph(
    "Dueño de la entrada: " + usuarioActivo.getLastName() + ", " + usuarioActivo.getFirstName(), font));
document.add(qr_image);
document.add(new Paragraph("Fecha del concierto: " + ticketActual.getConcert().getDate(), font));
Paragraph footer = new Paragraph(
    "Código entrada: " + compraActual.getIdBuy() + " " + (int) Math.floor(Math.random() * 10 * 20),
    font);
footer.setAlignment(Paragraph.ALIGN_BOTTOM);
document.add(footer);
document.close();
```

Bootstrap

Bootstrap es un framework CSS y Javascript diseñado para la creación de interfaces limpias y con un diseño responsive. Además, ofrece un amplio abanico de herramientas y funciones, de manera que los usuarios pueden crear prácticamente cualquier tipo de sitio web haciendo uso de los mismos.

Jquery

JQuery es una librería de JavaScript (JavaScript es un lenguaje de programación muy usado en desarrollo web). Esta librería de código abierto, simplifica la tarea de programar en JavaScript

CSS

Las siglas CSS (Cascading Style Sheets) significan «Hojas de estilo en cascada» y parten de un concepto simple pero muy potente: aplicar estilos (colores, formas, márgenes, etc...) a uno o varios documentos (generalmente documentos HTML, páginas webs) de forma masiva.

MYSQL

MySQL es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales.

5. Bibliografía

<https://stackoverflow.com/>

<https://openwebinars.net/>

<https://spring.io/>

<https://spring.io/projects/spring-boot>

<https://itextpdf.com/en>

<https://www.thymeleaf.org/>

<https://www.baeldung.com/>