

# DOCUMENTACION

#### ÍNDICE

[DOCUMENTACION 2](#_Toc158374741)

[INICIOS 3](#_Toc158374742)

[LOS PRIMEROS PASOS: 3](#_Toc158374743)

[LA CREACIÓN DE LA PÁGINA: 4](#_Toc158374744)

[ASIGNATURAS 5](#_Toc158374745)

[Base de datos 5](#_Toc158374746)

[Git 5](#_Toc158374747)

[Creación de la página: 5](#_Toc158374748)

[Formación y Orientación Laboral 8](#_Toc158374749)

[Java 9](#_Toc158374750)

[Planes para el futuro 13](#_Toc158374751)

[ENTORNOS DE DESARROLLO Y HERRAMIENTAS QUE HEMOS UTILIZADO EN NUESTRO PROYECTO: 14](#_Toc158374752)

## INICIOS

### LOS PRIMEROS PASOS:

En nuestros primeros pasos creando un sistema de reservas para nuestra empresa, hemos deliberado lo primero de todo sobre que se dedicara, ha habido diferentes ideas como, por ejemplo; una web para encargar de modo online comida en una pastelería. Al final nos hemos decantado por hacer una página web de ventas de entradas de un cine.

A continuación, una vez teniendo claro sobre que sería nuestra página web necesitábamos un nombre y un logo para la página web. Asique ha sido lo segundo que hemos decidido grupalmente. Después de diferentes ideas el único nombre que nos ha convencido a todos los integrantes del grupo A, ha sido **MovieServe**, ya que nos ha parecido un nombre único, diferente y captador.

Una vez teniendo claro sobre que sería nuestra página web necesitábamos un logo y un lema para la página web. Por lo que lo siguiente y más costoso ha sido el logo con su lema. Para ello hemos acudido a una página de confianza para poder hacerlo: [https://www.brandcrowd.com](https://www.brandcrowd.com/) ,donde nos mostró diferentes opciones hasta llegar a la que más nos gustó a todos. Después logramos un buen lema que rimara con serve. El resultado final es el siguiente:



## ASIGNATURAS

### Base de datos

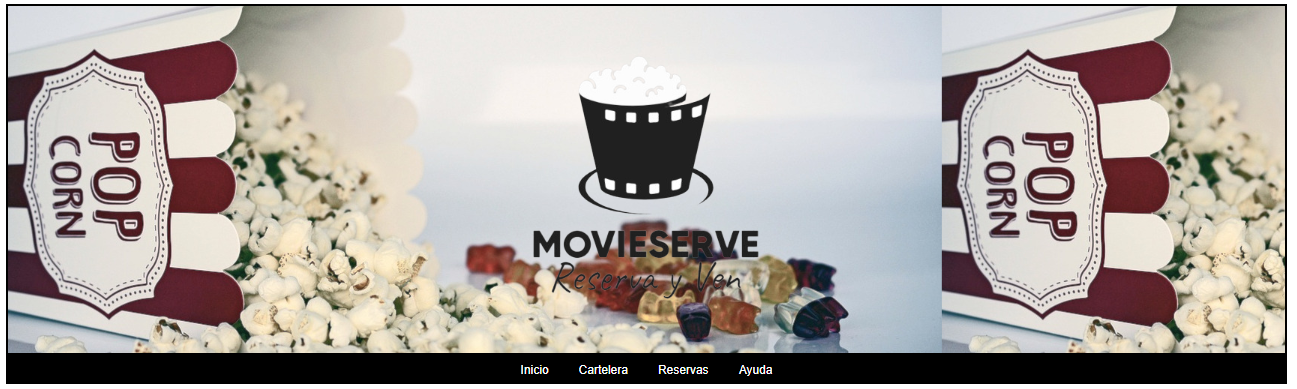
Al comienzo hicimos la base de datos en phpmyadmin teniendo en cuenta la entidad relación que hicimos al principio, creamos todas las entidades (salas, asientos, reservas, clientes y películas) sus respectivos atributos y por ultimo las relaciones necesarias. Tiempo después y ya cuando introducimos las páginas web al servidor EC2 en AWS, nos dimos cuenta que era imprescindible instalar una serie de paquetes en el mismo para así poder tener un servidor con una doble funcionalidad; la idea era que el servidor pudiese albergar servicio web y servicio de base de datos, esos paquetes eran indispensables ya que instalábamos con ellos php, MySQL, phpMyAdmin etc.

Debido a unas complicaciones durante el reto tuvimos que cambiar de phpmyadmin a workbench, el puerto 3306 estaba cerrado así que probamos a cambiarlo al 103, para ello, tuvimos que entrar en algunas de las carpetas de MySQL que se instalaban con el mismo paquete, pero tras varios intentos, no hubo éxito y tuvimos que cambiar el rumbo de idea de la base de datos; con lo cual exportamos la base de datos que habíamos creado en phpmyadmin y la importamos en workbench (otro cliente de MySQL). Después creamos un RDS en AWS, un servido específico de bases de datos, así pues, en vez de tener un único servidor con varios servicios, tuvimos que optar por la opción de servidores separados con un servicio en cada uno de ellos, esto era para crear un vínculo entre la página web que habíamos creado usando AWS con la base de datos de workbench. Por ultimo pusimos el enlace del servidor RDS (24h) tanto en java como en el archivo php y cambiamos el usuario.

### Git

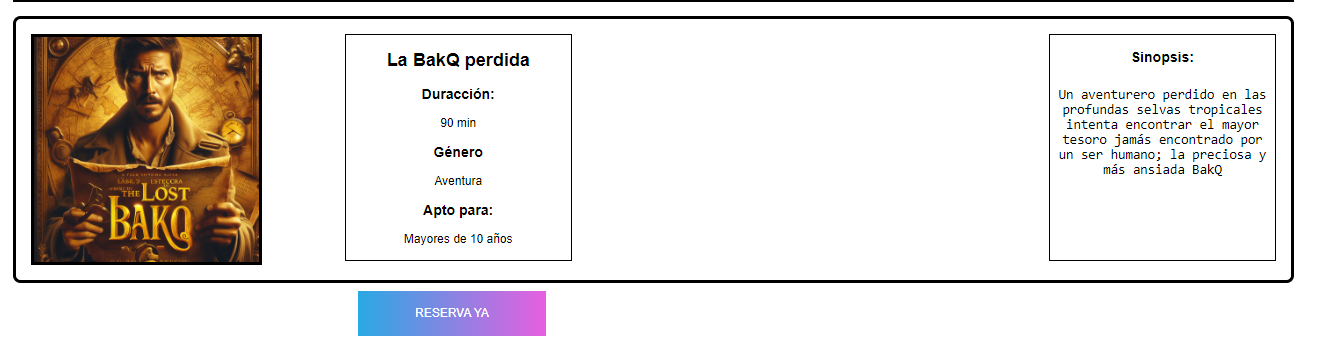
Lo primero que hicimos después de crear el repositorio de GitHub fue compartirlo a todos los integrantes del grupo para de esta manera todos poner tanto subir nuevos cambios como para descargar los cambios de los demás compañeros. Sin embargo 2 compañeros del grupo tuvimos problemas con Git Bash, al cambiarnos de sitio para estar más cerca a la hora de hacer el reto, el Git Bash de esos ordenadores ya estaban configurados con el usuario y Email de otro alumno. No fuimos capaces de encontrar una solución y pedimos ayuda a la profesora la cual tampoco encontró una solución, con lo cual 2 integrantes del grupo no pudieron usar Git Bash. Entonces para mandar nuestros cambios no nos quedó más remedio que usar Email. Es por esto, que solo desde un ordenador hicimos los cambios del proyecto y los fuimos subiendo a GitHub.

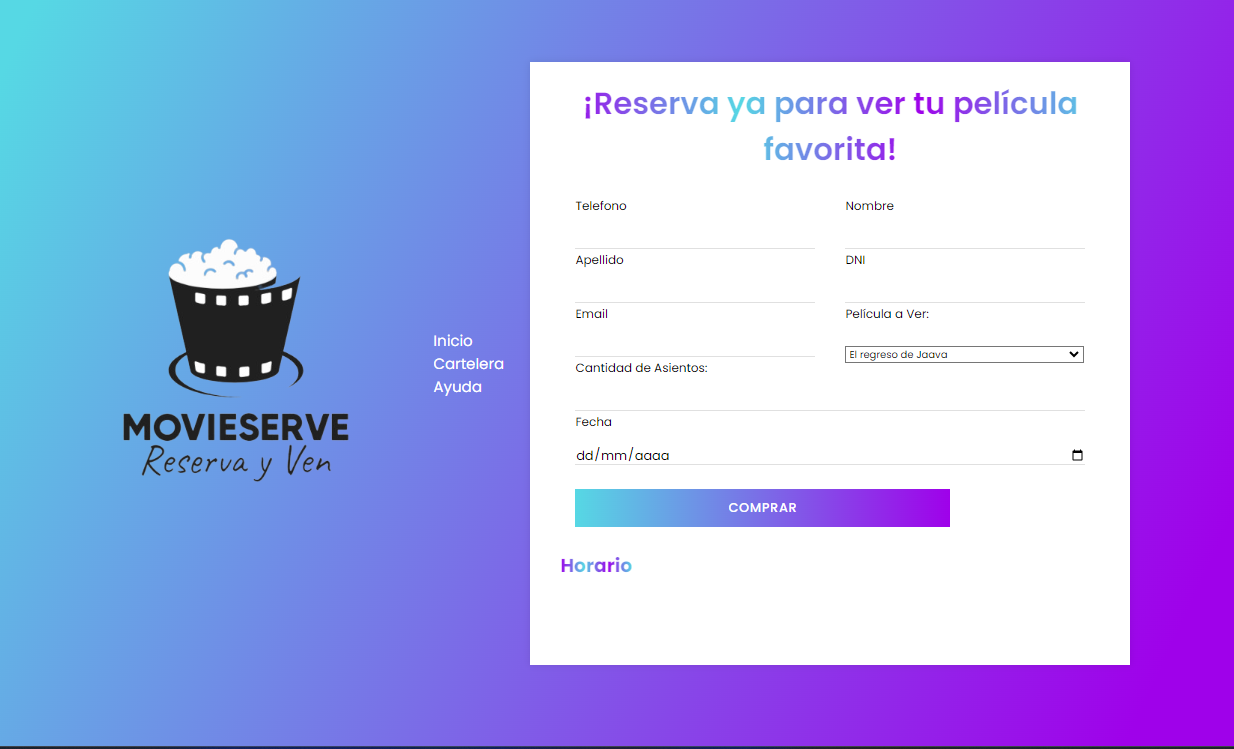
### Creación de la página:

Lo primero fue crear la página de inicio(index.html). La página como título en la parte superior tendría el logo que habíamos creado previamente y en el fondo una foto de un cubo con palomitas, ya que las palomitas son la comida por excelencia de los cines. Justo abajo pegado está el menú, que están los nombres del resto de páginas que usando un “a href” hace posible acceder a cada una de ellas. Debajo del menú con un pequeño espacio de margen se encuentra una breve descripción de que es y cómo funciona MovieServe. A continuación, se encuentran 5 fotos de cosas relacionadas con MovieServe, y por ultimo 2 imágenes hechas por IA y posteriormente editadas en gimp. 

La segunda página que creamos fue la de cartelera, la parte superior de la página, es la misma que la del index.html. Decidimos que habría 12 películas, para darle apariencia de cartelera utilizamos table, tr y td. La cartelera está compuesta por 3 tr y en cada uno de los tr hay 4 td, en cada uno de los td hemos puesto una imagen como portada de cada película. Las portadas las hemos creado utilizando IA y después muchos intentos conseguimos unas buenas, aunque no perfectas portadas para nuestras 12 películas. Y, para terminar, cada película tiene su propia página individual a la cual se accede clicando encima de la portada de cada película. En estas páginas hay un pequeño div el cual contiene a la izquierda del mismo la portada de la película, en medio aparece el nombre, la duración, el género y la recomendación de edad de la película y a la derecha una sinopsis. Y debajo de este div aparece un botón en el cual pone “RESERVA YA” el cual accede a la tercera página llamada reservas.

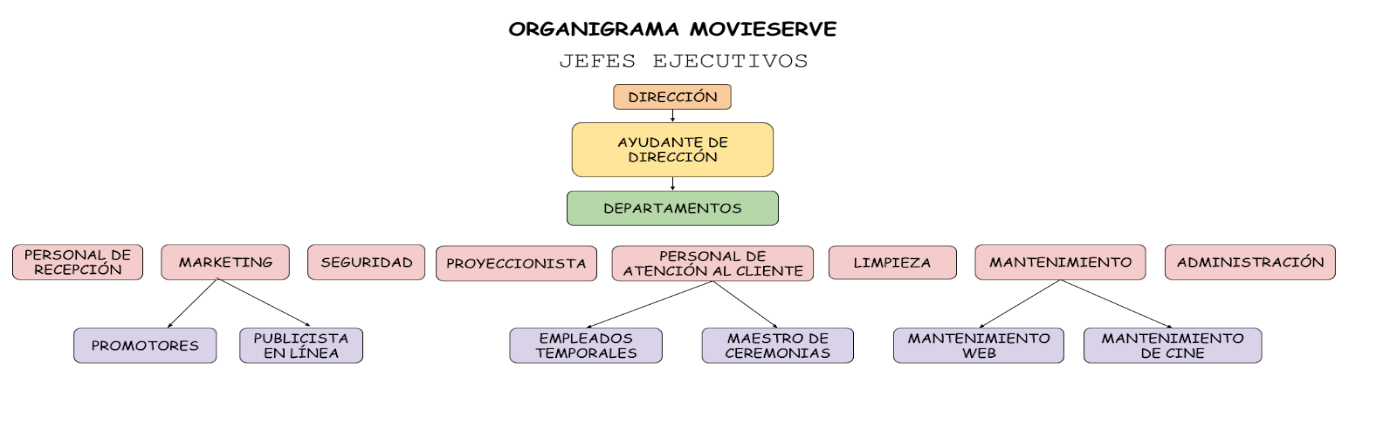
La página reservas como su propio nombre indica es para reservar/comprar una entrada para una película. El diseño de esta página es completamente distinto al de las otras páginas mencionadas anteriormente. En la parte izquierda aparece en grande el logo de MovieServe, en la parte central aparece el nombre del resto de las paginas para poder acceder a ellas y en la derecha está el formulario. El formulario tiene 8 campos a rellenar para reservar/comprar una entrada. Estos campos son: Teléfono, Nombre, Apellido, DNI, Email, Película a ver (usando “option value=” 12 veces conseguimos que aparezca un menú desplegable con todas las películas a elegir.), Cantidad de asientos y Fecha (usando “type=date” nos aparece un calendario en vez de tener que escribir la fecha). Debajo de todo esto aparece un botón en el que pone “COMPRAR” el cual después de rellenar todo el formulario de manera correcta, enviara toda esta información a la base de datos. Y por último debajo del botón “COMPRAR” esta la palabra “Horario” al clicar te lleva a una imagen png donde aparece el horario de ese día.

Para terminar, tenemos la página de ayuda, la parte superior de esta página es la misa que la de inicio y de cartelera, aparte de esto solo hay un div el cual usando otra vez. El div en cuestión contiene 10 preguntas que hemos considerado que pueden ser las más comunes y sus respectivas respuestas. 

### Formación y Orientación Laboral

Lo primero que hicimos de FOL fue la identificación de los diferentes puestos de trabajo. Al final identificamos 9 departamentos distintos: Dirección, administración, personal de recepción, seguridad, limpiadores, marketing, proyeccionista, personal de atención al cliente y mantenimiento. Dentro de dirección estarían los asistentes de dirección, dentro de marketing estarían el publicista en línea y los promotores, en el personal de atención al cliente el maestro de ceremonias y empleados temporales y en mantenimiento estarían el mantenimiento web y el mantenimiento de maquinaria. Y con toda esta información creamos el organigrama de nuestra empresa.

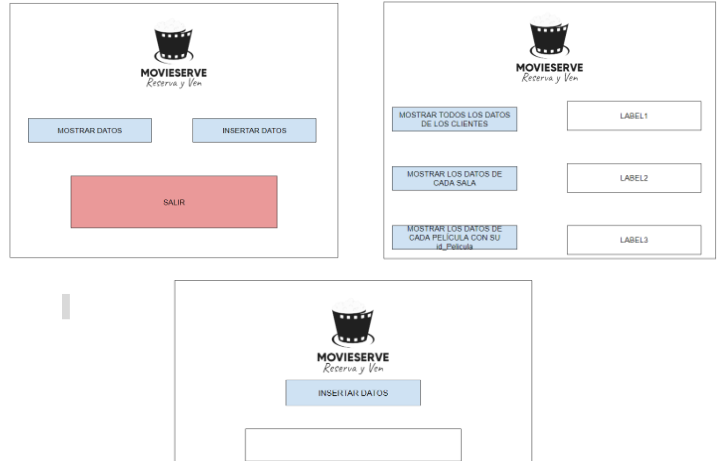


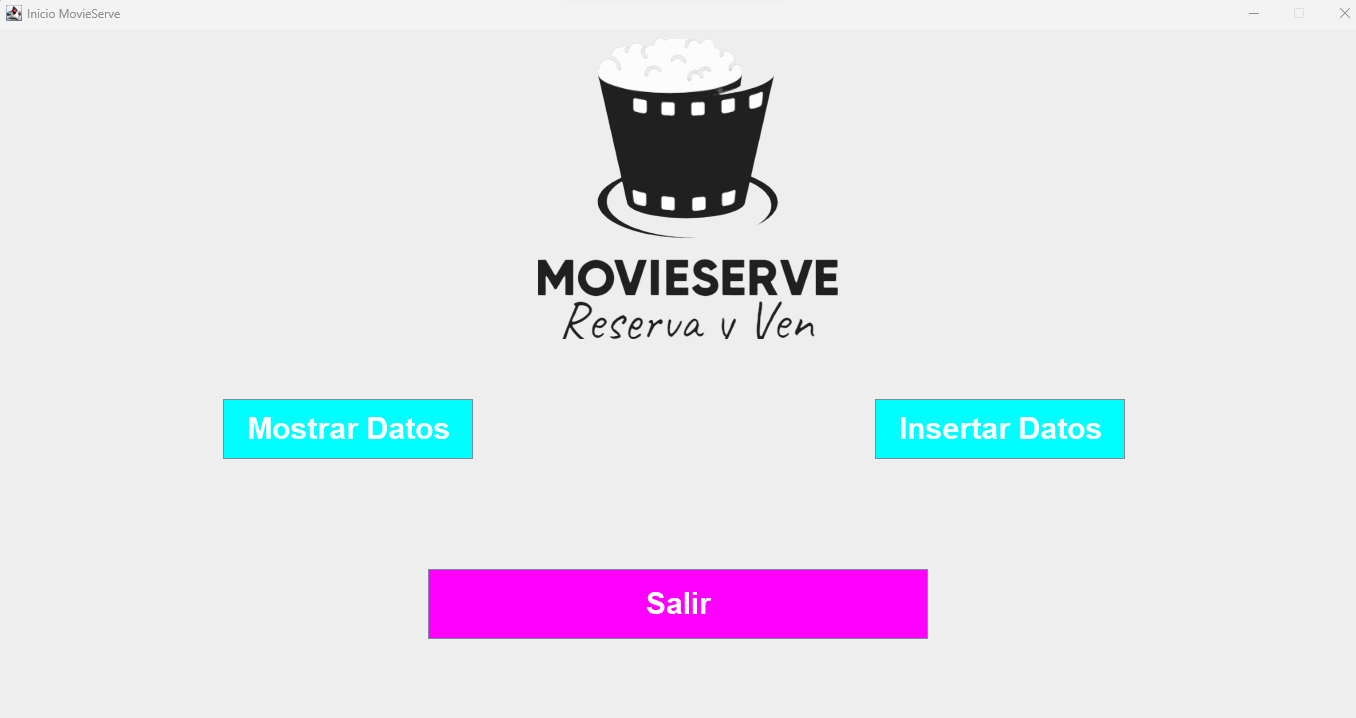
Continuamos haciendo la nómina de un trabajador de nuestra empresa. El ejemplo es Pedro un trabajador de personal de atención, le pusimos un salario mínimo que serían unos 2010’62€ como no tiene hijos el IRPF será del 14%. Después de rellenar toda la tabla y descontar lo necesario a Pedro se le quedan 1722’70€.

Lo siguiente fue identificar los posibles peligros que podrían tener cada uno de los trabajadores. Una vez identificado el peligro había que categorizar su probabilidad y su severidad para determinar la calificación del riesgo y por ultimo una forma de prevenir o solucionar este peligro.

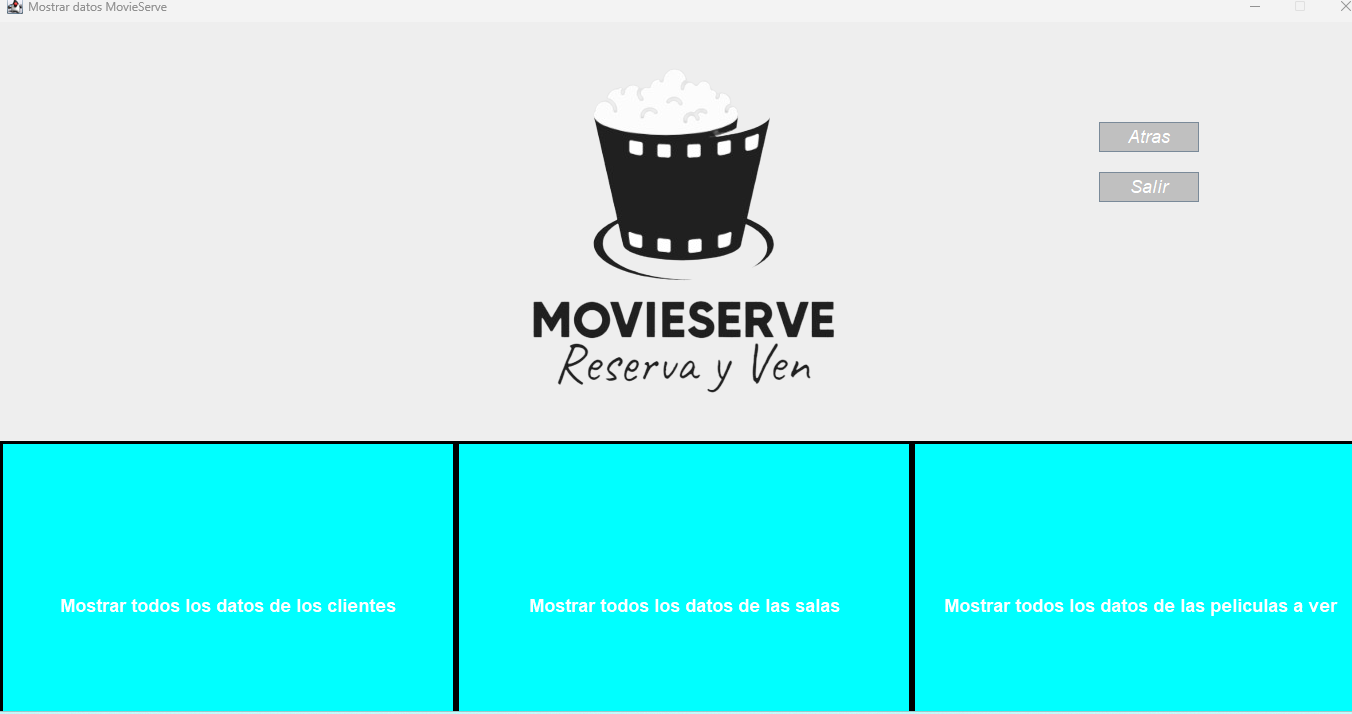
Por ultimo hemos hecho 2 protocolos relacionados con la prevención laboral. Uno de ellos serviría cuando una de nuestras limpiadoras accidentalmente tenga un derrame químico, y tenga contacto con los ojos, con la piel o inhalación. Y después realizamos uno para en el caso de que algún trabajador se sienta acosado psicológicamente.

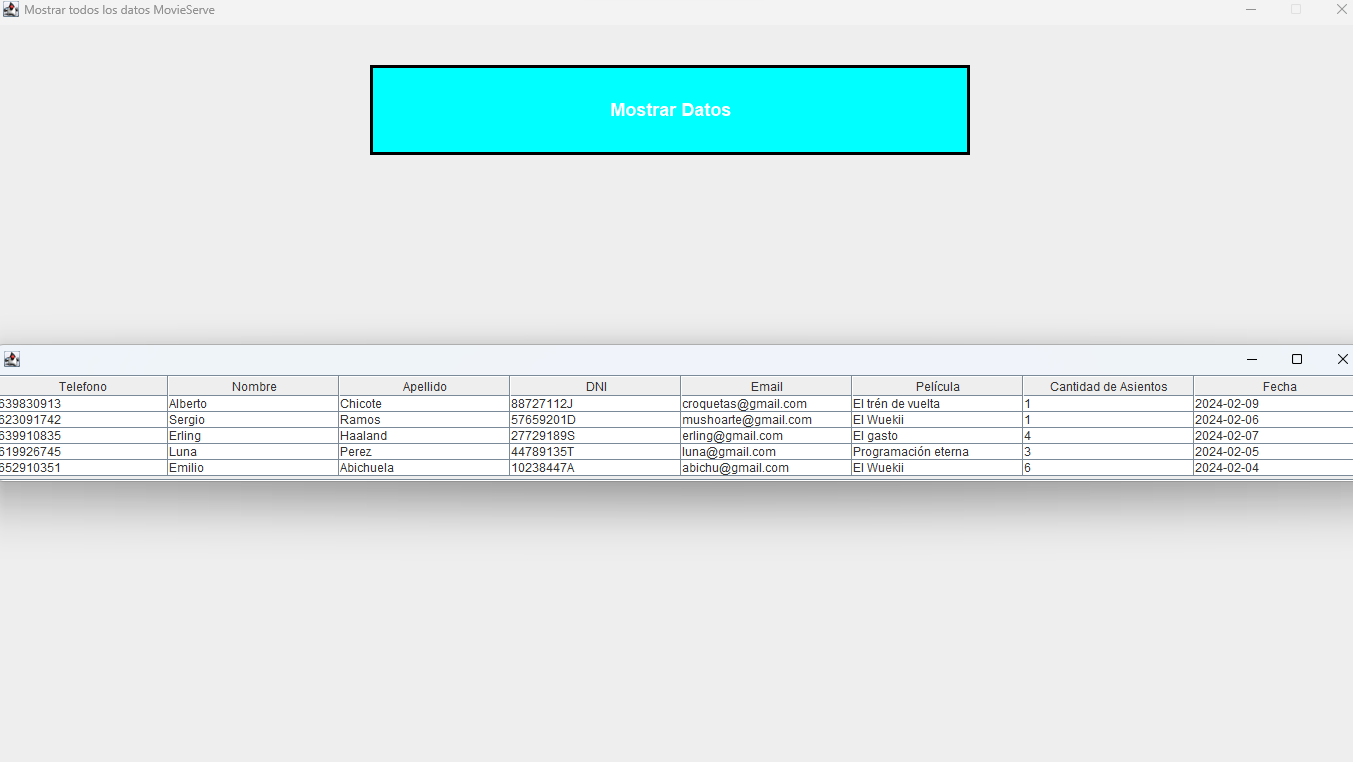
### Java

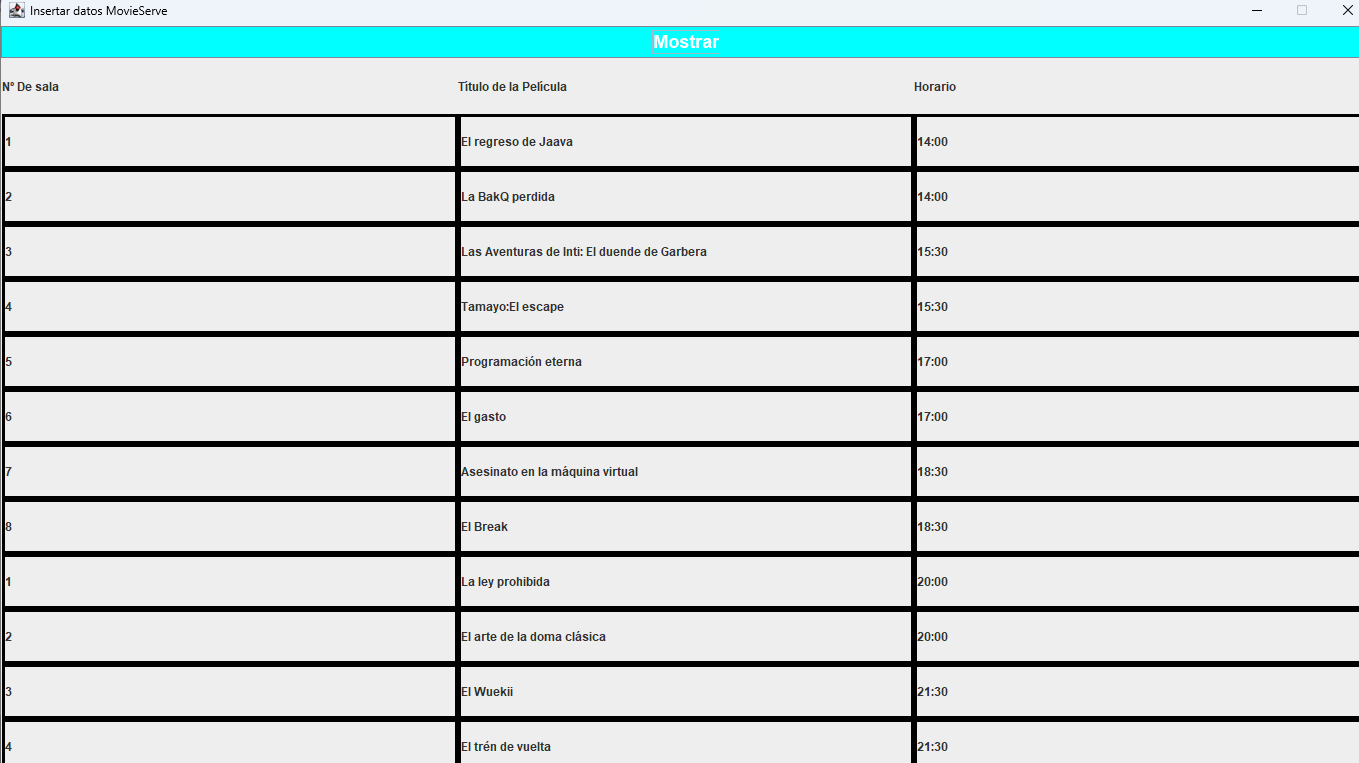
El apartado de Java lo dejamos para el final del proyecto, un par de días antes de empezar con ello, el diseño. He aquí los primeros bocetos de las ventanas gráficas: 

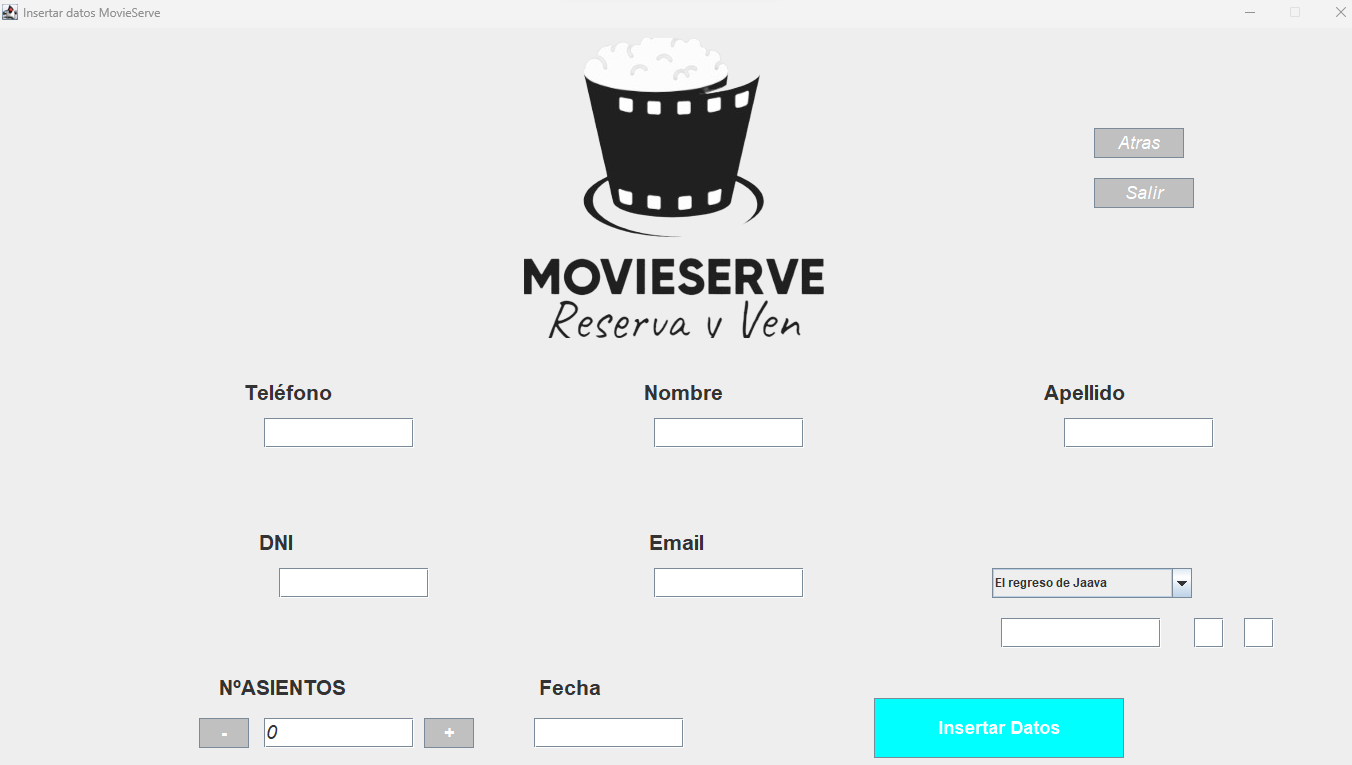
Tras alguna modificación de diseño e implementarlo en código, Lo primero que se muestra es una ventana de inicio con la cantidad de tres botones, uno de ellos para salir y los otros dos redirigen a distintas ventanas. Cabe recalcar que todos los archivos .java parten de la superclase de Jframe

Haciendo énfasis en la primera de las dos ventanas y tras pulsar el botón mostrar, se abre una nueva ventana con otros tres botones a elegir: mostrar todos los datos, mostrar datos de las películas y mostrar datos de las salas relacionadas con las películas y el horario. La intención de esta ventana es darnos la opción de mostrar distintos tipos de datos. Cada opción de los botones antes mencionados, nos abre una ventana distinta a los demás.

Si seleccionamos la primera opción, se mostrará una nueva ventana y en ella solo será visible un único botón con el texto “Mostrar” en la parte superior de la misma. Si presionamos este botón, una tabla será visible para todo aquel que haya realizado la acción antes mencionada. Para realizar esto, hemos tenido que emplear una tabla de encabezados para que se muestre encima que dato se está mostrando, una tabla bidimensional para obtener todos los datos de la base de datos (para ello nos hemos validado de una librería MySQL de java y un código de conexión a nuestra base de datos con nuestras credenciales) y otra tabla para poder ser visualizada. El objetivo principal de esta ventana es mostrar los cinco últimos registros de las reservas realizadas.

Si en cambio, seleccionamos la segunda opción, se nos mostrará una ventana con un botón de Mostrar y tras ser presionado se mostrarán tanto el género, título y duración de cada película. Ha sido indispensable el uso de un panel con el SetLayout como GridLayout para el diseño y la ubicación de todas las etiquetas (en esta ventana hemos preferido la opción de las etiquetas en vez de las tablas), obviamente, el código y la librería necesarias para conectarse a la base de datos están implementados. El objetivo principal de esta ventana, es mostrar todos los datos de las doce distintas películas.

Por otro lado, si hemos decidido presionar el tercer botón, otra ventana se va a mostrar con un botón de Mostrar y tras ser presionado se mostrarán tanto el título de la película, tanto el número de la sala y el horario de dicha película, como en la anterior ventana, hemos empelado varias etiquetas para obtener y visualizar los datos específicos de la base de datos, obviamente, el código y la librería necesarias para conectarse a la base de datos están implementados.Retrocediendo al menú de inicio y por último si seleccionamos aquel botón que nos quedó pendiente por pulsar, una ventana para realizar una reserva a última hora se hará visible. El código de esta ventana no está finalizado en cuanto a funcionalidad respecta, pero el diseño está finalizado. Al igual que hemos mencionado en todas las otras ventanas, tanto la librería para conectar Java con MySQL como el código para conectarlo con la base de datos son indispensables para el funcionamiento de este programa. El objetivo principal del mismo, es que un empleado puede encargarse de realizar una reserva a última hora a cualquier cliente (por ejemplo, si ocurre una llamada telefónica o si el cliente no sabe cuál es la dirección de la página web).

JavaDoc

Tras terminar con todo lo que queríamos hacer en java, hemos realizado la documentación de los códigos .Java para ello, hemos utilizado JavaDoc (es una utilidad de Oracle para la generación de documentación de APIs en formato HTML a partir de código fuente Java).

En cada archivo .Java que hemos empleado en este proyecto, hemos usado esta utilidad, poniendo las funcionalidades y objetivos de cada clase, los objetos añadidos en los métodos constructores, en los ActionListener los códigos para conectarlo a nuestra base de datos y más elementos para que el programa sea funcional y por último en los main hacer que se las ventanas se visualicen o se abran.

## Planes para el futuro

En una reunión del grupo hemos establecido las ideas que tenemos a realizar para un futuro o incluso ideas que han sido descartadas porque en el momento no nos pareció bien hacerlo por falta de tiempo o incluso por no ver necesario realizarlo en ese momento.

Para empezar de momento para hacer la reserva no pedimos ningún tipo de pago, es decir, no pedimos tarjeta de crédito ni nada. Y ya que es un cine de compra de entradas creemos que lo más lógico sería hacer un sistema para hacer que sea posible el pago de la entrada.

Además, nos gustaría que el trabajador desde la ventana de las consultas, lo que hemos realizado desde Java, sea posible insertar datos a nuestra base de datos sin tener que entrar a phpMyadmin. Tenemos realizado el código con la estructura, pero no hemos sido capaces de realizar esa consulta de inserción. Hemos buscado y preguntado en varios sitios y personas, pero no hemos llegado a realizarlo.

En Java también nos gustaría insertar en las tablas de las consultas botones para ver más de 5 consultas. Es decir, los siguientes 5 registros que hay en la base de datos. Ya que ahora mismo solo vemos las 5 últimas consultas y solamente nos quedamos con esos datos. Nos interesa el de siguiente, cuando nos salga eso si nos sale realizaremos el de anterior que solo habría que cambiar unos pocos datos. En realidad, sabemos más o menos como realizarlo porque nos lo han explicado por encima, pero por falta de tiempo y falta de código, no hemos podido realizarlo.

En Java también nos faltaría en alguna de las ventanas insertar botón de navegador, como ir a la anterior o cerrar esa ventana. Lo hemos realizado con las ventanas principales, pero con dos de ellas no ha sido posible hacerlo ya que lo hicimos a través de panel y si lo añadimos a esas pestañas no se ven como nosotros lo deseamos.

## ENTORNOS DE DESARROLLO Y HERRAMIENTAS QUE HEMOS UTILIZADO EN NUESTRO PROYECTO:

Para este proyecto hemos terminado usando bastantes programas relacionados con la conectividad y creación de servidores y bases de datos. Para empezar Drive ha sido necesario para hacer una de las relaciones de bases de datos. Además de tener ahí una carpeta en donde poner la información no oficial para consulta y tener todos acceso a ello. Después para todos los códigos hemos usado tanto git Bash como github. Ahí hemos estado subiendo cada día todas las novedades realizadas, principalmente los htmls de nuestra web. Para tener un seguro por si algo se borraba o se nos estropeaba hemos utilizado un USB en donde estaba todo nuestro proyecto actualizado.

Para la creación de imágenes hemos hecho uso de bing chat (una IA de Edge) y mucha imaginación para crear diferentes películas. También hemos utilizado el codium para editar los códigos html y css.

El servidor ha sido creado con la ayuda de Amazon AWS y subido al internet por WinScp. La base de datos de la empresa ha sido creada desde PHPmyadmin con la ayuda de XAMPP. Después hemos usado el buscador de google para buscar el modo de realizar diferentes acciones e información necesaria, sobre todo para la parte de FOL. Para hacer la nómina por ejemplo hemos usado estos dos links, que han sido facilitados por el profesor

<https://egoitza.gipuzkoa.eus/gao-bog/castell/bog/2023/04/25/c2302818.htm>

<https://www.gipuzkoa.eus/es/web/ogasuna/impuestos/retenciones/tabla-retenciones-rendimientos-trabajo-2024>

Para realizar la documentación hemos usado documentos Word y para presentar nuestro proyecto a los demás hemos utilizado el PowerPoint.