

PROGRAMACIÓN WEB

Antonio Díaz Barbancho
Carlos Marín Rodríguez
Carlos De la Torre Frías (GM2)
Daniel Grande Rubio (GM2)

Usuario de la base de datos: i82dibaa
Contraseña: default

Índice Práctica 3 Programación Web.

1. Introducción.	2
2. Tecnologías Utilizadas.	2
3. Reparto de Tareas.	3
4. Informes de trabajo.	4
Informe semanal 18/11	4
Informe semanal 25/11	4
Informe semanal 02/12	4
Informe semanal 09/12	4
Informe semanal 16/12	6
4. Mapa de Navegación.	7
5. Instrucciones de acceso.	8
6. Bibliografía.	8



1. Introducción.

En esta práctica, se integran y aplican los conocimientos adquiridos a lo largo del curso para desarrollar una aplicación web funcional y profesional que gestione reservas de pistas de baloncesto. Este proyecto combina el diseño y desarrollo de **back-end** y **front-end**, empleando tecnologías clave como **Java**, **J2EE** y **MySQL**. El objetivo es implementar funcionalidades siguiendo las buenas prácticas estudiadas, con una sólida arquitectura basada en el patrón **Modelo-Vista-Controlador (MVC)**.

Para ello, partiremos de la práctica anterior como base la cual ha sido modificada con las sugerencias de la profesora con el fin de entregar un producto más robusto y compacto en su desarrollo y organización. Además, eliminaremos en esta la sección de menús para posteriormente utilizar las tecnologías necesarias para implementar el patrón Modelo Vista Controlador.

2. Tecnologías Utilizadas.

En cuanto a tecnologías utilizadas respecta, hemos seguido los guiones prácticos aportados en moodle para integrar las nuevas funcionalidades.

Para comenzar, nuestra IDE de desarrollo siempre ha sido Eclipse la cual nos ha permitido desarrollar el proyecto de manera directa, simplemente teniendo los conocimientos necesarios para el despliegue y ejecución correcta de nuestro aplicativo.

Desde el primer momento, hemos querido buscar siempre que la implementación del patrón Modelo Vista Controlador, estuviera presente en nuestro código permitiéndonos separar la lógica de negocio, la interfaz de usuario y el control de flujo de la aplicación. De esta manera, nos permite trabajar cada sección de manera robusta y compacta.

Como servidor de aplicaciones hemos utilizado Tomcat v.7.0 tal y como se muestra en los guiones prácticos para alojar nuestro aplicativo web.

De lado del servidor, hemos utilizado los Servlets, los cuales son programas ejecutados de lado del servidor de aplicaciones, son muy eficientes, robustas y escalables gracias a el uso de Java en ellas, más concretamente, haciendo uso de la API estándar de Java (J2EE) para su implementación.

Por otro lado, hemos utilizado Java Server Pages (JSP) para implementar las vistas de nuestro aplicativo permitiéndonos insertar código Java dentro de páginas HTML. A su vez, también se ha utilizado los Javabeans para implementar la separación de funcionalidades para el cliente y para el administrador del aplicativo.

Por último, en cuanto a apariencia respecta, hemos utilizado hojas de estilo css para poder incrementar el apartado visual de nuestro aplicativo, permitiendo así una navegación limpia y clara para el usuario final.

3. Reparto de Tareas.

Con las instrucciones básicas aportadas en clase, decidimos organizar el trabajo de manera que cada integrante asumiera responsabilidades específicas. Esta división de tareas permite optimizar el proceso de desarrollo, garantizando que cada aspecto del proyecto reciba la atención necesaria. La siguiente tabla muestra las partes específicas que abordó cada integrante.

Integrante	Tarea
Antonio Díaz Barbancho	<ul style="list-style-type: none">• Nueva estructuración del proyecto dinámico.• Pantalla de inicio (opciones de acceso y registro).• Desarrollo de acceso y registro de usuario con validación de datos.• Bean de usuario (Customer Bean).• Definir tabla Usuarios en la bbdd.• Testing.
Carlos Marín Rodríguez	<ul style="list-style-type: none">• Restructuración de Gestores y DAOS.• Controlador y Vista JSP para menú Cliente y Administrador.• Opciones de desconectar y modificar usuarios.• Documentación.• JavaDoc completo.• Eliminar reservas.• Testing.
Daniel Grande Rubio	<ul style="list-style-type: none">• Dar de alta pistas.• Dar de alta materiales.• Dar de baja pistas.• Dar de baja materiales.• Listar usuarios.• Listar pistas• Listar materiales.• Asociar materiales a pistas• Modificar estado de pistas• Modificar estado de materiales• Testing
Carlos de la Torre Frías	<ul style="list-style-type: none">• Buscar pistas disponibles según el tipo.• Consultar reservas entre fechas.• Realizar una reserva de pista.• Adquirir bono de reservas y asociarlo a nuevas reservas.• Cancelar reservas con sus restricciones.• Modificar reservas con sus restricciones.• Diseño de las páginas realizadas.• Testing.

Tabla 1: Reparto de Tareas.

Cabe destacar que en caso de que cualquier integrante tuviera dificultades con la implementación de alguna de sus tareas asociadas, el resto del equipo actuaría de manera

proactiva con el fin de solucionar dicho error, dificultad o problema para realizar la correcta implementación.

4. Informes de trabajo.

Nuestros informes de trabajo vienen representados de forma semanal, en la que desglosamos los puntos principales a los que nos dedicamos en dicha semana.

Informe semanal 18/11

En la primera semana de esta nueva práctica, decidimos instalar la versión de Eclipse Enterprise, para poder implementar el servidor de aplicaciones Tomcat de manera directa con nuestro proyecto. Es por ello que esta semana solo la dedicamos a aprender sobre el funcionamiento de Tomcat junto con la ejecución de los códigos de prueba aportados en moodle para comprender el funcionamiento de la lógica a implementar.

Informe semanal 25/11

En esta segunda semana tuvimos la presentación de la última parte de la práctica y la dedicamos a conocer cómo funcionaba la nueva lógica de manera completa, abordando nuevamente los códigos prácticos aportados en moodle con el fin de tener una perspectiva global de las nuevas tecnologías. A su vez nos reunimos para poner en común las dudas que nos surgieron con el fin de resolverlas.

Informe semanal 02/12

Una vez tuvimos todos los puntos de la lógica claros, decidimos comenzar con la implementación de las nuevas tecnologías. Es por ello que Antonio comenzó a reestructurar todo el proyecto, implementando el MVC dentro del webapp, creando la paquetería de servlets y eliminando la lógica de menús la cual utilizaba la interfaz por consola.

A su vez, Antonio también implementó el customerBean para separar la lógica del usuario cliente y usuario administrador añadiendo a su vez las vistas primitivas de usermenu.jsp y adminmenu.jsp para realizar una comprobación global de la nueva estructura.

Por último, los demás integrantes estuvieron resolviendo dudas y comprobando el funcionamiento de la nueva estructura de trabajo.

Informe semanal 09/12

A lo largo de esta semana, Antonio terminó de desarrollar la parte de registro de usuarios y acceso, modificando consigo la tabla Usuarios para implementar el tipo de usuario (Cliente o Administrador) y la contraseña asociada. En este punto, Antonio se dedicó a la parte de comprobación de errores, para sanitizar de manera exhaustiva casuísticas conflictivas (Registrar un correo existente, comprobación de fechas...)

Por otro lado, Carlos Marín implementó el apartado de desconexión del usuario, permitiendo cerrar sesión. A su vez, también implementó la modificación de datos del usuario, permitiendo que si el usuario es un cliente, solamente pueda modificar los datos de su correo electrónico asociado. En caso de que el usuario fuese un administrador, este contaría con la lógica completa de modificar cualquier dato de cualquier usuario, permitiendo así el control de total de usuarios. Por último, Carlos Marín se dedicó a integrar de manera más robusta los paneles de administrador y usuario, permitiendo una sanitización de acceso más exhaustiva ya que por el momento se podía acceder al panel de administrador siendo cliente (endpoint no sanitizado).

Daniel Grande Rubio, durante la semana se dedicó a la implementación de las funciones de creación de pistas y materiales, dichas funciones las rescatamos de la anterior práctica, pero las adaptó para que devolvieran un número dependiendo del error que podía llegar a dar, por ejemplo, que el máximo de jugadores en una pista no fuera un número negativo, que no puedan crearse dos pistas con el mismo nombre, etc.

Por otro lado, esta semana también la dedicó a las funciones de listar pistas, listar materiales y listar usuarios, realizando la misma tarea para el control de errores que con las funciones de crear pistas y materiales. Una vez terminado el controlador de estas funciones y la correcta implementación de las propias funciones, creó un .css para que todas las listas tuvieran una visualización más sencilla y entendible en forma de tabla.

Por último esta semana la dedicó a la correcta implementación de los controladores y los .jsp con el archivo web.xml para que las direcciones aparecieran de forma correcta.

Carlos de la Torre durante esta semana estuvo desarrollando la función de realizar las reservas junto a todas sus restricciones implícitas como que no se pueden realizar reservas en un plazo menor de 24 horas, ni reservar en fechas pasadas, ni adjuntar un tipo de pista incompatible según el tipo de reserva que se requiere, ni adjuntar un número de participantes superior al permitido. Aquí se implementó un script de Javascript que genera un dinamismo a la página dependiendo del tipo de reserva que se requiera. Si se quería solicitar una reserva para adultos o una reserva infantil, solamente había una opción a rellenar según el número de participantes. Mientras que si se elegía la opción de familiar se despliegan dos nuevas opciones para rellenar que se corresponden con el número de niños y número de adultos. También elaboró la funcionalidad de los bonos, que permitían crear nuevos bonos dependiendo del tipo que eligiera el usuario, y realizar una reserva asignando un bono de descuento, con sus restricciones necesarias (como que se deben realizar reservas del tipo de reserva del bono y las ya mencionadas anteriormente de la reserva).

A su vez, como la profesora amplió el plazo de entrega de la práctica, dedicamos el resto de semana a que cada integrante comprobará de manera más completa sus funcionalidades en específico detectando y corrigiendo errores.

Por último, la profesora corrigió la práctica 2 y nos dio una serie de indicaciones como la relación Pista-Material o la implementación de mensajes en los DAOs pertenecientes a la práctica 2, por lo que comenzamos a trabajar en ellas con el fin de corregir implementaciones anteriores

Informe semanal 16/12

A comienzos de esta semana, nos dedicamos a implementar las funcionalidades restantes en nuestro sistema (modificar Estados de pista y material, eliminar reservas, bonos reserva). Es por ello que Antonio se dedicó a ayudar con la implementación de dichas funciones, Carlos Marín se dedicó a reestructurar la correcta implementación de la lógica de los Gestores, sanitizó los DAOs siguiendo las correctas indicaciones de la profesora y buenas prácticas. Además, implementó la funcionalidad de eliminar Reservas de otro integrante y por último, creó los controladores `adminMenuController.java` y `userMenuController.java` ya que hasta el momento tirábamos únicamente de las vistas asociadas sin controlador.

Al inicio de esta semana Daniel Grande Rubio se dió cuenta que no habíamos contemplado la funcionalidad de la eliminación de pistas y materiales por parte del administrador, por lo que es con lo que comenzó, para ello siguió el mismo formato que en las funciones de creación de pistas y materiales, pero cambiando las restricciones, de forma que no se pudiera eliminar una pista o un material.

Una vez terminadas estas funcionalidades, siguió con la función de asociar materiales a pistas, para ello aprovechó la función que ya teníamos de antes, pero la mejoró para aplicar las restricciones necesarias y que fuera de tipo `int` para que devolviera un valor en función del error que pudiese llegar a aparecer. Entre las restricciones se encontraban algunas como la de no poder asociar materiales inexistentes a pistas, no poder asociar materiales a pistas inexistentes o la de no poder asociar un material de interior a una pista que fuera exterior y viceversa.

Tras la correcta integración de esta función con el gestor y el DAO, Daniel comenzó con las funciones de modificación de pistas y materiales, ambas con el mismo formato. En estas funciones el formato era parecido al de creación de pistas y materiales ya que te lanzaba un formulario para rellenar los diferentes campos ya con las opciones a modificar actualizadas. Algunas de las restricciones más importantes en estas funciones eran la de no poder modificar pistas y materiales inexistentes o la de no poder poner números negativos como nuevo tamaño máximo de pista.

Una vez terminadas todas las funciones comenzó a testear todas las funciones creadas en el administrador comprobando que pasaba si se las sometía a errores y corrigiendo en caso de algún error desconocido.

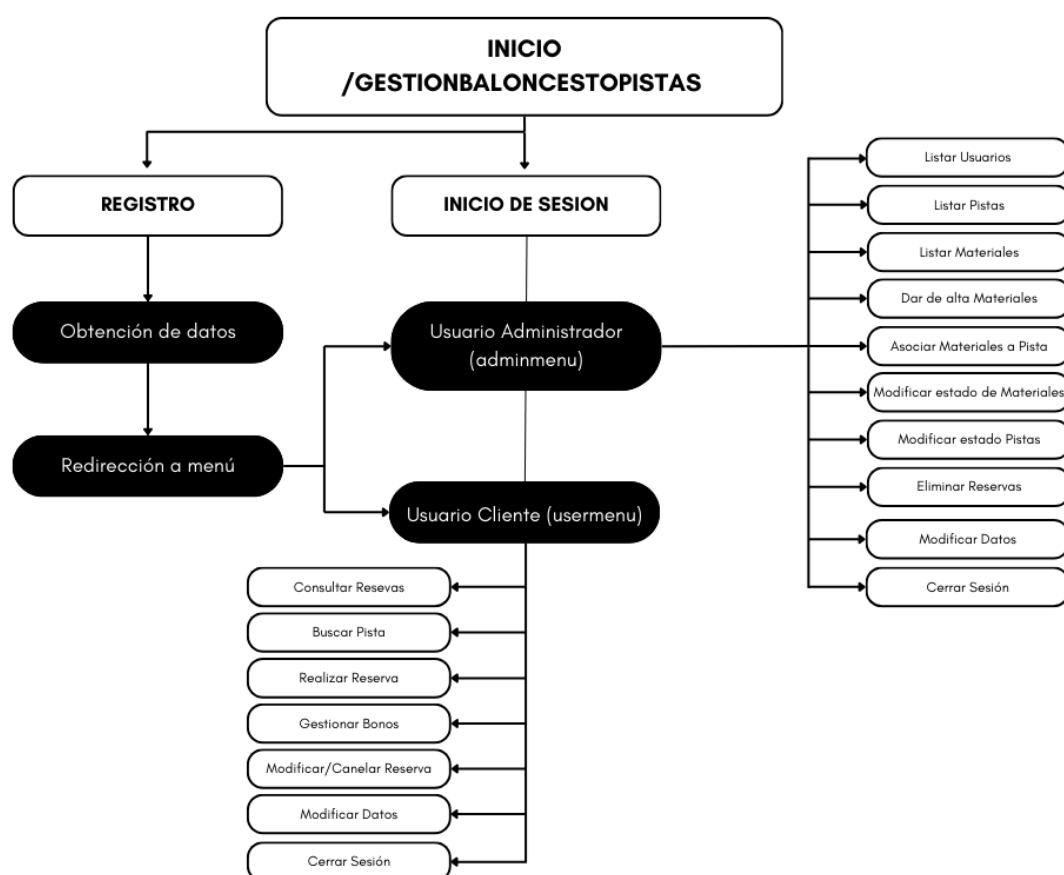
Por otra parte, Carlos de la Torre consiguió terminar la funcionalidad de buscar pistas disponibles según el tipo de pista y una fecha concreta. Asimismo desarrolló la funcionalidad de consultar las reservas que previamente ha solicitado ese usuario en un intervalo de dos fechas. En esta parte realizó el diseño de la página para que cuando una reserva ya ha finalizado, es decir que la fecha de la reserva es pasada, se muestre en un fondo rojo y cuando todavía no ha ocurrido la reserva se muestre en un fondo verde. También en esta opción se añadió que se pudiera ver el nombre de la pista y no su identificador. A su vez durante esta semana continuó con la implementación de la función de modificación de reservas, la cual primero tuvo que hacer una opción para buscar la reserva concreta según la fecha y la pista reservada, y una vez obtenida se modifican atributos como el número de participantes, la duración de la reserva (que modifica correctamente el precio de la reserva), la fecha y hora de la reserva, y la pista de reserva, la cual tiene que

tener la restricción de que debe de ser de un tipo compatible con la reserva ya realizada. Estos campos son opcionales, ya que si no se rellenan se mantienen los valores por defecto que tenían previamente. Se tuvo que añadir ciertas restricciones como que no se supere el límite de participantes de la pista anterior o de la nueva pista que se introduce, que no se realice una modificación de reserva antes de las 24 horas de su realización ni que se modifique la fecha de reserva a una menor de 24 horas respecto a la fecha en que se realice la modificación. Finalmente concluyó la opción de cancelar una reserva la cual tenía la misma interfaz inicial que la funcionalidad de modificar reserva, ya que había que buscar primero la reserva según la fecha y la pista. A estas dos últimas funcionalidades se incluye la restricción de que cada usuario solo puede modificar y cancelar las reservas que él había realizado previamente. Además el diseño de estas dos últimas páginas y la página que origina la redirección a estas dos opciones fueron implementados por Carlos de la Torre.

Para concluir el desarrollo del aplicativo, Carlos Marín desarrolló el javadoc correspondiente del aplicativo ya que con la nueva estructuración del proyecto, se perdió las buenas prácticas de documentación.

Para finalizar, comprobamos la compilación del ejecutable WAR en los servidores de la UCO.

4. Mapa de Navegación.



5. Instrucciones de acceso.

En cuanto a instrucciones de acceso respecta, hemos creado una funcionalidad extra en el registro la cual permite al usuario registrarse como usuario cliente o administrador con el fin de comprobar libremente el correcto funcionamiento de acceso y registro de ambos tipos de cuentas.

NOTA: Somos conscientes de que en un despliegue y salida a producción, esta funcionalidad debe de eliminarse de forma inmediata por problemas obvios de seguridad.

De manera adicional, se aportan una serie de usuarios existentes en la base de datos para el correcto inicio de sesión y funcionamiento del aplicativo.

Correo	Contraseña	Tipo de Usuario
admin@admin.es	admin	administrador
administrador@gmail.es	administrador	administrador
test@test.es	test	cliente
cliente@gmail.com	cliente	cliente

Tabla 1: Credenciales de Usuarios.

6. Bibliografía.



[Práctica 3 - Programación en J2EE/JSP y MVC](#)



[Práctica 3 - Programación de servlets en J2EE](#)



[Ejemplos JSP](#)



[Ejemplos Servlets](#)



[Documentación Apache Tomcat](#)



[Documentación J2EE](#)