|  |  |  |
| --- | --- | --- |
| **Objetivos**   1. Realizar un ejercicio que permita incorporar java web 2. Almacenar información en la memoria temporal 3. Serializar la información de un aplicativo 4. Demostrar la comunicación en la web | **Número de participantes**  participante.jpgparticipante.jpgparticipante.jpg | **Materiales**   * Internet * Eclipse o Netbeans * Apache Tomcat * Maven |
| **Duración de la actividad**  **4:00** |
| **Proyecto:**  Desarrollo de una aplicación web para una exposición canina  **Descripción:**  En una exposición canina se realizan desfiles de ejemplares de diferentes razas. De cada ejemplar se manejan la siguiente información: nombre (el cual debe ser único en toda la exposición), raza, edad y puntos otorgados en la exposición.  El cliente necesita una aplicación que permita manipular la información de los participantes de los desfiles caninos. Además de la información básica, se tiene también la foto de cada perro.  La aplicación debe:  1.- Mostrar la lista de los perros registrados en la exposición, ordenada por raza, puntos o edad  2.- Mostrar la información de un perro específico  3.- Registrar un nuevo perro  4.- Localizar un perro por su nombre  5.- Buscar el perro ganador de la exposición (el que tiene un mayor puntaje asignado)  6.- Buscar el perro con el menor puntaje  7.- Buscar el perro más viejo de todos (con mayor edad).  La información inicial de los perros se obtiene de un archivo de propiedades. Al ejecutar el programa, se cargaesta información  **Autoría**   |  |  | | --- | --- | | Proyecto Curso: | Estructuras de datos I | | Ejercicio: | Aplicación web para una exposición canina | | Autores: | 1. Neider Stiven Burbano Castro | | 1. Cristhian David Padilla Delgado | |  | | Fecha realización: |  |   **Listado de Requerimientos:**   |  |  | | --- | --- | | Nombre | Agregar usuario | | Resumen | Permite agregar un usuario | | Entradas | Nombre, Apellido, Cedula Correo, Contraseña | |  |  | | Resultados | Es agregado el usuario con sus respectivas propiedades | |  | |  |  |  | | --- | --- | | Nombre | Agregar registro PQRS | | Resumen | Permite agregar un registro pqrs | | Entradas | Nombre, Apellido, Cedula, Celular, Correo, Solicitud, Descripción, PDF | |  |  | | Resultados | Se agrega el registro | |  | |  |  |  | | --- | --- | | Nombre | Visualizar registros del usuario | | Resumen | Permite visualizar todos los registros de PQRS que realiza un usuario | | Entradas |  | |  |  | | Resultados | Se muestran los registros del usuario |  |  |  | | --- | --- | | Nombre | Ver estado del registro PQRS | | Resumen | Permite ver el estado en el que se encuentra un registro PQRS realizado por un usuario | | Entradas |  | |  |  | | Resultados | Se mira el estado del registro hecho por el usuario |  |  |  | | --- | --- | | Nombre | Editar un registro PQRS | | Resumen | Permite editar un registro mientras no esté revisado | | Entradas | IdRegistro | |  |  | | Resultados | El registro es modificado |  |  |  | | --- | --- | | Nombre | Eliminar un registro PQRS | | Resumen | Permite eliminar un registro mientras no esté revisado | | Entradas | IdRegistro | |  |  | | Resultados | El registro es eliminado |  |  |  | | --- | --- | | Nombre | Visualizar todos los registros | | Resumen | Permite ver todos los registros hechos por todos los usuarios | | Entradas |  | |  |  | | Resultados | Se miran todos los registros hechos por los usuarios |  |  |  | | --- | --- | | Nombre | Cambiar de estado | | Resumen | Permite cambiar de estado un registro | | Entradas | IdRegistro | |  |  | | Resultados | El registro cambio de estado |  |  |  | | --- | --- | | Nombre | Editar Usuario | | Resumen | Permite editar la información de un usuario | | Entradas | IdUsuario | |  |  | | Resultados | Se edito la información de un usuario | |  | |  |  |  | | --- | --- | | Nombre | Responder PQRS | | Resumen | Permite a un administrador responder con un correo un registro de pqrs realizado por un usuario | | Entradas | IdUsuario, IdRegistro | |  |  | | Resultados | Se envía un correo de respuesta al usuario sobre su PQRS | |  | |  |  |  | | --- | --- | | Nombre | Ordenar Registros | | Resumen | Permite ordenar los registros realizados por un usuario según su tipo | | Entradas | IdRegistro, IdOpcion, IdUsuario | |  |  | | Resultados | Se ordena los registros realizados por un usuario según su tipo | |  | |  |  |  | | --- | --- | | Nombre | Ordenar Registros Totales | | Resumen | Permite ordenar los registros de todos los usuarios según su tipo | | Entradas | IdRegistro, IdOpcion | |  |  | | Resultados | Se ordena todos los registros que hayan realizado todos los usuarios | |  | |  |  |  | | --- | --- | | Nombre | Eliminar Usuario | | Resumen | Permite a un admin eliminar a un usuario registrado | | Entradas | IdUsuario | |  |  | | Resultados | El usuario es eliminado correctamente | |  | |   **Link repositorio en GitHub: https://github.com/CristhianDPadillaD/PQRS**  **Diagrama Relacional**    **Diagrama de clases**    **Requisitos Técnicos:**  1. Utilizar el framework Maven para gestionar las dependencias del proyecto.  2. Implementar la serialización de objetos para almacenar y recuperar la información de la exposición de perros  3. Aplicar conceptos de programación orientada a objetos para modelar los datos y la funcionalidad del reproductor.  4. Crear una interfaz web amigable y responsiva basada en boostrap donde los usuarios puedan ver la exposición de perros  5. Utilizar un ArrayList para almacenar los objetos de manera temporal  6. Realizar un adecuado uso de las excepciones  **Entregables Esperados:**  1. Debe crear un repositorio en GIT para el proyecto  2. Documentación que explique la estructura del proyecto, boceto de la interfaz de usuario  3. Capturas de pantalla que muestren la aplicación en funcionamiento  **Criterios de Evaluación:**  Los proyectos serán evaluados en función de la correcta implementación de los requisitos funcionales y técnicos, la aplicación adecuada de la programación orientada a objetos, así como la usabilidad y apariencia de la interfaz web. Los estudiantes también deben ser capaces de explicar y justificar las decisiones de diseño y tecnología que tomaron durante el desarrollo del proyecto. Con ello:   1. Todos los requisitos funcionales desarrollados 2. Utilización de la programación orientada a objetos 3. Funcionalidad, el programa debe funcionar en su totalidad sin ningún error 4. Usabilidad, el programa debe ser fácil de utilizar, no debe dejar dudas al usuario 5. El archivo en un repositorio GIT de forma adecuada 6. Código documentado en su totalidad 7. Puntualidad en la entrega 8. Esta documentación completa con el prototipo de la interfaz | | |