

EIF209 Programación 4

Proyecto #1

Prof. Georges Alfaro Salazar (coordinador)

Porcentaje de evaluación:

15%

Puntaje total:

100 pts.

DESCRIPCIÓN DEL PROBLEMA.

Los estudiantes del curso de programación deben formar grupos de trabajo, para realizar tareas y proyectos. Para esto, se necesita construir un sitio *web* donde se puedan formar los grupos fácilmente y registrar la información de manera simple y segura.

Los estudiantes podrán ingresar al sitio del curso luego de haberse autenticado debidamente, y ahí podrán formar los diferentes grupos. Los datos sobre las personas que han ingresado al sitio y la manera en que se han integrado los grupos se mantendrán actualizados constantemente.

Se requiere implementar las siguientes funcionalidades:

1. El sitio debe mostrar una página de ingreso (*login*), donde el usuario se identificará con su número de cédula y clave. Todos los usuarios tendrán una clave automática para ingresar por primera vez al sitio. Dicha clave corresponderá con el mismo número de cédula. Después de que el usuario ha ingresado, podrá cambiar su clave de acceso.
2. Ningún usuario puede consultar las páginas del sitio (excepto la página de ingreso) si no se ha autenticado debidamente. En caso de que el usuario intente entrar a una página directamente sin autenticarse, se mostrará un mensaje de error y se redirigirá a la página de ingreso.
3. **Para verificar los datos del usuario en cada página del sitio, se usarán etiquetas JSP** en lugar de código Java.
4. Una vez que el usuario haya ingresado al sitio, contará con al menos tres (3) páginas (formación de grupos, consulta de grupos y usuarios) a las cuales puede acceder a través de un menú adecuado. Todas las páginas usarán un formato consistente entre ellas.
5. En una página se mostrará una lista de todos los usuarios del sistema, indicando cuáles se encuentran activos (ingresados al sistema) y cuáles no. La página se actualizará con una frecuencia de un segundo.
6. Un estudiante puede abrir un grupo nuevo en cualquier momento o unirse a un grupo ya existente, pero en ningún caso un grupo tendrá más de 5 integrantes. Cuando un estudiante abre un nuevo grupo, le dará un nombre distintivo, diferente a los demás. El diseño de la interfaz para crear grupos nuevos o unirse a alguno se deja a criterio de los estudiantes, pero existirá una única página para ejecutar cualquiera de las dos acciones. Un mismo estudiante no puede formar parte de dos grupos al mismo tiempo.
7. Un estudiante puede también abandonar un grupo en cualquier momento. Cuando el último estudiante del grupo sale, este grupo será eliminado. Observe que puede ser cualquier estudiante el que cierre el grupo de esta manera, no sólo el estudiante que lo creó. No existe una opción especial para cerrar el grupo. Simplemente, los integrantes dejarán el grupo hasta que esté vacío.
8. En una página aparte se mostrarán todos los grupos formados, usando una tabla aparte para cada uno, que muestre el nombre del grupo y los datos de los integrantes, arreglados en un formato de **dos columnas**, tal como se muestra más abajo en el diagrama.

Grupo1	
Los invencibles	
Burgos Alfaro	Jonathan
Arce Cubillo	Adriana
Araya Estrada	Alejandro
Gamboa Torres	Oscar

Grupo 2	
Hipster guys	
Arce Fonseca	Joseph
Barquero Artavia	Carolina
Espinoza Solís	Fabián

Grupo 3	
Oompa Loompas	
Barrantes Quirós	Víctor
Fonseca Bolaños	Rebeca
Corella García	Jazmín

Grupo 4	
La selección	
Céspedes Castro	Joan
Abarca Alfaro	Eduardo
Alvarado Villalobos	Daniela
Alemán Barahona	Kenneth
Araya Rosales	Mauricio

Grupo 5	
Los hijos de Batman	
Esquivel Dobles	Bernal
Alpizar Vargas	Mario
Garita Campos	José
Castro Moreira	José

Grupo 6	
El mejor de todos	
Chinchilla Hidalgo	Jorge
Barrantes Zúñiga	Evelyn
Chacon Bolaños	Kevin
Díaz Jiménez	José
González Sánchez	Sofía

Se pueden incluir botones o vínculos para las operaciones de actualización desde esta página, si es conveniente.

Al igual que la lista de usuarios, se usarán peticiones al servidor donde se recibirán los datos en formato JSON, para mantener la lista de grupos constantemente actualizada. Incluya una opción para ver la lista de usuarios junto con la información del grupo de trabajo, ordenada por: grupo de trabajo (1), grupo matriculado (2), número de identificación (3) o apellidos y nombre (4).

Es importante considerar que varios usuarios pueden estar utilizando el sitio de manera simultánea.

El usuario puede salir (*logout*) del sistema en cualquier momento o luego de 5 minutos de inactividad.

OBJETIVOS DEL PROYECTO.

La elaboración del proyecto tiene como objetivo que los estudiantes apliquen los conceptos teóricos y prácticos de programación *web* estudiados en clase. Se pretende familiarizar al estudiante con diferentes temas detallados en los contenidos del curso, según la carta al estudiante.

En particular, se busca:

- Organizar correctamente una aplicación *web* utilizando Java y JSP.
- Estudiar el uso de Java *beans* para el manejo de información en el servidor de una aplicación *web*.
- Implementar adecuadamente la persistencia de datos por medio un motor de base de datos y SQL.
- Utilizar solicitudes asíncronas como técnica para mantener actualizada la información presentada al usuario de una aplicación *web*.
- Emplear correctamente técnicas de control de errores y uso de excepciones.
- Usar técnicas básicas de control de acceso y control de sesiones para una aplicación *web*.

CONSIDERACIONES DE IMPLEMENTACIÓN.

Construya un programa para resolver el problema y efectué las pruebas necesarias para verificar el funcionamiento de los métodos de cada clase. Defina también un procedimiento de prueba para poder verificar el funcionamiento general de la aplicación, según los requerimientos descritos.

No se sugiere ningún tipo de interfaz especial para el programa, salvo las indicaciones anteriores, pero las opciones deben organizarse con un enfoque dirigido a tareas. Es decir, las funciones que se realizan simultáneamente en la operación normal del sistema, deberían parecer juntas en la interfaz.

Deberá necesariamente entregar el proyecto con datos de prueba ya incluidos, para todas las clases solicitadas.

Para hacer la revisión del proyecto, se utilizará una base de datos MySQL con un formato definido. Con el propósito de realizar pruebas y verificar el funcionamiento del programa, se incluirán con este enunciado los *scripts* necesarios para recrear la base de datos.

CRITERIOS DE EVALUACIÓN.

Todo el código debe organizarse correctamente, separando las páginas, hojas de estilo y código JavaScript en archivos separados, y usando paquetes (carpetas) diferentes para el código Java, incluyendo clases regulares, *beans* y *servlets*. Todas las páginas serán documentos JSP.

Criterio de evaluación	Porcentaje
Control de ingreso de usuarios (validación y cambio de claves)	10%
Control de acceso restringido a las páginas (utilizando etiquetas JSP)	10%
Lista de estudiantes activos	10%
Tabla de grupos	5%
Lista ordenada de estudiantes (4 criterios de ordenamiento)	10%
Creación y eliminación de grupos	15%
Adhesión y retiro de grupos (incluyendo control de integrantes)	20%
Definición, organización e implementación de <i>beans</i> de acceso a datos.	15%
Organización general del sitio (formato y usabilidad)	5%
Puntaje total	100%

Todos los archivos de código incluirán el nombre de los autores del proyecto, el título del proyecto y la fecha. Por ejemplo:

```
<%--  
// ingresoUsuario.jsp  
  
//  
// EIF209 - Programación 4 - Proyecto #1  
// Abril 2019  
//  
// Autores:  
//      - 123456789      Marcela Benavides  
//      - 987654321      Adriana Castro  
//      - 135798642      Luis Carlos Montero  
//  
//  
--%>
```

ENTREGA DEL PROYECTO

La fecha de entrega es el día **viernes 26 de abril de 2019**, a más tardar. No se aceptará ningún proyecto después de esa fecha, ni se admitirá la entrega del proyecto por correo electrónico, ni tampoco en dispositivos.

Los programas se entregarán por medio del aula virtual, en el espacio y la fecha designados. Recuerden incluir los comentarios necesarios en el código del programa para describir de manera concisa y completa cada uno de las funciones definidas, así como los elementos de datos utilizados.

En caso de que la aplicación no funcione adecuadamente, efectúe un análisis de los resultados obtenidos, indicando las razones por las cuales la aplicación no trabaja correctamente, y cuáles son las posibles correcciones que se le puedan hacer al programa. Durante la revisión del proyecto, siempre que sea posible, es muy importante poder defender adecuadamente la solución propuesta.

OBSERVACIONES GENERALES

- El trabajo será realizado en grupos de dos o tres personas, como máximo.
- Los trabajos no se copiarán de ninguna llave USB u otro dispositivo en el momento de la entrega, sino que se deben entregar en el formato solicitado en el horario indicado.
- Cualquier trabajo práctico que no sea de elaboración original de los estudiantes (plagio) se calificará con nota 0 (cero) y se procederá como lo indiquen los reglamentos vigentes de la universidad.
- También, si el código entregado no compila correctamente por errores de sintaxis, se calificará con nota de 0 (cero) sin revisiones posteriores.
- Los trabajos que por causa de fuerza mayor se reciban después de la fecha señalada para su entrega, cuando sean excepcionalmente aceptados por el profesor, serán penalizados con un 30% de la nota por cada día de atraso.
- Deberá entregar la carpeta completa del proyecto en NetBeans y cualquier biblioteca adicional que haya utilizado. Si utiliza varios proyectos para la implementación de bibliotecas, asegúrese de incluirlos en la entrega. Puede usar cualquiera de las bibliotecas usadas en los ejemplos de clase y bibliotecas propias, pero no se permite el uso de bibliotecas o *frameworks* de terceros (jQuery, Bootstrap, Angular, Spring, Hibernate o cualquier otra biblioteca similar).
- Durante la primera semana deberá informar al profesor, por medio de un mensaje de correo electrónico, quienes integran el grupo de trabajo.
- El programa deberá seguir una arquitectura de tres capas (Presentación, Lógica y Datos), donde la capa de Presentación deberá ajustarse al patrón Modelo-Vista-Controlador.