

Trabajo final integrador “Bases de Datos Aplicadas”

Versión 0.1 - 13-06-2013

Objetivos

- Que los estudiantes apliquen los conocimientos teóricos aprendidos en clase para desarrollar una aplicación web que almacenará sus datos en una base de datos relacional.
- Que los estudiantes desarrollen los componentes web del lado del servidor y del cliente utilizando nuevas tecnologías y asegurando la independencia de la plataforma.
- Que los estudiantes justifiquen la elección de cada componente y de la arquitectura de software utilizados.
- Que los estudiantes aprendan a investigar para mejorar el conocimiento adquirido y aplicar nuevos conocimientos en pos de crear una aplicación que produzca en el usuario una buena experiencia de uso, que sea performante, fácil de mantener y escalable.
- Que los estudiantes trabajen en grupo a fin de aprender a coordinarse y trabajar en forma mancomunada.

Especificaciones de evaluación y presentación

- Pueden formarse grupos de hasta 3 alumnos.
- Deberá presentarse un documento impreso y en versión electrónica que contendrá entre otros puntos una breve explicación de la aplicación y que problema soluciona, deberá cumplir además con los siguientes requisitos:
 - El documento deberá estar correctamente escrito, utilizando lenguaje técnico, referencias bibliográficas y el contenido debe ser original.
 - El documento tendrá la profundidad suficiente para no crear ambigüedades en el lector.
 - Se podrán utilizar imágenes, capturas de pantalla y gráficos si es necesario.
 - El documento deberá ser estructurado con secciones, títulos, subtítulos, Índice y todo lo que se considere necesario para una mejor comprensión.
 - El documento debe presentarse en formato electrónico, además debe entregarse una carpeta conteniendo el trabajo impreso en formato A4, el mismo debe contener las siguientes secciones:
 1. Carátula : Nombre de la Universidad, nombre de la materia (Bases de Datos Aplicadas), nombre del proyecto, año, nombre del docente (Ing. Mariano, GARCÍA MATTÍO) y nombre de los estudiantes junto a sus correos electrónicos.
 2. Índice.
 3. Objetivos de la aplicación
 4. Alcances y límites
 5. Ámbito de aplicación
 6. Breve reseña del funcionamiento dividida por módulos.
 7. Conclusión
 8. Bibliografía
 9. Anexos: deberán existir al menos 5 anexos:
 - Casos de Uso
 - Diagrama de Clases del modelo principal del sistema
 - Diagrama de Entidad Relación (DER) del modelo de persistencia.
 - Principales librerías y frameworks utilizados con una mínima descripción de cada uno.
 - Detalle de despliegue y herramientas utilizadas: esto se refiere a toda la información necesaria para desplegar la aplicación, por ejemplo, servidor de aplicaciones, archivos de configuración, versión de la máquina virtual,

sistema de gestión de bases de datos, etc. Además se deberán especificar todas las herramientas de desarrollo y test que se han utilizado y sus configuraciones específicas.

- Deberá entregarse además una copia de seguridad de la base de datos, el esquema DDL y el código fuente de la aplicación completa.
- Se evaluará el trabajo en forma grupal e individual haciendo hincapié en el conocimiento individual .

Desarrollo

Crear una aplicación que contenga los siguientes puntos mínimos:

1. Subsistema de autenticación y control de acceso no autorizado.
 - Nombre de usuario, clave (cifrada en la base de datos), fecha de expiración de la clave (puede no expirar), tiempo de expiración de la sesión (-1 tomará el valor del sistema)
 - Deberá crear los métodos para cifrar y descifrar la clave.
 - El sistema deberá tener en cuenta (en el momento del login) la fecha de expiración de la clave si la tiene, en caso afirmativo el usuario deberá poder cambiar su clave.
 - Cada usuario podrá crear una cuenta en el caso de no poseerla.
2. Administración de tareas pendientes:
 - El sistema permitirá administrar tareas por usuario.
 - Una tarea debe contener al menos un texto (debe ser texto rico, en otras palabras, se soportará html) y una serie de tags.
 - Las tareas serán fáciles de buscar (por tags, por texto, etc)
 - Las tareas se podrán marcar como realizadas.
 - Las tareas tendrán como dato extra la fecha de inserción y de última modificación, aunque este dato deberá mostrarse de forma secundaria (por ejemplo como tooltip)
 - Se podrán agregar nuevas tareas
 - Las tareas podrán eliminarse y modificarse.
 - El subsistema de tareas deberá implementarse utilizando Ajax y controles que faciliten el uso.
 - El subsistema de tareas deberá “verse bien”, en otras palabras, deberá poseer un look&feel acorde a las aplicaciones actuales.
 - El subsistema de tareas NO deberá poseer interacciones engorrosas para interactuar con las tareas, esto es: nada de botones editar, agregar, etc, deberá hacerse de forma natural (al estilo gmail)

Recomendaciones

Se recomienda:

- La utilización del patrón DAO
- Cuando el usuario cree la cuenta el sistema deberá enviar un mail para que el usuario confirme su mail, solo confirmando el mail se activará la cuenta. El mail de confirmación tendrá 24 horas de validez.
- La utilización de un framework que facilite la creación de interfaces gráficas
- Que el Look & Feel sea configurable.
- Que la aplicación soporte internacionalización (I18N)
- Que la interfaz de usuario sea agradable, cómoda, fácil de usar, intuitiva y sirva para vender la aplicación
- Una política clara de acceso a datos: transacciones y niveles de aislamiento.