

**ESCUELA POLITECNICA NACIONAL**  
**ESCUELA DE FORMACION DE TECNOLOGOS**  
**ANALISIS DE SISTEMAS INFORMÁTICOS**  
**AMBIENTES NO PROPIETARIOS**  
**Ing. Edwin Salvador**

**EXAMEN – SEGUNDO BIMESTRE**  
**21 de julio de 2015**

**Puede utilizar material de apoyo como las diapositivas de clases o la documentación de las herramientas.**  
**NO se puede utilizar código de los deberes ni de los ejemplos hechos en clase.**

En el repositorio de la sesion16 se encuentra el esqueleto de la aplicación web sobre el cuál se basa este examen. Esta aplicación debe utilizar la plantilla de bootstrap: <http://getbootstrap.com/examples/signin/> para presentar un formulario de inicio de sesión de usuarios. Se debe reemplazar el checkbox que dice “Remember me” por un link que permita el registro de un nuevo usuario. La aplicación debe tener las siguientes funcionalidades y contar con los siguientes archivos:

**PARTE 1: CONFIGURACIÓN INICIAL DEL PROYECTO Y BASE DE DATOS (3 puntos).**

- 1.1. El archivo “.htaccess” deberá ser configurado para que trabaje con el directorio: “C:\xampp\htdocs\anpexamen” y deberá definir la variable de configuración “auto\_prepend\_file” para auto importar el archivo “inc\config.php”.
- 1.2. Los datos para la creación de la base de datos están definidos en el archivo “inc\config.php”. La conexión a la base de datos deberá hacerse dentro de este archivo utilizando las variables definidas:
  - a. \$host = 'localhost';
  - b. \$bdd = 'examen';
  - c. \$usuario = 'anp';
  - d. \$cont = '12345';

Esta base tendrá una sola tabla llamada “usuarios” con los siguientes campos:

- a. id: int(11), PRIMARY\_KEY, AUTO\_INCREMENT
  - b. nombre: varchar(80), NOT\_NULL.
  - c. usuario: varchar(30), NOT\_NULL, UNIQUE.
  - d. contraseña: varchar(30), NO\_NULL.
- 1.3. Dentro del archivo “inc\config.php” también se deben definir las funciones para interactuar con la base de datos:
  - a. `function get_usuario($usuario) ; // verifica si el usuario ya existe en la BDD y devuelve un booleano. Debe utilizarse antes de crear un nuevo usuario en la BDD.`
  - b. `function guardar_usuario($nombre, $usuario, contraseña) ;  
// guarda los datos de un nuevo usuario en la BDD devuelve un booleano (true si el usuario se guardó con éxito).`
    - i. La contraseña del usuario debe guardarse en la BDD encriptada. Para esto debe usar la función md5() de PHP. Pueden ver el uso y documentación aquí: <http://php.net/manual/es/function.md5.php>.
  - c. `function verificar_usuario() ; // verifica que la combinación de usuario y contraseña sean correctos para permitir el inicio de sesión.`

- i. Para verificar la contraseña del usuario se debe utilizar la función md5() anterior sobre el valor enviado en el formulario y se debe verificar que los dos strings md5 coincidan, ya que no es posible descriptar la cadena.

## **PARTE 2: FORMULARIO DE INICIO DE SESIÓN (2 puntos)**

- 2.1. El formulario de inicio de sesión debe contar con los siguientes elementos:
  - a. Usuario (text)
  - b. Contraseña (password)
  - c. Iniciar sesión (botón)
  - d. Registrar nuevo usuario (link elemento <a>)
  - e. Se debe validar que se llenen los campos “usuario” y “contraseña” antes de enviar el formulario al servidor (utilizar el jQuery validate).
- 2.2. Los usuarios deben validarse contra la base de datos, si el usuario o la contraseña no son válidos, debe mostrarse un mensaje con el error.
- 2.3. Si el usuario y contraseña son correctos, el sistema debe redirigir a la página **inicio.php**. En la página de inicio se debe mostrar el mensaje “Bienvenido <NOMBRE\_DE\_USUARIO\_SESION>”.
- 2.4. Se debe añadir la funcionalidad al botón “Cerrar sesión” ubicado en la esquina superior derecha. Al darle clic a este botón se deben destruir las variables de sesión creadas y no será posible volver a la página inicio.php sin antes iniciar otra sesión.

## **PARTE 3: FORMULARIO DE REGISTRO (4 puntos)**

- 3.1. En la página index.php se incluyó un link “Registro nuevo usuario”. Al hacer clic en este link se debe abrir una ventana modal de bootstrap que debe contener el formulario de registro el cual debe contener:
  - a. Nombre (text)
  - b. Usuario (text)
  - c. Contraseña (password)
  - d. Verificar contraseña (password)
  - e. Crear usuario (Botón)
- 3.2. Los campos deben ser validados antes de ser enviados al servidor (utilizar jquery validate)
  - a. Todos son obligatorios
  - b. El usuario debe tener mínimo 5 caracteres
  - c. El usuario no debe estar ya registrado en la base de datos. Esta validación debe realizarse mediante una llamada Ajax a la función get\_usuario()
  - d. Los dos campos de contraseña deben coincidir
- 3.3. Si todos los campos son válidos, estos deben ser enviados al servidor mediante Ajax para guardarlos en la base de datos. Si el usuario es guardado correctamente se debe mostrar un mensaje indicándolo. Cuando el usuario cierre la ventana modal debe ser capaz de iniciar sesión.

## **PARTE 4: REPOSITORIO (1 punto)**

- 4.1. El examen debe ser subido a su repositorio personal dentro del directorio “examenes/examen2”.