**GUIA Laboratorio 2 Evaluado Asíncrono.**

**(Diseño de un CRUD para una app web PHP MVC MYSQL).**

**Distinguidos estudiantes:**

A continuación, se presentan las indicaciones para la realización del laboratorio 2 evaluado **asíncrono**, se les solicita que lean detenidamente la información planteada.

**INFORMACION GENERAL**

* Desarrolle una aplicación web según las indicaciones. Utilice la herramienta Visual Studio Code y utilice las tecnologías PHP + HTML5 + CSS (Bootstrap y propio) + JAVASCRIPT + MySQL.
* Puede usar las imágenes que usted quiera siempre y cuando sean libres de derechos de autor (por ejemplo, puede descargarlas de pixabay).
* El segundo laboratorio evaluado tiene una ponderación de 20%.
* **Será un trabajo individual, la solución presentada no se debe parecer en ningún elemento, ni tabla de la base de datos, ni código a la presentada por otros estudiantes, debe ser un trabajo de su propia autoría y originalidad.**
* El ejercicio deberá desarrollarse de manera formal en cuanto a su contenido y presentación y cumplir con los elementos solicitados en esta guía.
* El objetivo principal del laboratorio es el diseño de una pequeña aplicación web que incluya conexión a una base de datos MySQL y realice las funciones solicitadas en el apartado CLAVE A REALIZAR.
* **Estas funcionalidades debe programarlas en lenguaje PHP nativo y entregar todo el código del sitio Web solicitado, el Script SQL de la base de datos y esta guía llena.**
* Su base de datos según el enunciado debe tener 4 tablas, no debe hacer menos que eso.
* Para todos los elementos de estilo de su sitio web utilice un archivo CSS llamado **CARNET\_L2\_ClaveX.css** y guárdelo en la carpeta CSS también debe utilizar bootstrap la última versión o si usa la extensión de visual studio bootstrap quick snippets puede usar la versión de bootstrap 5.3.2 que es la que esa extensión coloca en el código html.
* Es obligatorio que utilice el patrón de diseño modelo vista controlador **MVC** para diseñar su aplicación web tal como se ha enseñado en las tutorías (No se permite Frameworks de Backend, solo las tecnologías base enseñadas en la materia).
* Desarrollará un sitio Web según las indicaciones dadas más abajo, resuelva con la herramienta Visual Studio Code y programe en lenguaje php+html5+css y utilizando MySQL para la base de datos.
* En este laboratorio aplicará los conocimientos vistos en las video clases, materiales del aula y tutorías.

**EJERCICIO A PROGRAMAR**

**El tema del laboratorio 2 Asíncrono que le toca a cada alumno es el siguiente:**

**CLAVE ASINCRONA**

Desarrollar un CRUD para la gestión de citas médicas en una clínica.

**Base de Datos:**

Crea una base de datos llamada "L2\_ClinicaCARNET" en donde dice carnet pondrá su propio carnet en mayusculas.

Diseña y crea las tablas necesarias para almacenar información sobre pacientes, doctores, especialidades y citas.

Define las relaciones adecuadas entre las tablas usando claves foráneas.

Inserta al menos 5 registros de ejemplo en cada tabla para pruebas.

**Tablas a considerar:**

Tabla para Pacientes:

* Id de paciente: Entero auto incrementable, clave primaria.
* Nombre completo: Texto, tamaño máximo de 100 caracteres.
* Fecha de nacimiento: Campo de tipo fecha.
* DUI: Número entero de 9 dígitos (único por paciente).
* Teléfono: Número entero de 8 dígitos.

Tabla para Doctores:

* Id de doctor: Entero auto incrementable, clave primaria.
* Nombre del doctor: Texto, tamaño máximo de 100 caracteres.
* Especialidad: Clave foránea que referencia a la tabla Especialidades.
* Teléfono: Número entero de 8 dígitos.

Tabla para Especialidades:

* Id de especialidad: Entero auto incrementable, clave primaria.
* Nombre de la especialidad: Texto, tamaño máximo de 50 caracteres (ej. Cardiología, Dermatología, Pediatría).

Tabla para Citas:

* Id de cita: Entero auto incrementable, clave primaria.
* Fecha de la cita: Campo de tipo fecha.
* Hora de la cita: Campo de tipo hora.
* Id de paciente: Clave foránea que referencia a la tabla Pacientes.
* Id de doctor: Clave foránea que referencia a la tabla Doctores.
* Estado de la cita: Texto, tamaño máximo de 20 caracteres (programada, completada, cancelada).

NOTA: Los nombres que le pondrá a cada tabla debe pensarlos usted no serán tal y exactamente como se menciona arriba para evita copia.

**Funcionalidades de la Aplicación:**

1. **Registro de Pacientes**:
   * Implementa un formulario que permita registrar pacientes con su información completa.
   * Valida que el DUI del paciente sea único en la base de datos.
2. **Registro de Doctores y Especialidades**:
   * Desarrolla una vista para registrar doctores junto con su especialidad.
   * La especialidad debe seleccionarse de una lista desplegable con los registros de la tabla Especialidades.
   * Crear una vista que permita visualizar el listado de doctores y su especialidad.
3. **Registro de Citas**:
   * Crea una funcionalidad para agendar citas seleccionando al paciente, doctor y el horario correspondiente.
   * Valida que no se registren citas duplicadas (mismo doctor, paciente, fecha y hora).
   * Al agendar una cita, el estado debe ser "programada" por defecto ese estado se debe poder modificar a “finalizada o perdida”.
   * Crear una vista que permita visualizar el listado de citas con estado “programada” registradas en el sistema ordenadas por la fecha más antigua a la mas reciente.
4. **Gestión del Estado de las Citas**:
   * Desarrolla un módulo que permita cambiar el estado de una cita a "completada" o "cancelada" según sea necesario.
   * Las citas canceladas deben poder re agendarse si es requerido.

**ASIGNACION DE FUNCIONALIDADES DEL SITIO WEB A PROGRAMAR**

La base de datos que creará debe tener el nombre solicitado en cada clave, para cada funcionalidad cada alumno debe diseñar la vista (formulario o formularios que necesite, tablas html, etc.), el modelo (para conectarse a la base de datos y para ejecutar las consultas respectivas a las funcionalidades que se le soliciten), el controlador (para poder recibir los datos desde las vistas y ocuparlos en combinación con el modelo para realizar la función solicitada ya sea inserción, consulta, actualización o eliminación según cual funcionalidades que les haya tocado en su clave de laboratorio).

CREAR A PARTIR DE AQUÍ SU SOLUCION

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DATOS DEL ALUMNO** | | | | |
| **CARNET** | **Nombres** | **Apellidos** | **Grupo teórico** | **Fecha realización** |
|  |  |  |  |  |
| **Nombre del tutor** | |  | | |

**ETAPA1 Creación de la Base de datos**:

Creación de la base de datos: llame a la base de datos **L2\_ClinicaCARNET,** cree dentro de ella las tablas que diseñará para su ejercicio a entregar **el nombre de las tablas a diseñar debe pensarlo usted**, **así como los campos que contendrá** (en la clave se detalla cuantos campos tiene cada tabla) (los nombres de cada campo debe pensarlos usted tomando en cuenta que no deben repetirse e ir sin espacios en blanco, sin tildes, sin caracteres especiales y tomar como base los campos que se mencionan de cada tabla en la clave) (entregar script SQL, llámelo **CARNET-ClaveX.sql**). para comenzar con su diseño llene los siguientes datos:

Breve explicación de su diseño a elaborar.

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre de la Tabla a crear** | **Campos de la tabla a crear** | **Comentarios de diseño** | **Funcionalidades a desarrollar en su aplicación Web** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Datos que debe tener cada Tabla diseñada (Subrayar la clave primaria y ponerla como primer campo), las claves foráneas deben ser los últimos campos que digite:

Nombre tabla: **PONERACANOMBRETABLADELABASEDEDATOS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre de Campo** | **Tipo de dato** | **Descripción** | **Es obligatorio (Si/No)** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Nombre tabla: **PONERACANOMBRETABLADELABASEDEDATOS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre de Campo** | **Tipo de dato** | **Descripción** | **Es obligatorio (Si/No)** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Nombre tabla: **PONERACANOMBRETABLADELABASEDEDATOS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre de Campo** | **Tipo de dato** | **Descripción** | **Es obligatorio (Si/No)** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Nombre tabla: **PONERACANOMBRETABLADELABASEDEDATOS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre de Campo** | **Tipo de dato** | **Descripción** | **Es obligatorio (Si/No)** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Construcción de script SQL para usarlo con su aplicación web:**

Con esos datos que escribió en las tablas de arriba construya el archivo **CARNET-ClaveX.sql** con el script SQL para su laboratorio 2 asíncrono de acuerdo al diseño de campos que ha pensado (según las tablas de cada clave), luego de creado debe montarlo en MySQL de su servidor Web local para después comenzar a programar su aplicación Web MVC con PHP.

No se olvide de mencionarle a su tutor el nombre de usuario y clave usada para conectarse a su base de datos y en donde editarla en su aplicación ya que su tutor debe modificar dichos datos para montar su script SQL en su propia computadora y poder calificarles su laboratorio 2.

Aquí se les deja un ejemplo de un script SQL para crear una tabla para que tomen ideas de su sintaxis:

CREATE DATABASE BaseDatosEjemplo;

USE BaseDatosEjemplo;

CREATE TABLE Personas (

ID int(10) AUTO\_INCREMENT NOT NULL,

LastName varchar(50) NOT NULL,

FirstName varchar(50) NOT NULL,

RegDate DATE NOT NULL,

Age int(3) NOT NULL,

Sueldo DECIMAL(10, 2) NULL,

PRIMARY KEY (ID)

);

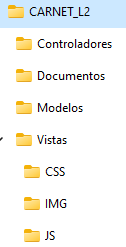
**ETAPA2 Construcción de la aplicación MVC**:

Una vez creada y montada la tabla de la base de datos en MySQL construya su aplicación Web MVC siguiendo las indicaciones dadas a continuación y de acuerdo a la clave que le tocó trabajar.

|  |  |
| --- | --- |
| Credenciales que se usó en su base de datos | |
| Usuario de la base de datos |  |
| Contraseña usada |  |

Indicaciones sobre el CRUD a crear según la clave:

* Debe crear del modelo mediante el cual se conectarán a la base de datos y realizarán las operaciones solicitadas inserción, selección, actualización y eliminación en la base de datos según se hayan solicitado.
* Debe crear y programar las funciones del controlador el cual se encargará de pasar los datos de la vista hacia el modelo y viceversa para realizar las operaciones solicitadas por el CRUD.
* Debe crear los formularios y tablas php-html de las vistas a crear cada funcionalidad es una página web, también **no se olvide crear una página principal** desde donde se pueda acceder a todas las demás funciones ya sea por botones o por un menú, utilice **bootstrap para su aspecto visual (La ultima versión)** y CSS personalizado por ejemplo para la imagen de fondo, tipos de letras, márgenes, etc. el diseño debe ser original de su propia autoría y tener los elementos organizados según la siguiente maqueta.
* La aplicación Web debe respetar la estructura de carpetas:



**La carpeta raíz de trabajo debe ser CARNET\_L2 en donde vea la palabra carnet debe sustituirlo por su propio carnet en mayúsculas, en la carpeta Documentos pondrá este archivo de Word, el archivo del script SQL, el resto de carpetas es para crear la aplicación usando el patrón MVC.**

**Solo se pueden usar las tecnologías vistas en la materia herramienta Visual Studio Code y utilice las tecnologías PHP + HTML5 + CSS (Bootstrap ultima versión y propio) + JAVASCRIPT + MySQL.**

**ENTREGA**:

Al final debe subir al aula virtual un archivo comprimido con el código fuente, carpetas y archivos de su sitio Web diseñado, el script SQL de la base de datos con el nombre de la base de datos y tablas que usó para su app web en el archivo con extensión .sql y también incluya esta guía con sus datos personales llenos y el diseño en las tablas de Word llenas que se piden.

**NOTA: Debe trabajar estrictamente de manera individual ya que el laboratorio 2 es asíncrono, cualquier parecido o copia será calificado con nota 0.0**

Cuando termine su laboratorio 2 comprima la carpeta con el contenido de su laboratorio terminado y súbalo al enlace **Laboratorio 2 en la pestaña evaluaciones**, el archivo se debe llamar **CARNET\_L2.ZIP** en donde la palabra carnet se refiere a su propio carnet en mayúsculas.

Dentro del archivo comprimido debe ir este mismo archivo de Word con los primeros datos de diseño de la base de datos y con los demás datos solicitados llenados en cada tabla (**CARNET\_Lab2Asincrono2024TPI.docx**), el archivo del script SQL **CARNET-ClaveX.sql** y la estructura de archivos y carpetas de la aplicación Web ya terminada.

Fecha de asignación: **lunes 4 de noviembre de 2024**.

Fecha de entrega: **martes 5 de noviembre de 2024 11:50 pm**

*“Huye de los elogios, pero trata de merecerlos”*