

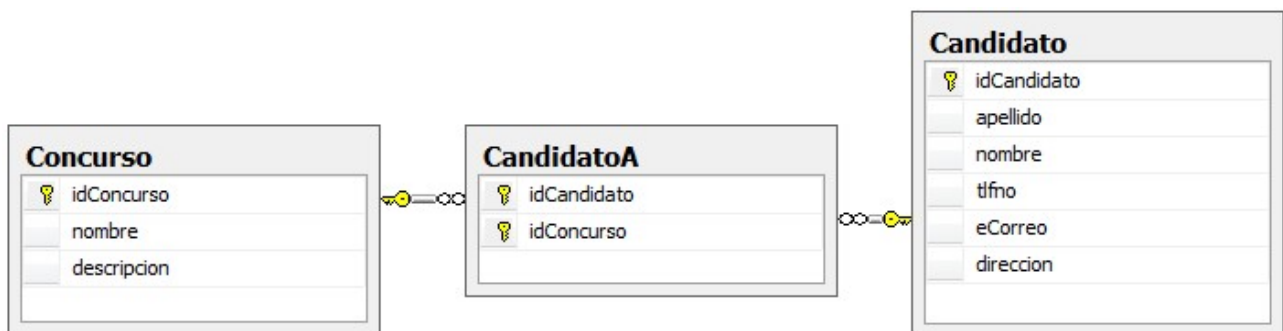
Trabajo de recuperación

De cara a recuperar el módulo debéis presentar antes de las 00 horas del día 3 de abril, un trabajo/proyecto sobre el tema que os planteo a continuación.

CONCURSOS DE TELEVISIÓN

Un operador de televisión, especializado en concursos, gestiona una aplicación web que los telespectadores utilizan para inscribirse como candidatos a participar en sus concursos.

En su SGBD SQL Server existe una base de datos llamada ConcursoTv con la estructura que se detalla en el diagrama que sigue:



Se pide:

1. Crear la base de datos concursoTV
2. Poblar la tabla Concurso con los datos que siguen:

Nombre del concurso	Descripción
Salto a la fama	En busca de jóvenes talentos que desean demostrar al gran público sus dotes particulares
Saber y concursar	¿Qué tan extensos son tus conocimientos culturales? El reto de contestar tanto a preguntas de actualidad como de hechos pasados
Queremos humor	Un desafío semanal en busca del mejor humorista que nos haga reír y olvidar las penas de la semana
Mira quién se acerca	¿Sabrías definir a esa persona que se acerca por la calle? ¿A qué se dedica? ¿Cuáles son sus gustos y aficiones? Participa y demuéstranoslo
Futbol, más futbol por favor	El futbol desde sus comienzos. Todas las competiciones. Todos los marcadores. Todos los astros. Acepta el reto);

3. Realizar una ventana de formulario que recoja los datos del solicitante a concursar. La ventana debe mostrar los concursos que se llevan a cabo actualmente (con su descripción). Cada aspirante puede inscribirse en más de un concurso. La ventana incluirá tantos campos como columnas tiene la tabla Candidato (excepto idCandidato) y los botones Enviar y Cancelar
4. La aplicación debe velar por la completitud de los datos enviados, sanearlos de alguna manera (para evitar ataques) y guardar la información del aspirante en las tablas Candidato (datos del candidato) y CandidatoA (concursos en los que desea participar. Si faltan datos, la aplicación deberá advertir al candidato de este hecho e invitarle a completar los datos que faltan
5. La página index.php debe mostrar una interfaz como la que sigue o mejor:



Notas:

- La solución **se debe** implementar mediante el patrón MVC en la estructura de directorios que acompaña a este documento, y crear las clases oportunas en el modelo (bajo los directorios `Modelo` y `Repositorio`) y en el controlador (bajo el directorio `Controlador`), así como las vistas que consideréis oportunas (bajo el directorio `plantillas`)
- La tabla `CandidatoA` es una tabla de relación.
- El resultado del trabajo/proyecto **debe** llevar a cabo, **correctamente**, cada uno de los puntos que se piden y almacenar la información recibida en las tablas que correspondan mediante una **transacción correcta**
- En el trabajo/proyecto **se deben** emplear Namespaces, y las diferentes clases que se creen bajo el directorio `fuentes` deben declarar el namespace al que pertenecen (recordar que la clase `defaultController.inc` ya existe y carece de namespace)
- Si implementáis la función `spl_autoload_register` para autocargar las clases a partir de los namespaces, y que os evite tener que especificar `include/require` para realizar la carga de las mismas, mejoraréis la nota en hasta un 1 punto

Qué debéis enviar:

1. Guion SQL con el que se crean las tablas de la base de datos `concursoTV`
2. Guion SQL con el que se puebla la tabla `Concurso`
3. El directorio `ConcursosTV` con los archivos de la solución y sin modificar la estructura de directorios y archivos base que os envié (agregar los archivos de solución que necesitéis)