**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN**

**DESARROLLO DE APLICACIONES WEB**

**SEGUNDA EVALUACIÓN - I TÉRMINO 2017**

***Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Matrícula: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

**TEMA 1 (20 puntos)**

Seleccione la respuesta correcta:

1. Si nosotros necesitamos obtener un recurso a través de un API REST, lo más apropiado sería utilizar:
   1. POST
   2. GET
   3. DELETE
2. El archivo WSDL sirve para especificar:
   1. Los métodos que posee un Web Service.
   2. Todos los web services que posee una compañía.
   3. La forma de conectarse al Web Service.
3. El patrón MVC sirve para:
   1. Crear aplicaciones web que utilizan AJAX
   2. Separar las responsabilidades de cada componente en una aplicación web
   3. Crear una conexión con la base de datos
4. Cuando se dice que el protocolo HTTP es stateless, quiere decir que:
   1. Tiene pocos estados web
   2. Cada requerimiento HTTP es independiente
   3. Permite definir métodos como GET, POST, PUT, DELETE
5. Un servidor puede responder a un requerimiento:
   1. Solo con JSON
   2. Con JSON, XML, o texto plano
   3. Con lo que se especifique en el Content-Type
6. El significado de URI es:
   1. Uniform Resource Identifier
   2. Unique Resource Identifier
   3. Uniform Request Identifier
7. Para manejar información de una manera más segura en una aplicación Web, se recomienda utilizar:
   1. Cookies
   2. Sesiones
   3. Campos ocultos

**TEMA 2 (30 PUNTOS)**

Conteste según corresponda:

1. Explique el uso de cada uno de los Métodos HTTP: POST, PUT, DELETE
2. Explique cómo se crea una sesión, donde se almacena una sesión (cliente o servidor), y ¿cómo identifica el servidor a un cliente?
3. ¿En qué consiste el patrón de arquitectura MVC?
4. ¿Qué es SOAP y para qué sirve el archivo WSDL?
5. ¿Cuál es la utilidad de un ORM?

**TEMA 3 (50 puntos)**

Los administradores de “La Bota” han decidido implementar una aplicación web para la compra de tickets de sus obras. Actualmente, los eventos son llevados en un registro no sistematizado por parte del personal del teatro. Para la realización de una obra es necesario conocer el nombre, el género (drama, comedia, etc.), nombre del organizador, una descripción, y una lista de horarios. Los horarios se detallan de acuerdo a la sala, a la fecha y a la hora, por ejemplo: viernes a las 10:00pm en la Sala A o domingos a las 11:30pm en la Sala B. Los administradores de “La Bota” desean llevar un registro de los actores; de los que se necesita saber el nombre, dirección, teléfono y correos electrónicos. Para la compra de un ticket es necesario que un cliente provea de su número de cedula, nombres completos, celular, correo electrónico, y deberá seleccionar un asiento (se le presenta una distribución de asientos del teatro).

Se le solicita:

1. Definir todos los modelos necesarios para su aplicación (Entidades)
2. Diseñar un API para el módulo de \_\_\_\_\_\_\_\_\_\_\_\_\_\_ (crear, obtener, borrar, actualizar) donde se especificará:
   1. La acción a realizar
   2. Método HTTP a utilizar
   3. Ruta o endpoint
   4. Parámetros a enviar
   5. Respuesta JSON
3. Se le solicita implementar la compra de un ticket por parte de un cliente.
   1. La ruta o endpoint de reserva de tickets será similar a la siguiente: “/obra/245/comprar”, donde “245” representa al id del evento.
   2. El método HTTP a utilizar.
   3. El Id del evento que está como parámetro en la URL.
   4. El Id del cliente que está en sesión.
   5. El horario recibido como parámetro.
   6. El asiento recibido como parámetro.