**Texto, Logotipo

Descripción generada automáticamente**

**UNIVERSIDAD PRIVADA DE TACNA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas**

**Proyecto Sistema Web "EventifyMe" para la gestión y reserva de eventos**

Curso: *Programación Web II*

*Docente: Mag. Patrick Jose Cuadros Quiroga*

Integrantes:

***Medina López, Marcelo José (2020066917)***

***Zevallos Purca, Justin Zinedine (2020066924)***

**Tacna – Perú**

***2025***

**Estándar de Programación para Sistema de Eventos “EventifyMe”**

**1. Estructura de Proyecto**

Mantén una estructura de proyecto clara y organizada:

* **Areas**: Define áreas específicas como Admin y Cliente con sus respectivos controladores y vistas.
* **Controllers**: Controladores deben estar agrupados por funcionalidades específicas.
* **Models**: Modelos deben estar en una carpeta dedicada.
* **Views**: Vistas deben estar organizadas en carpetas que correspondan a sus controladores.

Texto

Descripción generada automáticamente

**2. Nomenclatura**

* **Controladores**: Los nombres de los controladores deben seguir la convención NombreController.
* **Acciones en Controladores**: Las acciones deben ser descriptivas, como Index, DetalleEvento, Comentarios.
* **Modelos**: Los nombres de los modelos deben ser sustantivos en singular, por ejemplo, Evento, Reserva.

Texto

Descripción generada automáticamente

**3. Uso de MVC (Model-View-Controller)**

* **Modelos**: Encapsulan la lógica de negocio y la manipulación de datos.
* **Vistas**: Contienen la lógica de presentación.
* **Controladores**: Manejan la interacción del usuario, actualizan el modelo y seleccionan la vista.

**4. Convenciones de Codificación**

* **Indentación**: Usa cuatro espacios por nivel de indentación.
* **Comentarios**: Incluye comentarios para describir la funcionalidad del código y cualquier lógica compleja.
* **Propiedades de Clase**: Define propiedades con get y set y usa anotaciones de datos como [Required], [StringLength(255)].

Texto

Descripción generada automáticamente

**5. Métodos en Controladores**

* **Acciones GET**: Devuelven vistas o datos al cliente.

Texto

Descripción generada automáticamente

* **Acciones POST**: Manejan la lógica para modificaciones de datos.

Texto

Descripción generada automáticamente

6. **Manejo de Errores**

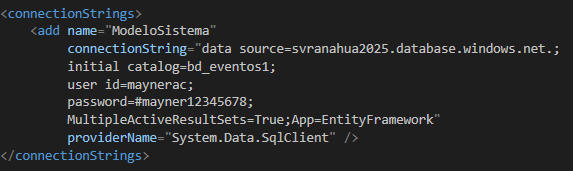
Usa try-catch para capturar y manejar excepciones.

Texto

Descripción generada automáticamente

**7. Conexión a la Base de Datos**

Define las conexiones de base de datos en el archivo web.config:



**8. Uso de Entity Framework**

* **Modelos**: Define las relaciones entre entidades usando virtual ICollection.
* **Consultas**: Usa Include para cargar datos relacionados.

Texto

Descripción generada automáticamente