

**UNIVERSIDAD PRIVADA DE TACNA**

**FACULTAD DE INGENIERIA**

**Escuela Profesional de Ingeniería de Sistemas**

**Estándar de Programación**

**Proyecto “Sistema de Gestión y Evaluación de Docentes”**

Curso: Programación Web II

Docente: Mag. Patrick Cuadros Quiroga

Integrantes:

**De La Cruz Choque, Ricardo Miguel**  **(2019063329)**

**Abraham Jesús Vela Vargas** **(2019063322)**

**Tacna – Perú**

**2025**

**Sistema de Gestión y Evaluación de Docentes**

**Estándar de Programación**

**Versión 1.1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CONTROL DE VERSIONES | | | | | |
| Versión | Hecha por | Revisada por | Aprobada por | Fecha | Motivo |
| 1.0 | RDC | PC | PC | 18/12/2024 | Plantilla del documento |
| 1.1 | RDC | PC | PC | 19/12/2024 | Versión Original |

**ÍNDICE**

[1. Introducción 4](#_Toc190963719)

[2. Arquitectura de Software 4](#_Toc190963720)

[2.1. Capa de Presentación 4](#_Toc190963721)

[2.2. Capa de Negocio 4](#_Toc190963722)

[2.3. Capa de Datos 4](#_Toc190963723)

[3. Convenciones de Nombres 4](#_Toc190963724)

[3.1. Nombres de Clases y Controladores 4](#_Toc190963725)

[3.2. Nombres de Vistas 4](#_Toc190963726)

[3.3. Nombres de Variables y Propiedades 4](#_Toc190963727)

[3.4. Nombres de Métodos y Funciones 4](#_Toc190963728)

[3.5. Nombres de Tablas y Campos en la Base de Datos 4](#_Toc190963729)

[4. Buenas Prácticas de Programación 4](#_Toc190963730)

[4.1. Formato del Código 5](#_Toc190963731)

[4.2. Comentarios 5](#_Toc190963732)

[4.3. Manejo de Errores 5](#_Toc190963733)

[4.4. Seguridad y Validaciones 5](#_Toc190963734)

[5. Control de Versiones 5](#_Toc190963735)

[6. Estándares para Desarrollo en C# 5](#_Toc190963736)

[6.1. Uso de Espacios de Nombres 5](#_Toc190963737)

[6.2. Estructura de una Clase en C# 5](#_Toc190963738)

[7. Estándares para Desarrollo en SQL Server 5](#_Toc190963739)

[7.1. Creación de Tablas 5](#_Toc190963740)

[7.2. Consultas SQL Parametrizadas 5](#_Toc190963741)

[8. Conclusión 5](#_Toc190963742)

# Introducción

Este documento define los estándares de programación para el desarrollo del "Sistema de Gestión y Evaluación de Docentes" en la Escuela de Odontología de la Universidad Privada de Tacna. Estos estándares buscan garantizar un código limpio, estructurado, eficiente y mantenible.

# Arquitectura de Software

El sistema se desarrollará bajo la arquitectura de tres capas (MVC - Modelo Vista Controlador)

* **Capa de Presentación**
* **Capa de Negocio**
* **Capa de Datos**

Utilizando ASP.NET con Visual Studio 2022, donde haremos uso de lo siguiente: ASP.NET MVC, HTML5, JavaScript, Bootstrap y C# (.NET Framework), implementando las reglas de negocio.

# Convenciones de Nombres

## Nombres de Clases y Controladores

Nombres en **PascalCase**.  
Los controladores llevarán el sufijo **Controller**. Ejemplo: DocentesController.

## Nombres de Vistas

* Las vistas se nombrarán de acuerdo con la acción a la que pertenecen en PascalCase.
* Se almacenarán dentro de la carpeta Views/Controlador/. Ejemplo: Views/Docentes/Index.cshtml.

## Nombres de Variables y Propiedades

* Variables locales y parámetros en camelCase.
* Propiedades de clases en PascalCase.
* Variables booleanas iniciarán con is o has para indicar su naturaleza. Ejemplo: isActivo.

## Nombres de Métodos y Funciones

Nombres en **PascalCase** y descriptivos, comenzando con verbos. Ejemplo: ObtenerDocentes(), GuardarEvaluacion().

## Nombres de Tablas y Campos en la Base de Datos

* Las tablas usarán PascalCase en singular. Ejemplo: Docente, EvaluacionDocente.
* Las columnas usarán PascalCase. Ejemplo: DocenteID, Nombre, FechaCreacion.
* Las claves primarias tendrán el sufijo ID. Ejemplo: DocenteID.
* Las claves foráneas incluirán el nombre de la tabla referenciada. Ejemplo: DocenteID en EvaluacionDocente.

# Buenas Prácticas de Programación

## Formato del Código

* Indentación de 4 espacios.
* Cada archivo de clase debe incluir comentarios XML de documentación.
* Evitar nombres de variables genéricos (ej. var1, temp), prefiriendo nombres descriptivos.

## Comentarios

* Incluir comentarios para explicar bloques de lógica complejos.
* Usar /// para documentar métodos y clases en C#.

## Manejo de Errores

* Implementar manejo de excepciones con try-catch.
* Los errores críticos se registrarán en una tabla de auditoría.

## Seguridad y Validaciones

* Validación del lado del cliente con JavaScript y del lado del servidor con Data Annotations en ASP.NET MVC.
* Autenticación basada en roles.
* Las consultas SQL deben evitar inyecciones con parametrización.

# Control de Versiones

* Se usará GitHub para el control de versiones.
* Se creará una rama principal main y ramas secundarias para cada funcionalidad.
* Se seguirá el flujo de trabajo de Git Flow.

# Estándares para Desarrollo en C#

## Uso de Espacios de Nombres

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web.Mvc;

## Estructura de una Clase en C#

public class Docente

{

public int DocenteID { get; set; }

public string Nombre { get; set; }

public string Apellido { get; set; }

public bool IsActivo { get; set; }

public void ActivarDocente()

{

this.IsActivo = true;

}

}

# Estándares para Desarrollo en SQL Server

## Creación de Tablas

CREATE TABLE Docente (

DocenteID INT PRIMARY KEY IDENTITY(1,1),

Nombre VARCHAR(200) NOT NULL,

Apellido VARCHAR(200) NOT NULL,

IsActivo BIT DEFAULT 1

);

## Consultas SQL Parametrizadas

string query = "SELECT \* FROM Docente WHERE DocenteID = @DocenteID";

using (SqlCommand cmd = new SqlCommand(query, connection))

{

cmd.Parameters.AddWithValue("@DocenteID", docenteId);

}

# Conclusión

Este documento establece los lineamientos que debemos seguir para desarrollar el "Sistema de Gestión y Evaluación de Docentes". Estos estándares están enfocados en asegurar que el código sea fácil de mantener, eficiente y seguro. Como equipo de desarrollo, es fundamental que todos sigamos estas pautas para garantizar que el software cumpla con los requisitos del sistema, sea escalable, y se pueda actualizar o modificar sin problemas en el futuro. Al adherirnos a estos estándares, también mejoramos la calidad del proyecto y facilitamos su evolución a lo largo del tiempo.