

Actividad 2 – Pagina web Matricula guardada en JSON BASICA

Andrés Felipe Gonzalez Noreña

Servicio Nacional de Aprendizaje – SENA

Programa: Análisis y Desarrollo de Software

Ficha: 3144585

Instructor: Luis Fernando

2025

Introducción

El presente documento registra la bitácora correspondiente al desarrollo de matricula, una aplicación diseñada con C#, cuyo propósito es gestionar la matrícula de estudiantes en diferentes asignaturas dentro de un entorno web. Durante el proceso se exploraron componentes fundamentales del patrón Modelo–Vista–Controlador. Este documento describe la organización interna del proyecto, las partes principales del código, tablas de resumen funcional y conclusiones del proceso.

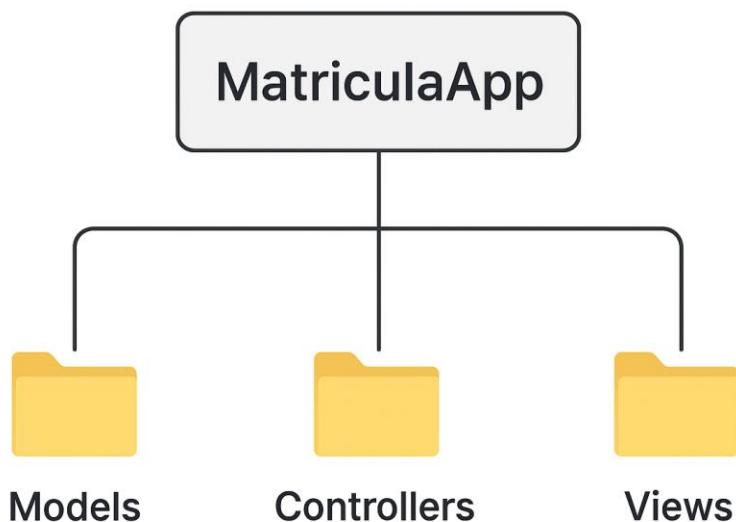
Estructura del proyecto

El Proyecto está organizado bajo el namespace MatriculaApp:

Estas son las principales carpetas de el proyecto las cuales contienen los archivos que usamos

- Models
- Controllers
- Views

Imagen de la estructura:



Models

Los **Models** representan las **entidades de datos** del sistema. En el proyecto, el modelo **Matricula** define la estructura de la información que se necesita para registrar a un estudiante.

El modelo incluye propiedades como *nombre*, *documento*, *curso*, *edad* y *email*, que permiten organizar y manejar los datos de cada matrícula dentro de la aplicación.

sirve como plantilla para almacenar y transportar la información que usan los controladores y las vistas.

```

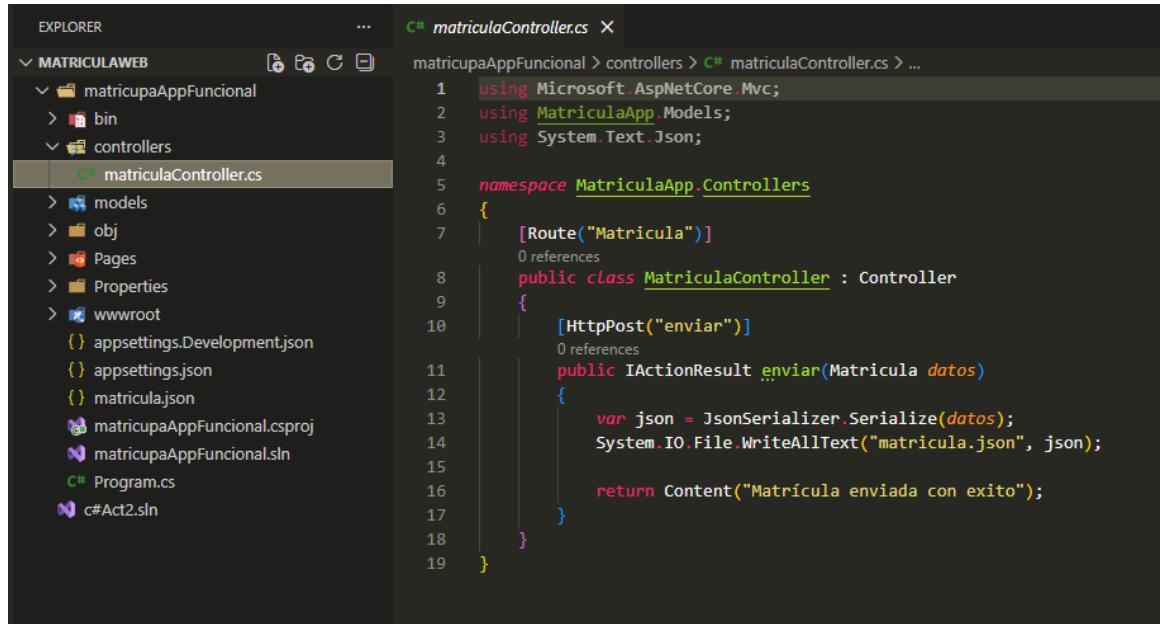
C# Matricula.cs X
matriculaAppFuncional > models > C# Matricula.cs > ...
1 namespace MatriculaApp.Models
2 {
3     public class Matricula
4     {
5         public string nombre { get; set; }
6         public string documento { get; set; }
7         public string curso { get; set; }
8         public int edad { get; set; }
9         public string email { get; set; }
10    }
11 }
12

```

Controllers

Controllers son la parte del proyecto encargada de manejar la lógica de la aplicación. Su función principal es recibir las solicitudes del usuario, procesar los datos utilizando los Models y devolver una respuesta apropiada (texto, vista, archivo, etc.).

el controlador actúa como un intermediario entre la información (Model) y lo que ve el usuario (Views).



```

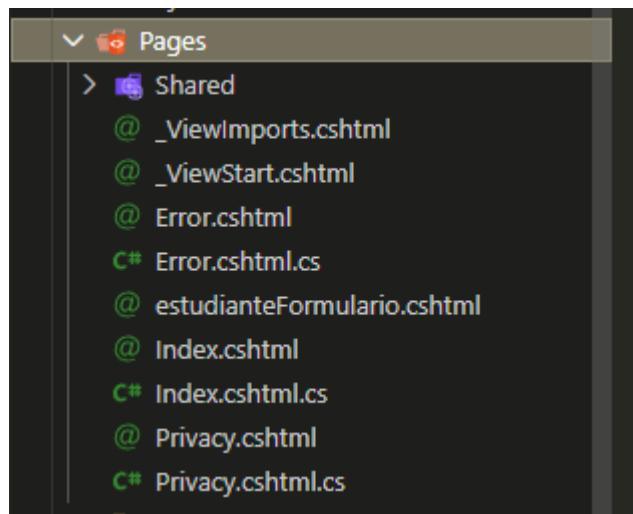
C# matriculaController.cs
matriculaAppFuncional > controllers > C# matriculaController.cs > ...
1 using Microsoft.AspNetCore.Mvc;
2 using MatriculaApp.Models;
3 using System.Text.Json;
4
5 namespace MatriculaApp.Controllers
6 {
7     [Route("Matricula")]
8     public class MatriculaController : Controller
9     {
10        [HttpPost("enviar")]
11        public IActionResult enviar(Matricula datos)
12        {
13            var json = JsonSerializer.Serialize(datos);
14            System.IO.File.WriteAllText("matricula.json", json);
15
16            return Content("Matrícula enviada con éxito");
17        }
18    }
19 }

```

Views/Pages

En este proyecto, las **Pages** cumplen la función de ser las **pantallas o páginas web donde el usuario ingresa y visualiza la información de la matrícula.**

Cada Page contiene un formulario o estructura visual que permite al usuario interactuar directamente con el sistema.



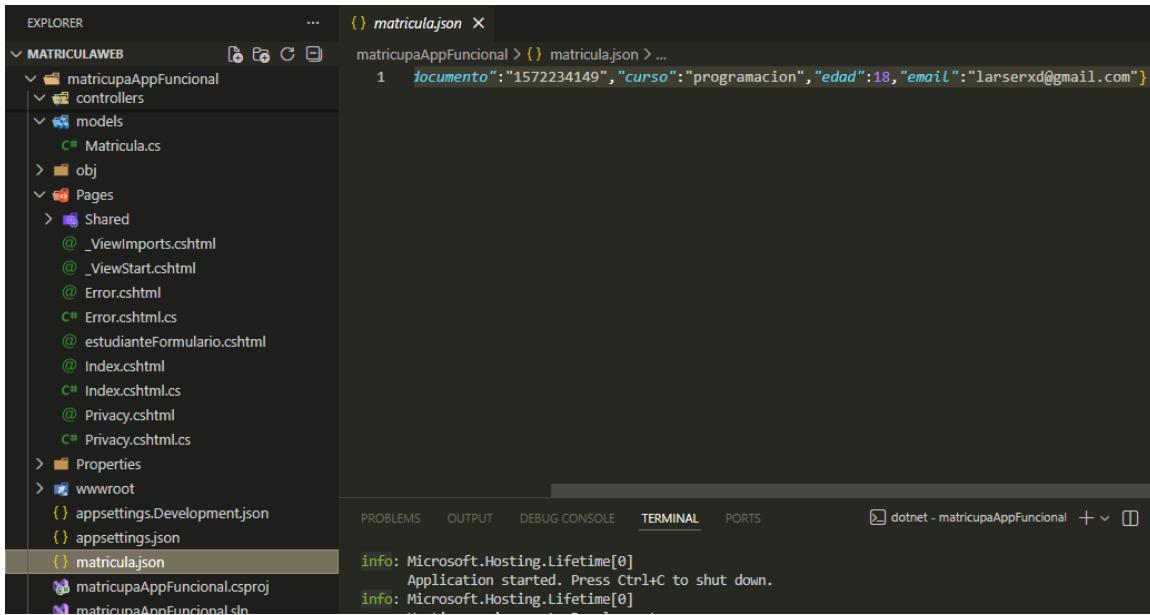
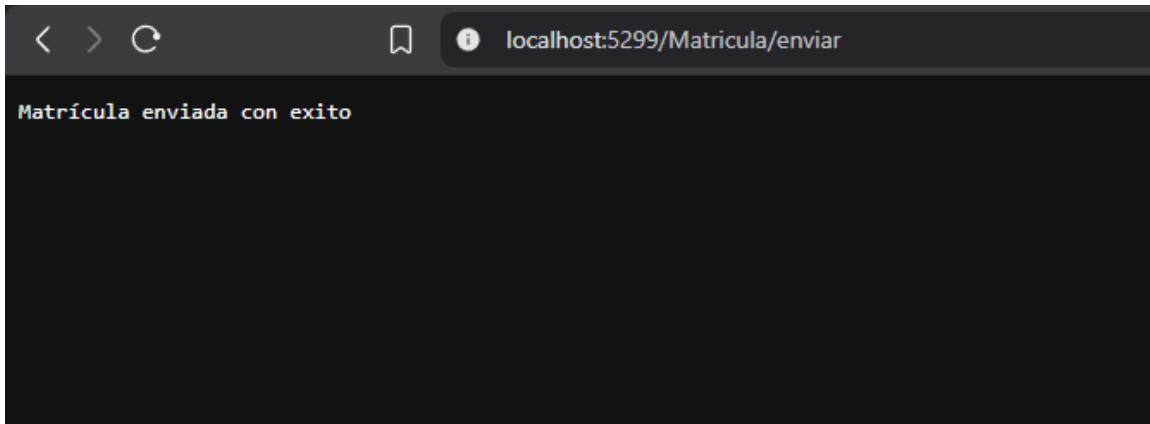
The screenshot shows a web browser window with the URL `localhost:5299`. The title bar says "matricupaAppFuncional". The main content area displays the word "Welcome" in large font, followed by a blue button labeled "Ver formulario de matrícula". At the bottom, there is a copyright notice: "© 2025 - matricupaAppFuncional - [Privacy](#)". On the right side, there is a link to "Activar Windows" with the subtext "Ve a Configuración para activar Windows".

The screenshot shows a web browser window with the URL `localhost:5299/estudianteFormulario`. The title bar says "matricupaAppFuncional". The main content area has a heading "formulario para registrarse". Below it is a form with the following fields:

- nombre del estudiante:
- documento:
- curso que desea estudiar:
- edad:
- email:
-

At the bottom, there is a copyright notice: "© 2025 - matricupaAppFuncional - [Privacy](#)".

Enviado al JSON



Tablas

Modelos

Modelo	Descripción	Función
Matricula	Guarda los datos de los estudiantes	Registrar procesos de matrícula

Tabla de Controladores

Controlador	Descripción	Funciones principales
MatriculaController	Gestiona las matrículas.	Registrar las inscripciones de los estudiantes en un JSON.

Tabla de Vistas

Vista	Uso	Función
Index	Página principal	Muestra la pagina principal la cual te redirige al formulario
EstudianteFormulario	Formulario	Permite crear la matricula del estudiante la cual luego es enviada al JSON

Tabla de Problemas Durante la Realización

Problema identificado	Descripción
Falta de experiencia en c#	La arquitectura del proyecto no es nada fácil de entender y mas el código, que aun que se básico es bastante complejo al menos para mi
Problemas al documentar	Hacer este documento no es fácil, ahorita ni se si esta bien pero le metí lo más que pude y trate de ser muy técnico con las definiciones
Problemas de rutas	Algunas veces copiaba mal las rutas y me tocaba corregirlas, COMO ODIE

Problema identificado	Descripción
	ESO

Referencias

ChatGPT: Asistencia para documentación técnica y dandome ideas en la estructuracion del documento, imagenes capturas y definiciones.

Conclusiones

Conclusion 1

En conclusion solo digo que es un proyecto bastante simple pero que tiene su grado de dificultad, cumple con lo solicitado que es guardar los datos en un-JSON, la pagina es lo mas simple y el formulario igual pero esto tomo tiempo y di mi maximo en esto, aprendi bastante y me parecio interesante el proceso de realizacion del codigo.

Conclusion 2

Este proyecto también me dejó una buena experiencia, porque, aunque es algo simple, igual requiere paciencia y dedicación para que todo funcione bien. Lograr que el formulario enviara los datos y se guardaran en JSON fue un reto, pero al final pude hacerlo. Aprendí cosas nuevas y me gustó ver cómo poco a poco el proyecto fue tomando forma