

Proyecto MVC

Instituto Tecnológico Superior del Sur de Guanajuato



19 de febrero de 2021

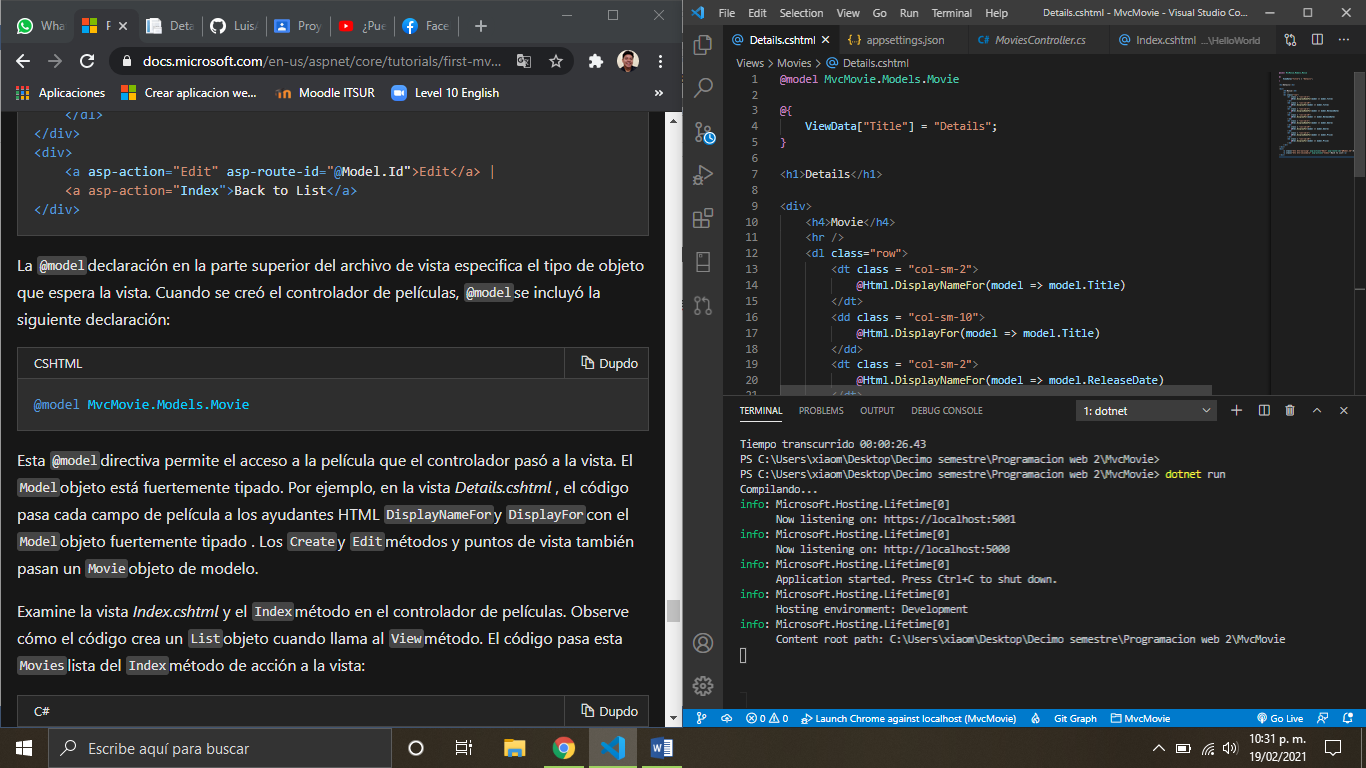
Progrmacion web 2

Alumno: Luis Alberto Ignacio Esteban

var movie = await \_context.Movie

.FirstOrDefaultAsync(m => m.Id == id);

Se pasa una expresión lambda a FirstOrDefaultAsyncpara seleccionar entidades de película que coincidan con los datos de ruta o el valor de la cadena de consulta.



La declaración de @model en la parte superior del archivo de vista especifica el tipo de objeto que espera la vista. Cuando se creó el controlador de películas, @model se incluyó la siguiente declaración:

@model MvcMovie.Models.Movie

Esta @modeldirectiva permite el acceso a la película que el controlador pasó a la vista. El Modelobjeto está fuertemente tipado.

public void ConfigureServices(IServiceCollection services)

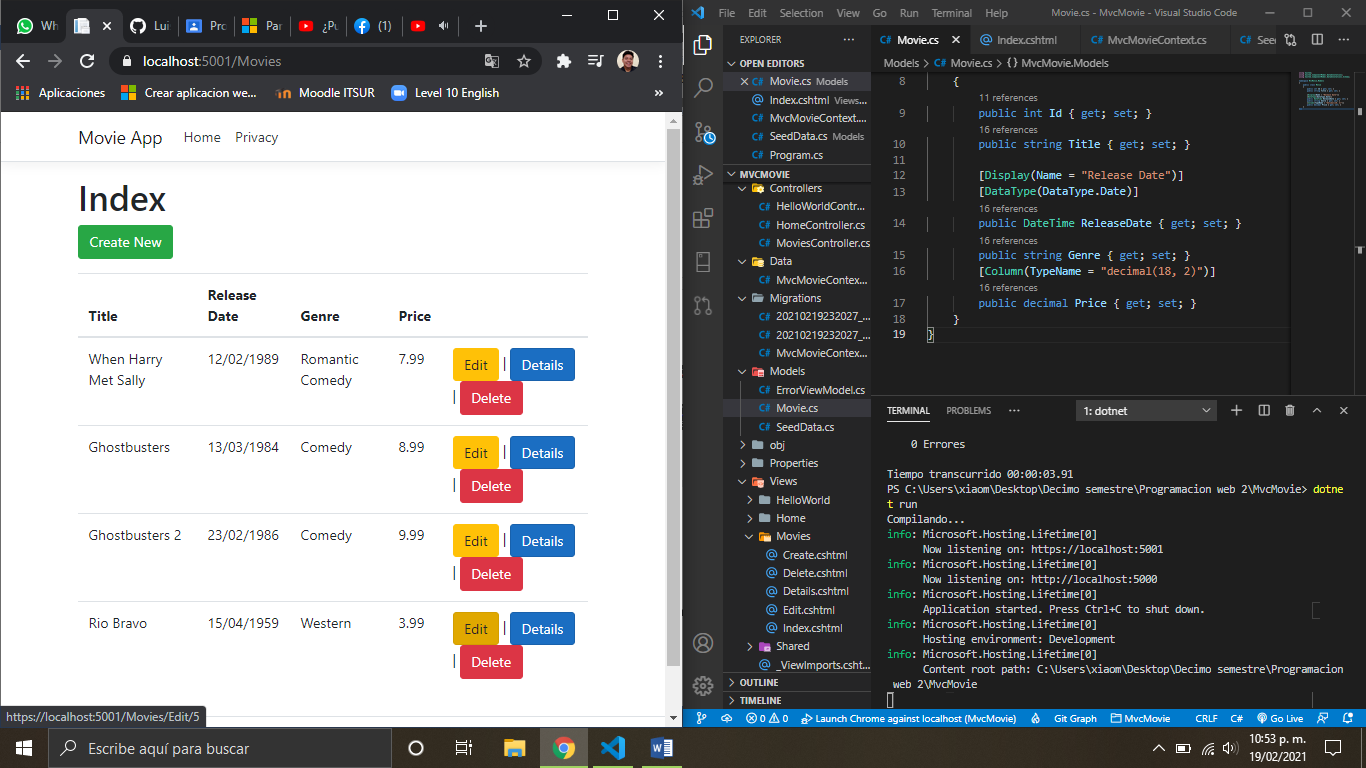
{

services.AddControllersWithViews();

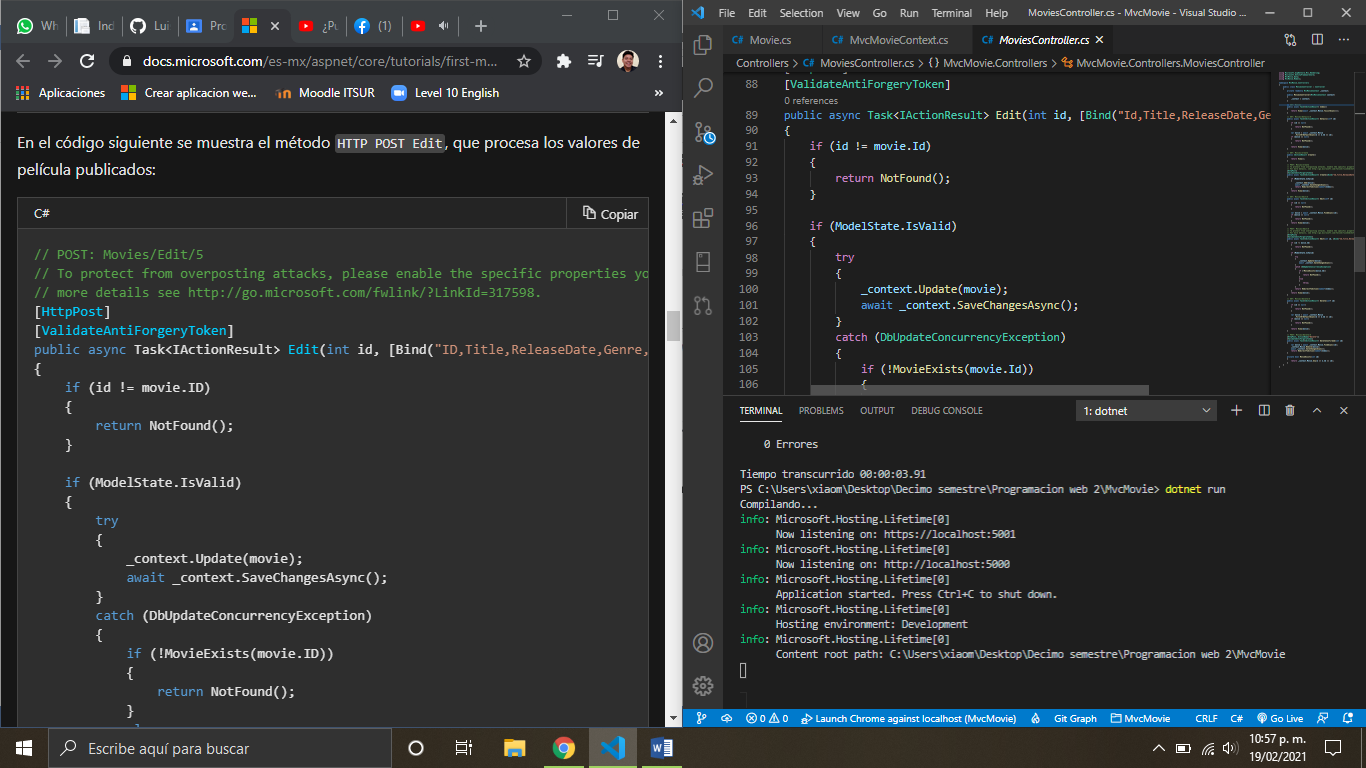
services.AddDbContext<MvcMovieContext>(options => options.UseSqlite(Configuration.GetConnectionString("MvcMovieContext")));

}

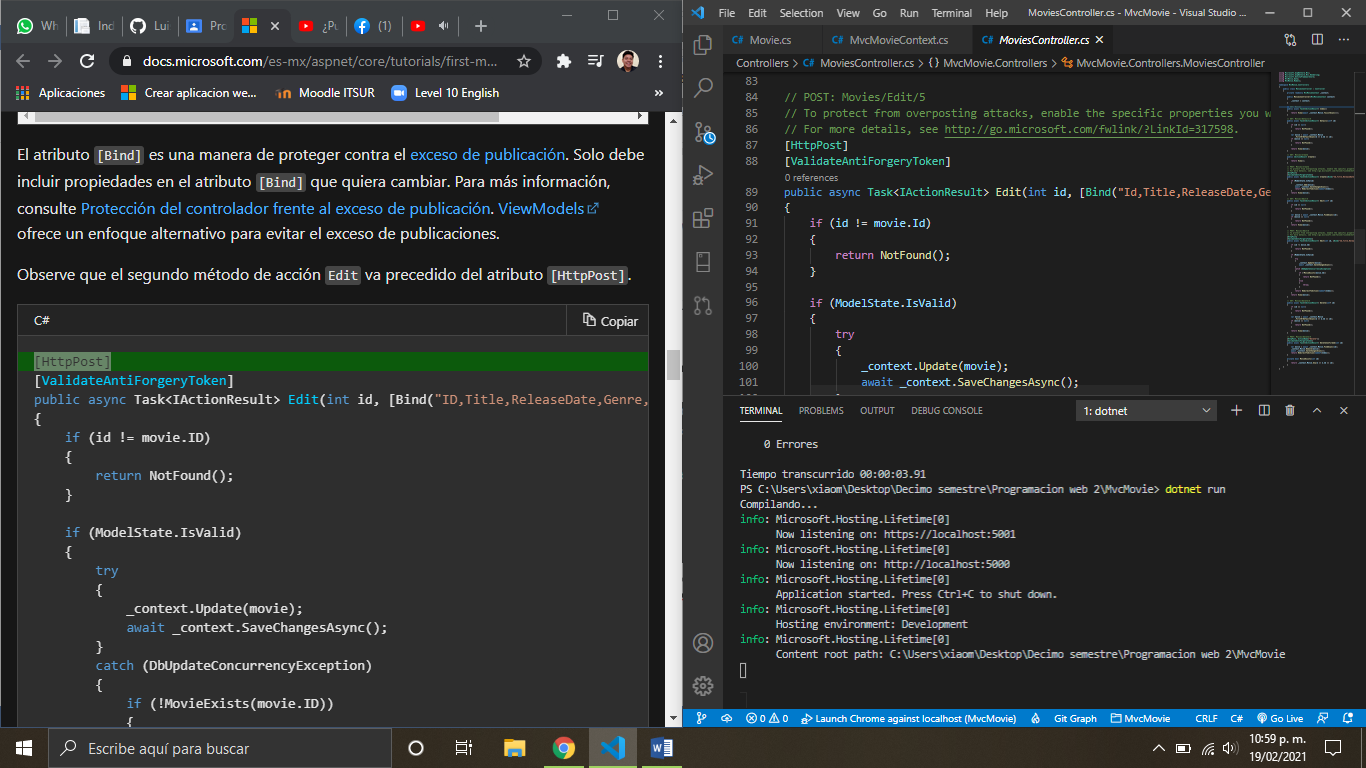
El objeto MvcMovieContext controla la tarea de conexión a la base de datos y asignación de objetos Movie a los registros de la base de datos. El contexto de base de datos se registra con el contenedor de inserción de dependencias en el método ConfigureServices del archivo Startup.cs:



Los vínculos Edit (Editar), Details (Detalles) y Delete (Eliminar) se generan mediante el asistente de etiquetas de delimitador de MVC Core en el archivo Views/Movies/Index.cshtml.

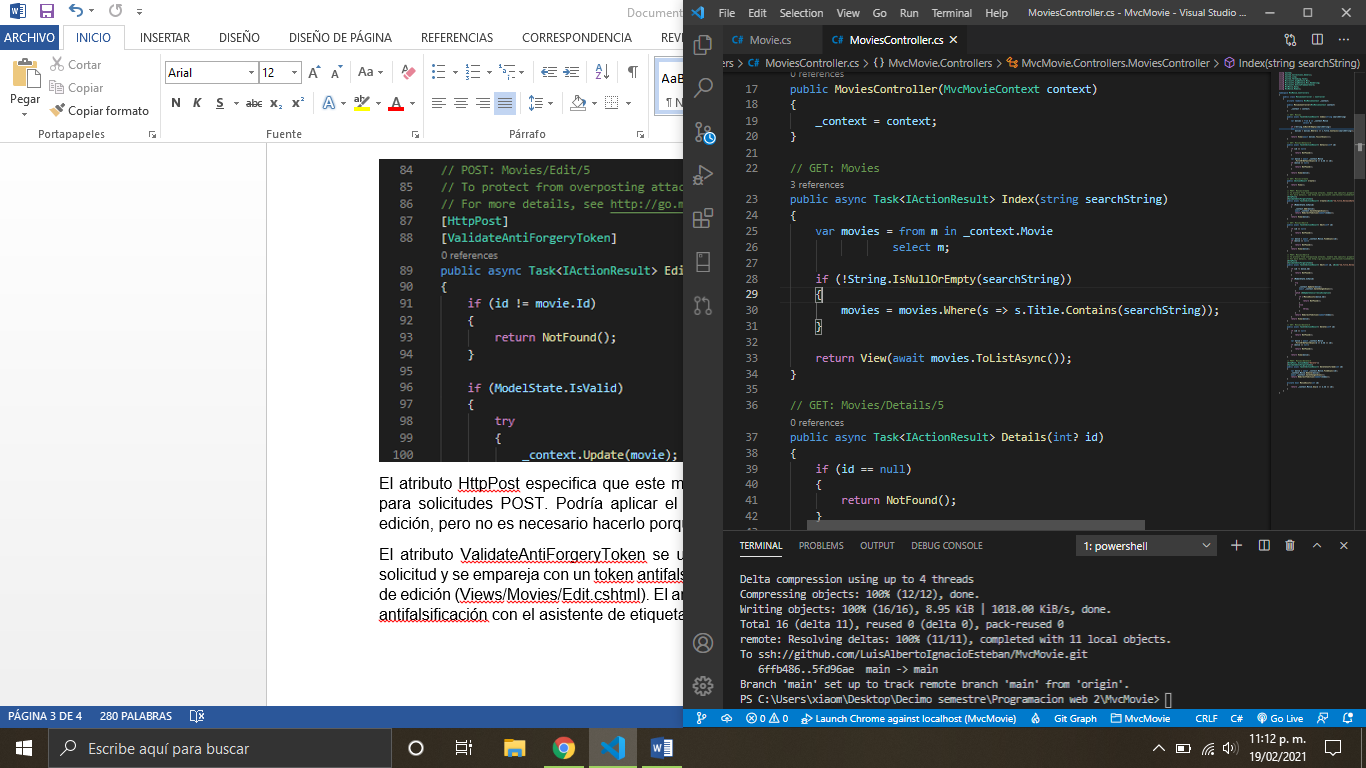


El atributo [Bind] es una manera de proteger contra el exceso de publicación. Solo debe incluir propiedades en el atributo [Bind] que quiera cambiar.

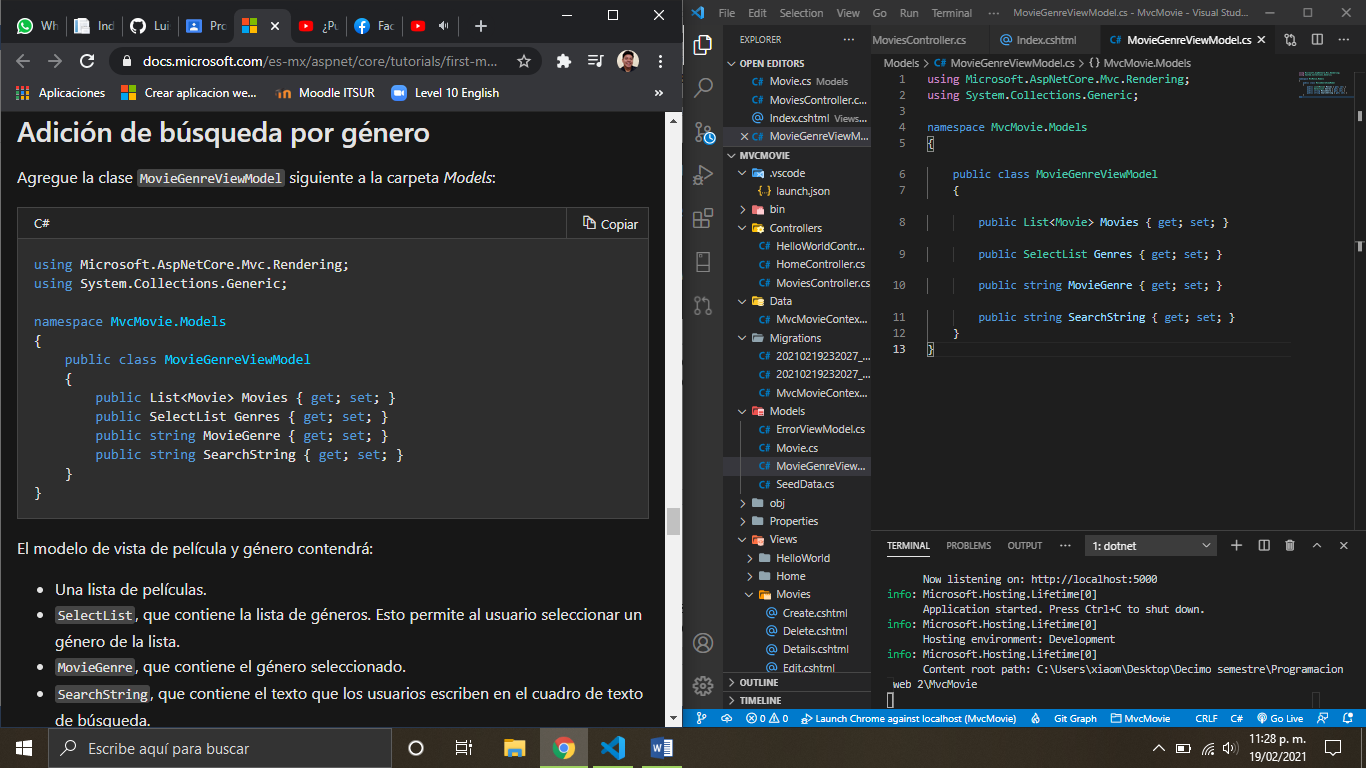


El atributo HttpPost especifica que este método Edit se puede invocar solamente para solicitudes POST. Podría aplicar el atributo [HttpGet] al primer método de edición, pero no es necesario hacerlo porque [HttpGet] es el valor predeterminado.

El atributo ValidateAntiForgeryToken se usa para impedir la falsificación de una solicitud y se empareja con un token antifalsificación generado en el archivo de vista de edición (Views/Movies/Edit.cshtml). El archivo de vista de edición genera el token antifalsificación con el asistente de etiquetas de formulario.



El código s => s.Title.Contains() anterior es una expresión Lambda. Las lambdas se usan en consultas LINQ basadas en métodos como argumentos para métodos de operador de consulta estándar, tales como el método Where o Contains (usado en el código anterior). Las consultas LINQ no se ejecutan cuando se definen ni cuando se modifican mediante una llamada a un método como Where, Contains u OrderBy. En su lugar, se aplaza la ejecución de la consulta. Esto significa que la evaluación de una expresión se aplaza hasta que su valor realizado se repita realmente o se llame al método ToListAsync.



El modelo de vista de película y género contendrá:

Una lista de películas.

SelectList, que contiene la lista de géneros. Esto permite al usuario seleccionar un género de la lista.

MovieGenre, que contiene el género seleccionado.

SearchString, que contiene el texto que los usuarios escriben en el cuadro de texto de búsqueda.

Repositorio github

https://github.com/LuisAlbertoIgnacioEsteban/MvcMovie