**Estructura de carpetas y archivos del proyecto C# usando ASP.NET Core con Entity Framework Core, aplicando el patrón Repository y Service**

**Estructura General del Proyecto**

Repository/ ← Nombre del proyecto/base del namespace

│

├── Controllers/ ← Controladores de API REST

│ ├── CoursesController.cs

│ └── StudentController.cs

│

├── Data/ ← Contexto de la base de datos

│ └── SchoolContext.cs

│

├── Models/ ← Entidades del dominio

│ ├── Course.cs

│ ├── Enrollment.cs

│ ├── Grade.cs

│ └── Student.cs

│

├── Repositories/ ← Interfaces de acceso a datos

│ ├── IGenericRepository.cs

│ ├── ICourseRepository.cs

│ ├── IStudentRepository.cs

│ └── Implements/ ← Implementaciones de los repositorios

│ ├── CourseRepository.cs

│ ├── StudentRepository.cs

│ └── GenericRepopsitory.cs

│

├── Services/ ← Interfaces de lógica de negocio

│ ├── IGenericService.cs

│ ├── ICourseService.cs

│ ├── IStudentService.cs

│ └── Implements/ ← Implementaciones de los servicios

│ ├── CourseService.cs

│ ├── StudentService.cs

│ └── GenericService.cs

│

├── appsettings.json ← Configuración general del proyecto (.NET)

├── Program.cs ← Punto de entrada del proyecto (configuración de servicios y middleware)

**Descripción de Archivos Clave**

**1. Controllers/**

* **CoursesController.cs / StudentController.cs**  
  Controladores de API REST para exponer los endpoints HTTP (GET, POST, PUT, DELETE) para Course y Student.

**2. Data/**

* **SchoolContext.cs**  
  Clase DbContext que configura las entidades y su mapeo a tablas mediante Fluent API. Usa EF Core.

**3. Models/**

* **Course.cs, Student.cs, Enrollment.cs, Grade.cs**  
  Entidades del dominio que representan la base de datos.  
  Se usan anotaciones como [DatabaseGenerated] y JsonIgnore.

**4. Repositories/**

* **IGenericRepository.cs**  
  Interface genérica con métodos CRUD asincrónicos.
* **ICourseRepository.cs / IStudentRepository.cs**  
  Interfaces específicas para las entidades Course y Student, que heredan de la genérica.

**5. Repositories/Implements/**

* **GenericRepopsitory.cs**  
  Implementación genérica de IGenericRepository usando EF Core.
* **CourseRepository.cs / StudentRepository.cs**  
  Implementaciones concretas para Course y Student.

**6. Services/**

* **IGenericService.cs**  
  Interface genérica de servicios (capa intermedia entre controladores y repositorios).
* **ICourseService.cs / IStudentService.cs**  
  Interfaces específicas que heredan de la genérica.

**7. Services/Implements/**

* **GenericService.cs**  
  Implementación genérica que delega al repositorio genérico.
* **CourseService.cs / StudentService.cs**  
  Implementaciones específicas de servicios.

**8. Program.cs**

* Configura:
  + EF Core con cadena de conexión a SQL Server.
  + Inyección de dependencias para repositorios y servicios.
  + Swagger para documentación de la API.
  + Middleware para redirección HTTPS y controladores.

**9. appsettings.json**

* Contiene:
  + Niveles de log.
  + Cadena de conexión a SQL Server (SchoolContext).