**User Guide – TechnoRishi :Library Management System**

**Introduction:-**

This guide provides step-by-step instructions to set up, configure, and run the Library Management System built with .NET, Entity Framework (Code-First), and SQL Server. Below is the Architecture diagram of project.

ToDo:- Authentication and Authorization

A diagram of a software application

AI-generated content may be incorrect.

**System Requirements:-**

1. Visual Studio 2019 or later
2. NET 8.0 SDK or later
3. SQL Server (LocalDB or SQL Server Management Studio)
4. Postman for API testing (optional), Swagger is implemented.

**Project Setup Instructions:-**

1. Extract the provided ZIP file to your local machine named “**TechnoRishi LMS**”.
2. Open the extracted folder
   * Go to `File` → `Open` → `Project/Solution`.
   * Navigate to the project folder and select the **TechnoRishi.sln** file.
3. Restore NuGet Packages
   * Right-click on the solution in the Solution Explorer.
   * Select `Restore NuGet Packages` to fetch all dependencies.

**Database Configuration and Migrations**

1. Update Connection String:
   * Open `appsettings.json`.
   * Modify the connection string to point to your local SQL Server instance:-

"ConnectionStrings": {

"LibraryDB": "Server=LocalDB ;Database=LibraryDB;User=sa;Password=password;TrustServerCertificate=True;"

}

1. Apply Migrations:-
   * Open the **Package Manager Console** from `Tools` → `NuGet Package Manager` → `Package Manager Console` and select default project as **“TechnoRishi.LMS.Repository”**
   * Run the following commands:

**Update-Database**

This will create the database schema based on the Entity Framework models.

1. Seed Initial Data (Optional):-

If you have seed data, it will be populated during migration. If not, you can add initial records manually using SQL or directly from the application.

**Running the Application Locally**

1. Start the Application:-

* click the `Start` button in Visual Studio.
* The application should launch in your default browser at `https://localhost:7060`.

1. Access Application Modules

* `Book`: Manage book records (CRUD operations).
* `Member`: Manage library members (CRUD operations).
* `BorrowedBooks`: Borrow a book and update its availability.

**Testing with Postman/Swagger**

1. Use Postman or Swagger to interact with API endpoints:

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a group of text boxes

AI-generated content may be incorrect.**

**Troubleshooting and Common Issues**

1. Database Connection Error:

* Ensure SQL Server is running.
* Check the connection string in `appsettings.json`.

1. Migrations Not Applying:

* Run `Add-Migration InitialCreate` before `Update-Database` if there are no migrations.

1. API Not Responding:

* Ensure the application is running and the correct port is open (default: 7098).

1. Postman Requests Failing:

* Verify the endpoint URL.