Conditions

You have to prepare a solution to the proposed problem in the defined period of time. The solution must comply with the requirements. For anything not explicitly listed, you are free to choose whatever technology/library/tool you feel comfortable with.

Once ready, you must send a package with the source code of the solution, so it can be built and reviewed by Musala Soft. Instructions how to use the solution must also be provided (resource names, SQL scripts to import test data, other scripts, etc.).

*If you have completed the task after the deadline has expired, you are still encouraged to submit a solution.*

Software Requirements

**Programming languages:** C# (Complete)

**Framework**: ASP.NET / ASP.NET Core (Complete)  
**Database: MSSQL** or in-memory (Complete)(MSSQL Use)

*https://github.com/valquiriacr21/GatewayDeviceAPI*  
**Automated build:** Solution of choice

**UI:** Angular (Complete)

*https://github.com/valquiriacr21/DeviceGatewayUI-Musala/settings*

Description

This sample project is managing gateways - master devices that control multiple peripheral devices.

Your task is to **create a REST service** (JSON/HTTP)for storing information about these gateways and their associated devices. This information must be stored in the database. (Complete)

When storing a gateway, any field marked as “to be validated” must be validated and an error returned if it is invalid.

* Also, no more that 10 peripheral devices are allowed for a gateway.(Complete)
* The service must also offer an operation for displaying information about all **stored gateways** (and their devices) (In Process)
* And an operation for displaying details for a single **gateway**.(In Process)
* Finally, it must be possible to add and remove a device from a gateway.(In Process)

Each gateway has:

* a unique serial number (string), *{SerialNumber[Key]}* (Complete)Varificar generacion
* human-readable name (string), *{Name (string)},*(Complete)
* IPv4 address (to be validated),*{IPv4 (string)[MaxLength(15)]****(validated)****},*(Complete)
* multiple associated peripheral devices.

Each peripheral device has:

* a UID (number),(Complete)
* vendor (string),(Complete)
* date created,(Complete)
* status - online/offline.*[MaxLength(6)][MaxLength(7)](validated)*(Complete)

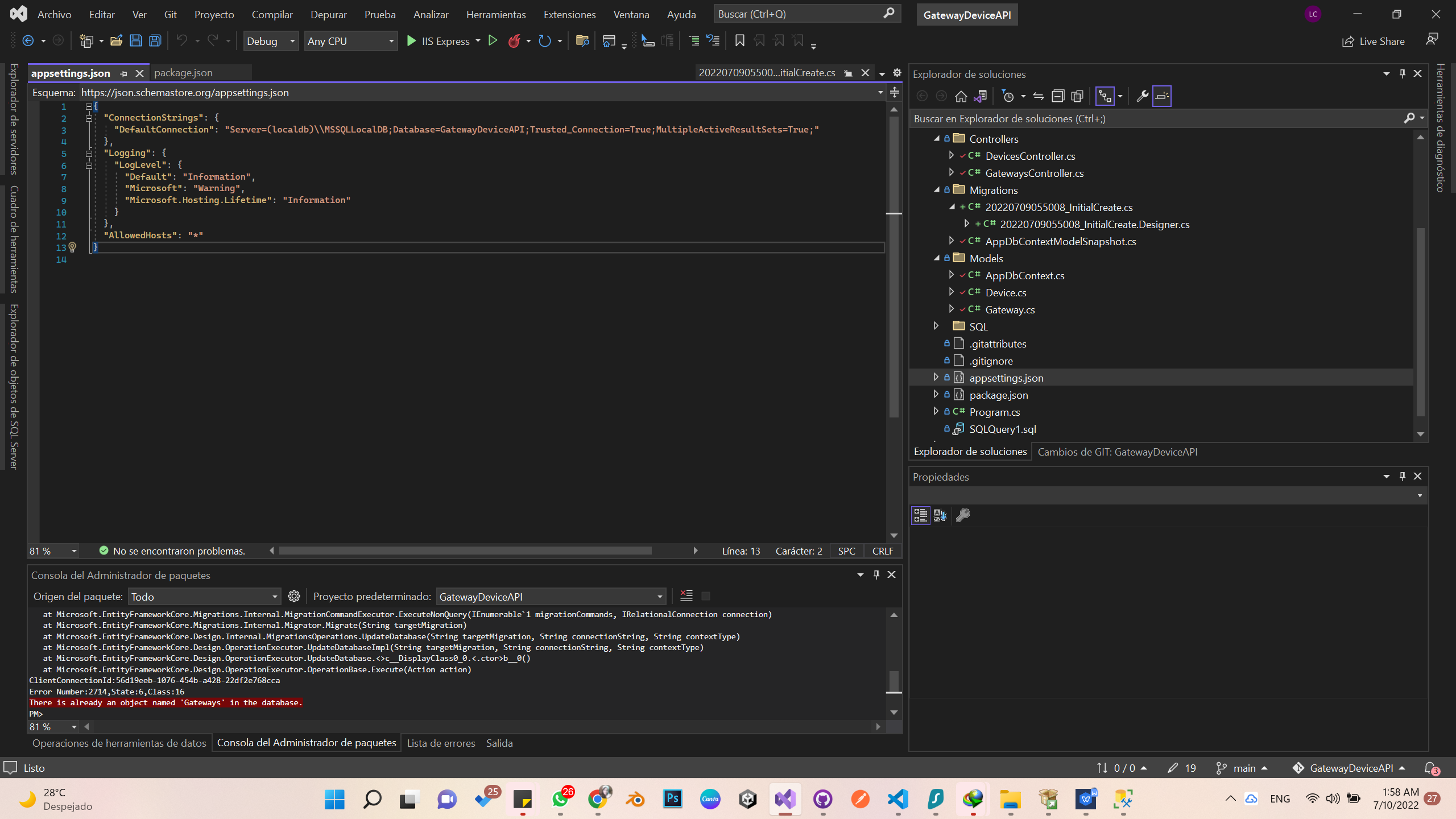
Other considerations

Please, provide

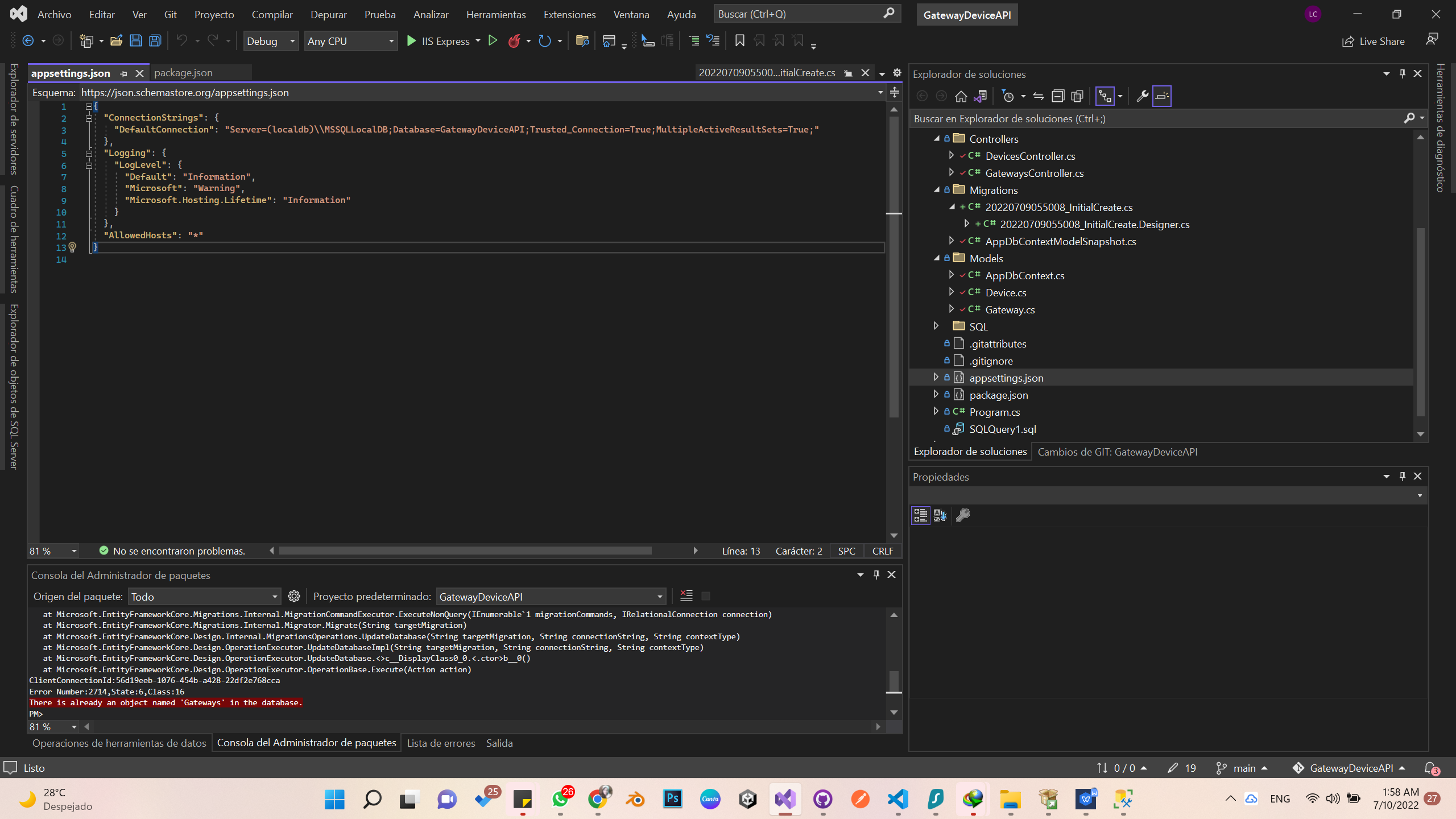
* Basic UI - recommended or (providing test data for Postman (or other rest client) if you do not have enough time.(Complete)(https://github.com/valquiriacr21/DeviceGatewayUI-Musala/settings)(add Swagger)
* Meaningful Unit tests.
* Readme file with installation guides.
* An automated build.

Readme file with installation guides.

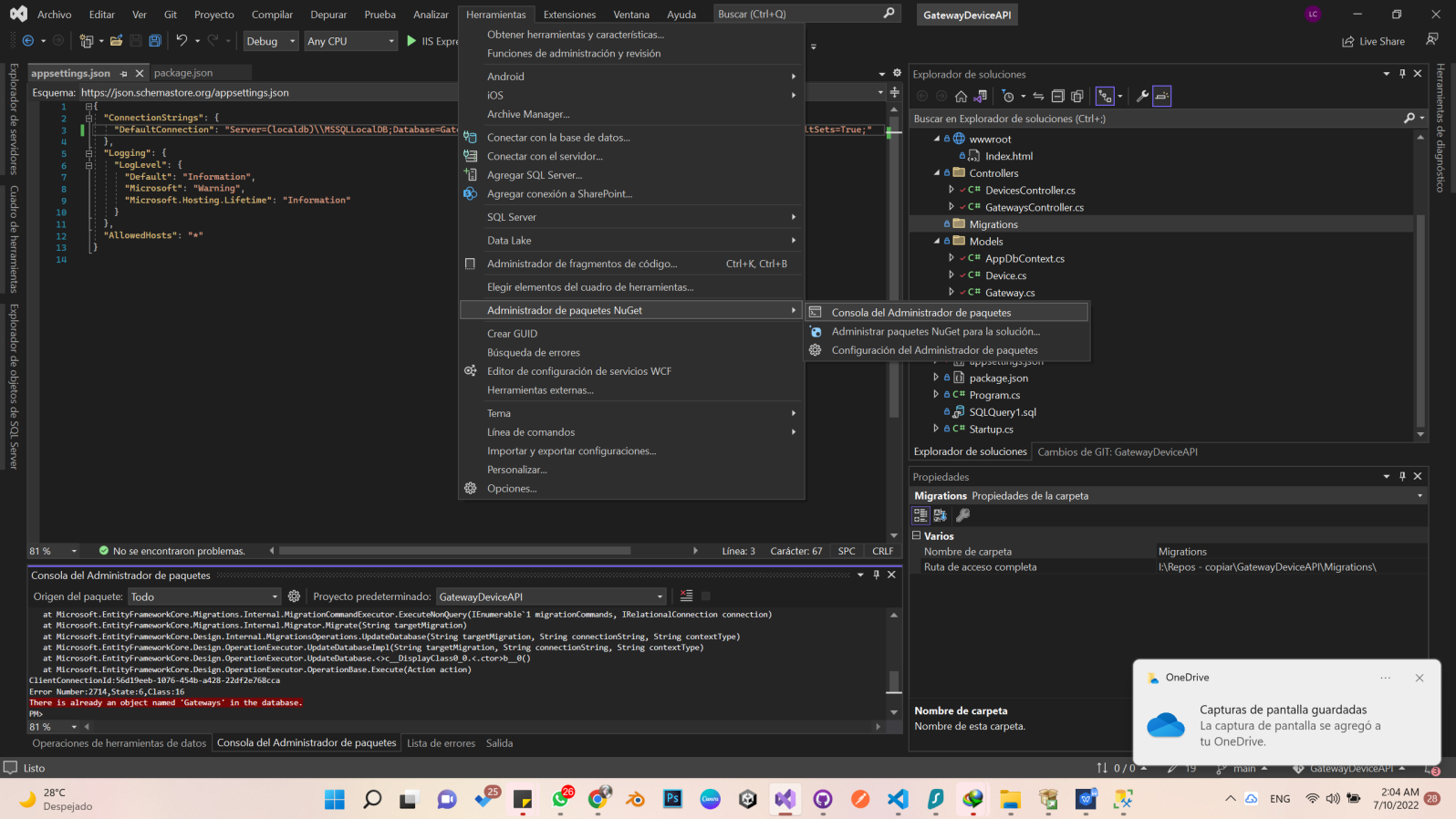
* -You must create a new database in SQL Server with the name "GatewayDeviceAPIMusalaSoft" or the name of your choice
* -Go to the file "appsettings.json to configure the local data server" in visual studio.



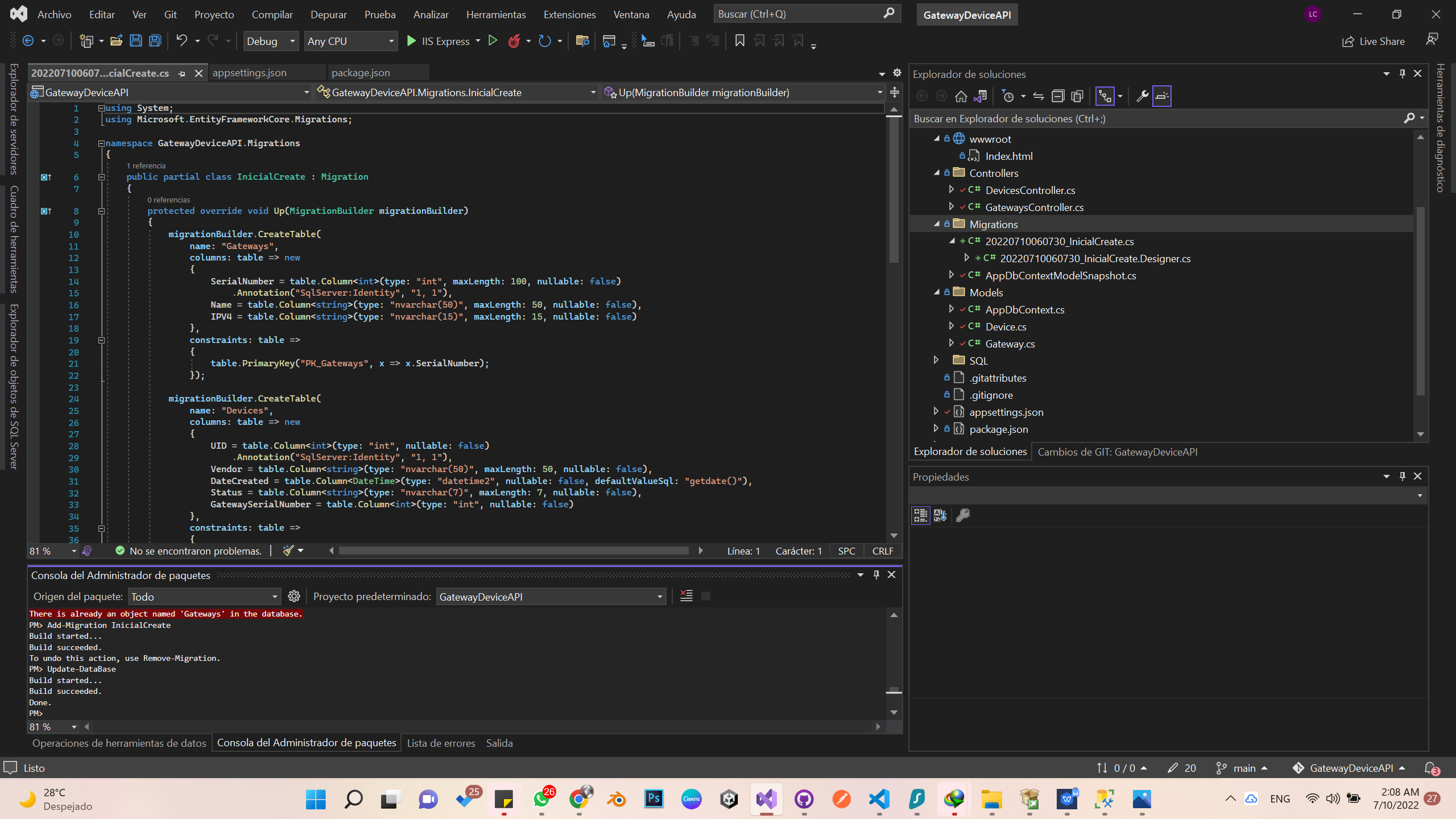
* -Review the “DefaultConnection” configuration and in Server put the name of your local server and in DataBase the name of the database that you just created. In my case it is “GatewayDeviceAPIMusalaSoft”.



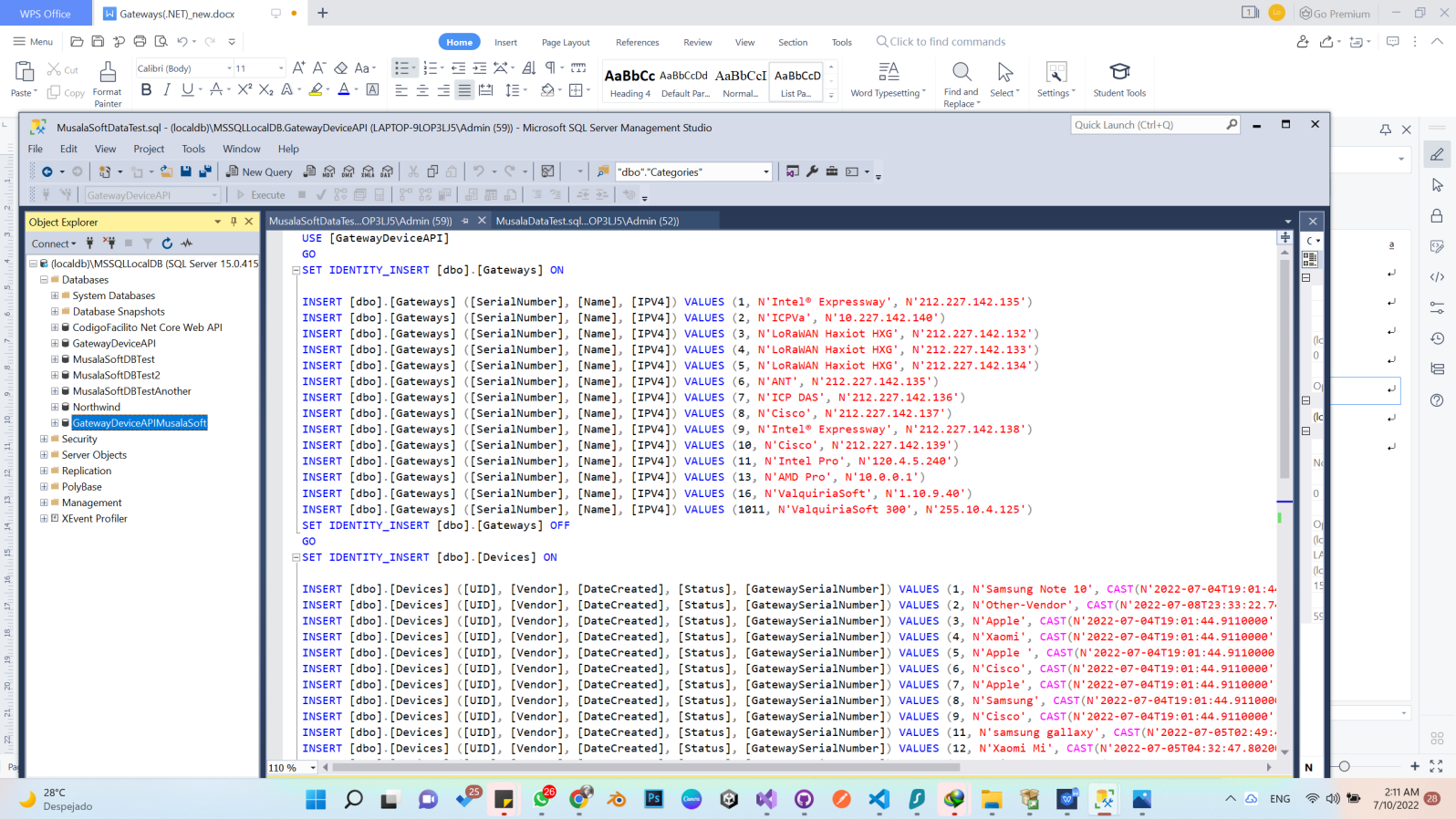
* -If there are elements in the migrations folder, they must be deleted.
* -Then they must do the migrations again, for this they must go to tools or tools / NuGet Package Management / Package Manager Console.



* -In the console after “PM>” write: Add-Migration InitialCreate
* -When PM comes out again> write: Update-DataBase



* Then go to the Microsoft SQL Management Studio



* And open the SQL (3. Musala Soft Insert into tables.sql) file that is in the API project inside the SQL folder: .
* Where it says USE in the code for the name of the database that you connected to the project, in my case it is the name that you read suggested: [GatewayDeviceAPIMusalaSoft]and you execute it.
* Then you go to the solution in visual studio and run the project.