

# Project management report

## DynamoDB and embedded data

Embedded data handling in .NET with Amazon DynamoDB

December 19, 2022

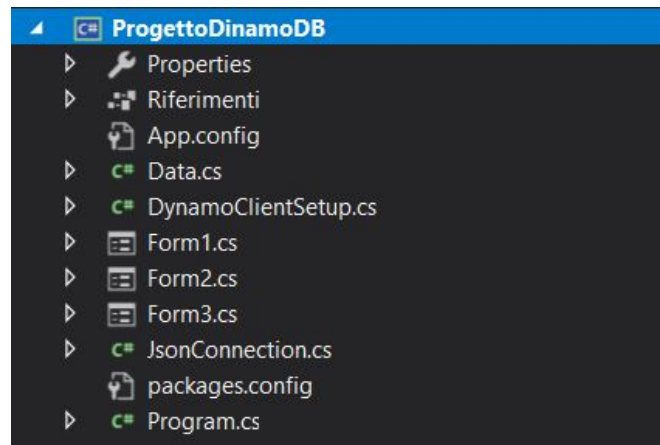
## 1 Instructions

The instructions of the project are to create an application that would automatically insert embedded river-type data within a database created through the AWS DynamoDB service.

The application also has allow the database to be queried through a graphical interface and display the results on the screen.

## 2 Classes Design

The approach for using the classes was to use each one for creating a resource or interacting with a service not already present.



The scheme of the project classes

The project is based mainly on the "DynamoClientSetup" class, which is needed to obtain and enter data into the Database, and the Data class, which is a class that contains parameters for everything that might be useful for river controls. The properties recorded are, in addition to the value of the measurement and an identifier for it (Time Hash), the values of light, temperature, humidity, and water level.

This data is collected through an Arduino, which provides it through a Json to the program C#, which checks every 5 seconds to see if there is new data input and, if so, it is entered into the Database.

### 3 The program functioning

The interface is based on three forms:

Login to AWS

DynamoDB Login

Access Key ID

Secret Access Key

Arduino Serial Port  
COM19

Login to AWS

The login interface, that allows to log with our credentials and select the Arduino COM's port

Dynamo Embedded Data DB

View Data Trend

From: domenica 18 dicembre 2022

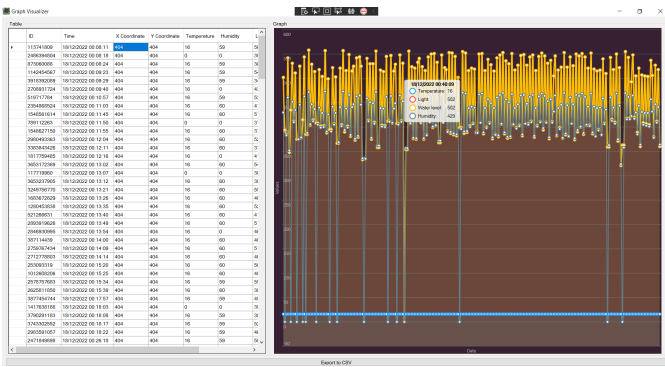
To: domenica 18 dicembre 2022

Show data trend

ID	Time	X Coordinate	Y Coordinate	Temperature	Humidity
2035859564	18/12/2022 02:08:26	404	404	16	0
2842428618	18/12/2022 02:08:20	404	404	16	59
2486329330	18/12/2022 02:08:08	404	404	0	0
1276475749	18/12/2022 02:08:03	404	404	17	59
1927402769	18/12/2022 01:56:19	404	404	16	60
1120833715	18/12/2022 01:56:13	404	404	16	60
1524052706	18/12/2022 01:56:04	404	404	16	60
2061554961	18/12/2022 01:55:59	404	404	16	60
3983869262	18/12/2022 01:55:50	404	404	16	60
92120957	18/12/2022 01:55:45	404	404	16	60
2061161745	18/12/2022 01:55:39	404	404	16	60
3983476046	18/12/2022 01:55:30	404	404	16	60
91727741	18/12/2022 01:55:25	404	404	16	59
495208876	18/12/2022 01:55:16	404	404	16	59
3983607118	18/12/2022 01:55:10	404	404	16	60
2417457641	18/12/2022 01:55:01	404	404	16	59
629361068	18/12/2022 01:54:56	404	404	16	60

Export to CSV

The daily data management screen



The data trend screen, that allows you to see the Data retrieved in a selected time period

## 4 Encountered issues

There were many problems encountered, but the main ones were two:

- The first was getting DynamoDB to communicate with the C# program. The documentation is lacking and sometimes contradictory, which made it particularly difficult to figure out what to do, and one often had to fumble around, succeeding only after many attempts, and it could have been done in much less time if there had been proper documentation and tutorials.
- The second was reading from serial and interpreting the documents, read as json, into c# classes. Here it was difficult to figure out how to do this, and even the Microsoft documentation, though always correct, was often lacking details needed to be able to work.

## 5 Conclusions

DynamoDB has proven to be a very powerful, though certainly unconventional, tool, and the possibilities for use are truly vast.

While this service is not without flaws, it proved to be truly functional and suitable for the project, which allowed us to gain new skills, both with Arduino, Json and DynamoDB, that will surely come in handy in the future and can be reused for new projects.