**STORE MANAGEMENT SYSTEM**

**Preliminary Requirements**

* What is the name of your team?

Full name: Store Management Center Team

Short name: SMC Team

* Who is your team leader?
* Who are your team members?

Stanislav Dinev

Vasil Yoshovski

* What is your project going to be about?
* What features will it consist of? Explain their purpose. (Try to be as granular as possible.)

Feature will provide information about:

* + - Clients
    - Suppliers
    - Products
    - Offers
    - Sales
    - Purchases
* Create a kanban board with the following data, fill it and keep it updated:
  + Name of Feature
  + Feature Owner (who will write it?)
  + Estimated time it would take (in hours, **don't overthink it**)
  + Actual time it took (in hours)
  + Estimated time it would take to unit test (in hours)
  + Actual time it took to unit test (in hours)

This information is available in Trello.com : https://trello.com/smc\_team/home

**Must Requirements**

* Use Code First approach
* Use Entity Framework Core 2.0+
* Use SQL Server 2017
* At least five tables in the SQL Server database
* Provide at least two type of relations in the database and use both attributes and the Fluent API (Model builder) for configuration
* The user should be able to manipulate the database through the client (basic CRUD)
* Provide some usable user interface for the client (preferably console)
* Write unit tests for the majority of your application's features. Unisolated tests are not considered valid.
* Follow the SOLID principles and the OOP principles. The lack of SRP or DI will be punished by death.

**Should Requirements**

* Load some of the data from external files (Either Excel, XML, JSON, zip, etc.) of your choice
  + For XML files should be read / written through the standard .NET parsers (of your choice)
  + For JSON serializations use a non-commercial free library / framework of your choice.
* Generate PDF reports based on your application's data. The PDF doesn't have to be pretty.
  + For PDF reports use a non-commercial free library / framework of your choice.

**Could Requirements**

* You could use Service Layer of your choice