Game Market

Text

Description automatically generated with medium confidenceResearch

12/06/2022 Eindhoven

Version: 1.0

Members:

Lars Kluijtmans: 4220269

Table of Contents

[Version 3](#_Toc103347099)

[Introduction 4](#_Toc103347100)

[Research Questions 5](#_Toc103347101)

[Problem 1: 5](#_Toc103347102)

[Major question 1: 5](#_Toc103347103)

[Minor questions 1: 5](#_Toc103347104)

[Problem 2: 8](#_Toc103347105)

[Major question 2: 8](#_Toc103347106)

[Minor questions 2: 8](#_Toc103347107)

# Version

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Date | Autor | Changes | Status |
| 0.1 | 03/04/2022 | Lars Kluijtmans | Made new document for research question. | Not finished |
| 1.0 | 12/06/2022 | Lars Kluijtmans | Made minor changes | Finished |
|  |  |  |  |  |

# Introduction

Game market is an online market place where you can buy other users and sell to other users used games. With a simple website that is easy to navigate and is reliable.

# Research Questions

## Problem 1:

What are model and DTO (Data transfer object) used for.

## Major question 1:

When to use a model and when to use a DTO

## Minor questions 1:

What is a model?

A model is used to separate the controller from accessing the data directly according to the MVC flow (Model-view-controller). It is described as representing the business layer of a application. (Choudary, 2021)

Others referee to it as an Model Object or MO for short, these are mapped closely to a corresponding database table and are often used as return values for a Data Access Object. (Java Practices, 2018)

Although the two definitions may seem similar one uses a model a object that can be transferred between the business and database layers but also refers to it as representing the business layer and the other describes it as closely related to the database even saying that it is “often map roughly to the records of a corresponding database table.“. (Java Practices, 2018)

I would have to conclude that a model differs in function depending on the contest it is used in. If it appears related to a MVC flow then it has the function of a transitional object that is used in transferring data from the database layer to the business layer. If found in the context of a Model Object or used in reference to a Data access object it will be more closely related to the database layer and even the database structure itself.

What is a DTO?

DTO is a abbreviation for data transfer object. The DTO pattern is used to reduce the number of methods called by combining multiple variables in one call, these contain no login and are used as storage. They are often composed of multiple models or parted of them. (Daeldung, 2022)

The DTO pattern is also mentioned often as a “objects that aggregate and encapsulate data for transfer”. It should contain serialization and deserialization mechanisms but no logic. The DTO are custom made to only send the data that is needed and to not include sensitive information increasing the security of the application. (Seniuk, 2020)

In conclusion a DTO is used as a collection of data that can be composed of multiple models or variables that include exclude unrequired or sensitive data and so increases security and speed. It also decreases the number of requests needed.

Are there alternatives to model and DTO?

After some research I found that there are other types of object except models and DTOs:

Entity: It MUST have its own identity and can be uniquely identified. Object that can be loaded from and saved to a database using a DataContext. Cannot exist without its DataBaseContext. (Joshi, 2019)

View model: Model that is exposed to View (Joshi, 2019)

I would conclude from this that there are different types of classes with different functions but none are an real replacement for the DTO class.

In regards to the model I think that the entity class might be a good substitute for it. It is more specific in what it does. Get the data from one table of the database and store it. While the model as seen before has a different function depending on who you ask.

Why would you convert a model to a DTO?

We should not expose the internal persistence models to the outside worlds. (PHALTANKAR, n.d.)

DTO give us benefit to hide our entity from client that using API and help us to maintain our code if we want to change our internal database structure. (adendrata, 2019)

The above statements are very clear, models(entities) are supposed to be hidden and never send out to the user. To be able to more easily enforce this models(entities) are converted to DTOs that hold only relevant data, preventing the exposing of none relevant or sensitive data to the outside world.

What are the advantages and disadvantages of a model?

Because the usage of the model is already vague in itself as seen in minor question 1, I can not answer this question. As different people us it for different purposes the actual use, advantages and disadvantages change depending on who you ask.

What are the advantages and disadvantages of a DTO?

I think this question has already been answered in the above questions but to recap:

Proper usage of DTOs prevents the leakage or confidential or sensitive information/data.

It shortens the time needed to send responses from the API because it only holds relevant data and cuts out anything unnecessary.

But it’s disadvantage is that it’s very specific and made only for one purpose. So they can not be reused.

When to convert a model to a DTO?

The service layer should to the conversion between DTO and Entities. (PHALTANKAR, n.d.)

The evidence is conclusive, the service layer is where the model(entity) should be converted to the DTO and vice versa.