

Service Oriented Architecture – GameStore

Horea Manita, 258

1. System Description

The application I developed is an Online GameStore that provides relevant information about current top selling games and also links for purchasing such games. It furthermore provides the possibility to add custom games with custom information as well.

The application is targeted at gaming enthusiasts, who are interested in getting their best prices for the games they are interested into buying.

2. System Architecture

The GameStore application has at its core the service-oriented architecture and consist of two main components:

- 1) The server, called GamesWs, is essentially a web application that also has two main functions:
 - i) Firstly, the GamesWs application firstly uses the web services provided by the SteamSpy application server (<https://steamspy.com/api.php>), that offers precise information about the current top 100 most popular games, their genres, developer and publisher teams, rating, current prices and discounted prices and also provides a link for buying the game from the official game store.
 - ii) Secondly, the GamesWs application offers all the data gathered from the application server mentioned at i) and exposes it as well through REST services. The application not only offers the current top 100 most popular games, but also has other operations such as getting more detailed individual information about the games in question, add new games to the list and update already existing game information.
- 2) The client, called GamesCl is the one that consumes the REST services exposed by the server described at ii). The GamesCl is also a web application that provides the user with an UI that is easy to understand and to interact with.

3. Models describing the system

UML Diagram

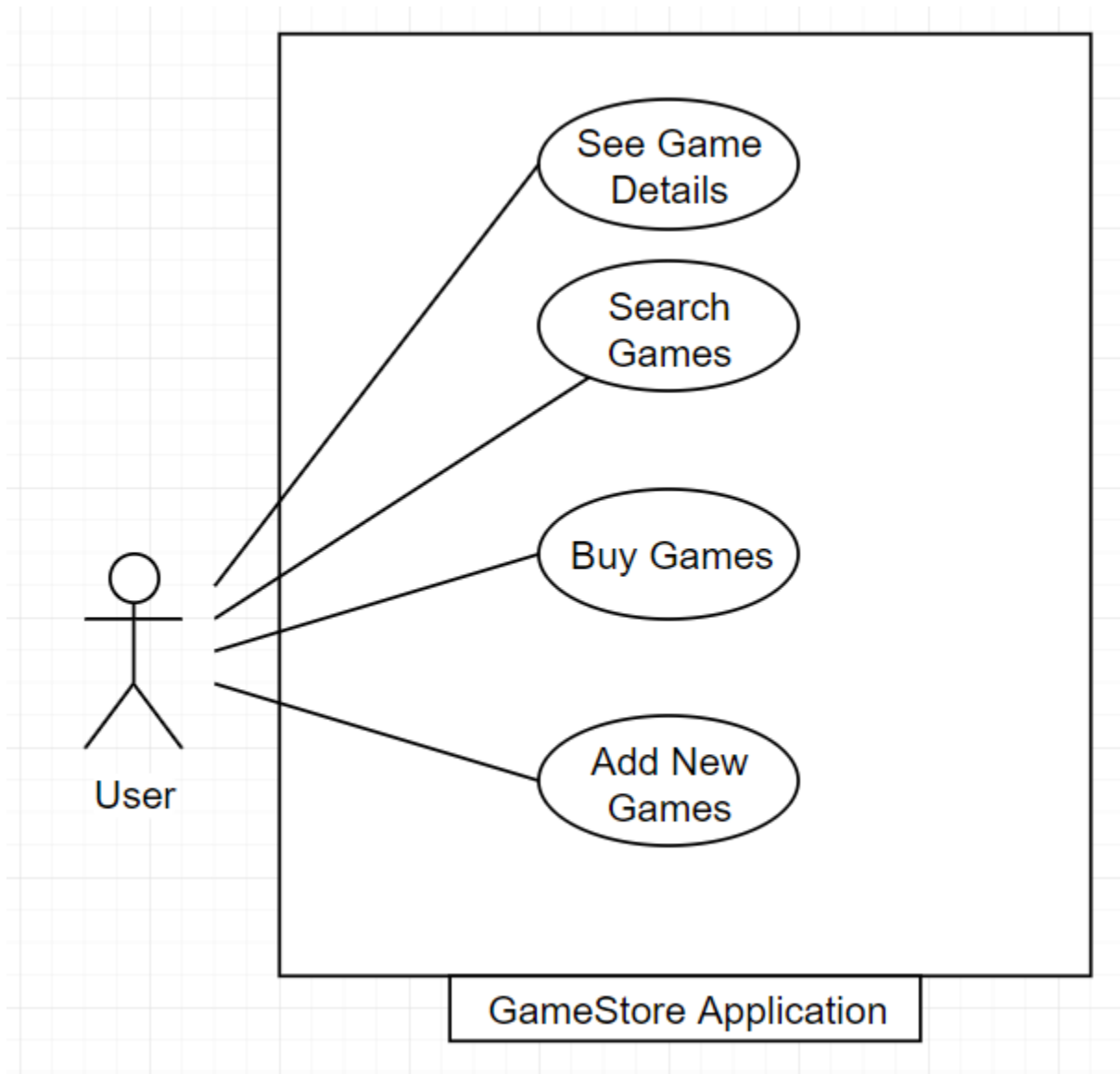


Figure 1 – Use Case UML Diagram

SoaML Diagram

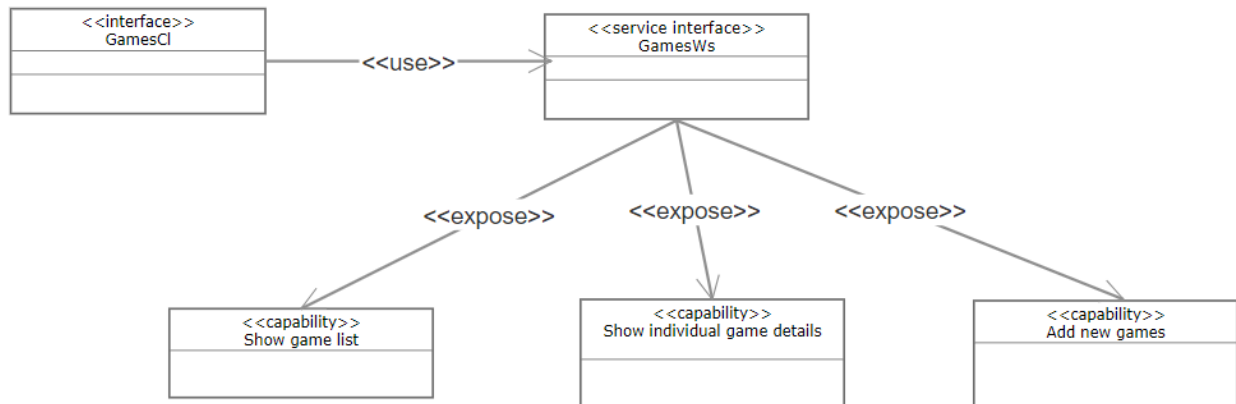


Figure 2 – Service Interface SoaML Diagram

BMM Diagram

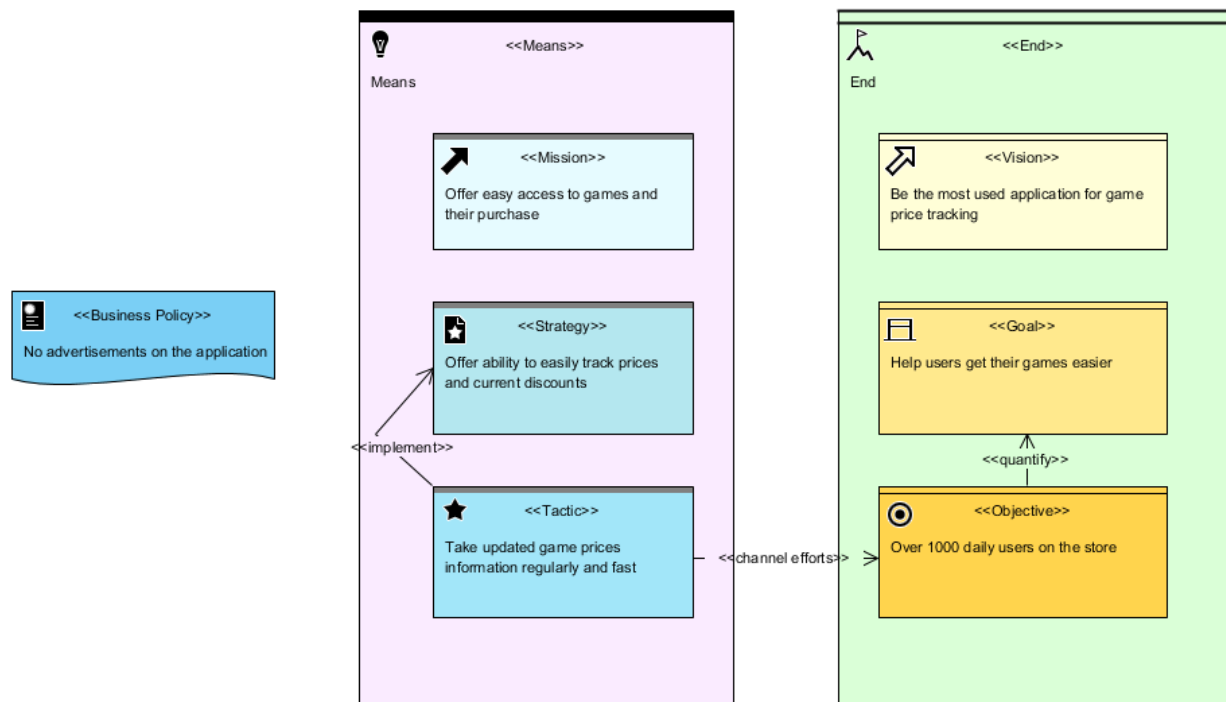


Figure 3 – BMM Diagram

BPMN Diagram

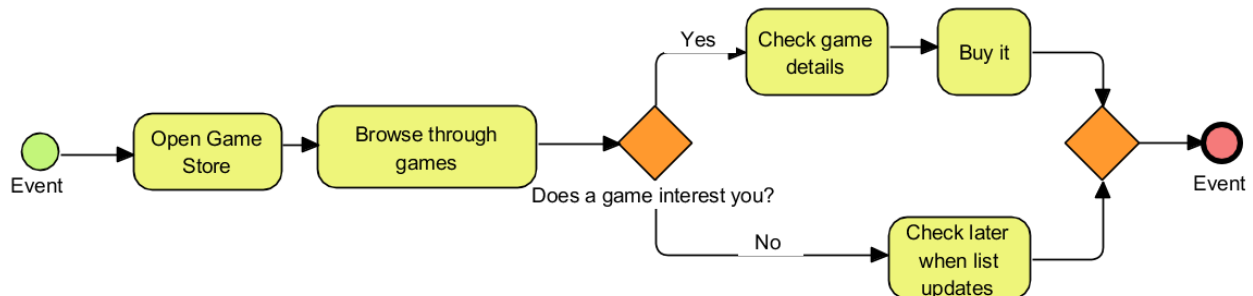


Figure 4 – BPMN Diagram

4. SOA Patterns

Service Façade Pattern

A service façade sits between a service and a contract. It eliminates the tight coupling between the service and its contract. This is intended to minimize changes to the service if the contract changes. In the case of the GameStore application, the façade calls the services from the GamesWs and sends the data to the Controller from the GamesCl.

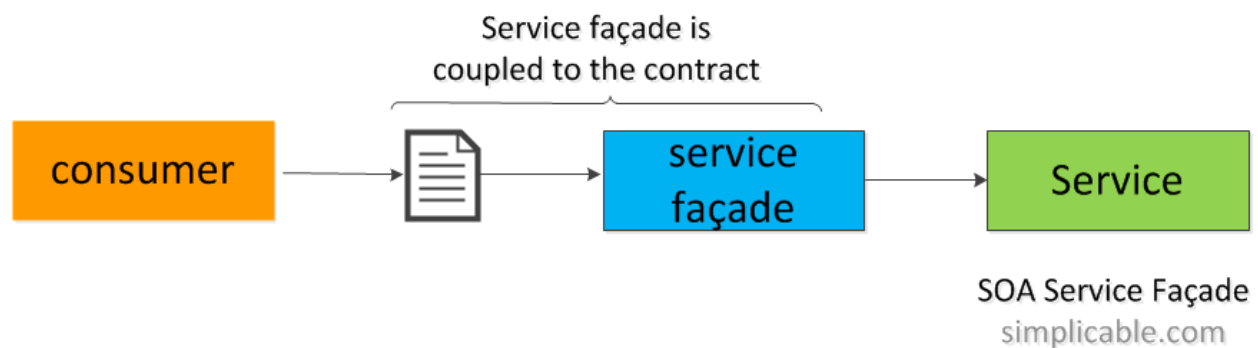


Figure 5 – Façade Pattern; source: simplicable.com

In our case the consume would be the GamesCl and the Service would be the GamesWs.