# SoftUni Game Store – 5 March 2017

The game store is a digital platform where the users can buy games. Your task is to create a web application that has back-end and front-end logic using the provided resources (**HttpServer**, **MVC Framework**, **Bootstrap, SQL script** and views). You are free to edit those provided resources to fit your needs.

## Data Models

Create the required **entities**. Use the appropriate **data types**.

* The system contains information about **users**, and **games**
* Users can **register** in the system. After successful registration, the user has **email, password, full name, list of games** and information whether he is an **administrator or not**.
* The **first registered user** becomes also an **administrator**. You can manually mark users as admins in the database.
* A **game** has **title**, **trailer** (YouTube Video Id), **image thumbnail** (URL), **size**, **price, description** and **release date. Each game can be owned by multiple users.**

## Functionality

* **All users** can view the home page.
* **All users** can view details page of each game
* **Guests** (anonymous users) can register an account with their own email and password
* **Guests** can login by email and password.
* **Logged-in users** can logout.
* **Logged in users** can buy games.
* **Administrators** can add, edit or delete games
* When guest user tries to access a page that is allowed only for logged in user he should be **redirected to the login page**
* When user tries to access the administrator pages he must be redirected to the home page.

## Design the Database

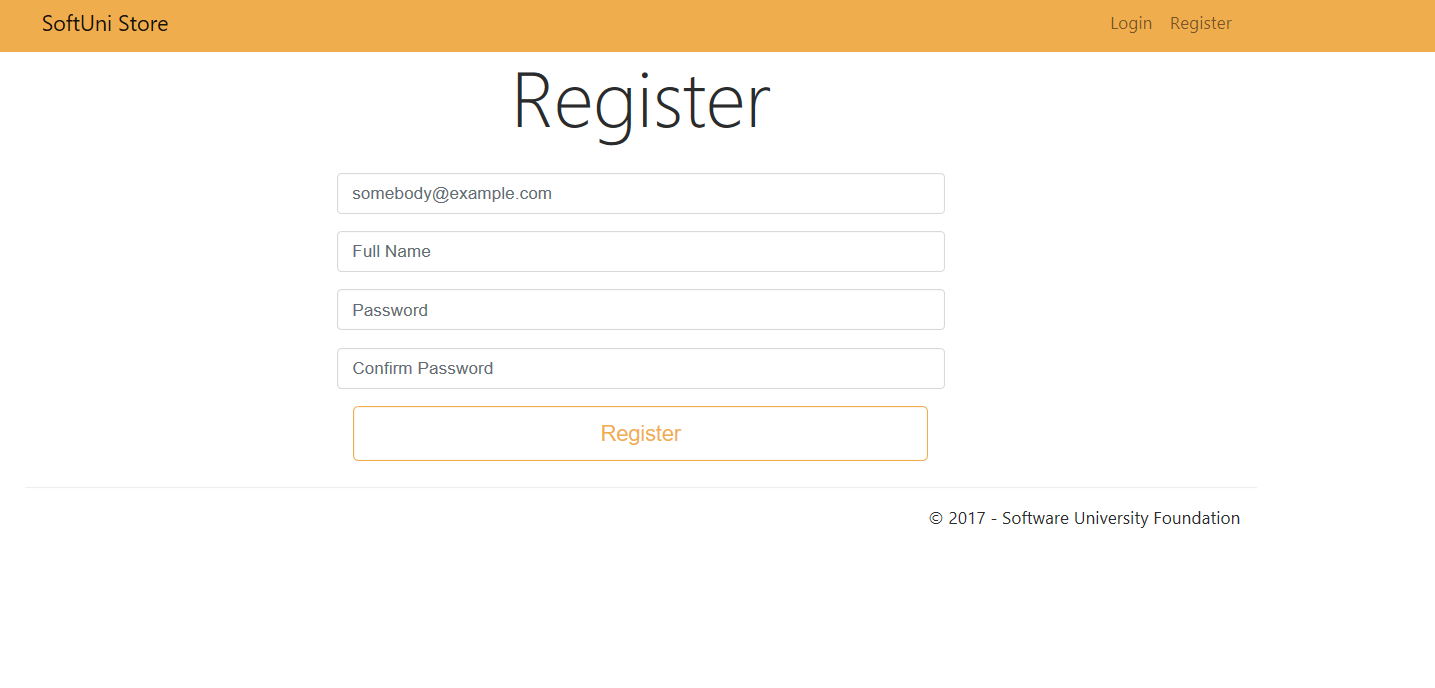
Design **entity classes** and create a **database** to hold the **users**, **games**.

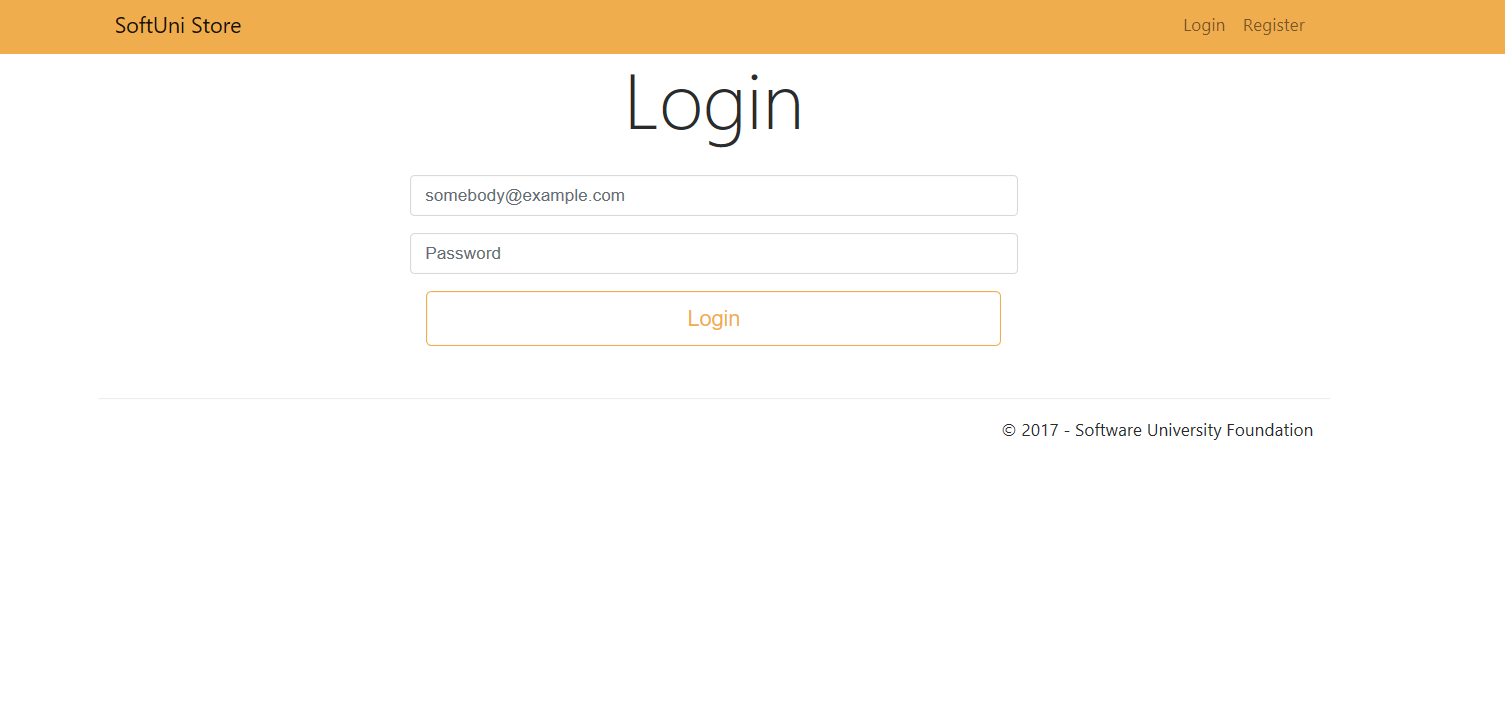
## Implement User Registration, Login and Logout

The guest users can register and log in the forum. He can provide to:

* **Register**
  + **Email** – must contain **@** **sign** and a **period**. It must be unique.
  + **Password** – length must be **at least 6 symbols** and must contain at least **1 uppercase**, **1 lowercase** letter and **1 digit**
  + **Confirm Password** – must **match** the provided password
  + **Full Name –** should **NOT** be an empty string
* **Login**
  + User can log in with **email** and **password**
* **Logout**
  + When logged in the user should have option to **log out**

After register, the user should be redirected to the login page. If the register is not successful, he should be redirected back to the register page. All validations should be performed on the back-end. If a registered user tries to access the GET/POST of Register or Login, he should be redirected to the homepage.





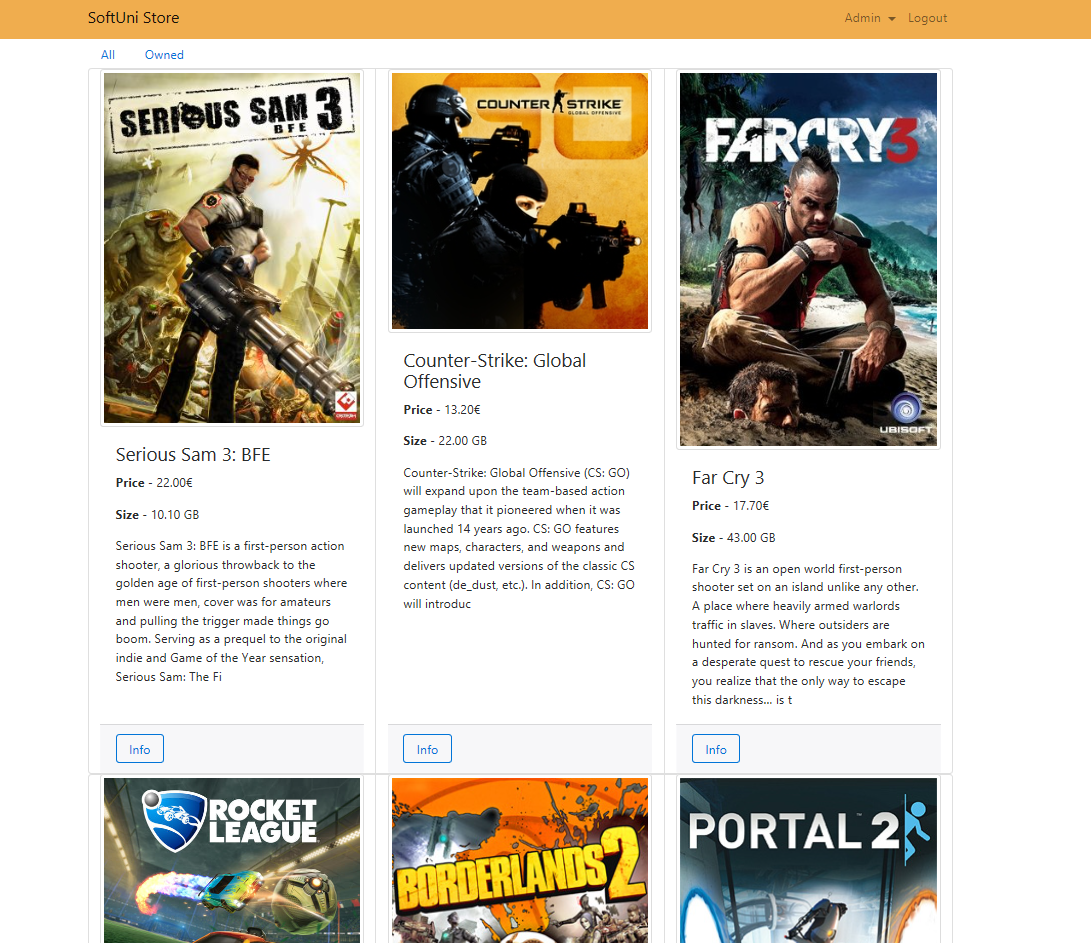
## Implement Home Page

Home page should be accessible for **both users and admins,** when clicking on the SoftUni store title. It should show a list of **all games** (thumbnail, title, price, size, description) with option to each game to **buy** it and see more **detailed information** for the game. In addition, there should be options to **filter** the games:

* All Games - default
* Owned Games

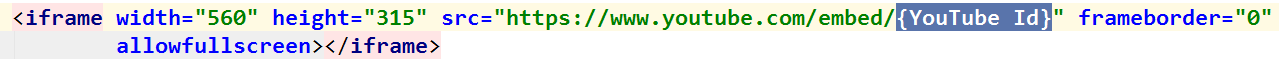
Only the first **300 characters** of the description for each game must be showed on the home page. There should be no more than **3 games per row**.

In the resource we have given you an SQL script that will populate your database with 20 sample games.

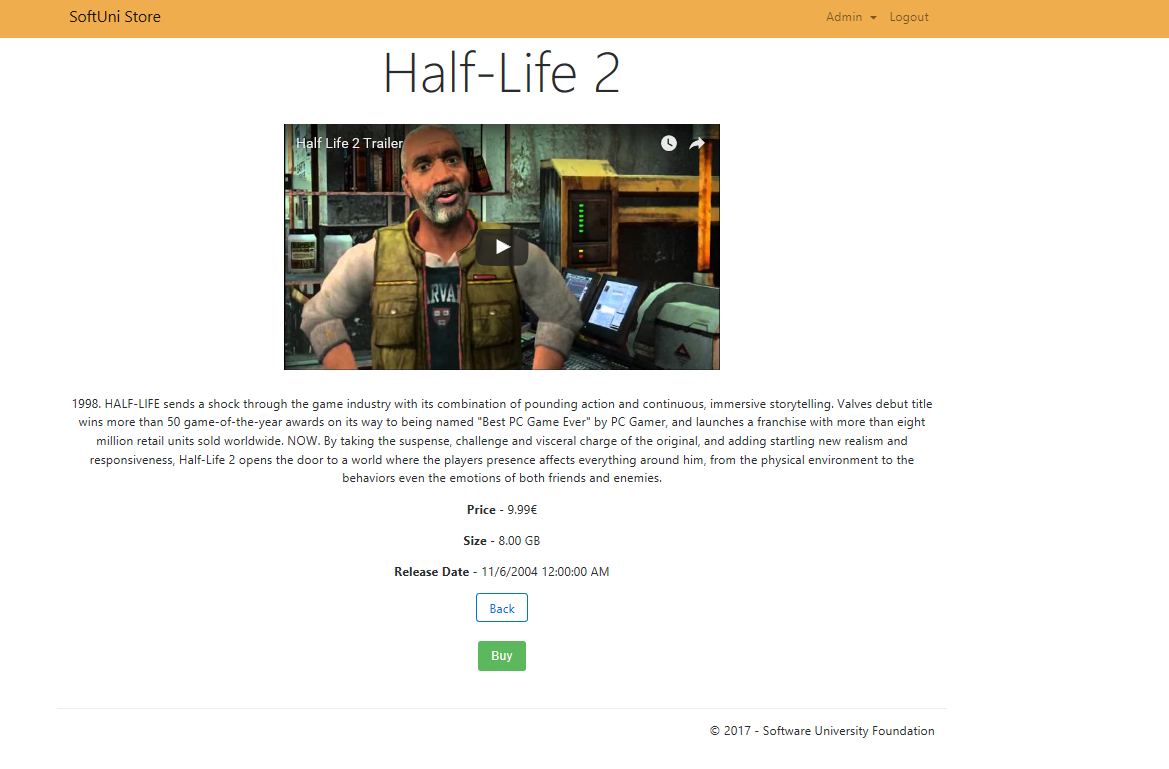


## Implement Game Details Page

Each game should have detailed page where the users can see **all details** for that game when click on the **info** button. This includes an embedded YouTube video. You can do it, using the YouTube Video Id, that you keep for each game. You are given the following template:



You need to replace the placeholder with the actual video id. Here is how it should look:



As you can see here is the functionality to buy the game for the current user. Implement it. Once the game has been bought, it should appear in the Owned (games) filter in the home page.

## Implement Managing Games

If the user is not admin, you should redirect him to the homepage when trying to access the pages below. As an admin, you have the option to **add games to the catalog**. A game should be added only to the catalog if matches the given criteria below, elsewhere the user should be redirected to the same page (newgame, editgame). If the adding/edit is correct, redirect to the all games page:

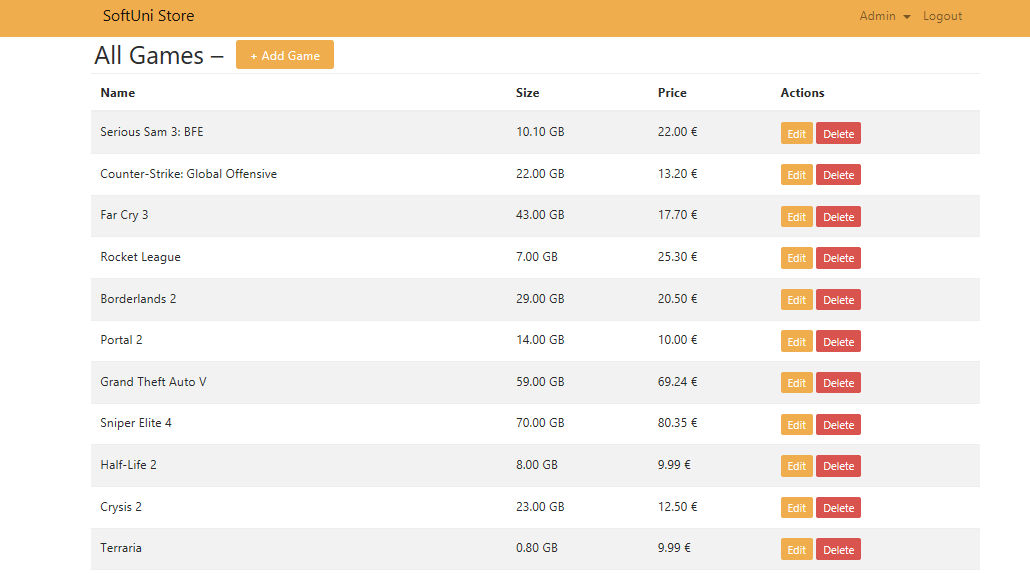
* **Title** – has to begin with **uppercase letter** and has length between **3 and 100 symbols** (inclusive)
* **Price** – must be a **positive number**
* **Size** – must be a **positive number**
* **Trailer**– only videos from YouTube are allowed and only their **ID** should be saved to the database which is a string of exactly **11 characters**.

For example, if the URL to the trailer is https://www.youtube.com/watch?v=edYCtaNueQY, the required part that must be saved into the database is edYCtaNueQY. That would be always the last 11 characters from the provided URL.

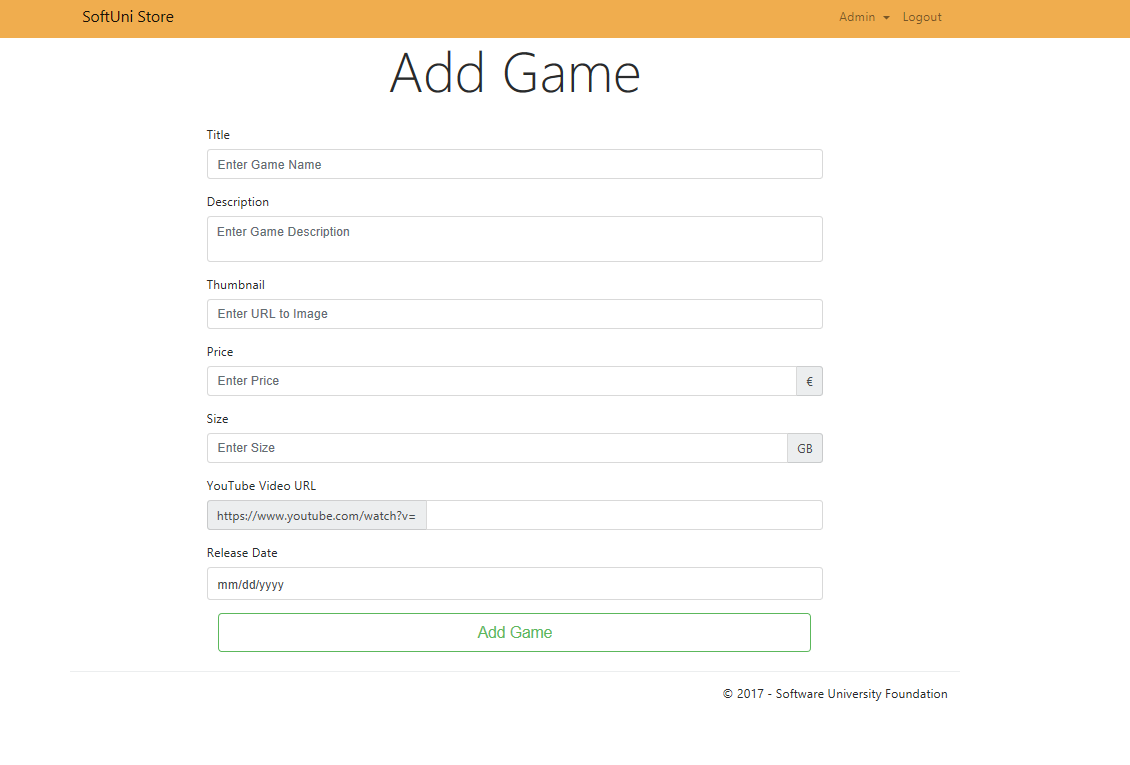
* **Thumbnail** **URL** – it should be a plain text starting with **http://**, **https://**
* **Description –** must be at least 20 symbols

All validations must be performed on the back-end.

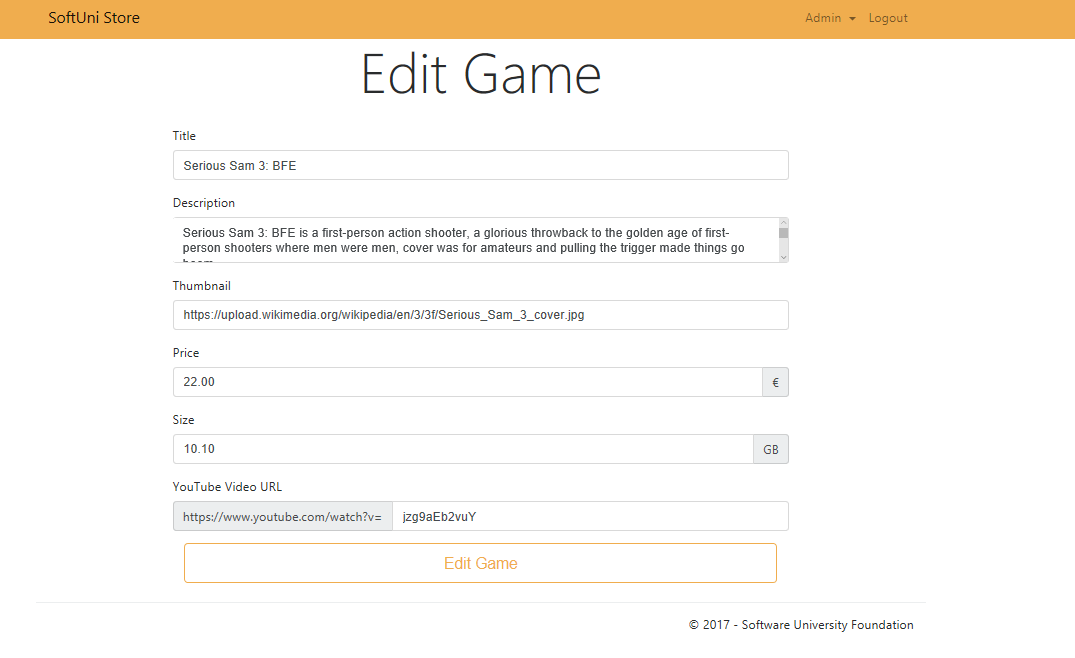
Listing all games:



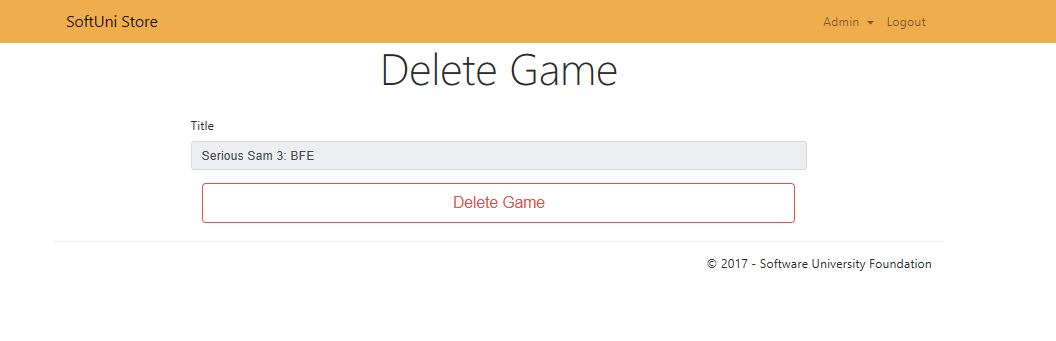
Adding a game:



Editing a game:



Deleting a game:



## Project Infrastructure Bonus

Bonus points code quality / good application structure / additional effort.

* Bonus points for implementing **separate data layer**.
* Bonus points for using **dependency inversion**.
* Bonus points for using **AutoMapper**.
* Bonus points for using **services** for different jobs.
* Bonus points for implementing **Repository** or **Unit of Work** pattern

-username – placeholder

-delete?

-details

-butonite! #