Online Game Store

TASK #2

EPM-OGST

2 TASK #2 DESCRIPTION

Extend functionality of the Game Store by add possibility to display and add new content.

2.1 GAME

2.1.1 Add Game

URL: /games/new

Method: GET

Expected Result:

- HTML page should contain HTML Form with all field which required to add new “Game”.

URL: /games/new

Method: POST

Expected Result:

- Execute server side validation (all fields are mandatory);
- If form validation passed successfully, proceed with creating new “Game” and redirect user to URL: “/games”;
- If form validation fails, stay on the current page and display validation error(s) as text.

Additional details:

Use Html.EditorForModel helper

2.1.2 View Game Details

URL: /game/{key}

Method: GET

Expected result:

- Display details of the game.
- Details should contain link to game download (see below).
- Change action result to view result.

Hint:

- Change action result to view result.

URL: /game/{gamekey}/download

Method: GET

Expected result:

- User should be redirected to the new tab.

URL: /game/{gamekey}/comments

Method: GET

Expected result:

- Under details of the game, show link to (change POST URL: /game/{gamekey}/comments action to allow GET URL: /game/{gamekey}/comments).

2.1.3 Update Game Entity

Extend “Game”. Now, it should contain the following fields:

Property name	Type	Comment
Key	String	unique; (alias for link crating)
Name	string	
Description	String	
Price	Money	
UnitsInStock	Smallint	
Discontinued	Bit	

Make sure that “Views” are updated accordingly.

2.2 COMMENTS

2.2.1 View Comments

URL: /game/{gamekey}/comments.

Method: GET

Expected result:

- Display list of comments on that view as HTML list.
- Below that list add form with target POST URL: /game/{gamekey}/newcomment.

Hint:

- Change action result to view result.

2.2.2 Add Comment

URL: /game/{gamekey}/newcomment

Method: POST

Expected result:

- Proceed server-side validation (all fields are required)
- if input is valid then proceed add comment and render view list of comments for game with new one, else render current view with list of comments and display validation error (as text message)

2.3 PUBLISHER

2.3.1 Entity Publisher

Create Entity Publisher. It should contain the following fields:

Property name	Type	Comment
CompanyName	nvarchar(40)	
Description	Ntext	
HomePage	Ntext	

2.3.2 View Publisher Details

URL: /publisher/{CompanyName}

Method: GET

Expected result:

- Display details of the publisher.
- Update pages with game details to view publisher name as a link to publisher details.

2.3.3 Add Publisher

URL: / publisher/new

Method: GET

Expected result:

- Should be html form on view with target POST URL: /publisher/new.
- Form should contains necessary inputs to create publisher.

2.4 ORDERDETAILS

Crete new entity OrderDetails. It should contain the following fields:

Property name	Type	Comment
Product ID	nvarchar(40)	
Price	Money	
Quantity	Smallint	
Discount	Real	

2.5 ORDERS

Create new Entity Orders.

Expected result:

- Each order should contain customer ID and Order Date;
- Entity “Order” has “1-to-many” relationship with “OrderDetails” entity;

2.6 BASKET

2.6.1 View Basket

URL: /busket

Method: GET

Expected result:

- Details of current order should be displayed;

2.6.2 Add Game into Basket

URL: /game/{gamekey}/buy

Method: GET

Expected result:

- Item should be added to the basket.

2.7 CACHE

- On the layout in top part of the page, display “total numbers of games”. Cache this data for 1 minute.
- Remove cache from “get all games” and “get game by key” actions.

2.8 GENERAL REQUIRMENTS

- Implement Views for all JSON responses;
- Implement CRUD for all domain entities;
- Add link to “basket” on all pages;

2.9 OPTIONAL

- Try to use TDD or BDD as development process;