

סביבות פיתוח – עבודה 4

בחרנו לממש חנות למכירת סרטים.

המערכת תשמש ל:

1. מכירת סרטים לפי קטגוריות (מתח, אימה, דרמה וכו').
2. ניהול מאגר משתמשים.
3. מעקב אחר רכישות והזמנות.

למערכת יהיו 3 ממשקים:

1. ממשק לקוח (Login + Registration).
2. ממשק מוצרים.
3. ממשק עגלת קניות.

<u>ID</u>	<u>Method Name</u>	<u>HTTP Method</u>	<u>Parameters</u>	<u>Returns</u>	<u>Explanation</u>
1	Login	POST	<pre>{ "UserName": , "Password": }</pre>	Bool	<p>This method login to the system.</p> <p>Using POST because the information which sent to the server is private.</p> <p>True = login succeeded.</p> <p>False = login failed.</p>
2	Registration	POST	<pre>{ "UserName": , "FirstName": , "LastName": , "Email": , "Address": , "Country": , "FavoritCategories": , "IsAdmin": , "Password": , "PetsName": }</pre>	Bool	<p>This method register guests to the system.</p> <p>Using POST because the information which sent to the server is private and we change information in the database.</p> <p>True = registration succeeded.</p> <p>False = registration failed.</p>
3	updateRating	POST	<pre>{ "MovieID": , "Rating": }</pre>	Bool	<p>This method update the rating of an existing movie.</p> <p>Using POST because we change information in the database.</p> <p>True = success.</p> <p>False = fail.</p>
4	getAllProducts	GET	-	<p>Return all products:</p> <pre>[{"MovieID": , "MovieName": , "MovieCategory": , "MovieDirector": , "Price": , "Month": , "Year": , "AmountInStock": , "Rating": },]</pre>	<p>This method return all movies.</p> <p>Using GET because there is no information to deliver to the server or any wish to change any information.</p>

5	getAllCategories	GET	-	Return all Movie's Categories: [{"CategoryName": }, {"CategoryName": },]	This method return all movie's categories. Using GET because there is no information to deliver to the server or any wish to change any information.
6	getAllCountires	GET	-	Return All Countries: { "ID": , "Name": },...	This method return all countries. Using GET because there is no information to deliver to the server or any wish to change any information.
7	getTop5HotProducts	GET	-	Returns top 5 hottest products (by rating) : [{"MovieID": , "MovieName": , "MovieCategory": , "MovieDirector": , "Price": , "Month": , "Year": , "AmountInStock": , "Rating": },]	This method return the top 5 hottest products. Using GET because there is no information to deliver to the server or any wish to change any information.
8	answerPersonalQuestions	POST	UserName, Address, PetName	Bool	This method send answers to the server. Using POST because the information which sent to the server is private. True = success. False = fail.
9	resetPassword	POST	Username, NewPassword, ConfirmPassword	Bool	This method reset user's password. Using POST because the information which sent to the server is private. True = success. False = fail.

10	getLatestMonthlyProducts	GET	-	Return all movies from the last month: [{"MovieName": , ... }]	This method return all new products from the last month. Using GET because there is no information to deliver to the server or any wish to change any information.
11	getProductsByCategory	GET	-	Return product and his category: [{"MovieName": , "MovieCategory": , },]	This Method return a list of products by categories. Using GET because there is no information to deliver to the server or any wish to change any information.
12	RecommendedProductsForUser	GET	UserName	[{ "MovieName": , ... }]	This Method return a list of recommended products for the specific user. Using GET because the information which sent to the server is not private.
12	getProductDetails	GET	MovieID	Return movie details: [{"MovieID": , "MovieName": , "MovieCategory": , "MovieDirector": , "Price": , "Month": , "Year": , "AmountInStock": , "Rating": }]	This Method return the product's details. Using GET because the information which sent to the server is not private and we don't change information in the database.
13	addProductToCart	POST	{ "UserName": , "MovieID": , "Amount": }	Bool	This Method add product to cart. Using POST because there is information to deliver to the server and we wish to change information in the database.
14	removeProductFromCart	DELETE	{ "UserName": , "MovieID": }	Bool	This Method remove product from cart.

					Using DELETE because the product is being deleted from user's cart.
15	showCart	GET	UserName	Return user's cart with the following fields: [{"MovieId": , "MovieName": , "Amount": , "Price": },]	This Method return the current products in cart. Using GET because the information which sent to the server is not private and we don't change information in the database.
16	getMyOrders	GET	UserName	Return user's orders with the following fields: [{"OrdreID": , "OrderDate": , "ShipmentDate": , "TotalAmount": , "TotalPrice": },]	This Method return all orders history. Using GET because the information which sent to the server is not private and we don't change information in the database.
17	Purchase	POST	UserName, CreditNum, ShipmentDate	Bool	This method add record to Orders Table and remove this record from Cart Table. Using POST because the information which sent to the server is private and we change information in the database. True = success False = fail
18	checkInStock	GET	MovieID	Bool	This method checks if the product exists in stock. Using GET because the information which sent to the server is not private and we don't change information in the database. True = success False = fail

19	getAllUsers	GET	-	Return all users: [{"UserName": , "FirstName": , "LastName": , "Email": , "Address": , "Country": , "FavoriteCategories": , "isAdmin": , "PetName": },]	This method return list of users. Using GET because there is no information to deliver to the server or any wish to change any information.
20	checkIsAdmin	GET	UserName	Bool	This method checks by the current login, if the user is Admin. Using GET because the information which sent to the server is not private. True = user is admin. False = user is not admin.
21	getOrdersReport	GET	-	[{ "OrderID"; "UserName"; "OrderDate"; "ShipmentDate"; "TotalAmount"; "TotalPrice": }]	This method return the order reports. Using GET because there is no information to deliver to the server or any wish to change any information.
22	editProductDetails הערה: המשתמש חייב להזין את כל הפרמטרים. גם אם ישנם שדות שהוא לא שינה.	PUT	movieID, Price, AmountInStock, Rating	Bool	This method change the product details. Using PUT because we updating existing resource.
23	getUser	GET	UserName	Return specific user: [{"UserName": , "FirstName": , "LastName": , "Email": , "Address": , "Country": , "FavoriteCategories": , "isAdmin": ,	This method return specific user details. Using GET because the information which sent to the server is not private and we don't change information in the database.

				"PetName": }]	
24	removeUser (Only admins)	DELETE	UserName	Bool	This method remove user from the system. Using DELETE because the user is being deleted from Users' table. True = success False = fail
25	addProductToStock	POST	{ "MovieName": , "MovieCategory": , "MovieDirector": , "Price": , "Month": , "Year": , "AmountInStock": , "Rating": }	Bool	This method return the updated stock by the product. Using POST because there is information to deliver to the server and we wish to change information in the database.
26	removeProductFromStock	DELETE	MovieID	Bool	This method remove product from the stock. Using DELETE because the product is being deleted from Products' table. True = success. False = fail.
27	checkAmountInStock	GET	-	Return the amount of each movie in stock: [{"MovieID": , "MovieName": , "AmountInStock": },]	This method return the amount of each movie in stock. Using GET because there is no information to deliver to the server or any wish to change any information.