**סביבות פיתוח – SERVER**

בעבודה זו אנו נבנה חנות וירטואלית לבגדים.

התייחסנו גם לסעיף הבונוס בעת המענה.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Explanation** | **Returns** | **Parameters** | **HTTP**  **Method** | **Method Name** | **ID** |
| Login cannot pass in GET.  The user entered sensitive details (password) and this is not secure, because other people can get to this details too. **If success returns the userName else returns userName or Password is wrong.** | Bool | Username: char, Password: char | POST | Login | **1** |
| Produces are related to the seller not to the client.  Using GET because there is no information to deliver to the server or any wish to change any information. | Returns all products : List<Product> | - | GET | AllProducts | **2** |
| Products are related to the seller not to the client.  Using GET because there is no information to deliver to the server or any wish to change any information.  **The server send the list of the 5 hot products.** | Returns the 5 hot Products from the last week:  List<Products> | - | GET | Hot5Products | **3** |
| Registration cannot pass in GET We create a new user, and its need to be stored in the server memory, so it’s a POST method.  **return True if the registration success and update the last entry, else if the registration failed return false.** | Bool | userName: char,  password : char,  firstName : char,  lastName : char,  gender: char,  email: char,  phone: int,  Birthdate : Date,  country : char,  address : char,  SecurityAnswer : char,  shirts: bool,  pants: bool , dresses : bool, skirts : bool , belts : bool , shorts : bool, underwears : bool,  hats : bool , bathing suits : bool ,  scarfs : bool . | POST | Registration | **4** |
| Forget Password cannot pass in GET  The server check the answers of the user for identification and then if success – show the password.  The user can not get the password without the answers, and when he succeeded to get it its need to be secured. So the POST method is the only option.  **if the answer matches to the username return the user’s password, else return an error message** | string | Username : char, answers : char | POST | ForgetPassword | **5** |
| Products are related to the seller not to the client.  Using GET because there is no information to deliver to the server or any wish to change any information.  **The server send the list of the new products from the last month.** | Returns list of all **new Products from the last month:**  List<Product> | - | GET | NewProducts | **6** |
| Products are related to the seller not to the client.  Using POST because there is information to deliver to the server.  **The user insert the category and the server returns the list of products in this category**. | **Returns list of products by category:**  List<Product> | Category : char | POST | ProductsByCategory | **7** |
| Using POST because there is no information to deliver to the server or any wish to change any information. | Returns list of sort **products by typeOfSort**:  List<Product> | sortedBy : char | POST | SortProducts | **8** |
| Products are related to the seller not to the client.  Using POSTbecause there is no information to deliver to the server or any wish to change any information.  **The user insert the id of himself and the server returns the list of products in the category that the user interested in.** | **Returns list of recommended Products by id of user:**  List<Product> | userId : int | POST | Recommended  Products | **9** |
| Products are related to the seller not to the client.  Using POST because there is no information to deliver to the server or any wish to change any information.  **The user enter the name of product that he want to get, and the server returns it to him.** | Returns the product with the same name  List<Product> | nameOfItem : char | POST | SearchProductBy  Name | **10** |
| Products are related to the seller not to the client.  Using GET because there is no information to deliver to the server or any wish to change any information.  **The user enter the color of product that he want to get, and the server returns a list of product with the same colors.** | Returns list of product with the same colors  List<Product> | ColorOfItem : char | POST | SearchProductBy  Color | **11** |
| Products are related to the seller not to the client.  Using POST because there is no information to deliver to the server or any wish to change any information.  **The user enter the size of product that he want to get, and the server returns a list of product with the same size.** | Returns list of product with the same size  List<Product> | SizeOfItem : char | POST | SearchProductBy  Size | **12** |
| Products are related to the seller not to the client.  Using POST because there is no information to deliver to the server or any wish to change any information.  **The user enter the size of product that he want to get, and the server returns a list of product with the same size.** | Returns list of product with the same size and color  List<Product> | SizeOfItem : char  ColorOfItem : char | POST | SearchProductBy  SizeAndColor |  |
| Orders are related to the server. using POST to get the username.  return the all orders of the user. | Return list of orders: List<Order> | UserName: char | POST | ShowPastOrders | **13** |
| Using POST to get details about the order and add a new order, remove the cart from the carts. | bool | Order Id: int, UserName: char, Order date: date, Shipment Date: date, Currency:varchar | POST | AddOrder | **14** |
| Using POST to get details about the user and the item and add a new item to cart.  returns true if success and false if failed. | Bool | UserName: char, ItemId: int, ammount: int | POST | AddToCart | **15** |
| Carts are related to the server. using POST to get the username.  return the all orders of the user. | Return list of products: List<Product> | UserName: char | POST | DisplayCart |  |
| Using POST to get details about the user and the item to remove and remove all the items with the itemId from the cart.  returns true if success and false if failed. | Bool | UserName: char, ItemId: int | POST | RemoveFromCart | **16** |
| Using POSTbecause there is no information to deliver to the server or any wish to change any information.  **The client enter the product id and the server returns if the products available or not.**  **return True if the product available else returns False.** | bool | Product id :int | POST | IsProductAvailable | **17** |
| Using GET because there is no information to deliver to the server or any wish to change any information.  **The server return list of the users in the system.** | Returns all users: Users[] | - | GET | ListUsers | **18** |
| Using Delete to delete information from the database of the server.  Return if the action success or fail.  **if the server succeeded to delete the user it return true, else it returns false.** | Bool | UserId : int | DELETE | DeleteUser | **19** |
| Create new user. Using POST to update new information in the server.  **return if the function success to add a new user or not .**  **returns : True = success**  **False = fail** | bool | userName: char,  password : char,  firstName : char,  lastName : char,  gender: char,  email: char,  phone: int,  Birthdate : Date,  country : char,  address : char,  SecurityAnswer : char  shirts: bool  pants: bool , dresses : bool, skirts : bool , belts : bool , shorts : bool, underwears : bool , hats : bool , bathing suits : bool , scarfs : bool . | POST | AddUser | **20** |
| Using Post because we should update the server with the new user that added  if the delete succeeded it returns true else the server returns False. | bool | Username : int | POST | DeleteProduct | **21** |
| Create new product, insert the details about this product so we use the POST method for insert the information to the server.  **if the add function was succeeded it returns true else the server returns False.** | bool | Id: char,  name : char  ,size: char ,  color:char ,  quantity : int,  price : money,  hot : bit, category :char,  day: int , month : int, year : int. | POST | AddProduct | **22** |
| Using PUT to update the information about product supply in the server. And then if the action succeeded or failed.  **The server update the quantity of an item. it the update was succeeded return true, else return false.** | bool | Product Id :int , quantity : int | PUT | UpdateStock | **23** |
| Add to cart new product .  we use the POST method for insert the information to the server.  **if the add function was succeeded it returns true else the server returns False.** | bool | Product id : int, User id : int | POST | AddToCart | **24** |
| Remove from cart a produce.  we use the POST method for insert the information to the server.  **if the remove function was succeeded it returns true else the server returns False.** | bool | Product id : int, User id : int | POST | RemoveFromCart | **25** |
| Update in the users the last Entry of the user to the system.  **returns true if it got updated or false if it was not.** | bool | userName: int | POST | LastEntry | **26** |

**ביצוע הAPI:**

לקוחה: דנה

מתכנתת: ירדן

**קשיים:**

הלקוחה התקשתה להגדיר את בקשותיה ולאפיין את המערכת באופן היקפי כך שאף דרישה לא תסתור דרישה אחרת.

המתכנתת חוותה קושי בקבלת החלטות לגבי שמירת המידע בצד הלקוח או בצד השרת ובהתאם לכך בניית הפונקציות בטבלה, לאפיין אילו מהשיטות יצטרכו גישה לשרת בצורה שתצמצם את מספר הפניות לשרת, ניסיון למידול השיטות בצורה גנרית.