**API – assignment 3.1  
Reut Chitayate id: 305573032, Maayan Hassidim id: 203869789**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Method Name** | **HTTP Method** | **Parameters** | **Returns** | **Explanation** | **הערות:** |
| **1** | exchangeRates | Get |  | Value of exchange rate - float | The function that called when the host page is required.  Will pass relevant exchange rates. |  |
| **2** | Login | Post | Username, password | User:  {  username:,  userPassword:,  userMail:,  categories[],  country:,  color:,  pet:  } | Login cannot pass in GET - we prefer pass private details in message's body. |  |
| **3** | Register | Post | Username, password, email, categories[], country, favorite Color, pet Name | True – if succeded  Else - Error message | Register cannot pass in GET - we prefer pass private details in message's body.  True= success  False=fail | Categories is string separated with ',' |
| **4** | listOfProducts | Get |  | Products array:  Products[] | 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.  Products interface:  We decided to write only this function which returns all products. The client will be responsible to filter the data by the different requests. |  |
| **5** | TopFiveProducts | Get |  | Products array:  Products[] | 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. |  |
| **6** | newProducts | Get |  | Products array:  Product[] | Products are related to the seller not to the client. In addition, only the client has the information about the publish date of every product.  Using GET because there is no information to deliver to the server or any wish to change any information. |  |
| **7** | Product | Get | productID | Product:  {  productID:,  productName:,  productCategory: ,  price:,  color:,  size:,  quantity:,  dateAdded:,  purchased:,  } | 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. | If failed – return error message  Last field counts how many times the product was purchased |
| **8** | RestorePassword | Get | Username, favoriteColor, petName | password | The password is store in the database so that the server is the only one who can get it.  Using GET because there is no information to deliver to the server or any wish to change any information. | If failed – return error message |
| **9** | UserByUserName | Get | userName | User:  {  username:,  userPassword:,  userMail:,  categories[],  country:,  color:,  pet:  } | Users are store in database so the server is the only one who can get it. The cookie store only the user id but in the process we require to use additional information.  Using GET because there is no information to deliver to the server or any wish to change any information. | If failed – return error message |
| **10** | RecommendedProducts | Get | userName | Products array:  Products[] | This function requires a calculation that can be done only in server (database needed).  Using GET because there is no information to deliver to the server or any wish to change any information. |  |
| **11** | Orders | Get | userName | Order:  {  orderID:,  orderDate:,  totalPrice:,  coin:,  delivaryDate:,  username:,  orderStatus:,  quantity:,  products[]  } | Orders are store in the database so that the server is the only one who can get it.  Using GET because there is no information to deliver to the server or any wish to change any information. | Array products contains only products ID's and all object |
| **12** | Buy | Post | products[]-a list of tuples (productID, quantity)  paymentDetails  deliveryDate  amount  coin  userName | OrderID if success. Else, false | Add new order to the orders table and remove all products from user's cart (in user table).  Using POST because there is an information to deliver to the server and a wish to change information.  Checking if the order is valid (i.e. product exist in stock) |  |
| **13** | AddProductByManager | Post | Product:  {  productID:,  productName:,  productCategory: ,  price:,  color:,  size:,  quantity:,  dateAdded:,  purchased:,  } | Boolean | Add new product to the products table.  Using POST because there is an information to deliver to the server and a wish to change information. | Bonus fanction |

**הנחות:**

* בטעינת דף האתר תקרא פונקציה שמחזירה את כל המוצרים והם יישמרו בצד הלקוח. כך סינונים וחיפושים יתאפשרו ללא קריאות API נוספות. זאת בהנחה שאין מספר רב של מוצרים במערכת.
* עגלת הקניות נשמרת בclient ובה נשמרים אובייקטים מסוג product. לכן, הצגה לפי מאפיינים, חישוב סה"כ המחיר ,הצגת הפרטים המלאים עבור מוצר בעגלה, מחיקה והוספה לעגלה יבוצעו בצד הלקוח.
* ביציאה מהמערכת, עגלת הקניות תשלח לשרת ותשמר בבסיס הנתונים.
* בכניסה למערכת באמצעות login נשלח האובייקט user ללקוח. האובייקט מכיל את עגלת הקניות של המשתמש שמיוצגת באמצעות מערך של product.
* בממשק רכישת מוצר, בצד הלקוח ייבדק אם תאריך המשלוח תואם לדרישות וישלח בקשת get לשרת רק אם תאריך המשלוח אכן תואם. בנוסף, הלקוח יחשב בעצמו את סכום ההזמה ע"פ שער המטבע המבוקש. סוג המטבע והסכום המחושב ישלחו בעת קריאת "Buy" לשרת ומשם ישמרו בDB.
* בטעינת דף האתר תקרא פונקציה שמחזירה מהשרת את שערי החליפין העדכניים של המטבעות הרלוונטיים.
* צביעת הרשומה באדום בהצגת המלאי עבור מוצר שקיימות פחות מ-3 יחידות ממנו תתבצע בעקבות בדיקה שיבצע צד הלקוח בהתאם לנתונים שנשלחו מהשרת.