

## סביבות פיתוח באינטרנט

### עבודה 4 – צד לקוח

בחרנו לבנות אתר עבור מוצרי קונדטוריה (עוגות)

להלן פירוט ה-API של השרת בעבודה :

ID	Method Name	HTTP Method	Parameters	Returns
1	/addOrder	POST	Order, cakes[]	Bool (Success or failure)
2	/users/addToCart	POST	Cart	Bool (Success or failure)
3	/users/register	POST	User, Questions and answers, categories ** The request structure described below	Bool (Success or failure)
4	/cakes/addCake	POST	Cake	Bool (Success or failure)
5	/users/login	POST	UserName , Password ** e.g.: { "UserName": "Avi", "Password": "123456" }	Login token (validation number) or NULL in case of failure
6	/users/restorePassword	PUT	userName, Answers[] ** e.g.: { "UserName": "igal", "Answer1": "vanila", "Answer2": "the beach" }	Password ** e.g.: { "Password": "123456" } If parameters are incorrect return "Bad request – 400"
7	/getDollarRate	GET	-	Double - dollar rate
8	/categories	GET	-	Categories[]
9	/supliers	GET	-	Supliers[]
10	/users/getAll	GET	-	Users[]
11	/users/recommandation /logged/{userName}	GET	-	Return recommended cakes : Cakes[]
12	users/orders/{userName}	GET	-	orders[]
13	/users/questions/{userName}	GET	-	questions ** e.g.: { "Question1" : "what is ..", "Question2" : "who...", }
14	/cakes/logged/getNew Cakes	GET	-	Cakes[]
15	/cakes	GET	-	Return all cakes : Cakes[]
16	/cakes/available/{Cake ID}/{Amount}	GET	-	Bool (if the quantity exist in the stock)
17	/cakes/top5	GET	-	Cakes[]
18	/cakesInOrder/{orderID}	GET	-	All the cakes and their quantity from this order

19	/users/deleteUser	DELETE	UserName	-
20	/cakes/deleteCake	DELETE	Cake	-
21	/users/deleteFromCart	DELETE	Cart	-
22	/cakes/byCategory/{categoryName}	GET	-	Returns all the cakes that are in the category
23	/cakes/byID/{cakeId}	GET	-	Returns the cake that has this cakeId
24	/cakes/byName/{cakeName}	GET	-	Returns the cakes that has this cakeName

### מבנה האובייקטים ב-JSON:

#### **Cake:**

```
{
  "CakeID": int
  "CakeName": varchar (20)
  "CakeCategory": varchar (200)
  "Description": varchar (200)
  "ImagePath": varchar (200)
  "StockAmount": int
  "SupplierID": int
  "price": float
}
```

#### **User:**

```
{
  "UserName": nvarchar (8) NOTNULL
  "Password": nvarchar (50) NOTNULL
  "FirstName": nvarchar (50) NOTNULL
  "LastName": nvarchar (50) NOTNULL
  "Adress": nvarchar (50)
  "City": nvarchar (50)
  "Country": nvarchar (50)
  "Phone": nvarchar (50)
  "Mail": nvarchar (50) NOTNULL
  "CreditCardNumber": nvarchar (50) NOTNULL
  "isAdmin": int
}
```

#### **Category:**

```
{
  "CategoryName": varchar (20)
}
```

#### **Orders:**

```
{
```

```
"UserName": nvarchar (8)
"OrderDate": datetime
"ShipmentDate": datetime
"Dollar": varchar
"TotalAmount": float
}
```

#### **Cart:**

```
{
  "UserName": nvarchar (8)
  "CakeID": int
  "Amount": int
}
```

#### **Supplier:**

```
{
  "SupplierID": int
  "SupplierName": varchar (100)
  "Description": varchar (255)
}
```

### **- מבנה בקשת Register:**

**/users/register/**

```
[{
  "UseName": "oren",
  "Password": "123456",
  "FirstName": "moshe",
  "LastName": "bond",
  "Adress": "rager",
  "City": "beer sheva",
  "Country": "egypt",
  "Phone": "0528341791",
  "Mail": "bond@gmail.com",
  "CreditCardNumber": "123456789",
  "isAdmin": "0"
  "Question1" : "what is you favorite flavore?", // varchar (200)
  "Question2" : "what is your favorite place?", // varchar (200)
  "Answer1" : "vanila", // varchar (200)
  "Answer2" : "the beach" // varchar (200)
},
{
  "category1" : "Moose",
  "category2" : "Fruit",
  "category3" : "Vanilla"
}]
```

### **- מבנה בקשת addOrder:**

**/addOrder**

```
[{
```

```
"UserName": "alon",
"ShipmentDate": "2017-06-04 09: 57: 51",
"Dollar": "d",
"TotalPayment": "100"
},
{
"CakeID": "10",
"Amount": "1"
},
{
"CakeID": "7",
"Amount": "11"
},
{
"CakeID": "6",
"Amount": "30"
}
}]
```

### הנחיות שימוש בשרת ובמערכת:

- פורט 3000, על מנת לראות את האתר, לאחר הרצת הקוד יש להיכנס אל localhost: 3000
- ניתן להירשם לאתר בדף ההרשמה, ולאחר מכן להתחבר בהתאם.
- ברישום משתמש, עליו לבחור 3 קטגוריות בלבד שהוא מעדיף.
- כל משתמש צריך להגדיר 2 שאלות ותשובות בלבד על מנת לשחזר סיסמא.
- לא קיימת אפשרות רכישה של עגלה.
- במידה והיה כיבוי והדלקה מחדש של השרת, יש לבצע התנתקות והתחברות של המשתמש מהאתר, על מנת לוודא שהשרת ישמור את ההתחברות של המשתמש מחדש.

### **שינויים שביצענו ב-API בעבודה זו:**

- הוספנו לכל הפונקציות בשרת שדורשות התחברות בURL את routen "logged" – תוספת זו אפשרה לנו להשתמש בmiddleware בשרת, וכך לבדוק בעזרת טוקן הנמצא אצל הלקוח, האם הבקשה נעשתה באמת ע"י לקוח מחובר.
- את טבלת קטגוריות שינינו כך שהמפתח בטבלה זו הוא שם הקטגוריה ולא ID, על מנת לפשט את מס' השליפות שעלינו לבצע, וכן על מנת להכריח כי לא קיימת אותה קטגוריה פעמיים.