Faculty of Engineering & Technology

Computer Science Department

Web Services Technology COMP4382

Assignment #1

Prepared by:

Yousef Injass   1200643

Instructor: Dr. Mohammad Kharmeh

Section: 1

Date: 5/6/2023

Customer :

HTTP request (method URI)	Operation	Description	HTTP status codes	Request sample	Respose sample

/getCustomers	GET	Returns all customers.	200 OK: Request, data returned.
			
			204 No Content: Request successful, no data returned.
			
			400 Bad Request: Request syntax or parameters are invalid.
			
			401 Unauthorized: Request lacks valid authentication.
			
			403 Forbidden: Request refused due to lack of authorization.
			
			404 Not Found: Requested resource not found.
					`[`
					  `{`
					   `"id": 1,`
					   `"firstName": "Yousef",`
					   `"lastName": "Injass",`
					   `"email": "yousef@gmail.com",`
					   `"address": "Ramallah",`
					   `"phone": "0595773402"`
					  `},`
					  `{`
					   `"id": 2,`
					   `"firstName": "Mohammad",`
					   `"lastName": "Hamed",`
					   `"email": "mohammad@gmail.com",`
					   `"address": "Selfit",`
					   `"phone": "0567823252"`
					  `}`
					`]`

/getCustomer/ {id}	GET	return a customer by ID	500 Internal Server Error: Unexpected server-side error occurred.		`{`
					  `"id": 1,`
					  `"firstName": "Yousef",`
					  `"lastName": "Injass",`
					  `"email": "yousef@gmail.com",`
					  `"address": "Ramallah",`
					  `"phone": "0595773402"`

					}

/addCustomer	POST	Creates a new customer		{
				
				    "firstName": "Ali",
				
				    "lastName": "Injass",
				
				    "email": "ali@gmail.com",
				
				    "address": "Kharbath",
				
				    "phone": "0598947775"
				
				}
					{
					 "id": 3,
					
					  "firstName": "Ali",
					
					  "lastName": "Injass",
					
					 "email": "ali@gmail.com",
					
					 "address": " Kharbath ",
					
					 "phone": "0598947775"
					
					}

/updateCustomer/{id}	PUT	Updates an existing customer by ID		{
				
				   "firstName": "Yousef",
				
				   "lastName": "Injass",
				
				  "email": "injasyousef@gmail.com",
				
				   "address": "Ramallah",
				
				   "phone": "0595773402"
				
				}
					{
					 "id":1
					
					   "firstName": "Yousef",
					
					   "lastName": "Injass",
					
					  "email": "injasyousef@gmail.com",
					
					  "address": "Ramallah",
					
					  "phone": "0595773402"
					
					}

customers/deleteCustomer/{id}	DELETE	Deletes a specific customer by ID			

item :

HTTP request (method URI)	Operation	Description	HTTP status codes	Request sample	Respose sample

items/getItems	GET	return all items	200 OK: Request, data returned.
			
			204 No Content: Request successful, no data returned.
			
			400 Bad Request: Request syntax or parameters are invalid.
			
			401 Unauthorized: Request lacks valid authentication.
			
			403 Forbidden: Request refused due to lack of authorization.
			
			404 Not Found: Requested resource not found.
					[
					  {
					   "id": 1,
					   "name": "book",
					   "description": "Programming book",
					   "price": 40.00,
					   "quantity": 22
					  },
					  {
					   "id": 2,
					   "name": "pencil",
					   "description": "pen",
					   "price": 2.50,
					   "quantity": 100
					  }
					]

items/getItem/{id}	GET	return a specific item by ID	500 Internal Server Error: Unexpected server-side error occurred.		{
					  "id": 1,
					  "name": "book",
					  "description": "Programming book",
					  "price": 40.00,
					  "quantity": 22
					}

items/createItem	POST	Creates a new item		{
				  "name": "pencil",
				  "description": "Red
					{
					  "id" :3

				pen",
				
				  "price": 3.50,
				
				  "quantity": 150
				
				}
					"name": "pencil",
					
					  "description": "Red pen",
					
					  "price": 3.50,
					
					  "quantity": 150
					
					  }

items/updateIte m/{id}	PUT	Update an existing item by ID.		{
				
				  "name": "pencil",
				
				  "description": "yallow pen",
				
				  "price": 4.50,
				
				  "quantity": 150
				
				}
					{
					
					  "id":3
					
					  "name": "pencil",
					
					  "description": "yallow pen",
					
					  "price": 4.50,
					
					  "quantity": 150
					
					}

/items/item/{ite mId}/customer	GET	Retrieves all customers associated with a specific item.			[
					
					  {
					
					  "id": 1,
					
					  "name": "Yousef",
					
					  "email": "yousef@gmail.com",
					
					  "address": "Ramallah",
					
					  "phone": "0595773402"
					
					  }
					
					  ]

inventory :

HTTP request (method URI)	Operation	Description	HTTP status codes	Request sample	Respose sample

/inventories/getInve ntories	GET	return all inventories.	200 OK: Request, data returned.
			
			204 No Content: Request successful, no data returned.
			
			400 Bad Request: Request syntax or parameters are invalid.
			
			401 Unauthorized: Request lacks valid authentication.
			
			403 Forbidden: Request refused due to lack of authorization.
			
			404 Not Found: Requested resource not found.
			
			500 Internal Server Error: Unexpected server-side error occurred.
					[
					
					  {
					
					   "id": 1,
					
					   "name": "Inventory1",
					
					   "location": "Ramallah-Kharbatha-Bani-Hareth"
					
					  },
					
					  {
					
					   "id": 2,
					
					   "name": "Inventory2",
					
					   "location": "Ramallah-Der-Qadis"
					
					  }
					
					 {
					
					   "id": 3,
					
					   "name": "Inventory3",
					
					   "location": "Ramallah-Belin"
					
					  }
					
					]

/inventories/GetInve ntory/{id}	GET	Return a specific inventory by ID			{
					
					  "id": 1,
					
					  "name": "Inventory1",
					
					  "location": "Ramallah- Kharbatha-Bani-Hareth "
					
					 }

/inventories/Create NewInventory	POST	Creates a new inventory		{
				
				  "name": "Inventory4",
				
				  "location": "
					{
					
					 "id":

				Ramallah-Nelin " }	"name": "Inventory4", "location": " Ramallah-Nelin " }

/inventories/updateInventory{id}	PUT	Updates an existing inventory by ID		{
				
				 "name": "Inventory3",
				
				 "location": "Nablus"
				
				}
					{
					
					 "id":3
					
					 "name": "Inventory3",
					
					 "location": "Nablus"
					
					}

/inventories/deleteInventory/{id}	DELETE	Deletes a specific inventory by ID			

Order:

HTTP request (method URI)	Operation	Description	HTTP status codes	Request sample	Respose sample

/orders/getOrders	GET	Return all orders	200 OK: Request, data returned.
			
			204 No Content: Request successful, no data returned.
			
			400 Bad Request: Request syntax or parameters are invalid.
			
			401 Unauthorized: Request lacks valid authentication.
			
			403 Forbidden: Request refused due to lack of authorization.
					[
					  {
					   "id": 1,
					   "CustomerID": 1,
					   "ItemID": 1
					   "date": "3-6-2024"
					  },
					  {
					   "id": 2,
					   "CustomerID": 2,
					   "ItemID": 2,
					   "date": "1-6-2024"
					  }
					]

/orders/getOrder/{id}	GET	Retrieves a specific admin by ID	404 Not Found: Requested resource not found.
			
			500 Internal Server Error: Unexpected server-side error occurred.
					{
					  "id": 1,
					  "CustomerID": "1",
					  "ItemID": "1"
					  "date": "3-6-2024"
					}

/orders/addNewOrder	POST	Creates a new Order		{
				  "CustomerID": "1",
				  "ItemID": "2"
				  "date": "5-6-2024"
				}
					{
					  "id": 3,
					  "CustomerID": 1,
					  "ItemID": 2,
					  "date": "5-6-2024"

					}

orders/updateOrder/ {id}	PUT	Updates an existing order by ID		{
				
				  "CustomerID": "1",
				
				  "ItemID": "2"
				
				  "date": "6-6-2024"
				
				}
					{
					
					   "id": 3,
					
					  "CustomerID": 1,
					
					  "ItemID": 2,
					
					  "date": "6-6-2024"
					
					}

orders/DeleteOrder/ {id}	DELETE	Deletes a specific order by ID			