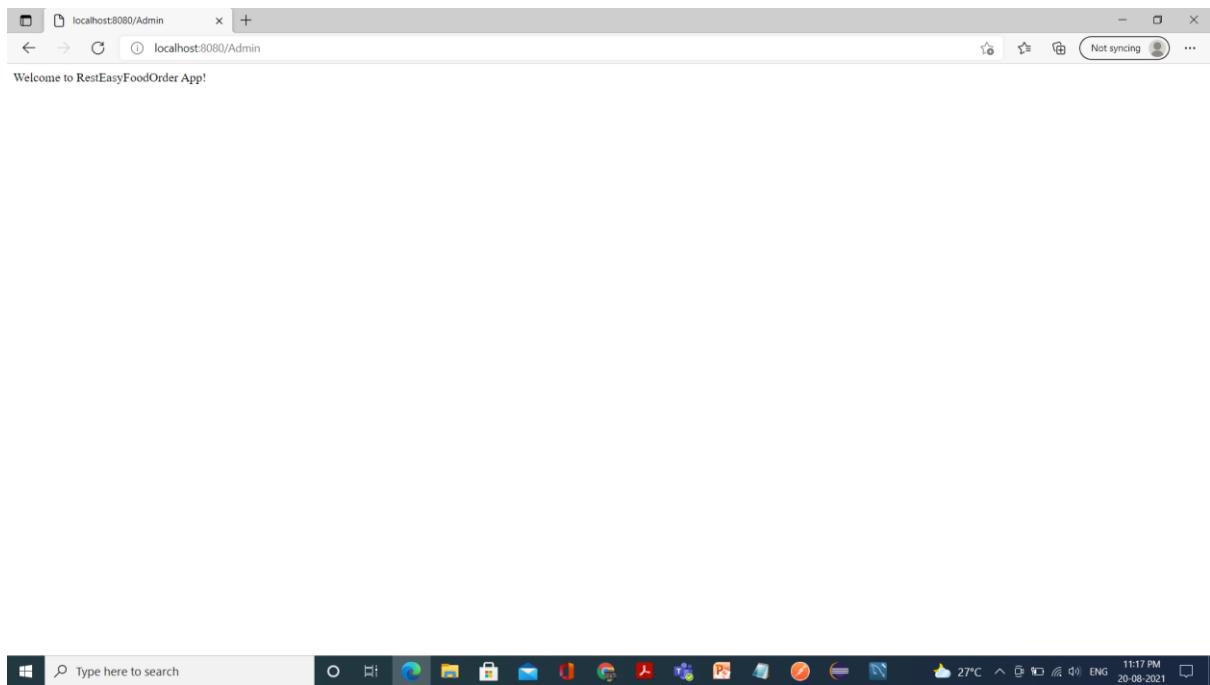


Welcome Screen



Admin user registered using Postman (POST method)

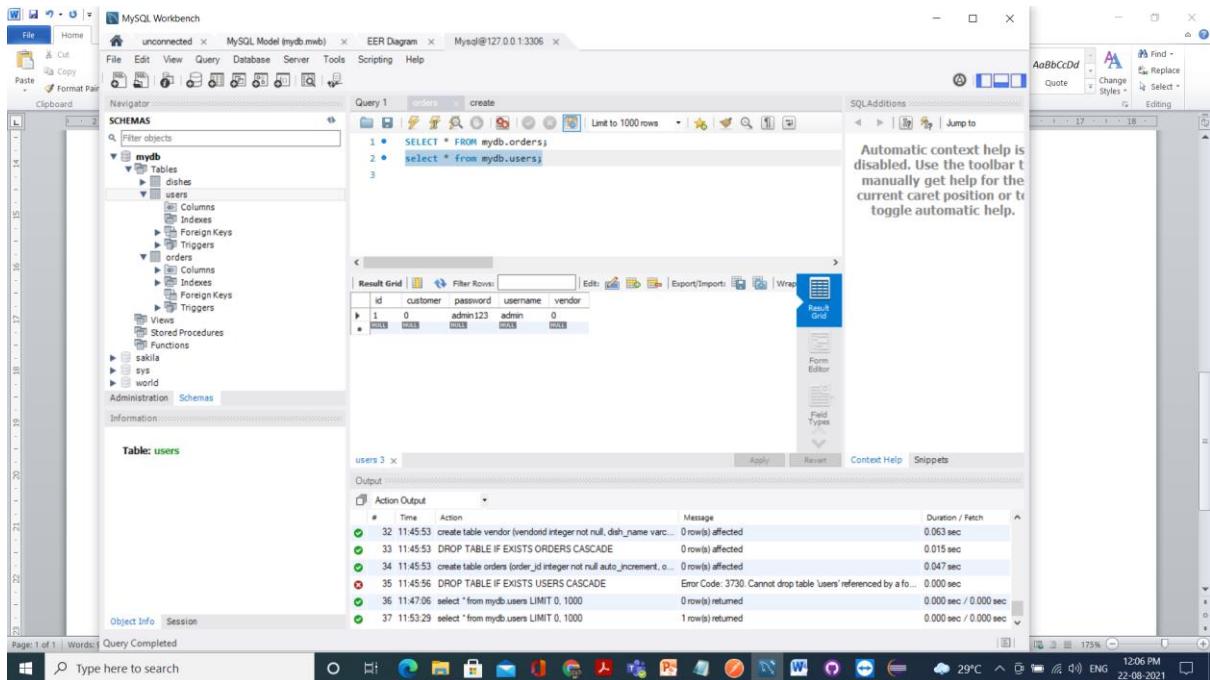
A screenshot of the Postman application. The left sidebar shows a collection named 'restEasyFoodOrder' with a single item 'Today' containing a POST request to 'http://localhost:8080/Admin/Register'. The request body is set to 'JSON' and contains the following data:

```
1  {
2   "vendor": false,
3   "customer": false,
4   "userName": "admin",
5   "password": "admin123"
6 }
```

The response tab shows a successful '200 OK' status with a response body identical to the request body.

Postman
File Edit View Help
Home Workspaces Reports Explore
restEasyFoodOrder New Import POST http://localhost:8080/Admin/Register
Params Authorization Headers (8) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON
1 {
2 "vendor": false,
3 "customer": false,
4 "userName": "admin",
5 "password": "admin123"
6 }
Body Cookies Headers (5) Test Results
Pretty Raw Preview Visualize JSON
1 {
2 "id": 1,
3 "userName": "admin",
4 "password": "admin123",
5 "vendor": false,
6 "customer": false
7 }

Value added reflected in database



Invoked vendor signin, without creating...

eclipse-workspace - restEasyFoodOrder/src/main/java/com/myProject/restEasyFoodOrder/Controller/VendorController.java - Eclipse IDE

```

File Edit Source Refactor Navigate Search Project Run Window Help
AdminControl... AdminJava application_... VendorJava AdminService... QuestionCon... SignUpUser... SignInRespo... CommonBusine... VendorContro... VendorServic... AdminDao.java
26
27     private VendorService vendorService;
28
29     @RequestMapping(value = "/Vendor/SignIn", produces = "application/json", method = RequestMethod.POST)
30     public Vendor add(@RequestBody Vendor vendorDetail) {
31         try {
32             Admin existUser = adminService.getUser(adminDao.findByName(vendorDetail.getVendorName()));
33             if (existUser != null) {
34                 Vendor vendor = new Vendor();
35                 vendor.setVendorID(existUser.getId());
36                 vendor.setVendorName(existUser.getUserName());
37                 vendor.setDishName(vendorDetail.getDishName());
38                 vendor.setVendorDishPrice(vendorDetail.getVendorDishPrice());
39
40                 return vendorService.save(vendor);
41             }
42         } catch (Exception nre) {
43             System.out.println(nre);
44         }
45     }
46     return null;

```

Console

```

RestEasyFoodOrderApplication [Java Application]
2021-08-22 13:42:21.041 INFO 25720 --- [           main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.52]
2021-08-22 13:42:21.291 INFO 25720 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[] : Initializing Spring embedded WebApplicationContext
2021-08-22 13:42:21.301 INFO 25720 --- [           main] w.s.web.context.ContextLoader : Root WebApplicationContext: initialization completed in 2229 ms
2021-08-22 13:42:21.428 INFO 25720 --- [           main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2021-08-22 13:42:21.966 INFO 25720 --- [           main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2021-08-22 13:42:22.056 INFO 25720 --- [           main] o.h.jpa.internal.util.LogHelper : HHH000204: Processing PersistenceUnitInfo [name: default]
2021-08-22 13:42:22.169 INFO 25720 --- [           main] o.h.hibernate.Version : HHH000412: Hibernate ORM core version 5.5.6.Final
2021-08-22 13:42:22.522 INFO 25720 --- [           main] o.h.hibernate.annotations.common.Version : HCAIM000001: Hibernate Commons Annotations (5.1.2.Final)
2021-08-22 13:42:22.663 INFO 25720 --- [           main] org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.hibernate.dialect.MySQL5Dialect
2021-08-22 13:42:23.429 INFO 25720 --- [           main] o.h.e.t.j.p.l.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
2021-08-22 13:42:23.554 INFO 25720 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-08-22 13:42:24.088 INFO 25720 --- [           main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed during view rendering. If this causes issues in your application, please set it to false.
2021-08-22 13:42:24.598 INFO 25720 --- [           main] c.a.r.RestEasyFoodOrderApplication : Started RestEasyFoodOrderApplication in 6.237 seconds (JVM running for 6.85)
2021-08-22 13:47:44.621 INFO 25720 --- [nio-8080-exec-2] o.a.c.c.C.[Tomcat].[localhost].[] : Initializing Spring DispatcherServlet 'dispatcherServlet'
2021-08-22 13:47:44.621 INFO 25720 --- [nio-8080-exec-2] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'
2021-08-22 13:47:44.623 INFO 25720 --- [nio-8080-exec-2] o.s.web.servlet.DispatcherServlet : Completed initialization in 1 ms
java.lang.NullPointerException: Cannot invoke "javax.persistence.EntityManager.createNamedQuery(String, java.lang.Class)" because "this.entityManager" is null

```

In postman, query is successful, but data not created

Postman

File Edit View Help

Home Workspaces Reports Explore

restEasyFoodOrder New Import

POST http://localhost:8080/Vendor/SignIn

POST http://localhost:8080/Admin/Register

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1
2   "vendorName": "udupi",
3   "dishName": "Dosa",
4   "vendorDishPrice":100
```

Body Cookies Headers (4) Test Results

Pretty Raw Preview Visualize Text

Status: 200 OK Time: 259 ms Size: 123 B Save Response

Find and Replace Console

Type here to search

Vendor registered

Postman

File Edit View Help

Home Workspaces Reports Explore

restEasyFoodOrder New Import

POST http://localhost:8080/Admin/Register

POST http://localhost:8080/Vendor/SignIn

POST http://localhost:8080/Admin/Register

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1
2   "vendor": true,
3   "customer": false,
4   "userName": "udupi",
5   "password": "vendordupi"
```

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

Status: 200 OK Time: 150 ms Size: 247 B Save Response

Find and Replace Console

Type here to search

Vendor registered

The screenshot shows the MySQL Workbench interface. On the left, the Navigator pane displays the database schema with a tree view of tables, views, and stored procedures under the 'mydb' schema. The central area contains a 'Query 1' window with the following SQL code:

```

1 • SELECT * FROM mydb.orders;
2 • select * from mydb.users;
3 • select * from mydb.vendor;
4
5

```

The results grid shows the following data for the 'users' table:

	id	customer	password	username	vendor
1	0	admin123	admin	0	
2	0	vendordupi	udipi	1	MALE

The status bar at the bottom right indicates the time as 02:28 PM and the date as 22-08-2021.

Customer registration

The screenshot shows the Postman application interface. On the left, the sidebar includes sections for Collections, APIs, Environments, Mock Servers, and Monitors. The main area shows a collection named 'restEasyFoodOrder' with a POST request to 'http://localhost:8080/Admin/Register'. The 'Body' tab is selected, showing the following JSON payload:

```

1 ...
2 ...
3 ...
4 ...
5 ...
6 ...
7 ...

```

The payload content is:

```

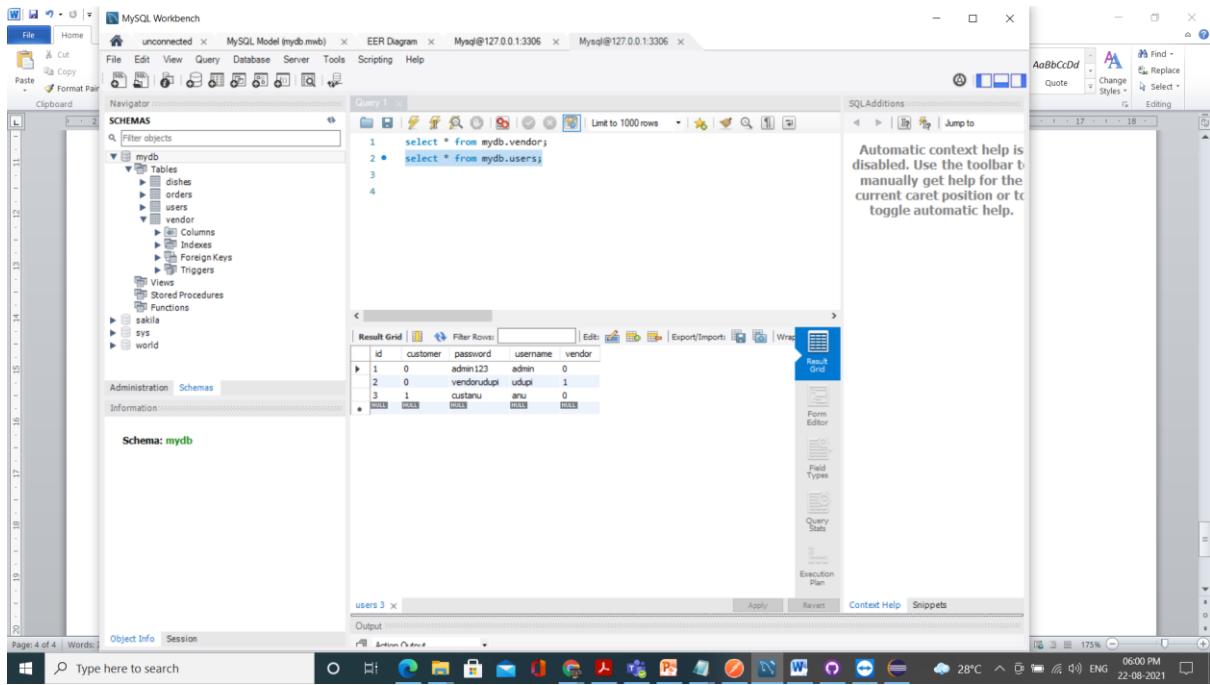
1 ...
2 ...
3 ...
4 ...
5 ...
6 ...
7 ...

```

The response panel shows the following details:

- Status: 200 OK
- Time: 150 ms
- Size: 241 B
- Save Response

The status bar at the bottom right indicates the time as 05:58 PM and the date as 22-08-2021.



Registering another customer

Postman Screenshot:

POST http://localhost:8080/Admin/Register

```

1 ...
2 ...
3 ...
4 ...
5 ...
6 ...
7 ...

```

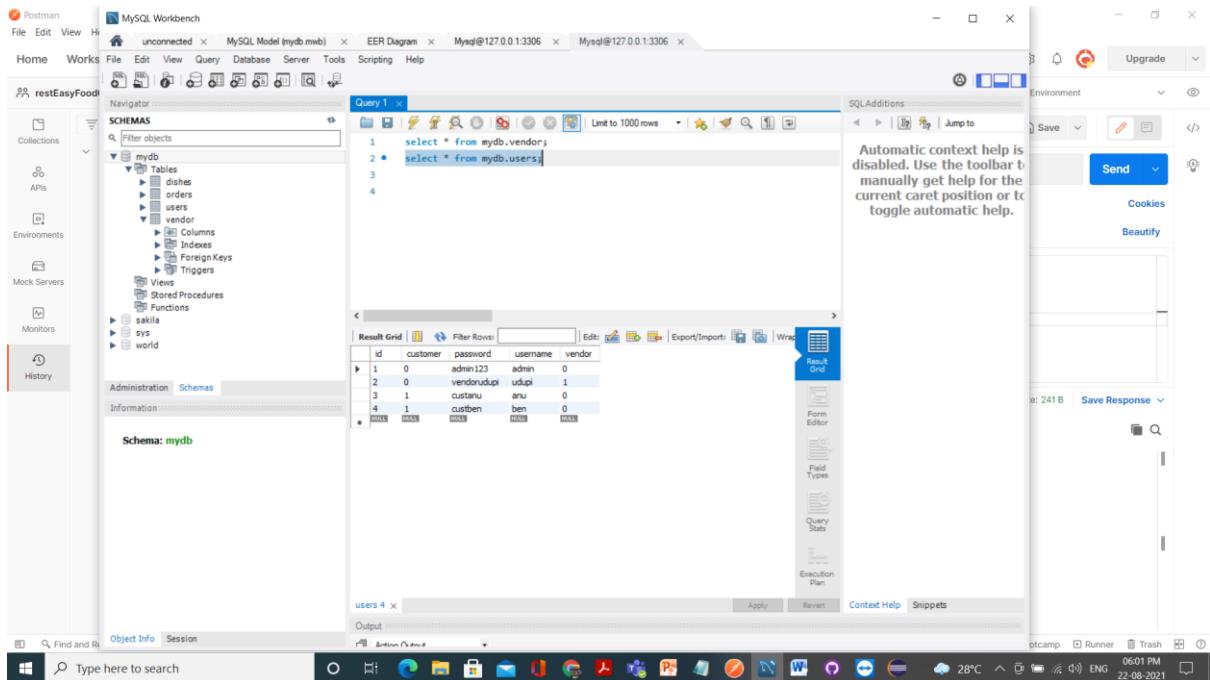
Body

```

1 {
2   ...
3   "vendor": false,
4   ...
5   "customer": true,
6   ...
7   "userName": "ben",
8   ...
9   "password": "custben"
10 }

```

Status: 200 OK Time: 23 ms Size: 241 B Save Response



Vendor signIn and updating vendor table

eclipse-workspace - restEasyFoodOrder/src/main/java/com/myProject/restEasyFoodOrder/Controller/VendorController.java - Eclipse IDE

```

19
20  @Autowired
21  private AdminService adminService;
22
23  //@Autowired
24  //private AdminDao adminDao;
25
26@  @Autowired
27  private VendorService vendorService;
28
29  @RequestMapping(value = "/Vendor/SignIn", produces = "application/json", method = RequestMethod.POST)
30  public Vendor add(@RequestBody Vendor vendorDetail) {
31      try {
32          Admin existUser = adminService.getUser(adminDao.findByName(vendorDetail.getVendorName()));
33          Admin existUser = adminService.getUser(vendorDetail.getVendorName());
34          if (existUser != null) {
35              Vendor vendor = new Vendor();
36              vendor.setVendorID(existUser.getId());
37              vendor.setVendorName(existUser.getUsername());
38              vendor.setDishName(vendorDetail.getDishName());
39              vendor.setDishCal(vendorDetail.getDishCal());
40              vendor.setDishPrice(vendorDetail.getVendorDishPrice());
41
42              Dishes dishes = new Dishes();
43              dishes.setDishName(vendorDetail.getDishName());
44          }
45      } catch (Exception e) {
46          e.printStackTrace();
47      }
48  }

```

Console

```

RestEasyFoodOrderApplication [Java Application] C:\Users\Lenovo.p2\pool\plugins\org.eclipse.jst\openjdkhotspot\jre\full\win32\x86_64_15.0.2.v20210201-0955\jre\bin\javaw.exe (22-Aug-2021, 9:12:38 pm)
2021-08-22 21:12:43.826 INFO 21576 --- [main] o.a.c.c.C.[Tomcat].[localhost].[] : Initializing Spring embedded WebApplicationContext
2021-08-22 21:12:43.826 INFO 21576 --- [main] w.s.c.WebServerApplicationContext : Root WebApplicationContext: initialization completed in 2290 ms
2021-08-22 21:12:44.073 INFO 21576 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2021-08-22 21:12:44.115 INFO 21576 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2021-08-22 21:12:44.115 INFO 21576 --- [main] org.hibernate.annotations.common.util.helpers.LogHelper : 
2021-08-22 21:12:44.924 INFO 21576 --- [main] org.hibernate.Version : HHH000412: Hibernate ORM core version 5.5.6.Final
2021-08-22 21:12:45.164 INFO 21576 --- [main] o.h.HibernateAnnotationsCommonVersion : HCAW000001: Hibernate Commons Annotations (5.1.2.Final)
2021-08-22 21:12:45.312 INFO 21576 --- [main] org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.hibernate.dialect.MySQL5Dialect
2021-08-22 21:12:46.103 INFO 21576 --- [main] o.h.e.t.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.N
2021-08-22 21:12:46.116 INFO 21576 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2021-08-22 21:12:46.873 WARN 21576 --- [main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database queries may be performed during view rendering.
2021-08-22 21:12:47.371 INFO 21576 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2021-08-22 21:12:47.388 INFO 21576 --- [main] c.m.r.RestEasyFoodOrderApplication : Started RestEasyFoodOrderApplication in 6.568 seconds (JVM running for 7.299)

```

Postman

File Edit View Help

Home Workspaces Reports Explore

restEasyFoodOrder

POST http://localhost:8080/Vendor/SignIn

Params Authorization Headers (8) Body Pre-request Script Tests Settings

Body (JSON)

```
1 "vendorName": "udupi",
2   "dishName": "Dosa",
3   "dishCal": 150.0,
4   "vendorDishPrice":100
5
6
7
8
```

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

Status: 200 OK Time: 623 ms Size: 264 B Save Response

MySQL Workbench

unconnected MySQL Model (mydb.mwb) EER Diagram Mysql@127.0.0.1:3306 Mysql@127.0.0.1:3306

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS mydb

Tables dishes orders users vendor

Columns vendorid vendor_name dish_name dish_cal vendor_dish_price

Indexs Foreign Keys Triggers

Views Stored Procedures Functions

Administration Schemas Information

Column: dish_cal

Definition: dish_cal float

Query 1

```
1 select * from mydb.vendor;
2 select * from mydb.users;
```

Result Grid

vendorid	vendor_name	dish_name	dish_cal	vendor_dish_price
2	udupi	Dosa	150	100

Object Info Session

Registering another vendor

The screenshot shows the Postman application interface. On the left, the sidebar includes sections for Collections, APIs, Environments, Mock Servers, and Monitors. The History section is currently selected. A search bar at the top right says "Search Postman". The main workspace displays a collection named "restEasyFoodOrder". A recent history of requests is listed on the left, including:

- POST http://localhost:8080/Admin/Register
- POST http://localhost:8080/Vendor/SignIn
- POST http://localhost:8080/Admin/Register
- POST http://localhost:8080/Admin/Register
- POST http://localhost:8080/Vendor/SignIn
- POST http://localhost:8080/Admin/Register
- POST http://localhost:8080/Admin/Register

The current request details are as follows:

- Method:** POST
- URL:** http://localhost:8080/Admin/Register
- Headers:** (8)
- Body:** (Green dot)
- Params:** (Grey dot)
- Authorization:** (Grey dot)
- Pre-request Script:** (Grey dot)
- Tests:** (Grey dot)
- Settings:** (Grey dot)

The body is set to "form-data" and contains the following JSON payload:

```
1
2   "vendor": true,
3   "customer": false,
4   "userName": "ananda",
5   "password": "vendoranantha"
6
```

The response panel shows the following details:

- Body:** Cookies Headers (5) Test Results
- Status:** 200 OK
- Time:** 50 ms
- Size:** 250 B
- Save Response:** (Green checkmark)

The response body is displayed in "Pretty" format:

```
1
2   {
3     "id": 5,
4     "userName": "ananda",
5     "password": "vendoranantha",
6     "vendor": true,
7     "customer": false
8 }
```

At the bottom, there are various system status icons and a search bar.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the AdminControl schema with tables like dishes, orders, users, and vendor. The central area has a query editor titled 'Query 1' containing the following SQL:

```
1 select * from mydb.vendor;
2 • select * from mydb.users;
```

A tooltip on the right states: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help." Below the query editor is a result grid showing data from the 'users' table:

	id	customer	password	username	vendor
1	0	admin123	admin	0	
2	0	vendordupi	udipi	1	
3	1	custanu	anu	0	
4	1	custben	ben	0	
5	0	vendaranatha	ananda	0	

The bottom status bar shows the date and time as 09:19 PM 22-08-2021.

Vendor signin

The screenshot shows the Postman interface. On the left, there's a sidebar with collections, APIs, environments, mock servers, and monitors. The main area shows a collection named "restEasyFoodOrder". A specific POST request to "http://localhost:8080/Vendor/SignIn" is selected. The "Body" tab is active, showing the following JSON payload:

```

1  "vendorName": "ananda",
2   "dishName": "Idli",
3   "dishCal": 50.0,
4   "vendorDishPrice":50
5
6
7
8

```

The response status is 200 OK, with a time of 39 ms and a size of 263 B. The response body is also displayed in JSON format:

```

1  {
2   "vendorID": 5,
3   "vendorName": "ananda",
4   "dishName": "Idli",
5   "dishCal": 50.0,
6   "vendorDishPrice": 50.0,
7   "id": 5
8

```

Now, the vendor table:

The screenshot shows the MySQL Workbench interface. The left sidebar displays the schema structure of the "mydb" database, including tables like "dishes", "orders", "users", and "vendor". The "vendor" table is expanded, showing columns: vendorid, vendor_name, dish_name, dish_cal, and vendor_dish_price. The "Query 1" tab contains the following SQL query:

```

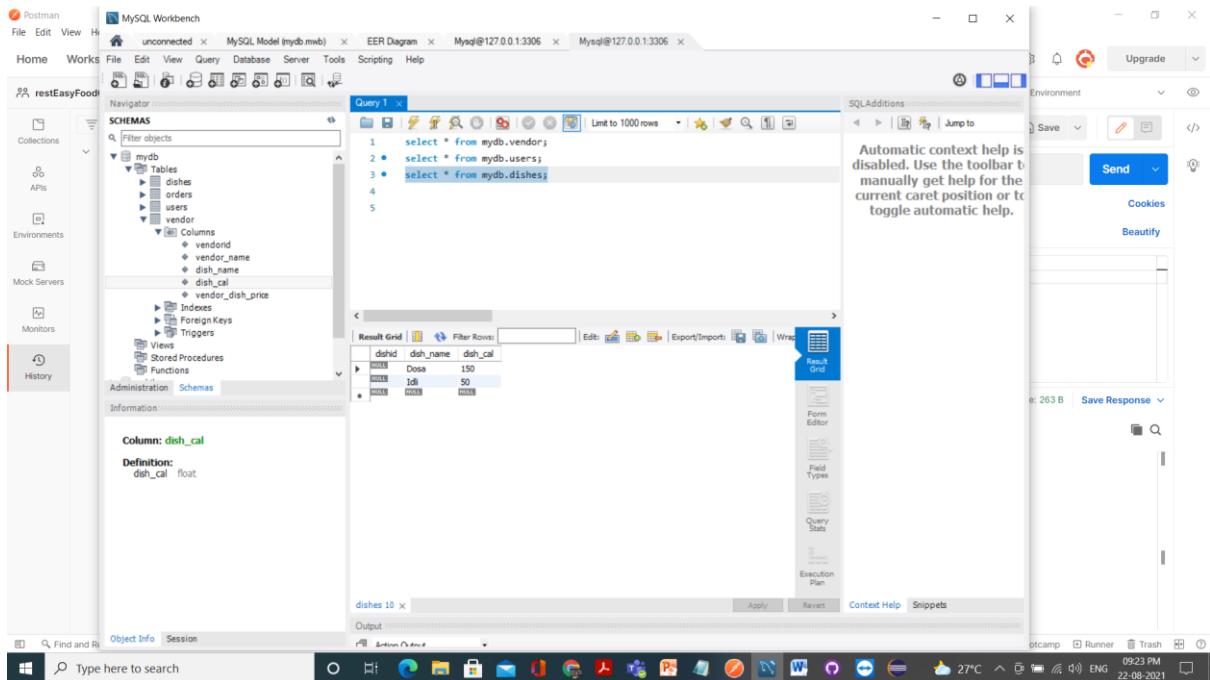
1 select * from mydb.vendor;
2 select * from mydb.users;
3
4

```

The results grid shows the data for the "vendor" table:

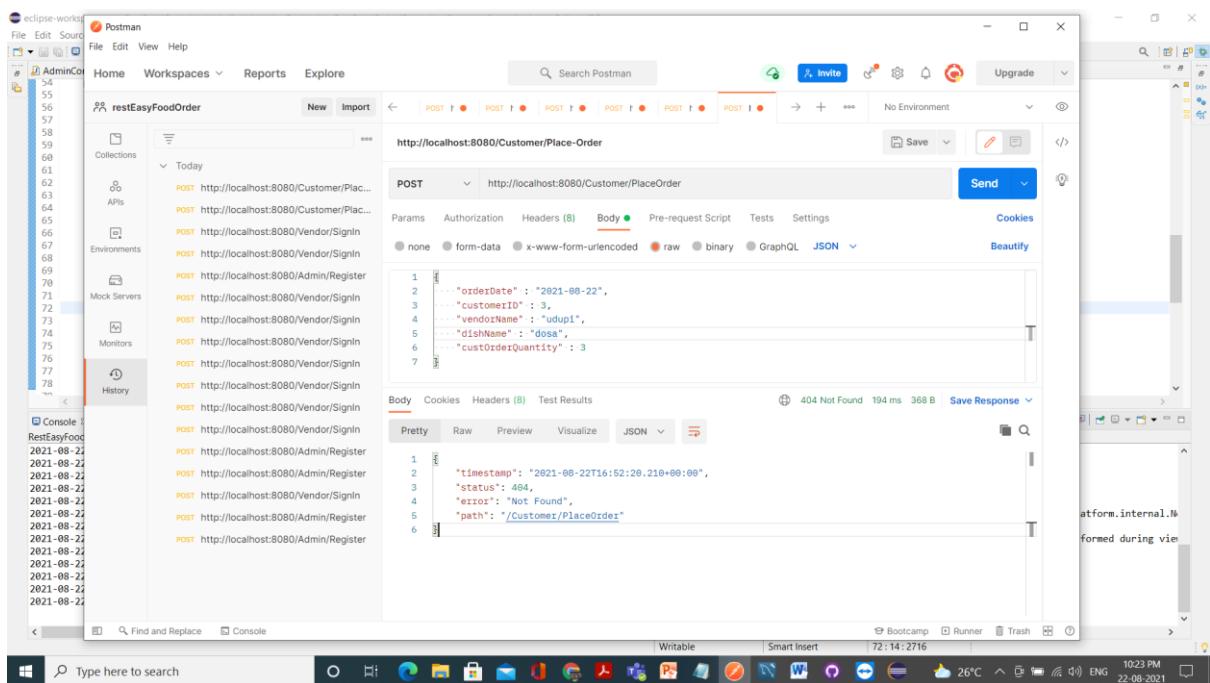
vendorid	vendor_name	dish_name	dish_cal	vendor_dish_price
2	udupi	Dosa	150	100
5	ananda	Idli	50	50
NULL	NULL	NULL	NULL	NULL

Listing all dishes:

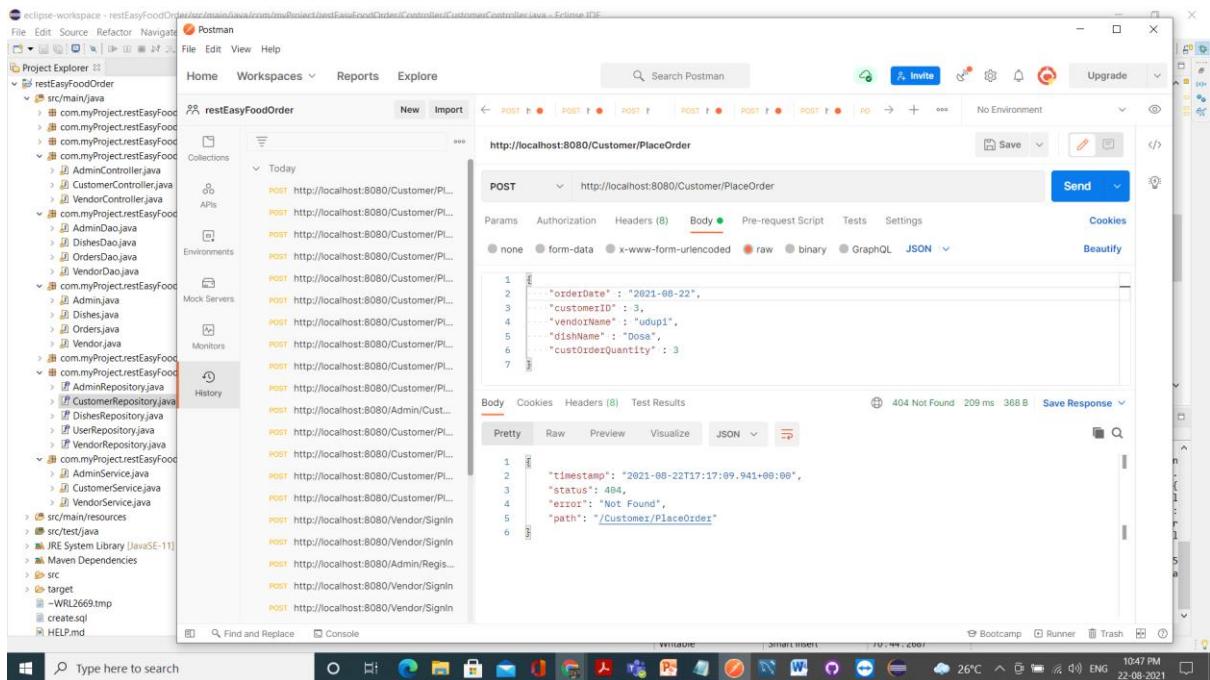


Placing Order

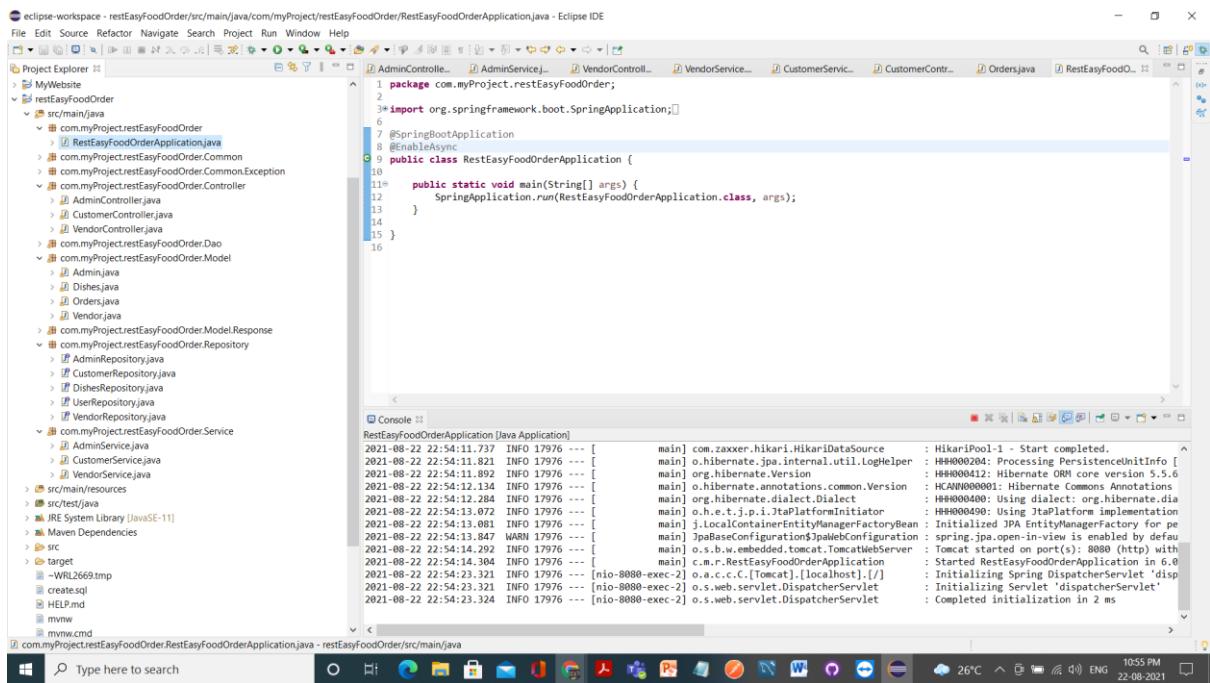
Here, order not successful as the dish name isn't a match



The problem still persists, and working on to rectify it...



Overall structure of project



All the required functionalities work, other than the Orders, and it is being still updated.

This is all updates till 11.15 pm, 22-08-2021

Anasuya.