




OlivApp - Olive Oil Mills Management App



CMPS356 Project Phase 1 – Web UI Design and Web API Implementation (15% of the course grade).

Group Id:	G1
Group Members:	Muhamad Irfan Fauzan(201609690) Thani Al-Thani(201517240) Ali El-Jaam Emails: mf1609690@student.qu.edu.qa; ta1517240@student.qu.edu.qa; student3@student.qu.edu.qa;

Grading Rubric - In the *Functionality* column please specify either: *Working (completed x%)*, *Not Working (completed x%)* or *Not done*.

Criteria	%	Functionality*	Quality of the implementation	Score
1) Application Design: <ul style="list-style-type: none"> 3 Class Diagrams for Entities, Repositories and Services [5 pts each]. Discussion of design rationale of 5 key design decisions [5 pts] 	20		- Good quality class diagrams with few issues highlighted in the comments.	19
2) Create test data JSON file for each of the entities.	5		Excellent work	5
3) [30%] Design and implement the Web UI using HTML and CSS for:				28
Main menu design	4		Excellent	4
 Login	4		Excellent	4
Register Customer	5		Good, no need for cancel button	5
Get Jobs	20		Good, but job details is not attribute. Also, the available buttons should be based on the job status.	19
Add Job	10		Good work	10
Update Job				
Cancel Job	1		Good	1
Add Invoice	8		Ok but could be better. See model solution.	7

Update Invoice				
View Invoice	6		Poorly done. This is not really an invoice! See the provided model solution.	4
Pay Invoice	5		It is better to display bit of details about the job and its invoice. See model solution.	4
Mill Calendar	4		Good	4
Add Job Note	5		Poor design	4
Get Job Notes	5		Poor design	4
Get Products (i.e., Products Catalog use case)	8		Excellent	8
Add / Update Product	7		Excellent	7
Jobs Summary Report	8		Excellent	8
[40%] Implement the repositories and the services (i.e., Web API) using JavaScript for:				38
Login	5		Using post for login is better to avoid revealing the password in the Url. Also, the returned user object should NOT have a password.	4
Add Customer	6		Excellent	6
Get Jobs	10		Good work	10
Add Job	6		Excellent	6
Update Job	5		Excellent	5
Cancel Job	5		Better to use delete method with url /api/jobs/:id	4
Add Invoice	6		Excellent	6
Update Invoice	5		Excellent	5
Get Invoice	10		Excellent	10
Pay Invoice	5		Better to use put whenever you update resource.	4

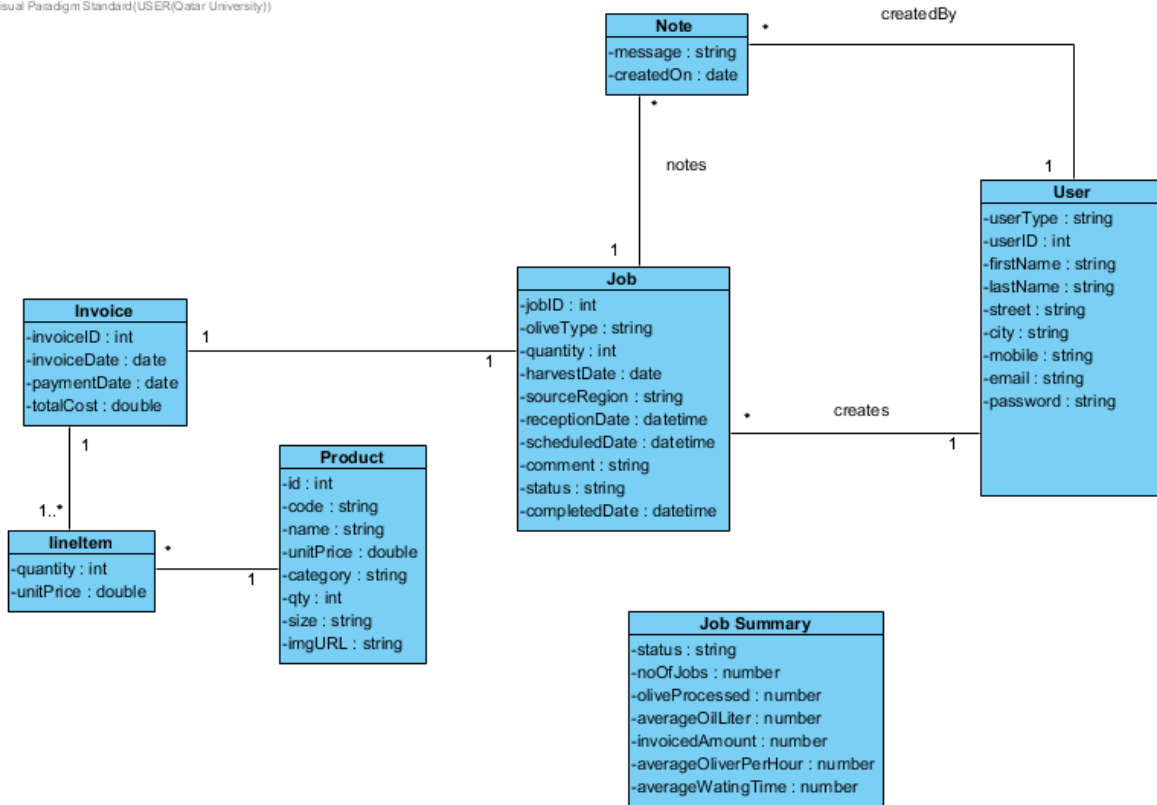
Get Jobs for Mill Calendar	8		Good work	8
Add Job Note	5		When adding a note, you should specify createdBy	4
Get Job Notes	5		Should return an array of notes	4
Get Products (i.e., Products Catalog use case)	8		Excellent	8
Add Product	6		Excellent	6
Update Product	5		Excellent	5
5) Testing documentation using screen shots illustrating the testing of Web UI and Web API.	5		Excellent	5
- Discussion of the project contribution of each team member [-10pts if not done]				
Total	100		Excellent work Ali El-Jaam made no project contribution (assigned grade = 0)	95
Copying and/or plagiarism or not being able to explain or answer questions about the implementation	- 100%			

* **Possible grading for functionality:** *Complete and Working* (get 70% of the assigned grade), *Complete and Not working* (lose 40% of assigned grade) and *Not done* get 0. The remaining grade is assigned to the quality of the implementation. In case your implementation is not working then 40% of the grade will be lost and the remaining 60% will be determined based on of the code quality and how close your solution to the working implementation. Quality **correct application of MVC**, includes meaningful naming of identifiers, no redundant code, simple and efficient design, clean code without unnecessary files/code, use of comments where necessary, proper white space and indentation. **Marks will be reduced** for code duplication, poor/inefficient coding practices, poor naming of identifiers and **unnecessary complex/poor user interface design**.

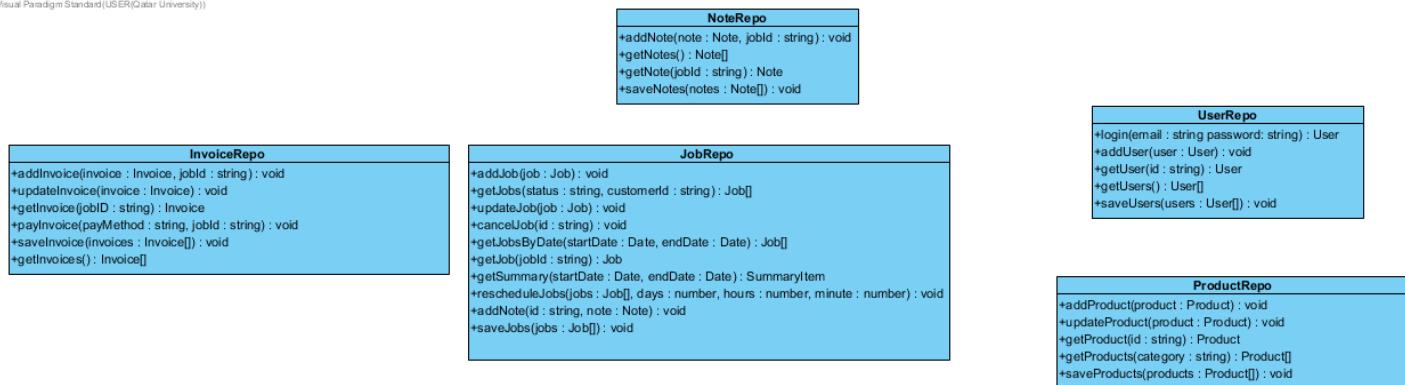
1. Design Documentation

1.1. Class Diagrams: Entities, Repositories and Services

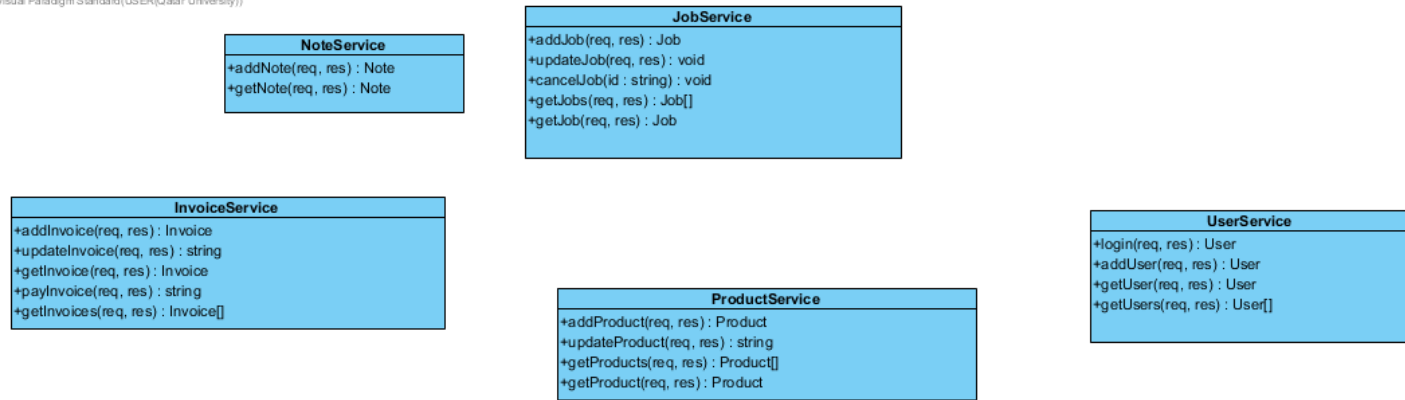
Entities Class Diagram



Repositories Class Diagram



Services

**Services Table**

JobService		
Verb	Url	Function
get	/api/jobs	getJobs
post	/api/jobs	addJob
get	/api/jobs/:id	getJob
put	/api/jobs/:id	updateJob
get	/api/jobs/:id/cancel	cancelJob

NoteService		
Verb	Url	Function
get	/api/jobs/:id/notes	getNote
post	/api/jobs/:id/notes	addNote

InvoiceService		
Verb	Url	Function
get	/api/jobs/:id/invoice	getInvoice
post	/api/jobs/:id/invoice	addInvoice
get	/api/jobs/:id/invoice	updateInvoice

get	/api/jobs/:id/invoice/:payment	payInvoice
get	/api/invoices	getInvoices

UserService		
Verb	Url	Function
get	/api/users	getUsers
post	/api/users	addUser
get	/api/login	login
get	/api/users/:id	getUser

ProductService		
Verb	Url	Function
get	/api/products	getInvoices
post	/api/products	addInvoice
get	/api/products/:id	getInvoice
put	/api/products/:id	updateInvoice

1.2. Discussion of design rationale of 5 key design decisions

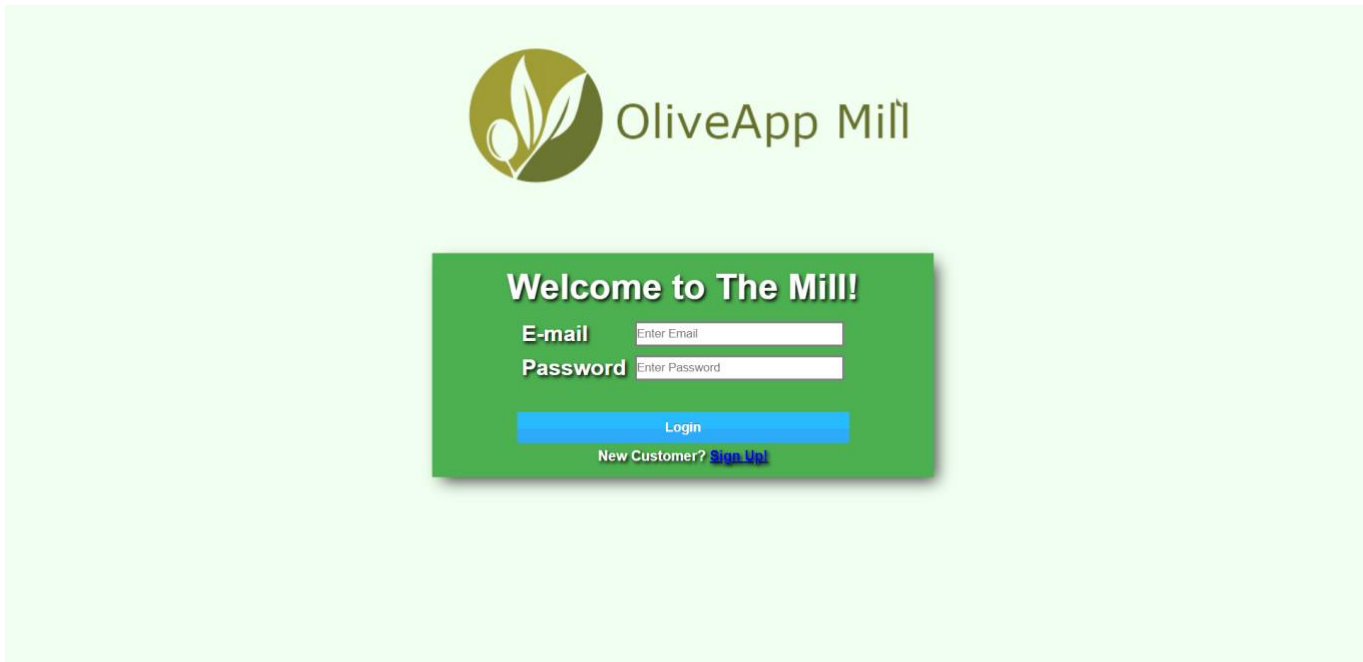
- 1.2.1. Our team has decided that for use cases that involves the use of forms, we will reusing the same web UI design for both the Add (Data) and Update (Data). We decided this to save ourselves development time from implementing similar HTML pages with minor changes and instead be manipulating the base pages later on with JavaScript depending on which use case will be selected by the user. Using JavaScript we will be changing the title of the form to match the use case being used and for the update use cases we will bill filling in the fields by default depending on what object the user chooses to update. We feel this will bring a level of consistency that the user might appreciate due to being familiar with the designs.
- 1.2.2. For the design of our UI, we have decided on a universal approach with some elements being shared across all html pages with the exception of the login. We have decided to keep the same Header and Navbar across all pages with the exception of the login page. We believe these elements are crucial for all the webpages and further emphasise our importance of consistency between pages in order to make our website more user-friendly. Our plan is to simply have a base page (the index) and will replace the body of this page with contents of our pages. These will be triggered through pressing the links on the navbar.
- 1.2.3. In order to make our website more appealing, we have decided to use some open-resources from the internet when we wish to implement a more complex design for one of our elements within a page. For example, we used a button generator from a website to get our

desired look for a button. Like the first point, this will mainly serve to save ourselves the development time in trying to reach the appropriate look we strive to achieve. However, it is important to note that we have also strictly limited our use of these resources in order for the design to better reflect our html and css skills. Our goal is to only use these resources in order to achieve a desired design while keeping the structure of our page to be dependent on our skills.

- 1.2.4. We have noticed that a lot of the use cases are dependent on the jobs. Each Invoice and Note are linked to only one job and unrelated to other jobs. This observation led to how we determined the URLs of the webAPI. In order to get an Invoice or Note, you have to go through the URL to access the related job first. We believe this is the appropriate measure to take in order to increase clarity as when you want to get an invoice or a note, you want to get them for a specific job. Other data which are less dependent on other data such as user have a unique URL that does not go through the URL of another data.
- 1.2.5. In our implementation of the webAPI, we have decided not to implement any services that will utilize DELETE. We have come to this conclusion after reading through the requirements and none of them requires any data to be deleted. The closest thing in our webAPI to deleting is the Cancel Job Use Case. In this case, we are simply changing the status of the job rather than removing that job entirely from our file. This will allow us to filter just the cancelled jobs and view all the jobs that were cancelled.
- 1.2.6. While designing some of the repositories, we noticed that some repositories are required for some functions of another repository. The Invoice Repo in particular uses the JobRepo in order for a job's status to be changed when dealing with the payment of an invoice. However, this could simply be due to the nature of working with files instead of a database and may change our implementation after implementing MongoDB for our work.

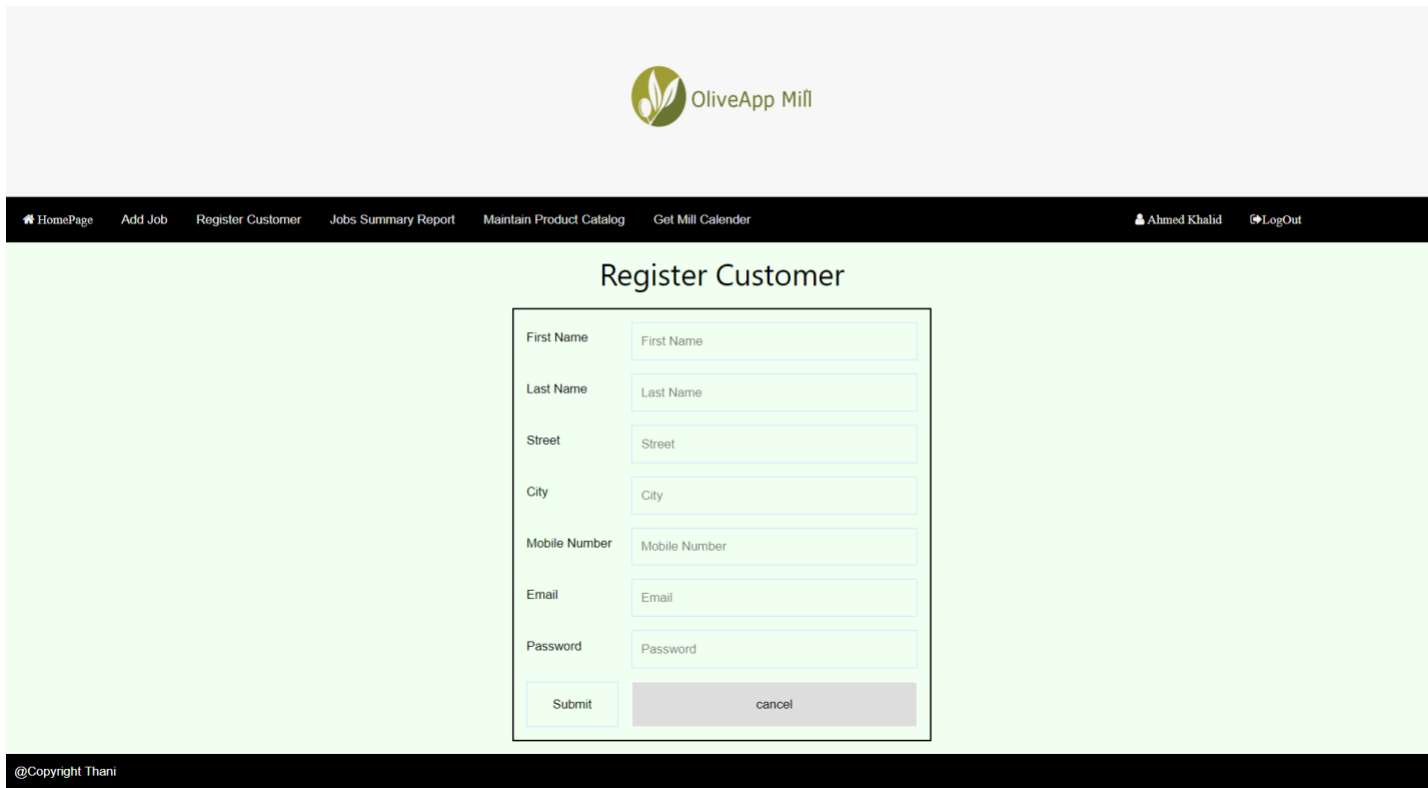
2. User Interface Testing

2.1. Login



The login page features the OliveApp Mill logo at the top center. Below the logo is a green rectangular box with a white border. Inside this box, the text "Welcome to The Mill!" is displayed in white. Below this text are two input fields: "E-mail" and "Password", each with a placeholder text "Enter Email" and "Enter Password" respectively. A blue "Login" button is positioned below the input fields. At the bottom of the green box, there is a link "New Customer? Sign Up!" in white text.


2.2. Register Customer:



The register customer page has a header with the OliveApp Mill logo. Below the header is a black navigation bar containing the following links: "HomePage", "Add Job", "Register Customer", "Jobs Summary Report", "Maintain Product Catalog", and "Get Mill Calendar". On the right side of the navigation bar, the user's name "Ahmed Khalid" and a "LogOut" button are displayed. The main content area is titled "Register Customer" and contains a form with the following fields: "First Name", "Last Name", "Street", "City", "Mobile Number", "Email", and "Password". Each field has a corresponding input box. At the bottom of the form, there are two buttons: "Submit" and "cancel".

@Copyright Thani

2.3. Get Jobs : Home page has the list of jobs



[Home Page](#) [Add Job](#) [Register Customer](#) [Jobs Summary Report](#) [Maintain Product Catalog](#) [Get Mill Calendar](#) Ahmed Khalid [Logout](#)


Jobs List

Job Status : Pending Customer list : Customer 1 [Search](#) [Reschedule](#)

	Job ID	Job Name	Job Date	Job Details	Status	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job

Note: Check Boxes only appear for Pending Jobs to allow for rescheduling

2.4. Add Job



[Home Page](#) [Add Job](#) [Register Customer](#) [Jobs Summary Report](#) [Maintain Product Catalog](#) [Get Mill Calendar](#) Ahmed Khalid [Logout](#)

Add Job

Customer

Add Customer

Olive Quantity

Olive Quantity

Olive Type

Harvest Date

mm/dd/yyyy

Source Region

reception date time

mm/dd/yyyy --:-- --

scheduled milling date time

mm/dd/yyyy --:-- --

Job Detail

Comments

Container

Customer

Mill

Stainless Steel 50L

Number


Add

Submit



cancel

@Copyright Thani

2.5. Update Job : Same template as add job expect when using javascript it will be populated with data, and it is not available in the folder given since it will be created with javascript folder for next stage.

 OliveApp Mill

[HomePage](#)
[Add Job](#)
[Register Customer](#)
[Jobs Summary Report](#)
[Maintain Product Catalog](#)
[Get Mill Calendar](#)

 Ahmed Khalid
 Logout

Update Job

Customer

Add Customer

Add Customer

Olive Quantity

Olive Quantity

Olive Type

Harvest Date

mm/dd/yyyy

Source Region

reception date time

mm/dd/yyyy --:-- --

scheduled milling date time

mm/dd/yyyy --:-- --

Job Detail

Comments

Job Status

Pending

Container

Customer

Mill

Stainless Steel 50L

Number

Add

Oil Liters

Enter Liters Produced

Submit

cancel

2.6. Cancel Job: Doesn't have a page since it will be simple button on the table of home page Which will prompt(alert) if your sure you want to delete it. In second phase it will be a popup small window (more colourful).

localhost:63342 says

Are you sure you want to delete this Job?

OK
Cancel


🏠 Home Page Add Job Register Customer Jobs Summary Report Maintain Product Catalog Get Mill Calendar 👤 Ahmed Khalid 🔒 Log Out

Jobs List

Job Status : Pending Customer list : Customer 1 Search Reschedule

■	Job ID	Job Name	Job Date	Job Details	Status	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
❑	12131313	Milling	29/3/2018	▶ Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
❑	12131313	Milling	29/3/2018	▶ Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
❑	12131313	Milling	29/3/2018	▶ Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
❑	12131313	Milling	29/3/2018	▶ Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
❑	12131313	Milling	29/3/2018	▶ Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
❑	12131313	Milling	29/3/2018	▶ Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
❑	12131313	Milling	29/3/2018	▶ Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
❑	12131313	Milling	29/3/2018	▶ Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
❑	12131313	Milling	29/3/2018	▶ Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job

2.7. Add Invoice

 OliveApp Mill

[HomePage](#) [Add Job](#) [Register Customer](#) [Jobs Summary Report](#) [Maintain Product Catalog](#) [Get Mill Calender](#) Ahmed Khalid [Logout](#)

Invoice

Jobs Olive Quantity1000

Unit Price10 QR

Containers

PlasticAddRemove

Added Containers : Plastic : 3Stainless Steel : 4

Total price : 1250 QR

SubmitCancel

@Copyrights Thani

2.8. Update Invoice: Same as page above.

2.9. View Invoice: Pop-up , click on update invoice to change.

StainlessSteel Container x 11
Plastic Container x 3
Quantity of Olive Processed : 1100
Unit Price : 12.5 QR

Total Price : 2350 QR

HomePageAdd

Jobs List

Job Status : PendingCustomer list : Customer 1SearchReschedule

	Job ID	Job Name	Job Date	Job Details	Status	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job

2.10. Pay Invoice:

Payment Method :
Cash


HomePageAdd JobRegister CustomerJobs Summary ReportMaintain Product CatalogGet Mill CalendarAhmed KhalidLogout

Jobs List

Job Status : PendingCustomer list : Customer 1SearchReschedule







	Job ID	Job Name	Job Date	Job Details	Status	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job
<input type="checkbox"/>	12131313	Milling	29/3/2018	► Dig A Hole	Completed	Add Invoice	Update Invoice	View Invoice	Pay Invoice	Update Job	Cancel Job

2.11. Mill Calendar: Fail safe added as <a> link after using javascript (phase 2) other methods will ensure calender get loaded.




[Home](#) [Add Job](#) [Register Customer](#) [Jobs Summary Report](#) [Maintain Product Catalog](#) [Get Mill Calendar](#) Ahmed Khalid [Log Out](#)

Milling Calendar

Today   March 2019    

Sun	Mon	Tue	Wed	Thu	Fri	Sat
24	25	26	27	28	1 Mar	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30 1:30am Milling Job # 1
31 1:30am Milling Job # 2	1 Apr	2	3	4	5	6


Events shown in time zone: Arabian Standard Time - Qatar 

@Copyright Thani

2.12. Add Job Note: Available as <detail> on Home page. Check Job Detail Column

2.13. Get Job Notes: Available as <detail> on Home page. Check Job Detail Column

2.14. Get Products (i.e., Products Catalog use case):




[Home](#) [Add Job](#) [Register Customer](#) [Jobs Summary Report](#) [Maintain Product Catalog](#) [Get Mill Calendar](#) Ahmed Khalid [Log Out](#)

Product Catalog

Milling Service										
Product ID		Product Code		Product Name		Product Category		Product Unit Price	Update Service	
1		MillingFee		Milling fee (per Kg)		Service		1.5	Update	
Mill Containers										
Product ID	Product Code	Product Name		Product Image	Product Category	Product Size	Product type	Product Unit Price	Product Quantity	Remove Product
2	SS001-50L	50L Sansone Stainless Steel Container with Tap			Container	50L	Stainless Steel	50	500	Remove
3	SS001-30L	30L Sansone Stainless Steel Container with Tap			Container	30L	Stainless Steel	30	300	Remove
4	SS001-10L	10L Sansone Stainless Steel Container with Tap			Container	10L	Stainless Steel	10	100	Remove
5	PL001-50L	50L Plastic Oil Container			Container	50L	Plastic	25	250	Remove
6	PL001-30L	30L Plastic Oil Container			Container	30L	Plastic	15	150	Remove
7	PL001-10L	10L Plastic Oil Container			Container	10L	Plastic	5	50	Remove

[Add Product to Catalog](#)

2.15. Add / Update Product: update product is the same exact javascript would be used to adjust which fields would be available and will be populated using javascript.

 OliveApp Mill

[Home Page](#)
[Add Job](#)
[Register Customer](#)
[Jobs Summary Report](#)
[Maintain Product Catalog](#)
[Get Mill Calendar](#)
Ahmed Khalid
[Log Out](#)

Add Product to Catalog

Product Code

Product Name

Category

Service

Size

Available Quantity

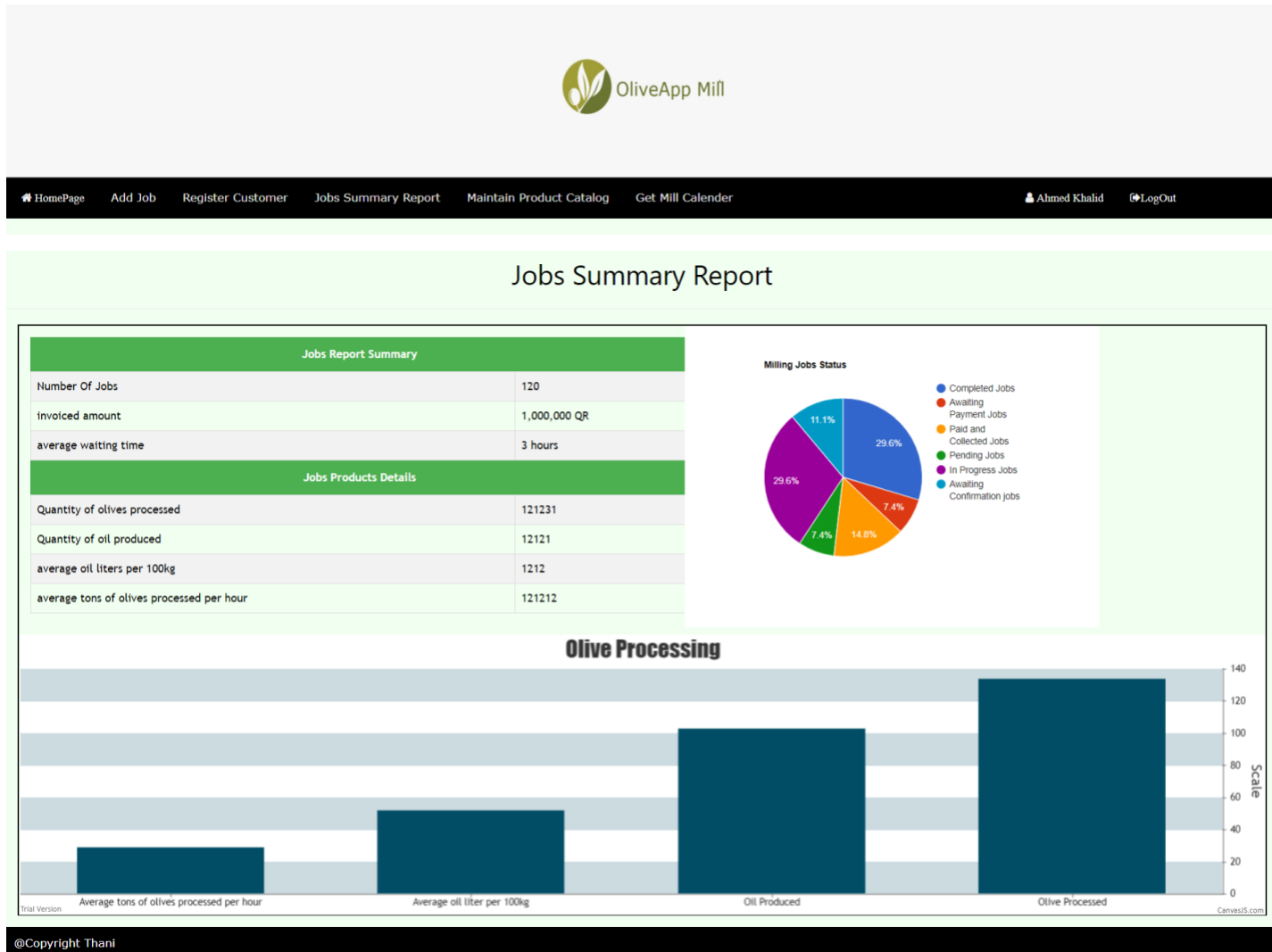
Unit Price

Image

No file chosen

@Copyright Thani

2.16. Jobs Summary Report



3. Web API Testing

3.1. Login

GET http://localhost:8000/api/login?email=testing@gmail.com&password=test

Params Authorization Headers (1) Body Pre-request Script Tests

Query Params

KEY	VALUE	DESCRIPTION
email	testing@gmail.com	
password	test	

Body Cookies Headers (6) Test Results Status: 200 OK Time: 44 ms Size: 391 B

Pretty Raw Preview JSON

```
1 {
2   "userType": "Customer",
3   "id": 1,
4   "firstName": "Ali",
5   "lastName": "Hussaini",
6   "street": "Arab League Street",
7   "city": "Doha",
8   "mobile": 66338094,
9   "email": "testing@gmail.com",
10  "password": "test"
11 }
```

3.2. Add Customer

Params Authorization Headers (1) Body Pre-request Script Tests

none form-data x-www-form-urlencoded raw binary JSON (application/json)

```
1 {
2   "userType": "Customer",
3   "firstName": "Roberto",
4   "lastName": "Carlos",
5   "street": "Arab League Street",
6   "city": "Doha",
7   "mobile": 66338094,
8   "email": "test@yahoo.com",
9   "password": "test2"
10 }
```

Body Cookies Headers (6) Test Results Status: 201 Created Time: 26 ms Size: 396 B

Pretty Raw Preview JSON

```
1 {
2   "userType": "Customer",
3   "firstName": "Roberto",
4   "lastName": "Carlos",
5   "street": "Arab League Street",
6   "city": "Doha",
7   "mobile": 66338094,
8   "email": "test@yahoo.com",
9   "password": "test2",
10  "id": 4
11 }
```

3.3. Get Jobs

Note: Temporarily added just the userId as a property of job. After we implement this using database, it will have a reference to the user from the Users database

http://localhost:8000/api/jobs

GET http://localhost:8000/api/jobs Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary JSON (application/json) Beautify

```
1 {
2   "oliveType": "Picholine",
3   "quantity": 6,
4   "status": "Pending",
5   "harvestDate": "2011-12-12T00:00:00.000-0800",
6   "sourceRegion": "Al Wakra",
7   "receptionDate": "2011-12-12T00:00:00.000-0800",
8   "scheduledDate": "2011-12-12T00:00:00.000-0800",
9   "userId": 3
10 }
```

Body Cookies Headers (6) Test Results Status: 200 OK Time: 42 ms Size: 1.49 KB Download

Pretty Raw Preview JSON

```
1 [
2   {
3     "jobId": 1,
4     "oliveType": "Picholine",
5     "quantity": 4,
6     "status": "In Progress",
7     "harvestDate": "2011-12-12T00:00:00.000-0800",
8     "sourceRegion": "Al Wakra",
9     "receptionDate": "2011-12-12T00:00:00.000-0800",
10    "scheduledDate": "2011-12-12T00:00:00.000-0800",
11    "userId": 1
12  },
13  {
14    "jobId": 2,
15    "oliveType": "Picholine",
16    "quantity": 4,
17    "status": "In Progress",
18    "harvestDate": "2011-12-12T00:00:00.000-0800",
19    "sourceRegion": "Al Wakra",
20    "receptionDate": "2011-12-12T00:00:00.000-0800",
21    "scheduledDate": "2011-12-12T00:00:00.000-0800",
22    "userId": 1
23  }
24 ]
```

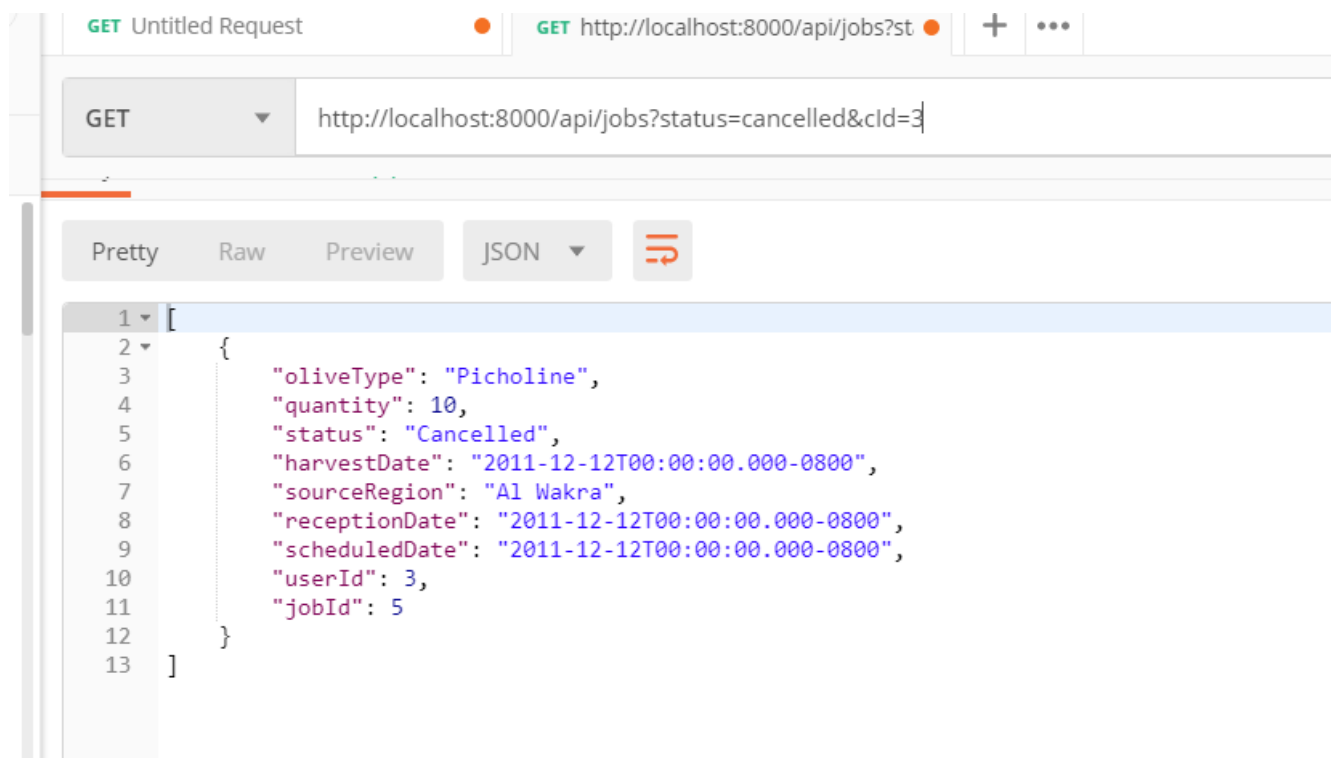
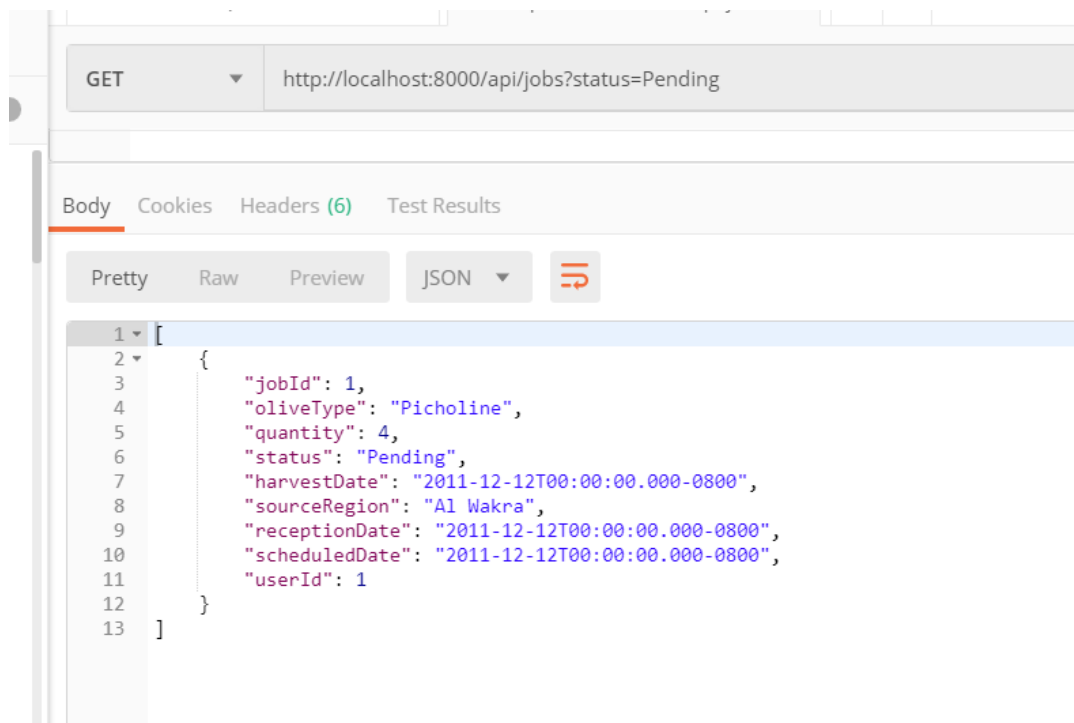
GET Untitled Request GET http://localhost:8000/api/jobs?st + ... No Environment

GET http://localhost:8000/api/jobs?cId=2

Body Cookies Headers (6) Test Results Status: 200 OK Time: 20 ms

Pretty Raw Preview JSON

```
1 [
2   {
3     "jobId": 4,
4     "oliveType": "Picholine",
5     "quantity": 6,
6     "status": "Cancelled",
7     "harvestDate": "2011-12-12T00:00:00.000-0800",
8     "sourceRegion": "Al Wakra",
9     "receptionDate": "2011-12-12T00:00:00.000-0800",
10    "scheduledDate": "2011-12-12T00:00:00.000-0800",
11    "userId": 2
12  }
13 ]
```



3.4. Add Job

POST http://localhost:8000/api/jobs

No Environment

http://localhost:8000/api/jobs

POST

http://localhost:8000/api/jobs

Send

Params

Authorization

Headers (1)

Body

Pre-request Script

Tests

Cookies

Code

Cor

none

form-data

x-www-form-urlencoded

raw

binary

JSON (application/json)

```
1 {
2   "oliveType": "Picholine",
3   "quantity": 10,
4   "status": "Pending",
5   "harvestDate": "2011-12-12T00:00:00.000-0800",
6   "sourceRegion": "Al Wakra",
7   "receptionDate": "2011-12-12T00:00:00.000-0800",
8   "scheduledDate": "2011-12-12T00:00:00.000-0800",
9   "userId": 3
10 }
```

Body

Cookies

Headers (6)

Test Results

Status: 201 Created

Time: 51 ms

Size: 462 B

D

Pretty

Raw

Preview

JSON

```
1 {
2   "oliveType": "Picholine",
3   "quantity": 10,
4   "status": "Pending",
5   "harvestDate": "2011-12-12T00:00:00.000-0800",
6   "sourceRegion": "Al Wakra",
7   "receptionDate": "2011-12-12T00:00:00.000-0800",
8   "scheduledDate": "2011-12-12T00:00:00.000-0800",
9   "userId": 3,
10  "jobId": 5
11 }
```

3.5. Update Job

PUT http://localhost:8000/api/jobs/6

No Environment

http://localhost:8000/api/jobs/6

PUT

http://localhost:8000/api/jobs/6

Send

Params

Authorization

Headers (1)

Body

Pre-request Script

Tests

Cookies

Code

Cor

none

form-data

x-www-form-urlencoded

raw

binary

JSON (application/json)

```
1 {
2   "oliveType": "Picholine",
3   "quantity": 5,
4   "status": "Pending",
5   "harvestDate": "2011-12-12T00:00:00.000-0800",
6   "sourceRegion": "Al Wakra",
7   "receptionDate": "2011-12-12T00:00:00.000-0800",
8   "scheduledDate": "2011-12-12T00:00:00.000-0800",
9   "userId": 3,
10  "id": 6
11 }
```

Body

Cookies

Headers (6)

Test Results

Status: 200 OK

Time: 66 ms

Size: 230 B

D

Pretty

Raw

Preview

HTML

```
1 Job Successfully Updated!
```

```
GET http://localhost:8000/api/jobs

No Environment

GET http://localhost:8000/api/jobs Send Save

Pretty Raw Preview JSON

24 },
25 {
26   "jobId": 3,
27   "oliveType": "Picholine",
28   "quantity": 6,
29   "status": "Cancelled",
30   "harvestDate": "2011-12-12T00:00:00.000-0800",
31   "sourceRegion": "Al Wakra",
32   "receptionDate": "2011-12-12T00:00:00.000-0800",
33   "scheduledDate": "2011-12-12T00:00:00.000-0800",
34   "userId": 1
35 },
36 {
37   "jobId": 4,
38   "oliveType": "Picholine",
39   "quantity": 6,
40   "status": "Cancelled",
41   "harvestDate": "2011-12-12T00:00:00.000-0800",
42   "sourceRegion": "Al Wakra",
43   "receptionDate": "2011-12-12T00:00:00.000-0800",
44   "scheduledDate": "2011-12-12T00:00:00.000-0800",
45   "userId": 2
46 },
47 {
48   "oliveType": "Picholine",
49   "quantity": 5,
50   "status": "Pending",
51   "harvestDate": "2011-12-12T00:00:00.000-0800",
52   "sourceRegion": "Al Wakra",
53   "receptionDate": "2011-12-12T00:00:00.000-0800",
54   "scheduledDate": "2011-12-12T00:00:00.000-0800",
55   "userId": 3,
56   "id": 6
57 }
58 1
```

3.6. Cancel Job

```
GET http://localhost:8000/api/jobs/5/

No Environment

http://localhost:8000/api/jobs/5/cancel

GET http://localhost:8000/api/jobs/5/cancel Send

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code C
● none ● form-data ● x-www-form-urlencoded ● raw ● binary JSON (application/json)

1 {
2   "oliveType": "Picholine",
3   "quantity": 10,
4   "status": "Pending",
5   "harvestDate": "2011-12-12T00:00:00.000-0800",
6   "sourceRegion": "Al Wakra",
7   "receptionDate": "2011-12-12T00:00:00.000-0800",
8   "scheduledDate": "2011-12-12T00:00:00.000-0800",
9   "userId": 3
10 }
```

Body Cookies Headers (6) Test Results Status: 200 OK Time: 13 ms Size: 231 B

Pretty Raw Preview HTML

```
{ 1 } Job Successfully Cancelled
```

```

40      },
47      {
48        "oliveType": "Picholine",
49        "quantity": 10,
50        "status": "Cancelled",
51        "harvestDate": "2011-12-12T00:00:00.000-0800",
52        "sourceRegion": "Al Wakra",
53        "receptionDate": "2011-12-12T00:00:00.000-0800",
54        "scheduledDate": "2011-12-12T00:00:00.000-0800",
55        "userId": 3,
56        "jobId": 5
57      }
58    ]
59  }

```

3.7. Add Invoice

POST http://localhost:8000/api/jobs/3
No Environment

http://localhost:8000/api/jobs/3/invoice

POST
http://localhost:8000/api/jobs/3/invoice
Send

Params
Authorization
Headers (1)
Body
Pre-request Script
Tests
Cookies
Code

none
form-data
x-www-form-urlencoded
raw
binary
JSON (application/json)

```

1 {
2   "lineItems": [
3     {
4       "productId": 1,
5       "quantity": 15,
6       "unitPrice": 1.5
7     },
8     {
9       "productId": 7,
10      "quantity": 3,
11      "unitPrice": 30
12    }
13  ]
14 }

```

ody
Cookies
Headers (6)
Test Results
Status: 201 Created
Time: 33 ms
Size: 401 B

Pretty
Raw
Preview
JSON

```

1 {
2   "lineItems": [
3     {
4       "productId": 1,
5       "quantity": 15,
6       "unitPrice": 1.5,
7       "price": 22.5
8     },
9     {
10      "productId": 7,
11      "quantity": 3,
12      "unitPrice": 30,

```


3.8. Update Invoice

The screenshot shows the Postman application with a PUT request to `http://localhost:8000/api/jobs/3/invoice`. The request body is a JSON object with two line items. The status bar indicates a 201 Created response.

Request:

```
PUT http://localhost:8000/api/jobs/3/invoice
```

Body (JSON):

```
{  "lineItems": [    {      "productId": 1,      "quantity": 15,      "unitPrice": 1.5    },    {      "productId": 7,      "quantity": 6,      "unitPrice": 30    }  ]}
```

Response: Status: 201 Created, Time: 8 ms, Size: 237 B

Test Results:

```
f 1 Invoice Successfully Update
```

The screenshot shows the Postman application with a GET request to `http://localhost:8000/api/jobs/3/invoice`. The response body is a JSON object with two line items, including a 'price' field for each. The status bar indicates a 200 OK response.

Request:

```
GET http://localhost:8000/api/jobs/3/invoice
```

Response (JSON):

```
{  "lineItems": [    {      "productId": 1,      "quantity": 15,      "unitPrice": 1.5,      "price": 22.5    },    {      "productId": 7,      "quantity": 6,      "unitPrice": 30,      "price": 180    }  ]}
```

Test Results:

```
f 1 Invoice Successfully Update
```

3.9. Get Invoice

GET http://localhost:8000/api/jobs/2/

No Environment

http://localhost:8000/api/jobs/2/invoice

GET

http://localhost:8000/api/jobs/2/invoice

Send

Save

Params

Authorization

Headers (1)

Body

Pre-request Script

Tests

Cookies

Code

Comments (0)

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body

Cookies

Headers (6)

Test Results

Status: 200 OK

Time: 53 ms

Size: 391 B

Download

Pretty

Raw

Preview

JSON

```
1 {
2   "lineItems": [
3     {
4       "productId": 1,
5       "quantity": 5,
6       "unitPrice": 1.5,
7       "price": 7.5
8     },
9     {
10      "productId": 7,
11      "quantity": 3,
12      "unitPrice": 30,
13      "price": 90
14    }
15  ],
16  "id": 1,
17  "jobId": 2,
18  "date": 1554147468651,
19  "total": 97.5
20 }
```

3.10. Pay Invoice

GET http://localhost:8000/api/jobs/2/

No Environment

http://localhost:8000/api/jobs/2/invoice/cash

GET

http://localhost:8000/api/jobs/2/invoice/cash

Send

Params

Authorization

Headers (1)

Body

Pre-request Script

Tests

Cookies

Code

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body

Cookies

Headers (6)

Test Results

Status: 201 Created

Time: 30 ms

Size: 235 B

Pretty

Raw

Preview

HTML

```
1 Invoice Successfully paid
```

GET http://localhost:8000/api/jobs/2/ + ... No Environment

http://localhost:8000/api/jobs/2/invoice

GET ▼ http://localhost:8000/api/jobs/2/invoice Send Save

Params Authorization Headers (1) Body ● Pre-request Script Tests Cookies Code Comments (0)

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (6) Test Results Status: 200 OK Time: 23 ms Size: 442 B Download

Pretty Raw Preview JSON ▼ ≡

```

1 {
2   "lineItems": [
3     {
4       "productId": 1,
5       "quantity": 5,
6       "unitPrice": 1.5,
7       "price": 7.5
8     },
9     {
10      "productId": 7,
11      "quantity": 3,
12      "unitPrice": 30,
13      "price": 90
14    }
15  ],
16  "id": 1,
17  "jobId": 2,
18  "date": 1554147468651,
19  "total": 97.5,
20  "paymentMethod": "cash",
21  "paymentDate": 1554148342904
22 }

```

3.11. Get Jobs for Mill Calendar

GET http://localhost:8000/api/jobs?st + ... No Environment

http://localhost:8000/api/jobs?startDate=2011-12-12&endDate=2011-12-12

GET ▼ http://localhost:8000/api/jobs?startDate=2011-12-12&endDate=2011-12-12 Send Save

Params ● Authorization Headers (1) Body ● Pre-request Script Tests Cookies Code Comments (0)

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> startDate	2011-12-12			
<input checked="" type="checkbox"/> endDate	2011-12-12			
Key	Value	Description		

Body Cookies Headers (6) Test Results Status: 200 OK Time: 58 ms Size: 2.35 KB Download

Pretty Raw Preview JSON ▼ ≡

```

1 [
2   {
3     "jobId": 1,
4     "oliveType": "Picholine",
5     "quantity": 4,
6     "status": "In Progress",
7     "harvestDate": "2011-12-12T00:00:00.000-0800",
8     "sourceRegion": "Al Wakra",
9     "receptionDate": "2011-12-12T00:00:00.000-0800",
10    "scheduledDate": "2011-12-12T00:00:00.000-0800",
11    "user": {
12      "userType": "Customer",
13      "id": 1,
14      "firstName": "Ali",
15      "lastName": "Hussaini",
16      "street": "Arab League Street",
17      "city": "Doha"
18    }
19  }
20 ]

```

3.12. Add Job Note

POST http://localhost:8000/api/jobs/2/notes

Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary JSON (application/json) Beautify

```
1 {  
2   "message" : "Will need to be extended"  
3 }
```

Body Cookies Headers (6) Test Results Status: 201 Created Time: 54 ms Size: 310 B Download

Pretty Raw Preview JSON Beautify

```
1 {  
2   "message": "Will need to be extended",  
3   "createdBy": "unknown",  
4   "date": "1554053769520",  
5   "jobId": "2"  
6 }
```

3.13. Get Job Notes

GET http://localhost:8000/api/jobs/1/notes

Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary JSON (application/json) Beautify

```
1 {  
2   "message" : "Will need to be extended"  
3 }
```

Body Cookies Headers (6) Test Results Status: 200 OK Time: 17 ms Size: 316 B Download

Pretty Raw Preview JSON Beautify

```
1 [  
2   {  
3     "createdBy": "Unknown",  
4     "date": "2011-12-12T00:00:00.000-0800",  
5     "message": "Progress Going Well",  
6     "jobId": 1  
7   }  
8 ]
```

3.14. Get Products (i.e., Products Catalog use case)

http://localhost:8000/api/products

GET http://localhost:8000/api/products Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (6) Test Results Status: 200 OK Time: 46 ms Size: 1.58 KB Download

Pretty Raw Preview JSON ≡

```
1 [
2   {
3     "id": 1,
4     "code": "MillingFee",
5     "name": "Milling fee (per Kg)",
6     "category": "Service",
7     "unitPrice": 1.5
8   },
9   {
10    "id": 2,
11    "code": "SS001-50L",
12    "name": "50L Sansone Stainless Steel Container with Tap",
13    "imageUrl": "https://cmps356s19.github.io/images/SS001-50L.jpg",
14    "category": "Container",
15    "size": "50L",
16    "type": "Stainless Steel",
17    "unitPrice": 50,
18    "qty": 500
19  },
20 ]
```

3.15. Add Product

POST http://localhost:8000/api/products + ... No Environment 👁

http://localhost:8000/api/products

POST http://localhost:8000/api/products Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary JSON (application/json) Bea

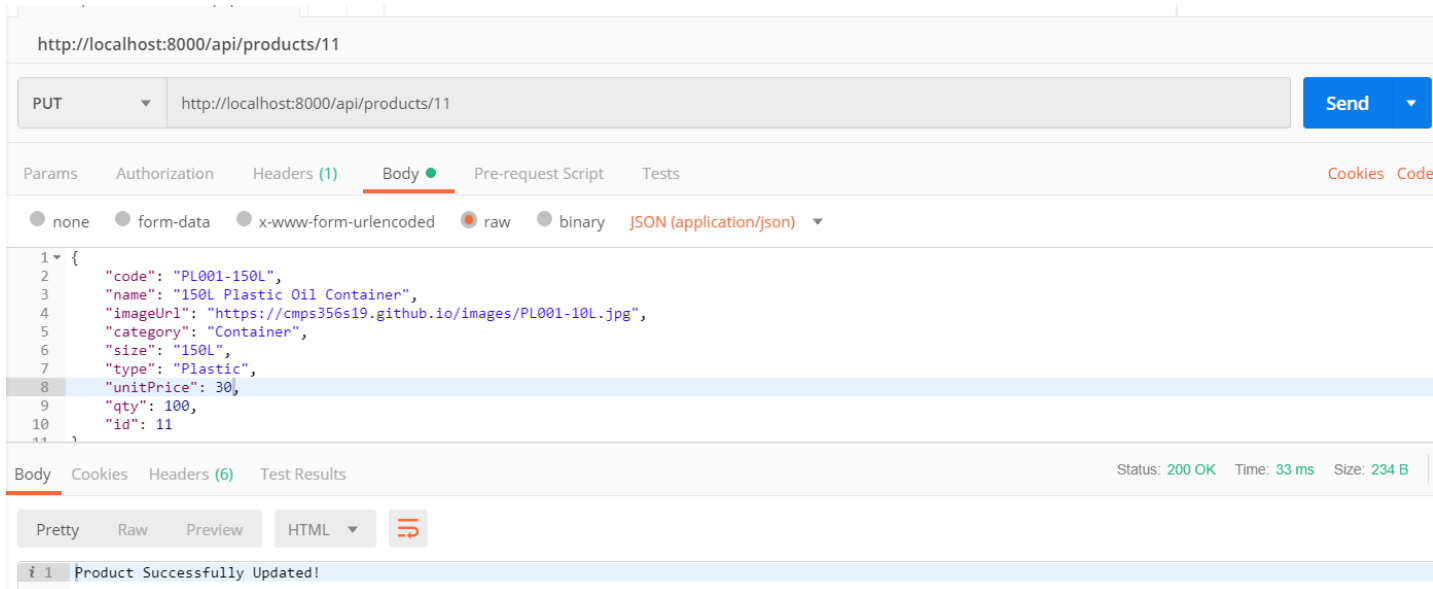
```
1 {
2   "code": "PL001-150L",
3   "name": "150L Plastic Oil Container",
4   "imageUrl": "https://cmps356s19.github.io/images/PL001-10L.jpg",
5   "category": "Container",
6   "size": "150L",
7   "type": "Plastic",
8   "unitPrice": 20,
9   "qty": 100
10 }
```

Body Cookies Headers (6) Test Results Status: 201 Created Time: 10 ms Size: 425 B Downl

Pretty Raw Preview JSON ≡

```
1 {
2   "code": "PL001-150L",
3   "name": "150L Plastic Oil Container",
4   "imageUrl": "https://cmps356s19.github.io/images/PL001-10L.jpg",
5   "category": "Container",
6   "size": "150L",
7   "type": "Plastic",
8   "unitPrice": 20,
9   "qty": 100,
10  "id": 11
11 }
```

3.16. Update Product



Student Contributions

Muhamad Irfan Fauzan

For phase 1, my role in the team was to develop the WebAPI. To do this first I had to design the 3 class diagrams as well to give myself an idea on how I would want to implement the WebAPI. After getting feedback on the class diagrams and make adjustments to them, I started working on the repositories first. I decided on 5 major repositories and used my knowledge from the lab and experience to code appropriate functions that can fulfill the required use cases. Also, I had to write some test files that can be read and written to in order for me to store the datas that were being used for the repositories. After finishing the repositories, I worked on the services. Services just implemented the functions of the repositories but to be used through the web. After organizing my different service classes, I finally moved on to route.js to figure out the necessary urls that would be appropriate to access the various functions within the services. Over the creation of the repositories and services, I made modifications to the class diagram whenever I felt a change was necessary or was more suitable. I also helped give feedback on the web UI whenever I feel like an element of a web UI can be improved on. With the Monday extension, I was also able to contribute on the web UI by working on the login page. I was also the person who wrote the key decisions in the report.

Thani Al-Thani:

In this Phase I implemented the HTML & CSS and took feedback from Muhammed Fuzan.

We both shared and took opinions but mostly he did the web api and designs.

I implemented and designed the web pages. And took key decisions on implementations to simplify the User experience.

The discussion and reasoning was given in the main Document and will be more detailed on the upcoming office hour presentation.

Ali Jaam

In this phase, I have not contributed much to the project because the work was split between my 2 other members and I was not given a specific objective to work on. They have done all the work from HTML to CSS to WEB api to documentation. However, I will contribute in phase 2.