EXPERIMENT - 10

Software Engineering Lab

Aim

To perform various testing using the testing tool unit testing, integration testing for a sample code of the suggested system.

Syeda Reeha Quasar
14114802719
4C7

EXPERIMENT – 10

Aim:

To perform various testing using the testing tool unit testing, integration testing for a sample code of the suggested system.

Theory:

"Testing is the process of executing a program with the intent of finding errors."

The purpose of software testing is:

1. To demonstrate that the product performs each function intended.
2. To demonstrate that the internal operation of the product performs according to specification and all internal components have been adequately exercised.
3. To increase our confidence in the proper functioning of the software.
4. To show the product is free from defect.
5. All of the above.

Unit Testing — Checks each coded module for the presence of bugs. Unit testing's purpose is to ensure that each as- built module behaves according to its specification defined during detailed design.

Integration Testing ---- Interconnects sets of previously tested modules to ensure that the sets behave as well as they did as independently tested modules. Integration testing's purpose is to ensure that each as-built component behaves according to its specification defined during preliminary design.

Performance Instruction:

Unit test planning—

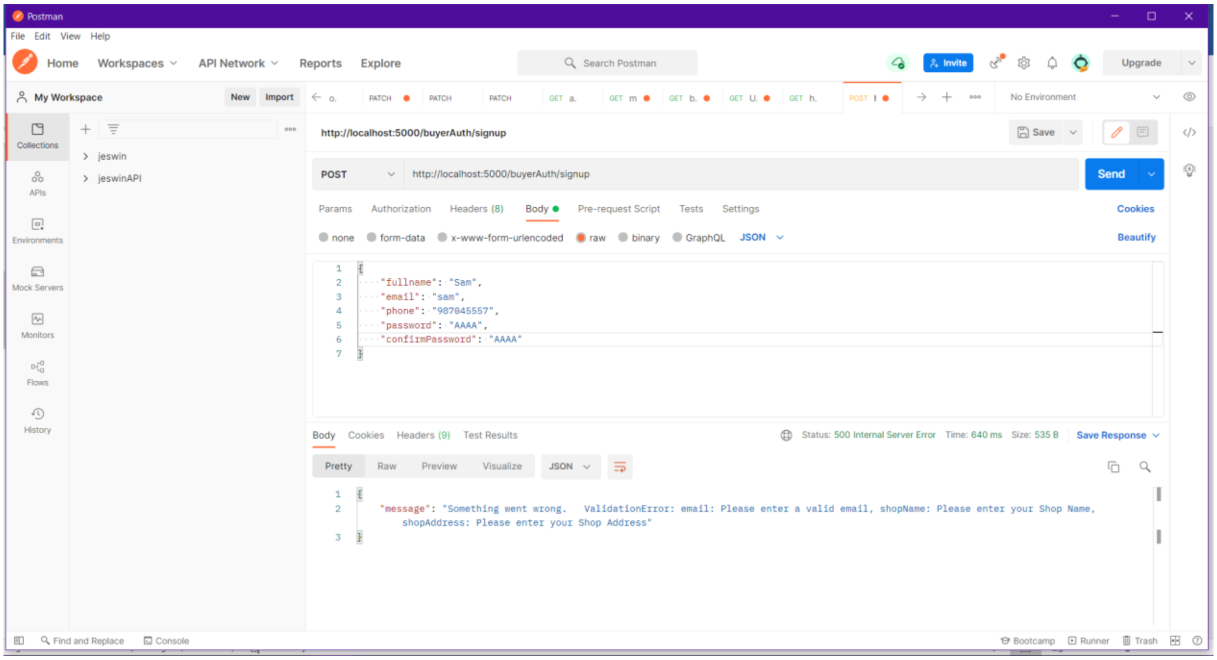
Generate plans and procedures to test each module independently and thoroughly.

Integration Test Planning—
Generate plans and procedures to effect orderly system integration.

Output:

Backend validations: - Authentication module -> (same for buyer, seller and deliverer)

Missing fields – "Something went wrong"



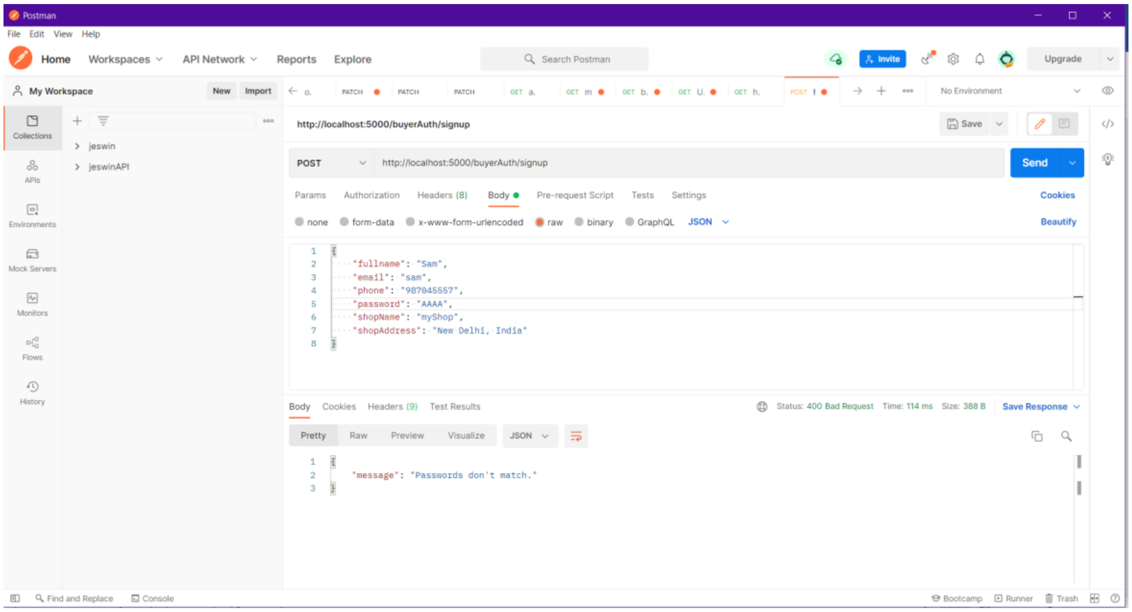
POST http://localhost:5000/buyerAuth/signup

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "fullname": "Sam",
    "email": "sam",
    "phone": "987045557",
    "password": "AAAA",
    "confirmPassword": "AAAA"
}
```

Status: 500 Internal Server Error Time: 640 ms Size: 535 B

```
{
    "message": "Something went wrong.   ValidationError: email: Please enter a valid email, shopName: Please enter your Shop Name,
        shopAddress: Please enter your Shop Address"
}
```



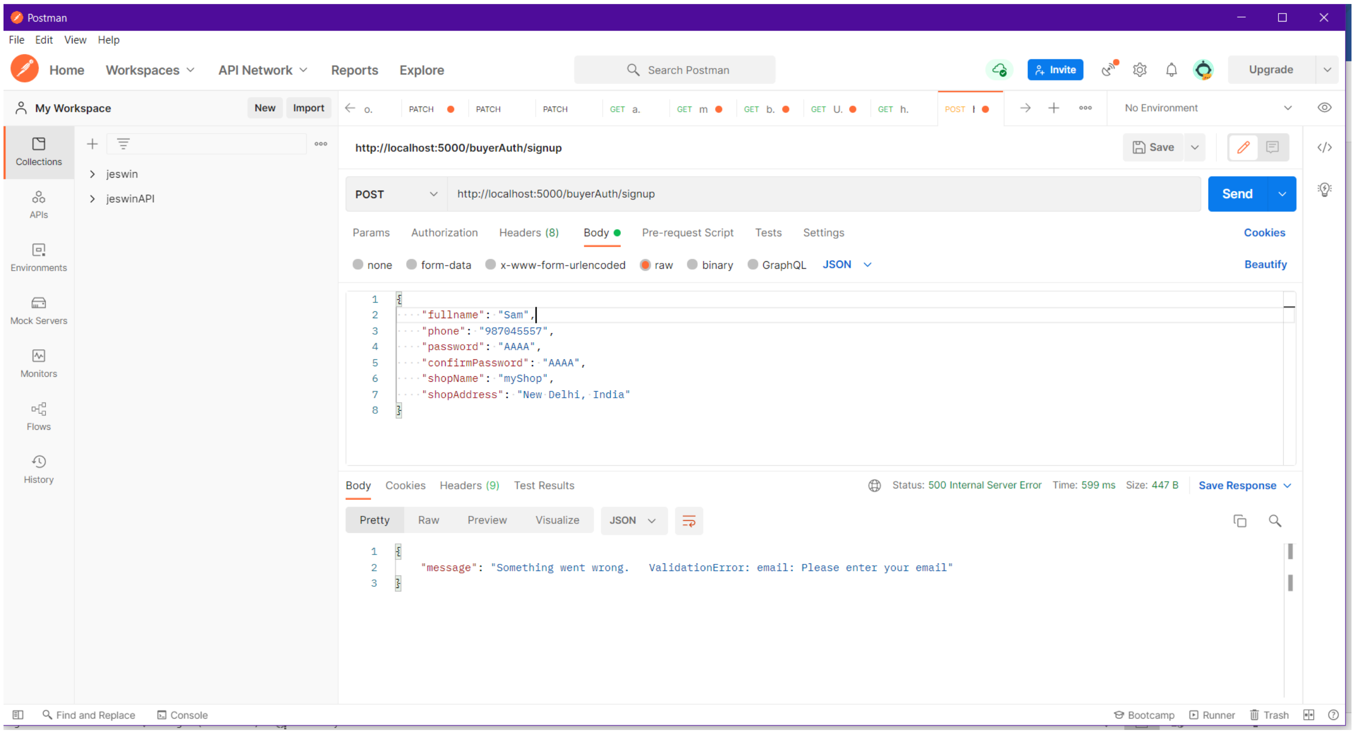
POST http://localhost:5000/buyerAuth/signup

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "fullname": "Sam",
    "email": "sam",
    "phone": "987045557",
    "password": "AAAA",
    "shopName": "myShop",
    "shopAddress": "New Delhi, India"
}
```

Status: 400 Bad Request Time: 114 ms Size: 388 B

```
{
    "message": "Passwords don't match."
}
```



http://localhost:5000/buyerAuth/signup

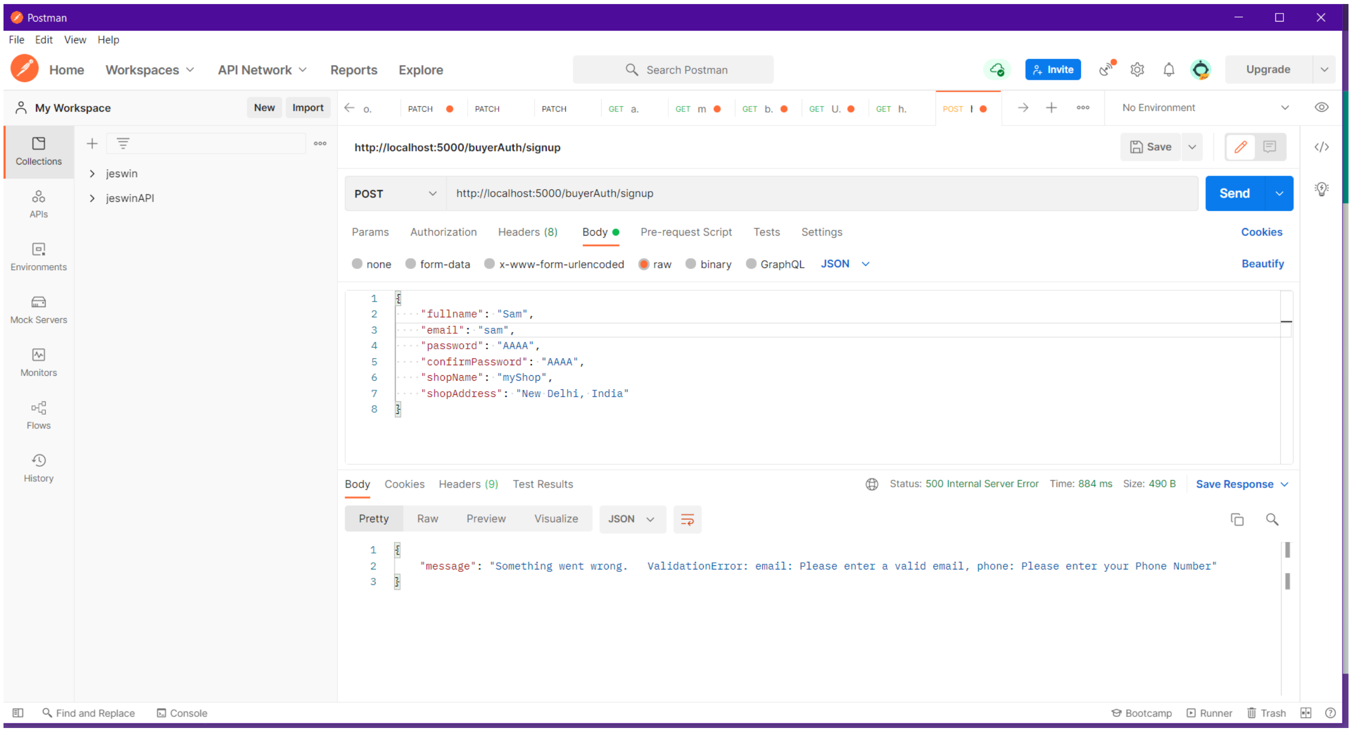
POST http://localhost:5000/buyerAuth/signup

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

```
{
    "fullname": "Sam",
    "phone": "987045557",
    "password": "AAAA",
    "confirmPassword": "AAAA",
    "shopName": "myShop",
    "shopAddress": "New Delhi, India"
}
```

Status: 500 Internal Server Error Time: 599 ms Size: 447 B

```
{
    "message": "Something went wrong.  ValidationError: email: Please enter your email"
}
```



http://localhost:5000/buyerAuth/signup

POST http://localhost:5000/buyerAuth/signup

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

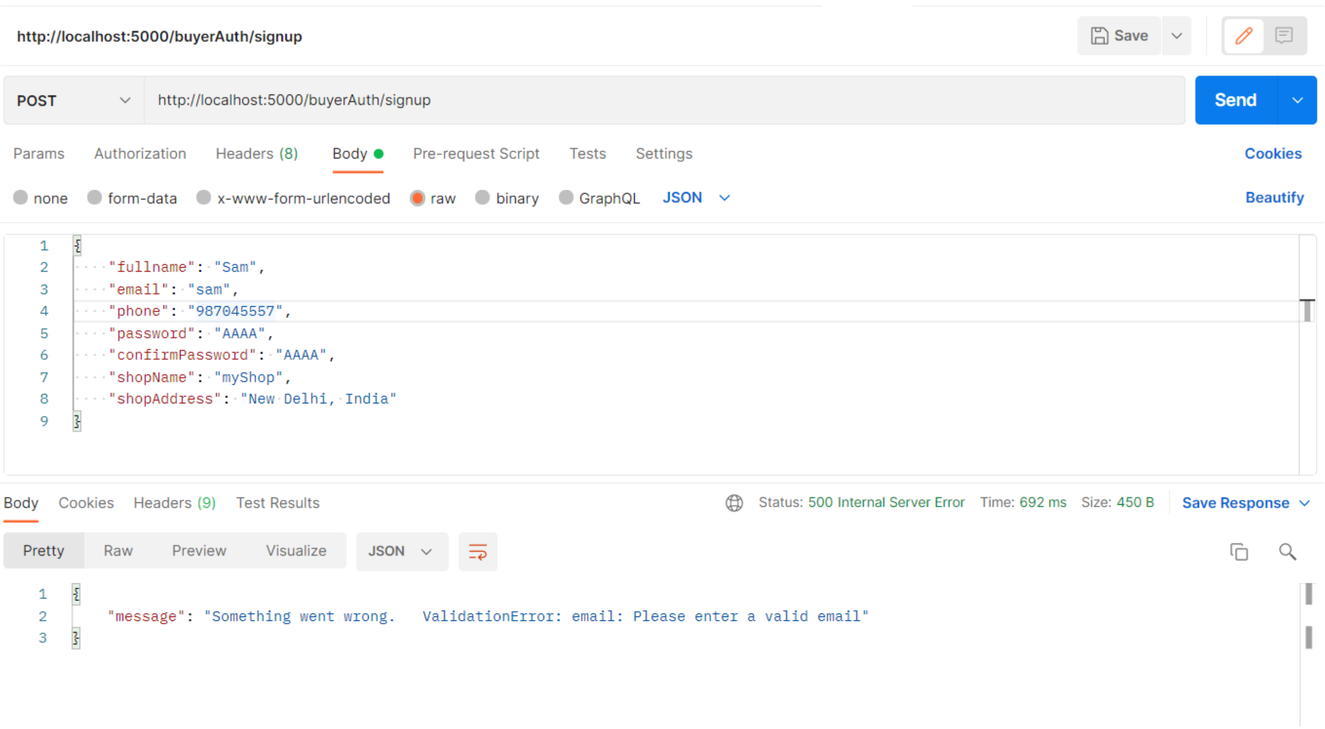
```
{
    "fullname": "Sam",
    "email": "sam",
    "password": "AAAA",
    "confirmPassword": "AAAA",
    "shopName": "myShop",
    "shopAddress": "New Delhi, India"
}
```

Status: 500 Internal Server Error Time: 884 ms Size: 490 B

```
{
    "message": "Something went wrong.  ValidationError: email: Please enter a valid email, phone: Please enter your Phone Number"
}
```

Not a valid email or phone

“Validation errors”



http://localhost:5000/buyerAuth/signup

POST http://localhost:5000/buyerAuth/signup

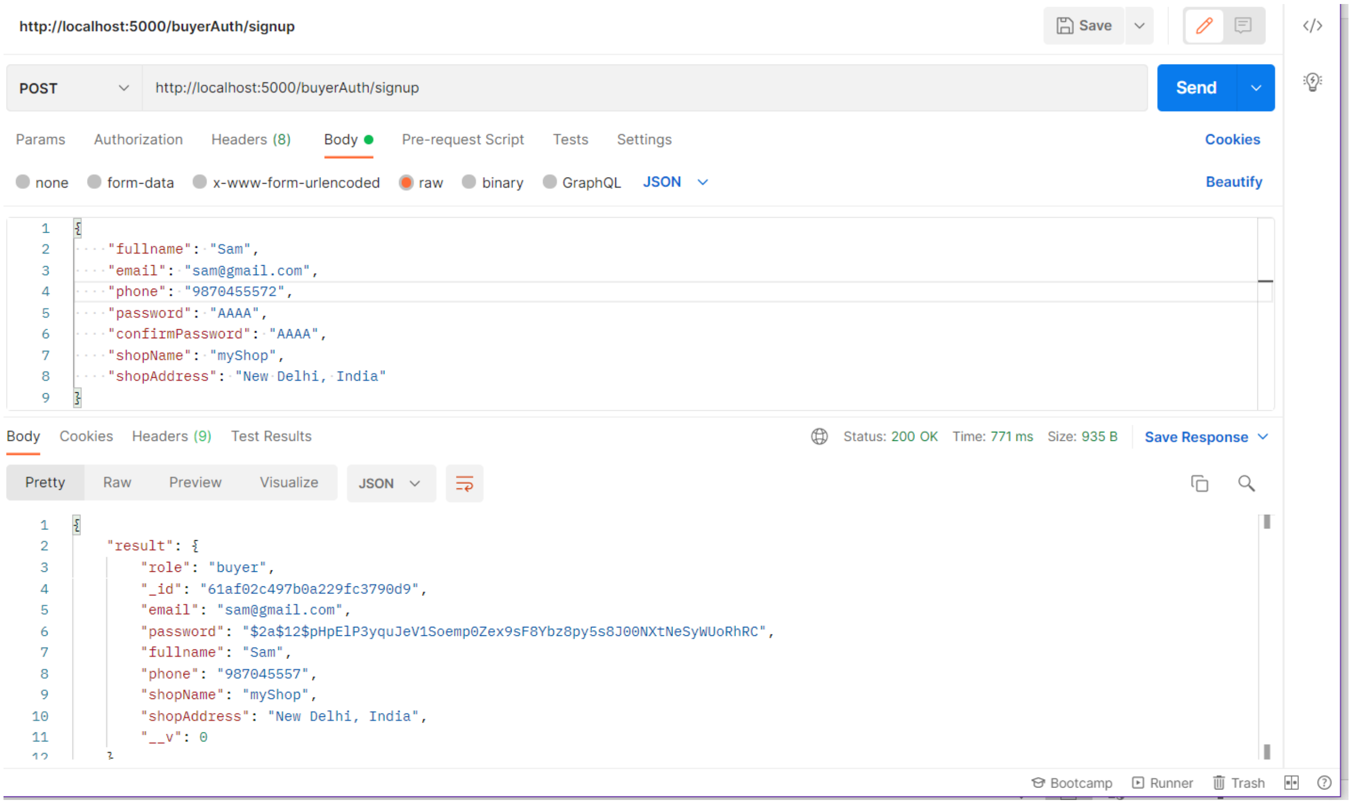
Send

```
{
    "fullname": "Sam",
    "email": "sam",
    "phone": "987045557",
    "password": "AAAA",
    "confirmPassword": "AAAA",
    "shopName": "myShop",
    "shopAddress": "New Delhi, India"
}
```

Status: 500 Internal Server Error Time: 692 ms Size: 450 B

```
{
    "message": "Something went wrong.   ValidationError: email: Please enter a valid email"
}
```

When everything is valid user get’s signed up



http://localhost:5000/buyerAuth/signup

POST http://localhost:5000/buyerAuth/signup

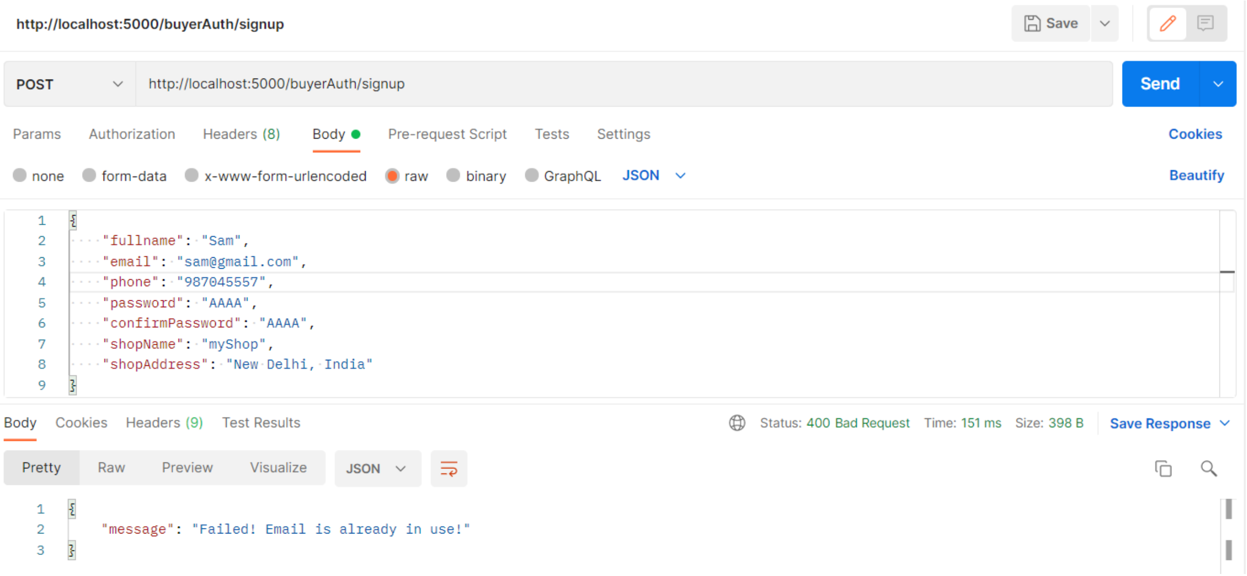
Send

```
{
    "fullname": "Sam",
    "email": "sam@gmail.com",
    "phone": "9870455572",
    "password": "AAAA",
    "confirmPassword": "AAAA",
    "shopName": "myShop",
    "shopAddress": "New Delhi, India"
}
```

Status: 200 OK Time: 771 ms Size: 935 B

```
{
    "result": {
        "role": "buyer",
        "_id": "61af02c497b0a229fc3790d9",
        "email": "sam@gmail.com",
        "password": "$2a$12$pHpElP3yquJeV1Soemp0Zex9sF8Ybz8py5s8J00NXtNeSyWUoRhRC",
        "fullname": "Sam",
        "phone": "987045557",
        "shopName": "myShop",
        "shopAddress": "New Delhi, India",
        "__v": 0
    }
```

If current email and phone number already in use!



http://localhost:5000/buyerAuth/signup

POST http://localhost:5000/buyerAuth/signup

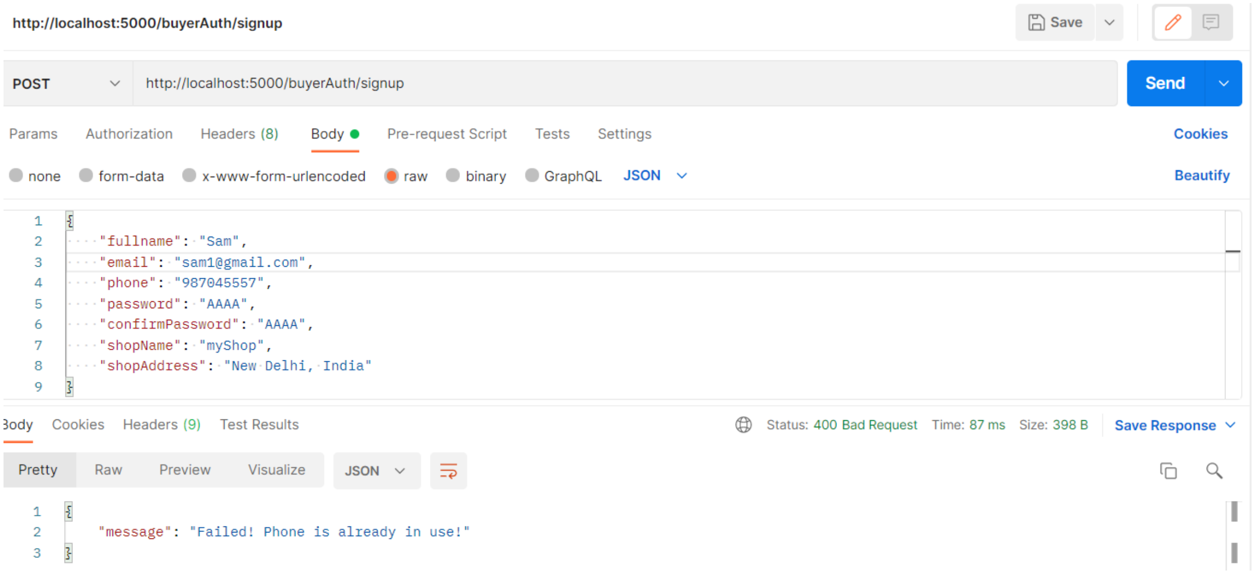
○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "fullname": "Sam",
    "email": "sam@gmail.com",
    "phone": "987045557",
    "password": "AAAA",
    "confirmPassword": "AAAA",
    "shopName": "myShop",
    "shopAddress": "New Delhi, India"
}
```

Status: 400 Bad Request Time: 151 ms Size: 398 B

JSON

```
{
    "message": "Failed! Email is already in use!"
}
```



http://localhost:5000/buyerAuth/signup

POST http://localhost:5000/buyerAuth/signup

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

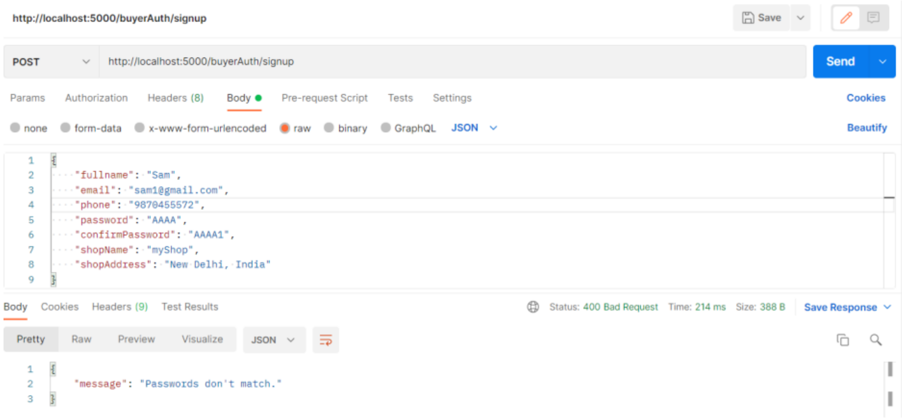
```
{
    "fullname": "Sam",
    "email": "sam1@gmail.com",
    "phone": "987045557",
    "password": "AAAA",
    "confirmPassword": "AAAA",
    "shopName": "myShop",
    "shopAddress": "New Delhi, India"
}
```

Status: 400 Bad Request Time: 87 ms Size: 398 B

JSON

```
{
    "message": "Failed! Phone is already in use!"
}
```

Password doesn't match error



http://localhost:5000/buyerAuth/signup

POST http://localhost:5000/buyerAuth/signup

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

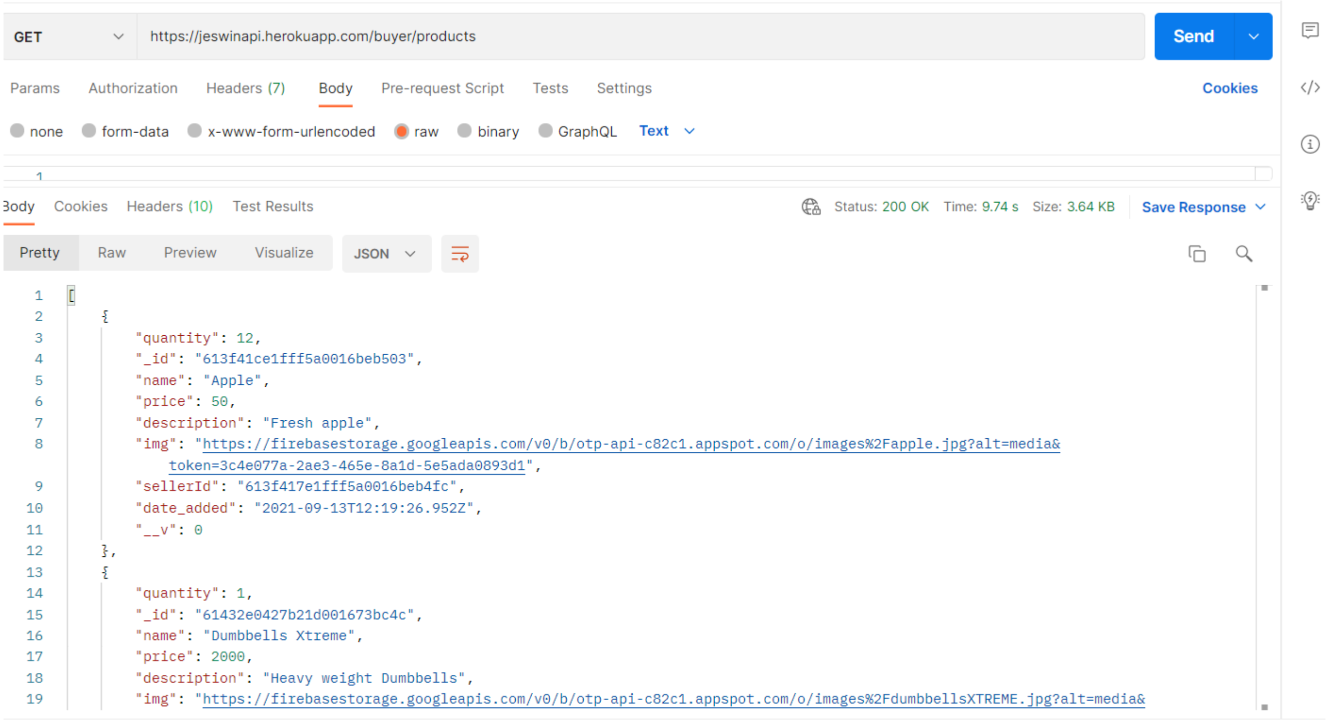
```
{
    "fullname": "Sam",
    "email": "sam1@gmail.com",
    "phone": "9870455572",
    "password": "AAAA",
    "confirmPassword": "AAAA1",
    "shopName": "myShop",
    "shopAddress": "New Delhi, India"
}
```

Status: 400 Bad Request Time: 214 ms Size: 388 B

JSON

```
{
    "message": "Passwords don't match."
}
```

Similarly for all APIs there is different error msgs in string and is displayed to cover edge cases



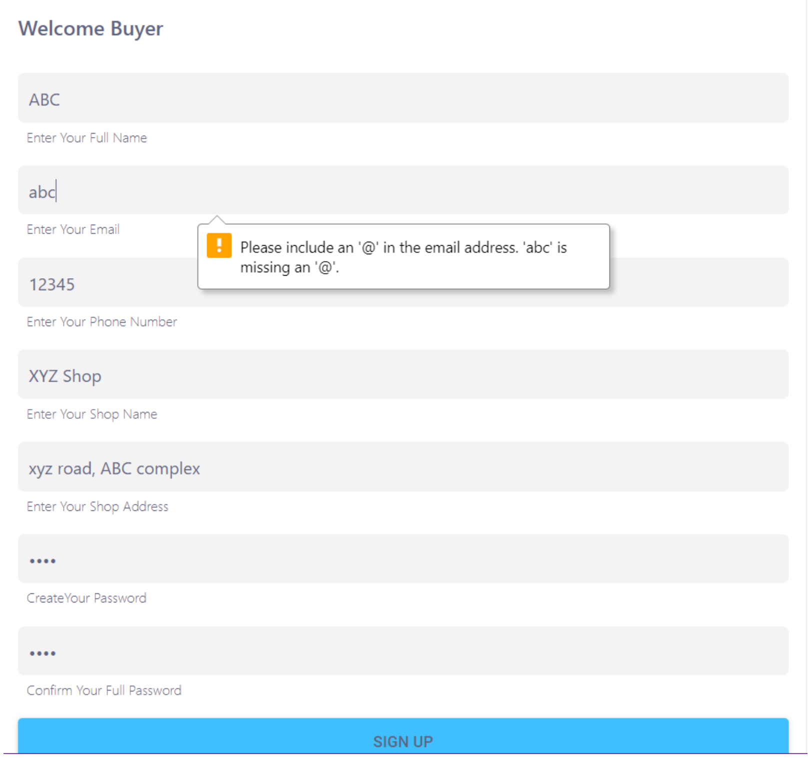
GET https://jeswinapi.herokuapp.com/buyer/products

Body: raw, Text

Status: 200 OK Time: 9.74 s Size: 3.64 KB

```
[
    {
        "quantity": 12,
        "_id": "613f41ce1fff5a0016beb503",
        "name": "Apple",
        "price": 50,
        "description": "Fresh apple",
        "img": "https://firebasestorage.googleapis.com/v0/b/otp-api-c82c1.appspot.com/o/images%2Fapple.jpg?alt=media&
            token=3c4e077a-2ae3-465e-8a1d-5e5ada0893d1",
        "sellerId": "613f417e1fff5a0016beb4fc",
        "date_added": "2021-09-13T12:19:26.952Z",
        "__v": 0
    },
    {
        "quantity": 1,
        "_id": "61432e0427b21d001673bc4c",
        "name": "Dumbbells Xtreme",
        "price": 2000,
        "description": "Heavy weight Dumbbells",
        "img": "https://firebasestorage.googleapis.com/v0/b/otp-api-c82c1.appspot.com/o/images%2FdumbbellsXTREME.jpg?alt=media&
```

Frontend validations: - Authentication module -> (same for buyer, seller and deliverer)



Welcome Buyer

ABC
Enter Your Full Name

abc
Enter Your Email

Please include an '@' in the email address. 'abc' is missing an '@'.

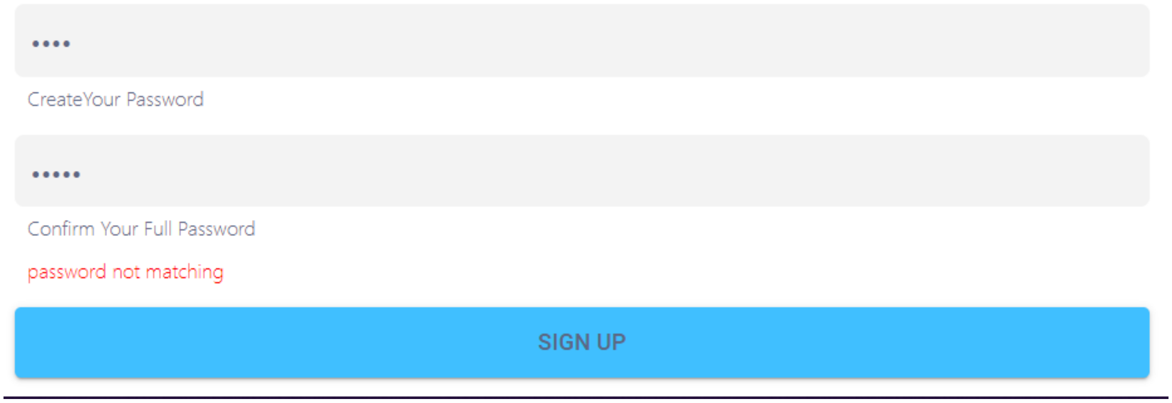
12345
Enter Your Phone Number

XYZ Shop
Enter Your Shop Name

xyz road, ABC complex
Enter Your Shop Address

••••
CreateYour Password

••••
Confirm Your Full Password



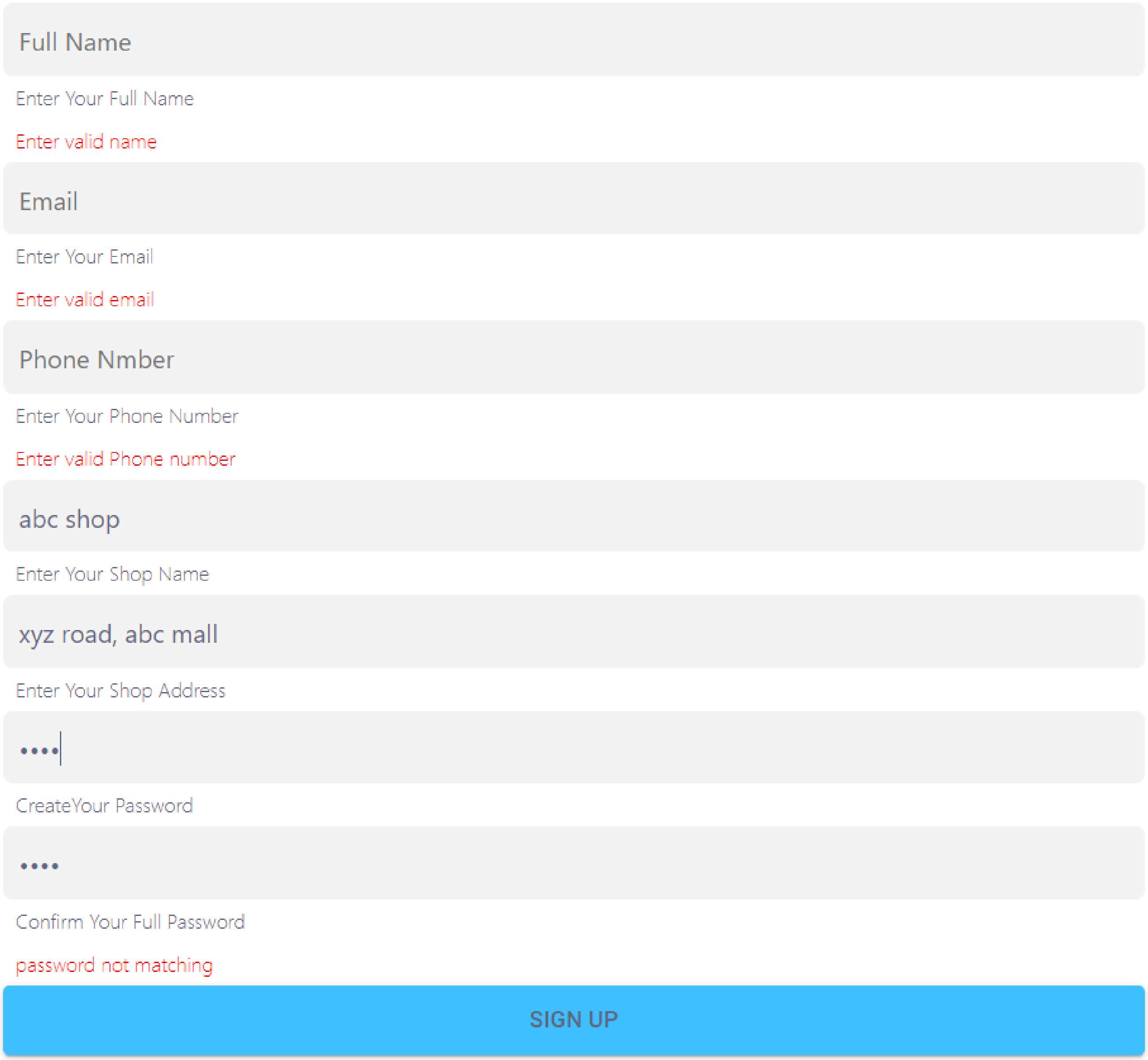
••••

CreateYour Password

•••••

Confirm Your Full Password

password not matching



Full Name

Enter Your Full Name

Enter valid name

Email

Enter Your Email

Enter valid email

Phone Nmber

Enter Your Phone Number

Enter valid Phone number

abc shop

Enter Your Shop Name

xyz road, abc mall

Enter Your Shop Address

••••

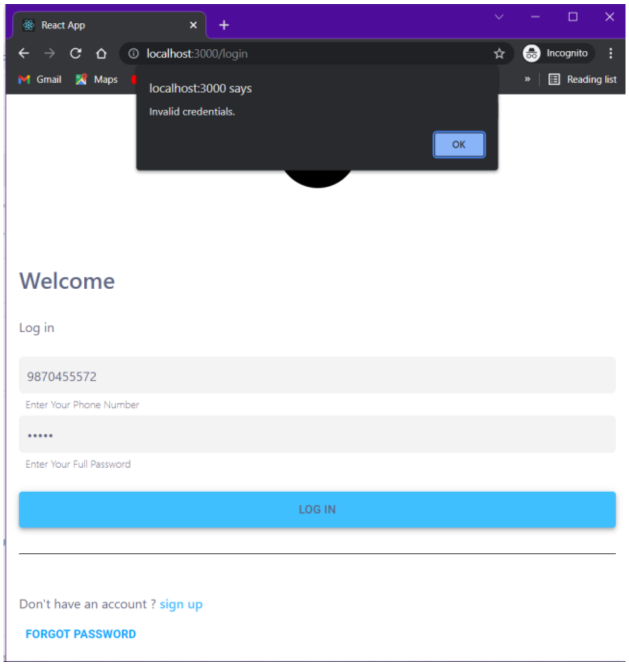
CreateYour Password

••••

Confirm Your Full Password

password not matching

Invalid Credentials



localhost:3000/login

localhost:3000 says

Invalid credentials.

Welcome

Log in

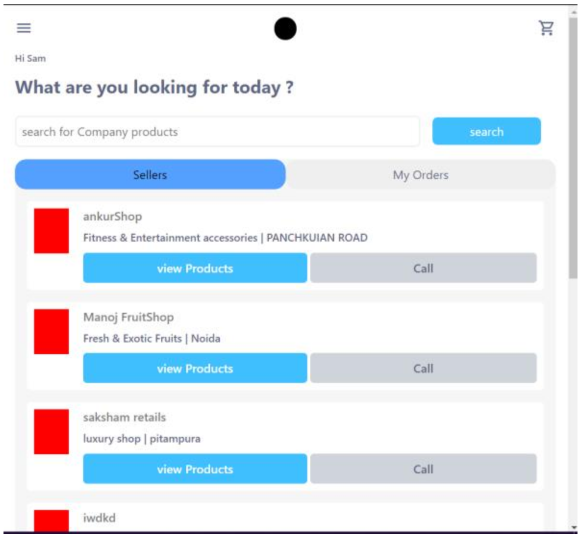
9870455572

Enter Your Phone Number

•••••

Enter Your Full Password

Don't have an account ? sign up



Hi Sam

What are you looking for today ?

search for Company products

ankurShop

Fitness & Entertainment accessories | PANCHKUIAN ROAD

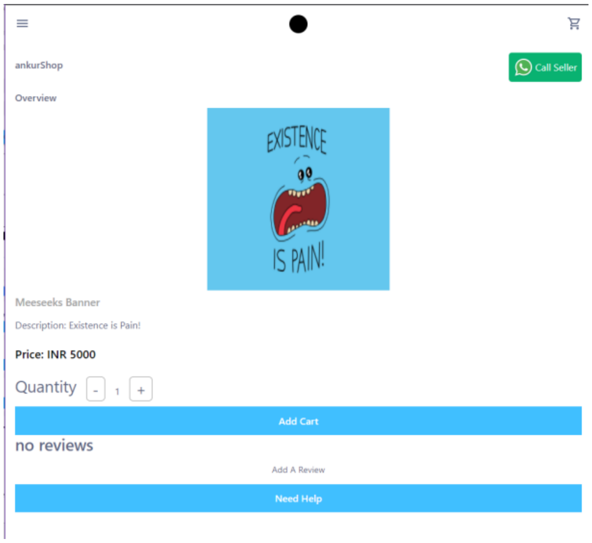
Manoj FruitShop

Fresh & Exotic Fruits | Noida

saksham retails

luxury shop | pitampura

iwdkd



ankurShop

Overview

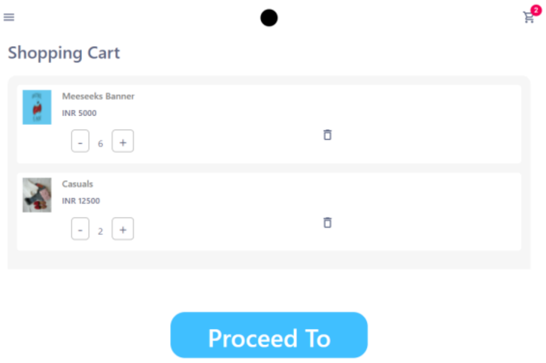
Meeseeks Banner

Description: Existence is Pain!

Price: INR 5000

Quantity - 1 +

no reviews



Shopping Cart

Meeseeks Banner

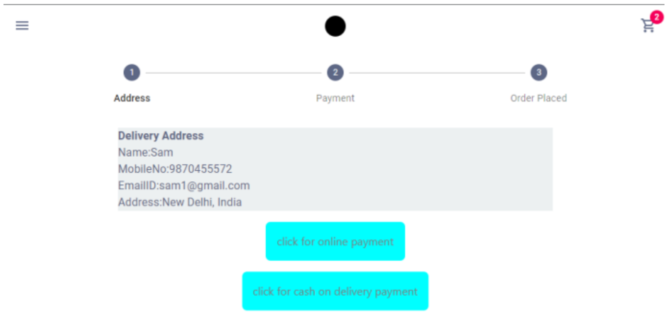
INR 5000

- 6 +

Casuals

INR 12500

- 2 +



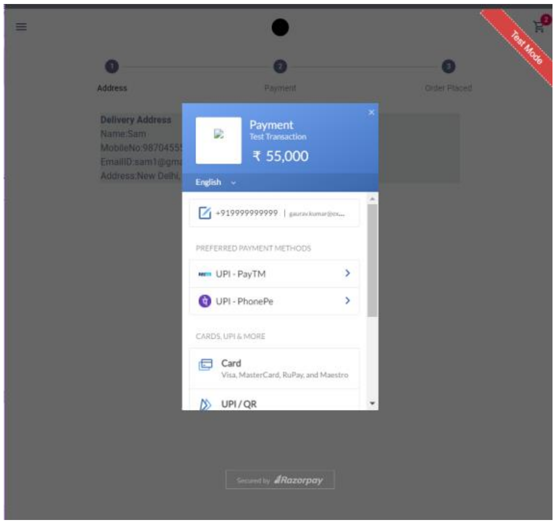
Delivery Address

Name:Sam

MobileNo:9870455572

EmailID:sam1@gmail.com

Address:New Delhi, India



Address
Payment
Order Placed

Delivery Address
Name:Sam
MobileNo:98704555
EmailID:sam1@gma
Address:New Delhi,

Payment
Test Transaction
₹ 55,000
English

+919999999999 | gaurav.kumar@ex...

PREFERRED PAYMENT METHODS

UPI - PayTM

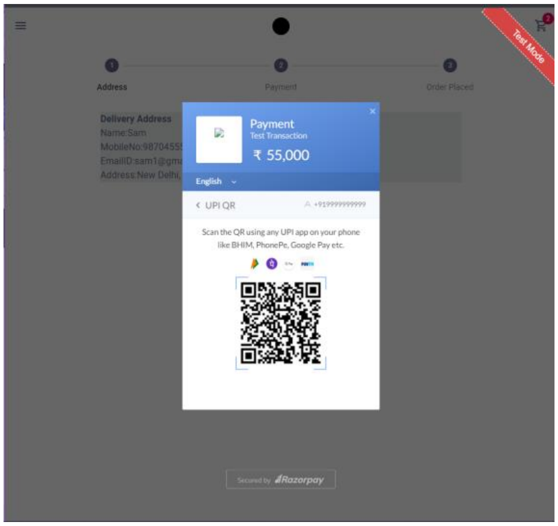
UPI - PhonePe

CARDS, UPI & MORE

Card
Visa, MasterCard, RuPay, and Maestro

UPI / QR

Secured by Razorpay



Address
Payment
Order Placed

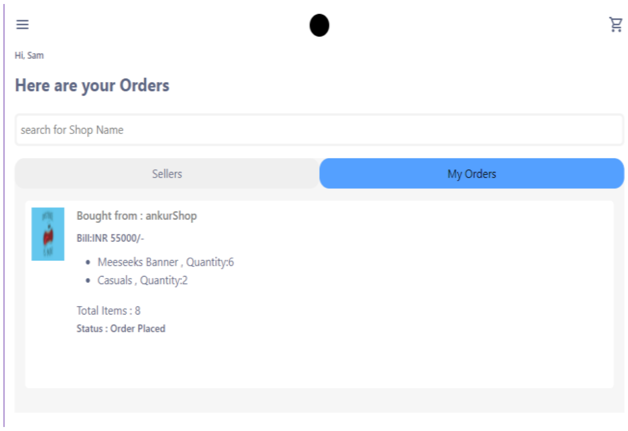
Delivery Address
Name:Sam
MobileNo:98704555
EmailID:sam1@gma
Address:New Delhi,

Payment
Test Transaction
₹ 55,000
English

UPI QR +919999999999

Scan the QR using any UPI app on your phone like BHIM, PhonePe, Google Pay etc.

Secured by Razorpay



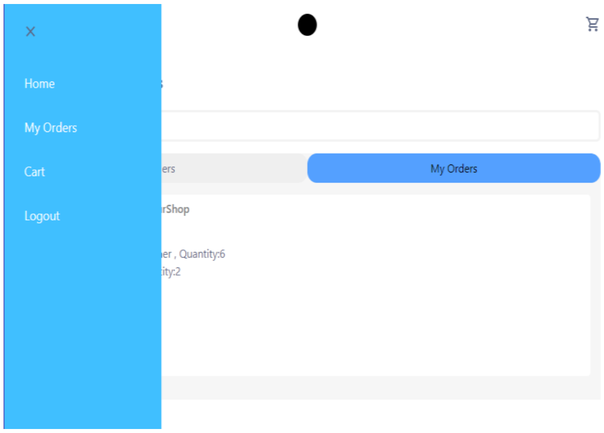
Hi, Sam

Here are your Orders

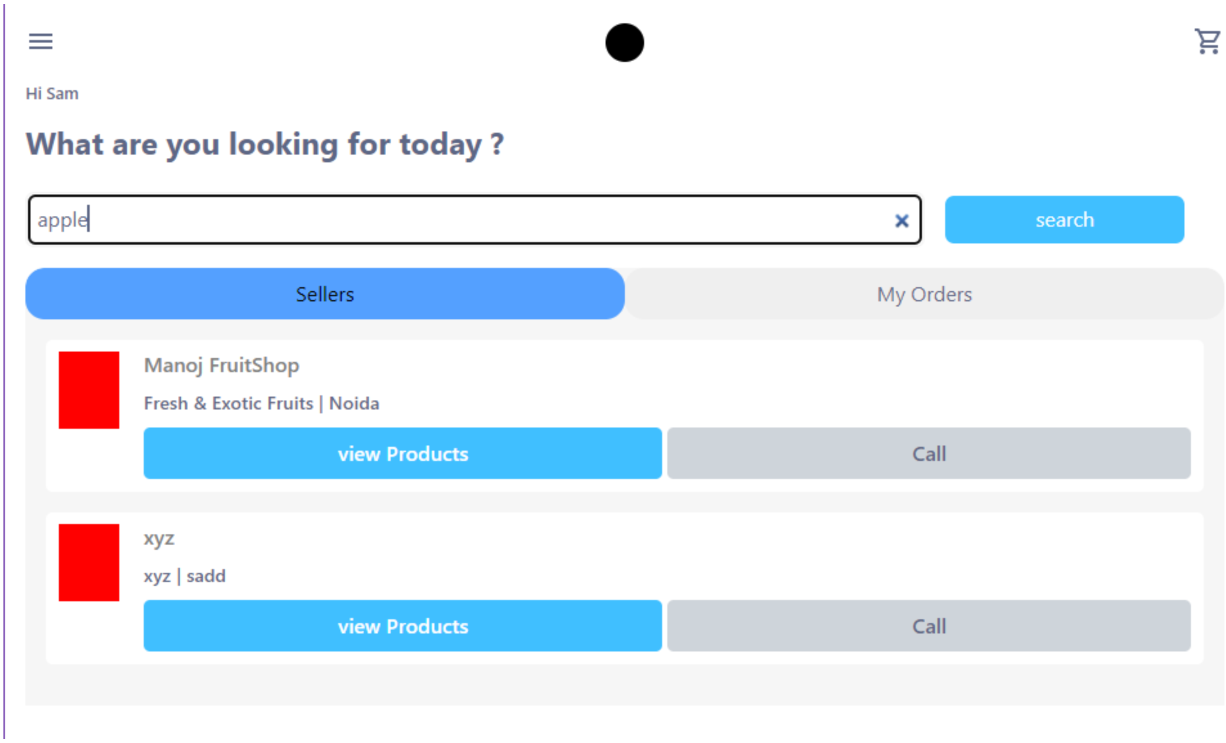
search for Shop Name

Bought from : ankurShop
Bill:INR 55000/-
- Meeseeks Banner , Quantity:6
- Casuals , Quantity:2

Total Items : 8
Status : Order Placed



rShop
er , Quantity:6
ity:2



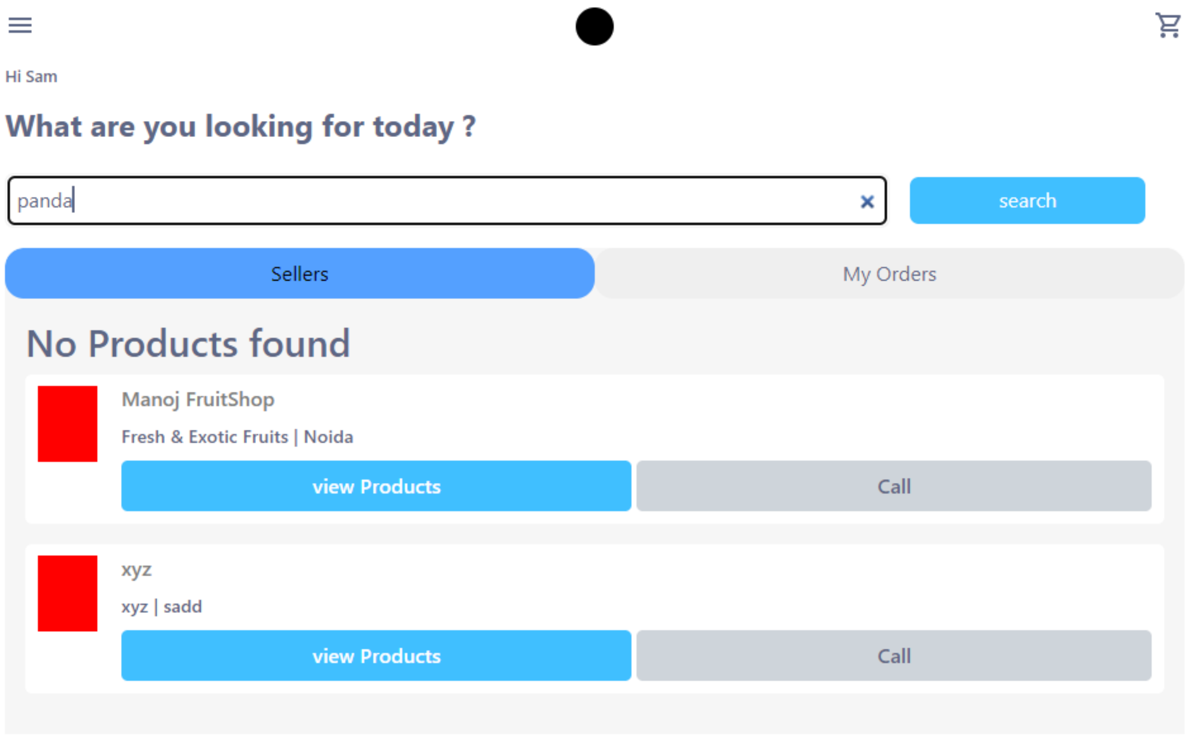
Hi Sam

What are you looking for today ?

apple

Manoj FruitShop
Fresh & Exotic Fruits | Noida

xyz
xyz | sadd



Hi Sam

What are you looking for today ?

panda

No Products found

Manoj FruitShop

Fresh & Exotic Fruits | Noida

xyz

xyz | sadd

Signup is valid if:

1. Email is valid.
2. Phone Is valid phone.
3. Password and confirm Password fields match.
4. Email used is not used by any other customer to signup.
5. Phone is not used by any other customer to signup.
6. Fields are not empty.

Signup is invalid if:

1. Email is invalid.
2. Phone is invalid.
3. Password and confirm password fields do not match.
4. Email already exists in the DB.
5. Phone already exists in the DB.
6. Any of the fields are empty.

Boundary Value Test Cases

TC	Full Name	Email	Phone	Shop Name	Shop Address	Password	Confirm Password	Role	Expected Output	Actual Output	Status
1	Abc	abc@test.com	99999999999	Abc shop	ABC Mall, Delhi	ABC@a#123	ABC@a#123	Seller	Seller Signedup	Seller Signed up	True
2	XYZ	xyz@gmail.com	1234567890	XYZ Shop	XYZ road, JJ colony	AAAA	AAAA	Seller	Seller Signedup	Seller Signed up	True
3	Reeha	reeha@google.com	0000000000	Home	Sector – 16, rohini	AAAA@1233	AAAAA@1233	Buyer	Buyer Signed up	Buyer Signed up	True
4	Saba	saba@ceo.in	1212121212	Home	House no – 122, …	saba@123	saba@123	Buyer	Buyer Signed up	Buyer Signed up	True
5	Queen	queen@elizabeth.com	1111111111	House	Villa no. 12, Delhi…	AAAA	AAAA	Deliverer	Deliverer Signedup	Deliverer Signed up	True
6	ABC sharma	Abc_s@yahoo.com	1000000000	ABC bakers	Ambience Mall, …	ABC@a#123	ABC@a#123	Seller	Seller Signedup	Seller Signed up	True
7	KBC	kbc@rediff.com	9870455572	Home	Golden colony, …	kbcccccccccc@	kbcccccccccc@	Buyer	Buyer Signed up	Buyer Signed up	True
8	Amitabh Bachan	Amitabh@cases.com	9958937319	Home	Appt – 19, delhi…	AAAA	AAAA	Buyer	Buyer Signed up	Buyer Signed up	True
9	My Name	khan@outlook.com	1231231231	House	House – 10, 3rd floor, Delhi….	123456789	123456789	Deliverer	Deliverer Signedup	Deliverer Signed up	True

*****Phone no. should be valid as OTP is sent for the verification purpose**

*****Phone no. should be valid as OTP is sent for the verification purpose**

As OTP is sent to verify the same

Robust Test Cases

TC	Full Name	Email	Phone	Shop Name	Shop Address	Password	Confirm Password	Expected Output	Actual Output	Status
1.	xyz	abc	9870455572	Abc_shop	X	AAAA	AAAA	Inavlid	Inavlid	True
2.								Invalid	Invalid	True
3.	xyz	abc@gmail.com	9870455572	a	X	AAAA	AA	Invalid	Invalid	True
4.	xyz	abc@gmail.com	9870455	abc	X	AAAA	AA	Invalid	Invalid	True
5.	Abc	abc@gmail.com	99999999999	Abc shop	ABC Mall, Delhi	ABC@a#123	ABC@a#123	Valid	Valid	True
6.	Xyz	abc@gmail.com	1111111111	Xyz shop	Xyz road	AA@123	AA@123	Invalid	Invalid	True
7.	Abc	Abc1@gmail.com	99999999999	Abc shop	ABC Mall, Delhi	ABC@a#123	ABC@a#123	Invalid	Invalid	True
8.	Abc	Abc1@gmail.com	98704555572	Abc shop	ABC Mall, Delhi	ABC@a#123	ABC@a#123	Valid	Valid	True
9.	XYZ	abc@	00000000000	xyz	xyz	AAAA	AAAA	Invalid	Invalid	True
10.	XYZ	abc@test.com	00000000000	xyz	xyz	AAAA	AAAA	Valid	Valid	True
11.	abc	Abc@test.com	XXXX	123	123	saba	saba	Invalid	Invalid	True
12.	abc	Abc@test.com	9313305723	My shop	Karol bagh, shop no. 12	abc@123@abc$	abc@123@abc$	Valid	Valid	True
13.	reeha	reeha@google.com	9899234567	Pacific	Rohini, delhi	123abc@omg	123abc@omg	Valid	Valid	True

Conclusion:

Test Cases for given code was made successfully.

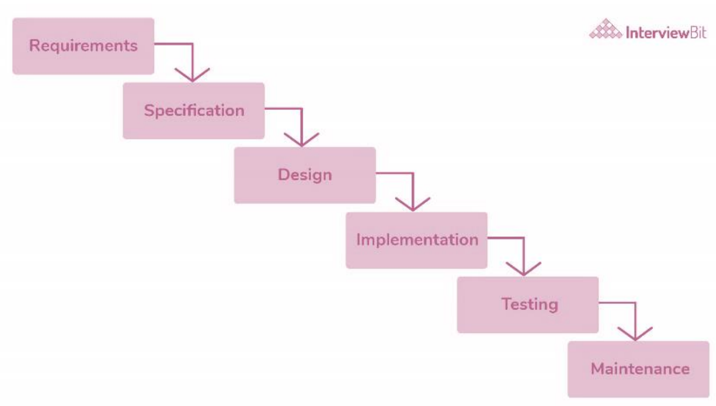
Viva Questions

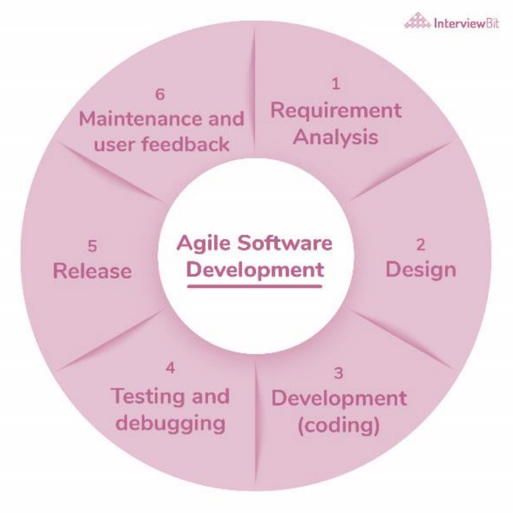
1. Explain the role of testing in software development?

Ans.

Software testing comes into play at different times in different software development methodologies. There are two main methodologies in software development, namely Waterfall and Agile.

In a traditional waterfall software development model, requirements are gathered first. Then a specification document is created based on the document, which drives the design and development of the software. Finally, the testers conduct the testing at the end of the software development life cycle once the complete software system is built.





Waterfall Software Development Model

An agile software development model works in small iterations. You test the software in parallel as it is getting built. The developers build a small functionality according to the requirements. The testers test it and get customer feedback, which drives future development.

2. How much testing is sufficient? Or, is it possible to do exhaustive testing of the software?

Ans.

It is impossible to exhaustively test software or prove the absence of errors, no matter how specific your test strategy is.

An extensive test that finds hundreds of errors doesn't imply that it has discovered them all. There could be many more errors that the test might have missed. The absence of errors doesn't mean there are no errors, and the software is perfect. It could easily mean ineffective or incomplete tests. To prove that a program works, you'd have to test all possible inputs and their combinations.

Consider a simple program that takes a string as an input that is ten characters long. To test it with each possible input, you'd have to enter 2610 names, which is impossible. Since exhaustive testing is not practical, your best strategy as a tester is to pick the test cases that are most likely to find errors. Testing is sufficient when you have enough confidence to release the software and assume it will work as expected.

3. Why developers shouldn't test the software they wrote?

Ans.

Developers make poor testers. Here are some reasons why:

- They try to test the code to make sure that it works, rather than testing all the ways in which it doesn't work.
- Since they wrote it themselves, developers tend to be very optimistic about the software and don't have the correct attitude needed for testing: to break software.
- Developers skip the more sophisticated tests that an experienced tester would perform to break the software. They follow the happy path to execute the code from start to finish with proper inputs, often not enough to get the confidence to ship software in production.

However, it doesn't mean that developers shouldn't test the software before sending it to the tester. Developer testing helps find many bugs that are caused by programming errors. These are hard to find for a tester because they don't always have access to the source code.

4. What is functional testing?

Ans.

Functional testing is a form of black-box testing. As the name suggests, it focuses on the software's functional requirements rather than its internal implementation. A functional requirement refers to required behavior in the system, in terms of its input and output.

It validates the software against the functional requirements or the specification, ignoring the non-functional attributes such as performance, usability, and reliability.

Functional testing aims to answer the following questions, in particular:

- Does the software fulfill its functional requirements?
- Does it solve its intended users' problems?

5. What is non-functional testing?

Ans.

Non-functional testing tests the system's non-functional requirements, which refer to an attribute or quality of the system explicitly requested by the client. These include performance, security, scalability, and usability.

Non-functional testing comes after functional testing. It tests the general characteristics unrelated to the functional requirements of the software. Non-functional testing ensures that the software is secure, scalable, high-performance, and won't crash under heavy load.