

# Testing, Automation Scripts and Readme file

## 1. Testing

### Black box testing

Testing can be divided into two types broadly: functional testing and structural testing. Structural testing, also known as white-box testing, involves examining the internal implementation. It tests the design used by the implementation to verify its correctness. In contrast, Functional testing, sometimes referred to black box testing, is testing on the functionality of the system based on the specified requirement. The test itself has little knowledge about the testing target's internal structure. The following table shows the test cases for black box testing.

Test Code	Test Case	Test Steps	Expected Result	Actual Result	Pass/Fail
Test 1	Checking the book a table functionality	1.Go to the main website.  2.Either scroll down till book table section or Click book a table option in the navbar.	1.Fill the details and press submit.  2. If successful then you would get an alert and the restaurant manager will get an email of the message.	As expected	Pass
Test 2	Checking	1.Go to the main	1.Fill the details	As	Pass

	the contact us functionality.	website. 2.Either scroll down till contact us section or Click contact option in the navbar.	and press submit.  2. If successful then you would get an alert and the restaurant manager will get an email of the message.	expected	
Test 3	Checking the cart functionality.	1.Go to the main website.  2.Click the cart button in the navbar.	1. Press add to cart button for the item you want to add to cart.  2.If successful your items quantity and total price would increase.	As expected	Pass
Test 4	Entering incorrect format in forms.	1. Go to the website and enter a invalid email address in any of the forms.	You should get the message stating wrong email address.	As expected	Pass
Test 5	Adding the same item in cart more than once.	1.Go to the main website.  2.Click the cart button in the navbar.	1. Press add to cart option for a product more than once.  2. If successful you'd have the same product in the cart the times you've pressed add to	As expected	Pass

			cart and amount would reflect accordingly.		
Test 6	Checking out.	1. Go to the main website and then press the cart button and then add the products you would like to add.	1.Fill the checkout form. 2.To submit press the button checkout. 3.If successful you'd get a prompt stating order received, details will be sent through email and the restaurant manager will get an email of the order details.	As expected	Pass

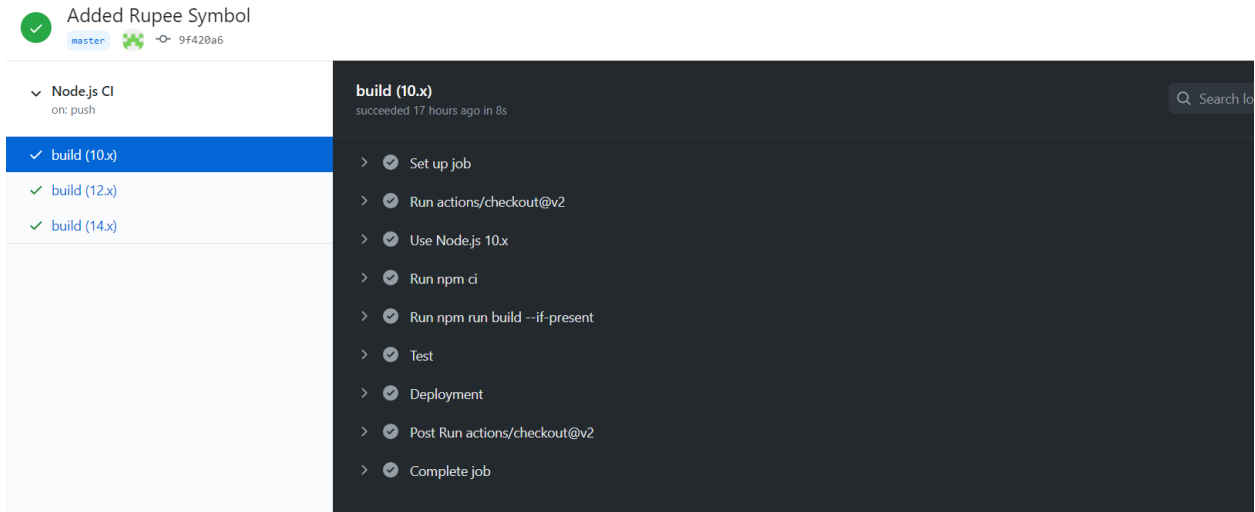
## 2. Automation Scripts:

CI/CD pipeline was made using Github Actions. The website was deployed on heroku. Whenever there's a push in the project repository first our CI/CD pipeline runs and it gets automatically deployed on heroku only if it passes through the CI/CD pipeline. The github CI/CD scripts are:

```
4 name: Node.js CI
5
6 on:
7   push:
8     branches: [ master ]
9   pull_request:
10    branches: [ master ]
11
12 jobs:
13   build:
14
15     runs-on: ubuntu-latest
16
17     strategy:
18       matrix:
19         node-version: [10.x, 12.x, 14.x]
20         # See supported Node.js release schedule at https://nodejs.org/en/about/releases/
21
22     steps:
23     - uses: actions/checkout@v2
24     - name: Use Node.js ${ matrix.node-version }
25       uses: actions/setup-node@v2
26       with:
27         node-version: ${ matrix.node-version }
28     - run: npm ci
29     - run: npm run build --if-present
30
31     - name: Test
32       run: echo Tested the app
33
34     - name: Deployment
35       run: echo The app will be deployed after testing automatically on heruko
```

This CI/CD pipeline basically means that when there is a push on the master branch the build job is to be run on a ubuntu machine environment. This job is to be run for three different node versions 10.x, 12.x, 14.x. The job has the following steps, the first step is checkout which means basically requiring the code for the next steps, then setting up the node version, building the app using the command npm run and then the app would already have been tested by the person beforehand, so we just echo it. And then as we have enabled the auto deploy option on github, after all the jobs have been done and code has been tested, our website will be deployed.

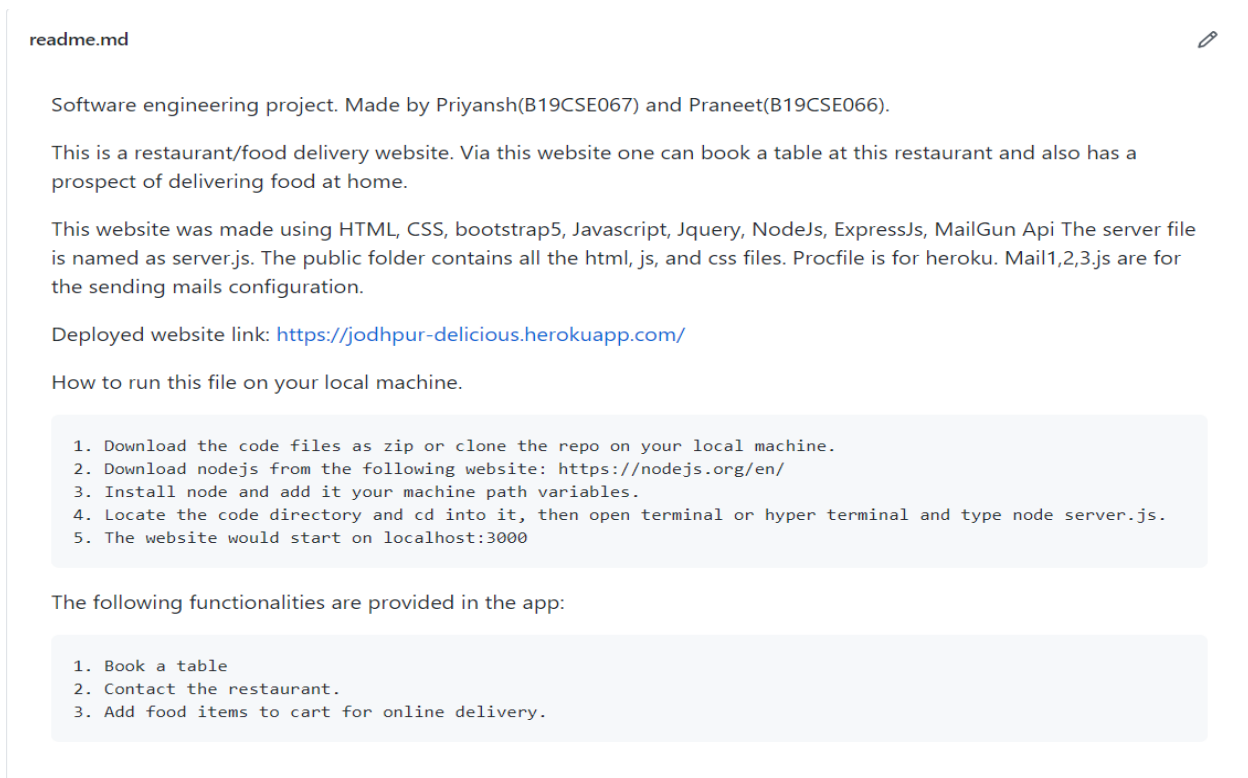
For example, when I pushed some changes into the repository the CI/CD pipeline was triggered as follows:



### 3.Readme file

Readme file was made and added to the github repository. The link for which is [https://github.com/priyansh2011/SE\\_project\\_food](https://github.com/priyansh2011/SE_project_food).

A snapshot of the the read me is:



We would also attach the readme.md file in the form.