

YAKSHA HEALTH APP WITH PYTHON AND PYTEST

SN o.	Summary	Action	Expected Result
1	Handle Alert on Pharmacy Module	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Pharmacy/Order/GoodsReceiptList 2. Add Good Receipt Modal opens. 3. Click on "Print Receipt" button.	Handle 2 alerts : 1) Please select supplier 2) Please enter Invoice no.
2	Verify to get the validation message when Click on "print receipt" without filling any details	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Pharmacy/Order/GoodsReceiptList 2. Add Good Receipt Modal opens 3. Click on add new items button. 4. Fill all the mandatory details (Item Name : Test, BatchNo: any random 5 digit value, Item Qty : 2, Rate: 24) 5. Now click on save button. 6. Fill Supplier name : Ashar & Company, Invoice no : any random 3 digit number 7. Click on "Print Receipt" button.	Verify success message popup - "Goods Receipt is Generated and Saved."
3	Verify that entering a keyword matching existing records in the search bar returns the corresponding data.	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Medical-records/InpatientList 2. Click on MR Outpatient List Tab. 3. Enter date in from field (01-01-2024) and click on "OK" button. 4. Enter the keywords "Female" in search bar.	Data should be present for the search, and the 'Gender' column should contain only patients in the "Female" category.

4	Add and Verify a New Dynamic Template in Settings Module	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Settings 2. Click on Dynamic Templates Tab. 3. Click on Add Template Button. 4. Add Template Modal opens and fill all the mandatory details. (Template Type : Discharge Summary , Template Name : Test Template, Template Code : Temp_9898) 5. Enter "test print" in print format text field. 6. Click on "Add" button.	A success confirmation popup with the message: "Template Saved." should appear, indicating that the template has been added successfully.
5	Create and Verify Inventory Requisition in Substore Module	1. Navigate to https://healthapp.yaksha.com/Home/Index#/WardSupply/Inventory/Stock 2. Click on Inventory Requisition Tab. 3. Click on Create Requisition Button. 4. Fill requisition details: . Select target inventory as General-Inventory from the dropdown . Enter item name as "soap" . Enter required quantity as "1". 5. Click on "Request" button.	A success confirmation popup with the message: "Requisition is Generated and Saved" should appear.
6	verify Maternity Allowance Report is visible	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Maternity 2. Click on Reports 3. Click on Maternity Allowance. 4. Enter date in from field (01-01-2024) and click on "show report" button.	Report should be visible when click on show more button.
7	verify Imaging	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Doctors	A success confirmation popup with the message:

	and lab order add successfully	2. Click on InPatientDepartment 3. Click on the search bar and search patient "Devid8 Roy8" 4. Click on Imaging under the action column. 5. Select imaging for order dropdown. 6. Select "USG Chest" in the search order item. 7. Click on proceed button. 8. Click on sign Button.	"Imaging and lab order add successfully" should appear.
8	Verify to data range by select "Last 3 months" option from drop down	1. Navigate to https://healthapp.yaksha.com/Home/Index#/Radiology/ImagingRequisitionList 2. Click on the date range button. 3. select "Last 3 Months" option from the drop down. 4. Click on "OK" button	Data should be present as per the selected date range using dropdown The 'Requested on' column date must fall within the "Last 3 months".

MAKE SURE TO HAVE THE JSON FILE AVAILABLE IN THE VM DESKTOP WITHIN THE CREATED PROJECT FOLDER

EXPECTATIONS:

Learners will gain experience in building strongly-typed applications using Python, Selenium, and managing their dataflow. They'll learn how to define interfaces, use types for error prevention, and improve code maintainability. app. Key skills include:

- **Browser Automation:** Interacting with web elements and testing multiple browsers.
- **Assertions & Validations:** Ensuring app behavior meets expected results.

- **End-to-End Testing:** Automating real user interactions and validating overall app functionality.

STEPS TO FOLLOW

- After you login from in to the virtual machine, you need to open the created folder and open the command prompt in that directory
- Using the command code . open VS code
- Write the code in the template as the requirements given above and run the code
- Open the Terminal
- To run the code, python filename.py
- Execute the following command to run all test cases:
- pytest

2. Run all test cases:

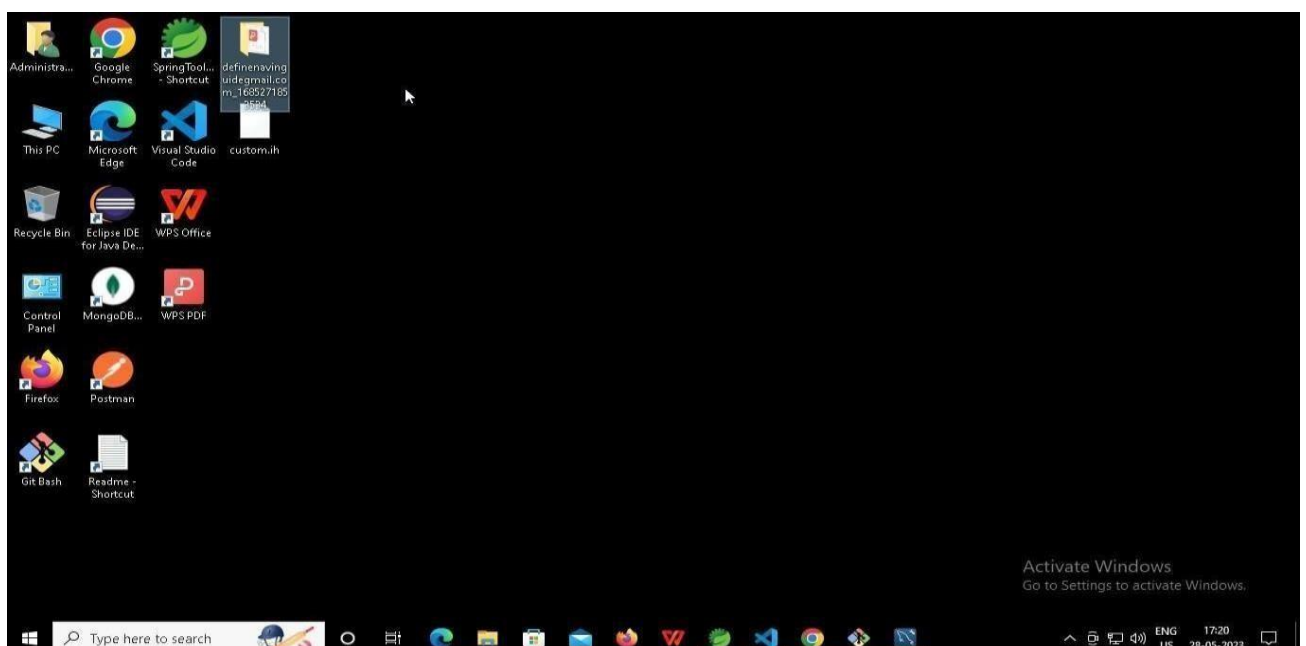
Using the command `pytest`

Screenshot to push the application to github

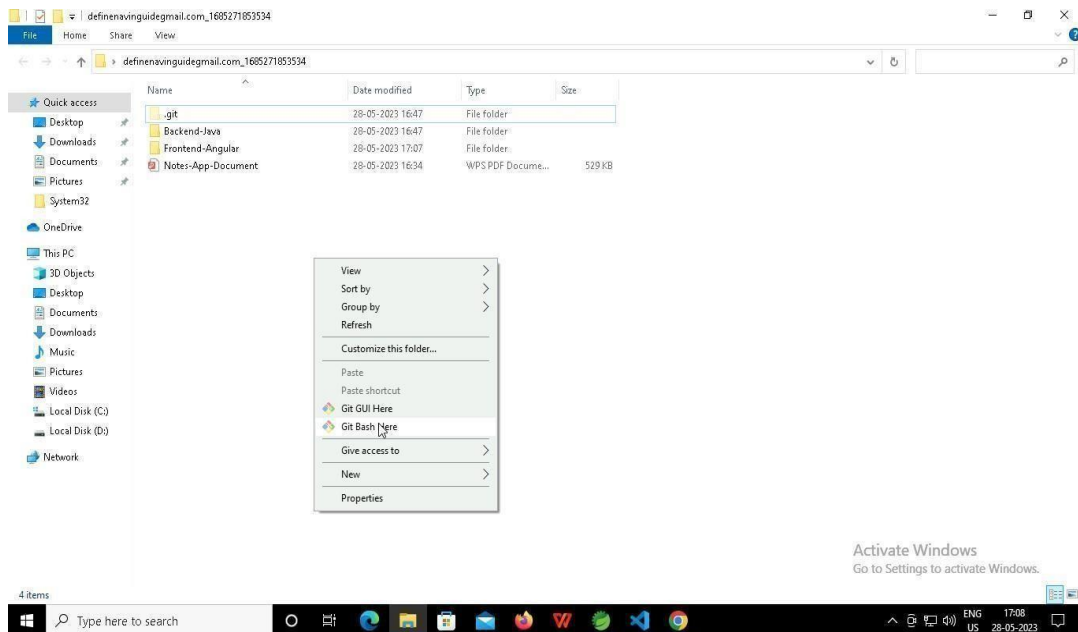
X -----

You can run test cases as many numbers of times and at any stage of Development, to check howmany test cases arepassed/failed and accordingly refactor your code.

1. Make sure before final submission you commit all changes to git. For that open theproject folder available on desktop



- Right click in folder and open Git Bash



b. In Git bash terminal, run following commands

c. git status

```
Administrator@2a5ee7ad258f58c MINGW64 ~/Desktop/tesgmailcom_1728449791471 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    templatespark.py

no changes added to commit (use "git add" and/or "git commit -a")
Administrator@2a5ee7ad258f58c MINGW64 ~/Desktop/tesgmailcom_1728449791471 (main)
$
```

d. git add .

```
Administrator@2a5ee7ad258f58c MINGW64 ~/Desktop/tesgmailcom_1728449791471 (main)
$ git add .
```

e. git commit -m "First commit"

(You can provide any message every time you commit)

```
Administrator@2a5ee7ad258f58c MINGW64 ~/Desktop/tesgmailcom_1728449791471 (main)
$ git commit -m "first commit"
[main f97ce24] first commit
1 file changed, 91 deletions(-)
delete mode 100644 templateespark.py
```

f. git push

```
Administrator@2a5ee7ad258f58c MINGW64 ~/Desktop/tesgmailcom_1728449791471 (main)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 212 bytes | 212.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/IIHTDevelopers/tesgmailcom_1728449791471.git
alc1905..f97ce24 main -> main
```

After you have pushed your code Finally click on the final submission button
