

Part 1:

You were given the following user story and design to be used to create test cases, Carefully, Read the user story and Review the design, and please answer the Questions coming later.

User Story

Submitting Annual Leave Request:

As a user, I should be able to:

1. See my available balance.
2. Choose my leave start date.
3. Choose my leave end date, or;
4. Alternatively, decide the duration of the leave.
5. Choose a substitute colleague within the same department.
6. Choose the reason of my leave.

Design

10:31

Annual Leave Request

| | |
|-------------------|--------|
| Available Balance | X Days |
| Remaining Balance | X Days |

From Date

To Date

| | |
|--------------|--------|
| Leaving Days | X Days |
|--------------|--------|

Substitute Employee

Reason

Submit Leave Request

Questions:

1. Are all the requirements clearly presented? If not, please specify which point is ambiguous and how?
2. Are all the requirements were satisfied in the design? If not, please specify which one is not matched with the design, and suggest how it should be implemented/modified?
3. How many test cases you would need to write in order to cover all the requirements? Please list every possible test case.
4. Regards *Reason*, if it comes to your knowledge that in some cases there should be special calculation for the *Leaving Days* with Specific Reasons, could you think which cases it would happen?

Please provide the answers in a word/PDF file

Part 2:

Egyptian aviation company just launched a new mobile app, and for this occasion, they add an offer on the tickets submitted through this app. It would be a **10%** offer that applies only in case the following conditions are met:

1. The user is buying the ticket from Cairo to any international country.
2. The user is submitting his booking request and paid the amount during the period from **10 Dec 2022** to **25 Dec 2022**.
3. The user wants to travel within the period from **31 Dec 2022** to **31 Jan 2023**.

Questions:

1. Apply **Boundary Value Analysis** to this example and provide the cases.
2. Apply **Equivalence Partitioning** to this example and provide the cases.
3. Make a **Decision Table** that covers every possible case.

Please provide the answers in a word/PDF file

Part 3:

Find the Attached **Defects Classification.docx** file read it, please try to use your own logic/common sense to fill the table then reattach the file.

Part 4:

Visit the following URL:

[OrangeHRM \(orangehrmlive.com\)](https://orangehrmlive.com)

Build an automation framework to perform the following:

1. Login with the provided user

Username : Admin

Password : admin123

2. Scroll down to get this view



3. Navigate to Admin in the side panel

4. Click on Add

5. Add a User

6. Delet this User

Please upload your code to GitHub with public access and provide the URL

Part 5:

Visit the following URL and Read the API documentations:

[Restful-booker](#)

Create a Postman collection to test these APIs.

Build an automation framework to perform the following:

1. Use **Auth - CreateToken** API to get a token, use it in the other APIs.
2. Use **Booking - CreateBooking** API to create a booking
Assert that you got a "bookingid" and save it.
3. Use **Booking - UpdateBooking** API to update the entire object.
Assert that object is updated to the new values
4. Use **Booking - PartialUpdateBooking** API to update the name only.
Assert that object is updated to the new values
5. Use **Booking - DeleteBooking** API to delete this object
Assert that you got 201 status code

Please upload your code to GitHub with public access and provide the
URL