QA Assignment

Test Plan for CRM Module

Objective

To validate the functionality of the Manage Contacts and Manage Invoices features to ensure they meet the defined acceptance criteria and operate as expected in various scenarios.

Test Scope

- Functional testing of all CRUD operations for contacts.
- Functional testing of invoice creation, updates, payments, and returns.
- Validation of integration between contacts and invoices (e.g., balance adjustments).
- Ensuring proper handling of edge cases, errors, and invalid inputs.

Test Approach

- Manual Testing: For user interface and workflow validations.
- Automated Testing: For API endpoints and repetitive functional checks.
- Unit Testing: For individual components, e.g., balance calculations.
- Integration Testing: For interactions between contacts and invoices modules.

Test Cases

Manage Contacts

- Test Case ID: TC_01
- View all contacts
- Description: Verify that the user can view all contacts with optional pagination and search filters.

Test Steps:

1. Access the “View All Contacts” API or interface.
2. Provide valid pagination parameters (e.g., page=1, size=10).
3. Provide valid search parameters (e.g., name=Omar).

Expected Results:

List of contacts is displayed.
Pagination and search filters work as expected.

- Test Case ID: TC_02
- Retrieve a Contact by ID
- Description: Verify that the user can retrieve a specific contact by its unique ID.

Test Steps:

1. Use a valid contact ID.
2. Use an invalid contact ID.

Expected Results:

- Valid ID: Contact details are returned.
- Invalid ID: Appropriate error message is shown (e.g., 404 Not Found).

- Test Case ID: TC_03
- Create a New Contact
- Description: Verify that a new contact can be created with valid data.

Test Steps:

1. Submit a valid payload (e.g., name, email).
2. Submit an invalid payload (e.g., missing required fields).

Expected Results:

- Valid payload: Contact is created, and confirmation is returned.
- Invalid payload: Error message is displayed.

- Test Case ID: TC_04
- Update Contact Information
- Description: Verify that an existing contact can be updated.

Test Steps:

1. Update with valid data.
2. Update with invalid data.

Expected Results:

- Valid data: Contact is updated successfully.
- Invalid data: Error message is displayed.

- Test Case ID: TC_05
- Delete a Contact
- Description: Verify that a contact can be deleted.

Test Steps:

1. Use a valid contact ID.
2. Use an invalid contact ID.

Expected Results:

- Valid ID: Contact is deleted.
- Invalid ID: Error message is displayed.

- Adjust Contact Balance
- Test Case ID: TC_06
- Description: Verify that the balance of a contact can be adjusted

Test Steps:

1. Add a positive amount.
2. Add a negative amount.

Expected Results:

• Contact balance is updated correctly.

Manage Invoices

- Test Case ID: TC_07
- Create an Invoice
- Description: Verify that a new invoice is created and associated with a contact

Test Steps:

1. Submit valid invoice details and a valid contact ID.
2. Submit invalid details or invalid contact ID.

Expected Results:

- Valid details: Invoice is created, and contact balance is updated.
- Invalid details: Error message is displayed.

- Test Case ID: TC_08
- View List of Invoices
- Description: Verify that the user can view all invoices associated with their account.

Test Steps:

1. Access the GET "View all Invoices" API or interface.
2. Click execute

Pre-conditions:

- User should have created his invoices using POST command before he can view the list of his invoices

Expected Results:

- List of invoices is displayed with pagination.

- Test Case ID: TC_09
- Retrieve an Invoice by ID
- Description: Verify that the user can view details of a specific invoice

Test Steps:

1. Use a valid invoice ID.
2. Use an invalid invoice ID.

Expected Results:

- Valid ID: Invoice details are displayed.
- Invalid ID: Error message is shown.

- Test Case ID: TC_10
- Return an Invoice
- Description: Verify that an invoice can be returned and its status updated
  .
  Test Steps:
  1. Submit a valid invoice return request.
  2. Try paying the returned invoice.

  Expected Results:
  - Invoice is marked as returned, and balance is adjusted.
  - Payment for returned invoices is disallowed.

- Test Case ID: TC_11
- Pay an Invoice
- Description: Verify that an invoice can be paid, and the payment is reflected in the contact's balance.

  Test Steps:
  1. Pay with a valid amount.
  2. Pay with an amount exceeding the invoice total.
  3. Pay a returned invoice.

  Expected Results:
  - Valid amount: Invoice is marked as paid, and contact balance is updated.
  - Excess amount: Error message is displayed.
  - Returned invoice: Payment is disallowed.

Test Execution Plan

1. Environment: Development, staging, and production environments.
2. Tools: Swagger

Exit Criteria

- All high-priority test cases are passed.
- Critical defects are resolved.
- Regression testing shows no major issues.

Bug Reports

- Bug ID: Bug_01
- Invalid Email Accepted in Email Address Field
- Title: Invalid email address accepted in the email address field

Description:

When entering an invalid email address (e.g., 1@1), the system accepts it as a valid input despite it not meeting the standard email address format. This allows users to submit incomplete or incorrectly formatted email addresses.

Steps to Reproduce:

1- Navigate to the user registration or email input form.
2- In the "Email Address" field, enter an invalid email such as @gmail (missing the local part and domain).
3- Submit the form.
4- Notice that the form is submitted successfully without any error or validation message.

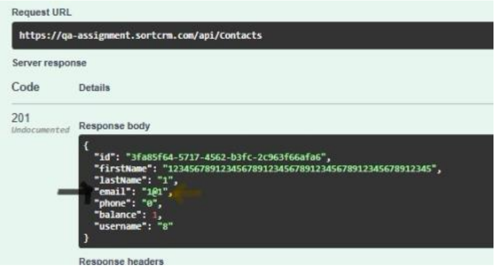
Expected Result:

The system should validate the email address field and reject any invalid email formats (e.g., missing local part, domain, or incorrect characters). The user should receive a clear error message indicating that the email address entered is invalid.

Actual Result:

The system accepts the invalid email address (@gmail) without any validation error.

Screenshots/Attachments:



```
https://qa-assignment.sortcrm.com/api/Contacts
```

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "firstName": "12345678912345678912345678912345678912345678912345",
  "lastName": "1",
  "email": "1@1",
  "phone": "0",
  "balance": 1,
  "username": "8"
}
```

Environment:

Browser: Google Chrome
Version: [130.00]
Operating System: Windows 10
Platform: Web

Severity: High
Priority: High

- Bug ID: Bug_02
- Blank Phone Number Field Triggers Error Despite Being Nullable
- Title: Phone number field incorrectly marked as required during registration despite being nullable

Description:

When a user leaves the phone number field blank during registration, the system incorrectly throws an error, requiring the phone number to be filled in. This issue occurs even though the phone number field is marked as nullable: true, meaning it should be allowed to remain blank without triggering an error.

Steps to Reproduce:

1- Navigate to the registration page.
2- Leave the Phone Number field blank (i.e., do not enter any value).
3- Complete the rest of the registration form and submit.
4- Observe that an error message appears, indicating that the Phone Number field is required, despite it being nullable.

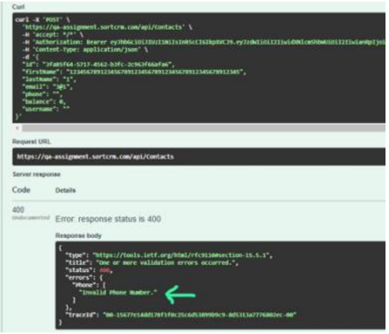
Expected Result:

The system should allow the Phone Number field to be left blank, as it is marked as nullable: true. No error should be triggered, and the user should be able to complete registration without a phone number.

Actual Result:

The system incorrectly requires the phone number, preventing the user from completing registration if the field is left blank, even though the phone number field is nullable.

Screenshots/Attachments:



Curl

```
curl -X 'POST' \
  'https://qa-assignment.sortcrm.com/api/Contacts' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI1IiwidXNlcm5hbWUiOiI1IiwianRpIjoi
  -H 'Content-Type: application/json' \
  -d '{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "firstName": "12345678912345678912345678912345678912345678912345",
  "lastName": "1",
  "email": "1@1",
  "phone": "",
  "balance": 0,
  "username": ""
}'
```

Request URL

```
https://qa-assignment.sortcrm.com/api/Contacts
```

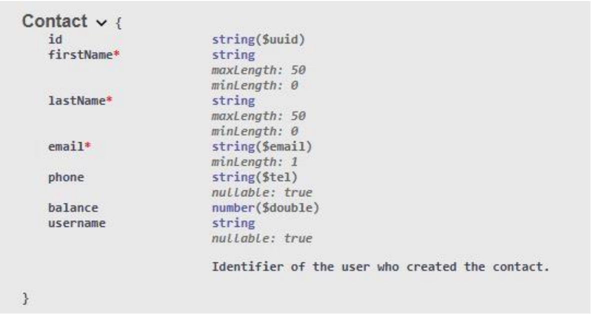
Server response

Code: 400 Undocumented

Details: Error: response status is 400

Response body

```
{
  "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
  "title": "One or more validation errors occurred.",
  "status": 400,
  "errors": {
    "Phone": [
      "Invalid Phone Number."
    ]
  },
  "traceId": "00-15677e14dd178f1f0c25c6d53099b9c9-8d5313a7776802ec-00"
}
```



```
Contact ⌄ {
   id                    string($uuid)
   firstName*            string
                         maxLength: 50
                         minLength: 0
   lastName*             string
                         maxLength: 50
                         minLength: 0
   email*                string($email)
                         minLength: 1
   phone                 string($tel)
                         nullable: true
   balance               number($double)
   username              string
                         nullable: true

                         Identifier of the user who created the contact.

}
```

Environment:
Browser: Google Chrome
Version: [130.00]
Operating System: Windows 10
Platform: Web

Severity: Low
Priority: Medium

Additional Information:
This issue prevents users from completing their registration if they choose not to provide a phone number, even though the field is optional. This could lead to frustration and hinder user sign-ups for those who do not wish to provide their phone number.

- Bug ID: BUG_03
- DELETE operation is not functioning as expected.
- Title: DELETE command to remove a customer info by ID, the subsequent GET command using the same customer ID still returns the customer's information

Description:
After executing a DELETE command to remove a customer by ID, the subsequent GET command using the same customer ID still returns the customer's information. This indicates that the DELETE operation is not functioning as expected.

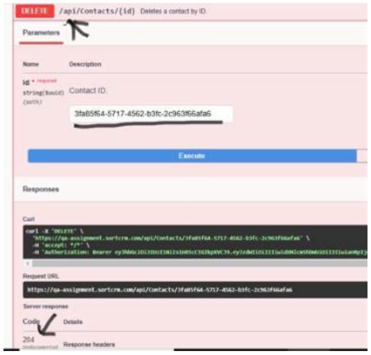
Steps to Reproduce:

1. Send a DELETE request to the endpoint to delete a customer by ID:
   Request: DELETE /customers/{customer_id}
   - Status code: 200 OK or 204 No Content
   - Message: "Customer informations deleted successfully."
2. Send a GET request to the same endpoint using the customer ID:
   Request: GET /customers/{customer_id}
3. Observe the response from the GET command.

Actual Result:
The GET command returns the customer's information, even though the DELETE command was successfully executed.

Expected Result:
The GET command should return a 404 Not Found or a similar error, indicating the resource does not exist after being deleted.

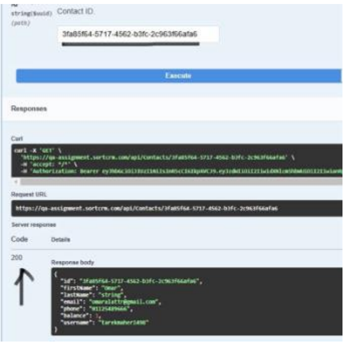
<u>Screenshots/Attachments:</u>



3fa85f64-5717-4562-b3fc-2c963f66afa6

```
curl -X 'DELETE' \
  'https://qa-assignment.sortcrm.com/api/Contacts/3fa85f64-5717-4562-b3fc-2c963f66afa6' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.[illegible]
```

```
https://qa-assignment.sortcrm.com/api/Contacts/3fa85f64-5717-4562-b3fc-2c963f66afa6
```



3fa85f64-5717-4562-b3fc-2c963f66afa6

```
curl -X 'GET' \
  'https://qa-assignment.sortcrm.com/api/Contacts/3fa85f64-5717-4562-b3fc-2c963f66afa6' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.[illegible]
```

```
https://qa-assignment.sortcrm.com/api/Contacts/3fa85f64-5717-4562-b3fc-2c963f66afa6
```

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "firstName": "Omar",
  "lastName": "string",
  "email": "omaralattr@gmail.com",
  "phone": "01125489664",
  "balance": 1,
  "username": "tarekmaher1498"
}
```

<u>Environment:</u>
Browser: Google Chrome
Version: [130.00]
Operating System: Windows 10
Platform: Web

<u>Severity</u>: High
<u>Priority</u>: High

- Bug ID:BUG_04
- the resulting balance retrieved using the GET command is incorrect. The added balance is doubled and then added to the existing balance
- Title: PUT Command Doubles the Added Balance When Updating Customer's Balance

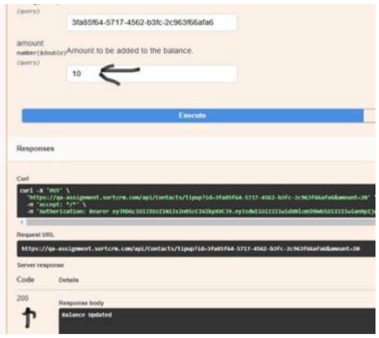
<u>Description:</u>
When a PUT command is used to update the balance of a customer by providing the customer ID and an amount to add, the resulting balance retrieved using the GET command is incorrect. The added balance is doubled and then added to the existing balance, leading to an inflated total.

<u>Steps to Reproduce:</u>

1. Use the GET command to retrieve the current balance of a customer.
   Request: GET /customers/{customer_id}
2. Use the PUT command to add a specific amount to the balance of the customer.
   Request: PUT /customers/{customer_id}/balance
3. Use the GET command again to retrieve the updated balance.
   Request: GET /customers/{customer_id}

Actual Result:
The customer’s balance is updated incorrectly. The added amount (10) is doubled (10* 2 = 20) and then added to the original balance (41+20= 60).

Expected Result:
The customer’s balance should increase by the exact amount provided in the PUT command (41 + 20 = 41).
Screenshots/Attachments:



(query)

3fa85f64-5717-4562-b3fc-2c963f66afa6

amount
number($double) Amount to be added to the balance.
(query)

10

Responses

Curl

```
curl -X 'PUT' \
  'https://qa-assigement.sortcrm.com/api/Contacts/tipup?id=3fa85f64-5717-4562-b3fc-2c963f66afa6&amount=20' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI2IiwidW5lcmShbWUiOiI2IiwianRpIj
```

Request URL

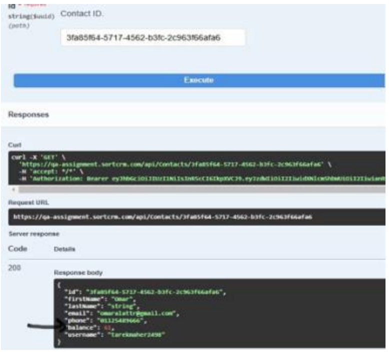
```
https://qa-assigement.sortcrm.com/api/Contacts/tipup?id=3fa85f64-5717-4562-b3fc-2c963f66afa6&amount=20
```

Server response

Code Details

200 Response body

```
Balance Updated
```



id * required
string($uuid) Contact ID.
(path)

3fa85f64-5717-4562-b3fc-2c963f66afa6

Responses

Curl

```
curl -X 'GET' \
  'https://qa-assigement.sortcrm.com/api/Contacts/3fa85f64-5717-4562-b3fc-2c963f66afa6' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI2IiwidW5lcmShbWUiOiI2Iiwian
```

Request URL

```
https://qa-assigement.sortcrm.com/api/Contacts/3fa85f64-5717-4562-b3fc-2c963f66afa6
```

Server response

Code Details

200 Response body

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "firstName": "Omar",
  "lastName": "string",
  "email": "omarelattr@gmail.com",
  "phone": "01125489666",
  "balance": 61,
  "username": "tarekmaher2498"
}
```

Environment:
Browser: Google Chrome
Version: [130.00]
Operating System: Windows 10
Platform: Web

Severity: High
Priority: High

- Bug ID:BUG_05
- The resulting Profit using the GET command is incorrect.
- Title: Incorrect Profit Calculation in Invoice System

Description:
The invoice system incorrectly calculates the profit when specific inputs are provided. The system computes a profit of 90, which is incorrect based on the given formula and inputs.

Steps to Reproduce:

1. Open the invoice system.
2. Input the following values:
   - Cost = 100
   - Discount Percentage = 10
   - Tax Amount = 10
3. Calculate the profit using the system.

Expected Result:

The profit should be calculated as follows:

1. Calculate the discount amount:

{Discount Amount} = {Cost} - {Discount Percentage}}={100} - {10%}{100} = 10

2. Calculate the total amount:

{Total amount} = ({Cost} - {Discount Amount}) + {Tax Amount} = (100 - 10) + 10 = 100

3. Calculate the profit:

{Profit} = {Total amount} - {Cost} = 100 - 100 = 0

Thus, the profit should be 0.

Actual Result:

The system calculates the profit as 90, which is incorrect.

Impact:

This issue leads to:

1. Misrepresentation of financial data.
2. Potential overestimation of business profitability.
3. Incorrect reporting and decision-making.

Root Cause (Suspected):

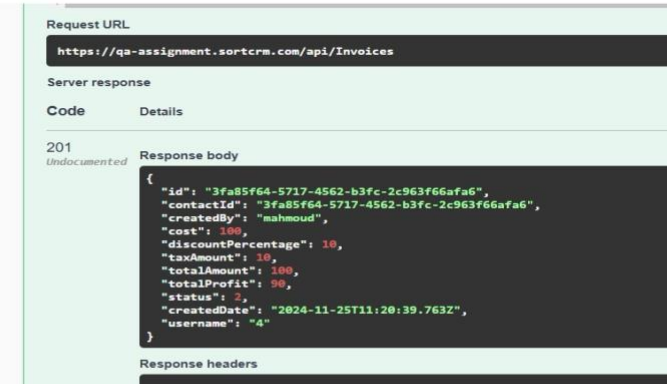
The system appears to calculate profit incorrectly by:

- Ignoring the cost in the calculation.
- Adding the tax amount directly to the profit without adjusting for discounts or costs.

Environment:
Browser: Google Chrome
Version: [130.00]
Operating System: Windows 10
Platform: Web

Priority: Critical
Severity:Critical

Screenshots/Attachments:



Request URL

```
https://qa-assignment.sortcrm.com/api/Invoices
```

Server response

Code Details

201 *Undocumented* Response body

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "contactId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "createdBy": "mahmoud",
  "cost": 100,
  "discountPercentage": 10,
  "taxAmount": 10,
  "totalAmount": 100,
  "totalProfit": 90,
  "status": 2,
  "createdDate": "2024-11-25T11:20:39.763Z",
  "username": "4"
}
```

Response headers

- Bug ID:BUG_06
- Invoice System Accepts Invalid Status Values
- Title: The invoice system incorrectly accepts a status value that is outside the defined enumeration of valid statuses

Description:
The invoice system incorrectly accepts a status value (7) that is outside the defined enumeration of valid statuses [0, 1, 2]. This leads to data integrity issues and potential logical errors.

Steps to Reproduce:

1. Open the invoice creation or update interface in the system.
2. Set the status of the invoice to 7.
3. Submit or save the invoice.

Expected Result:

The system should reject the status 7 and display an appropriate error message, such as:

> “Invalid status: The invoice status must be one of [0, 1, 2].”

Actual Result:

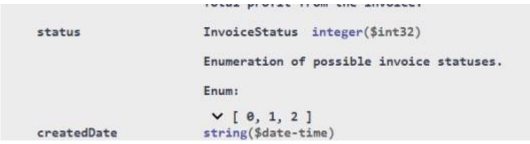
The system accepts the invoice with status 7 without any validation error.

Environment:

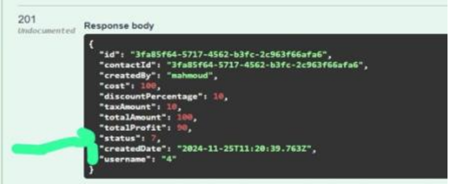
Browser: Google Chrome
Version: [130.00]
Operating System: Windows 10
Platform: Web

Priority: High
Severity:High

Screenshots/Attachments:



```
                     Total profit from the invoice.
status               InvoiceStatus  integer($int32)
                     Enumeration of possible invoice statuses.
                     Enum:
                      ⌄ [ 0, 1, 2 ]
createdDate          string($date-time)
```



201
Undocumented

Response body

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "contactId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "createdBy": "mahmoud",
  "cost": 100,
  "discountPercentage": 10,
  "taxAmount": 10,
  "totalAmount": 100,
  "totalProfit": 90,
  "status": 7,
  "createdDate": "2024-11-25T11:20:39.763Z",
  "username": "4"
}
```

- Bug ID:BUG_07
- Invoice payment issues
- Title: Payment Allowed with any Amount

Description:

The system allows users to pay any amount of money either greater or smaller than the invoice total. This behavior is unintended, as it could lead to financial discrepancies and customer confusion.

Steps to Reproduce:

1. Navigate to the Invoices section.
2. Use POST Command to pay invoice
2. Enter any amount exceeding the invoice total (e.g., invoice total = 95, payment entered = 10).
3. Confirm the payment.

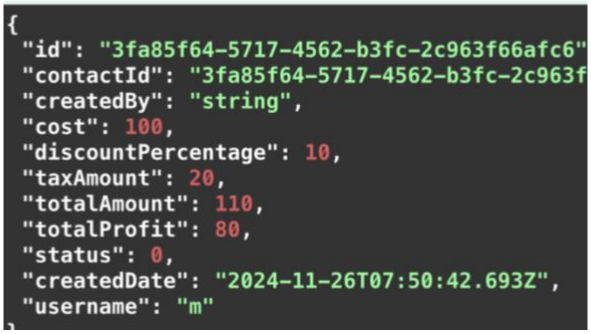
Expected Result:
The system should validate the payment amount and display an error message if the entered amount less than the invoice total.

Actual Result:
The system accepts any payment amount exceeding or less than the invoice total and processes the transaction without any warning or error message.

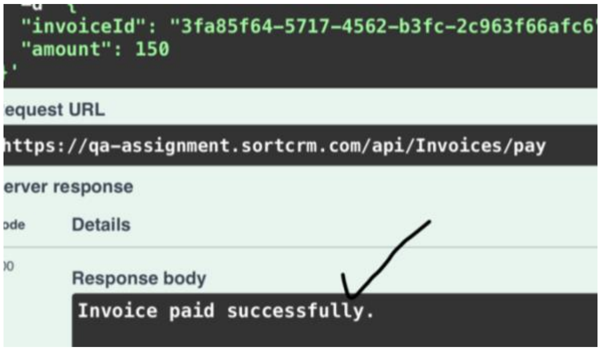
Environment:
Browser: Google Chrome
Version: [130.00]
Operating System: Windows 10
Platform: Web

Priority: Critical
Severity: Critical

Screenshots/Attachments:



```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afc6"
  "contactId": "3fa85f64-5717-4562-b3fc-2c963f
  "createdBy": "string",
  "cost": 100,
  "discountPercentage": 10,
  "taxAmount": 20,
  "totalAmount": 110,
  "totalProfit": 80,
  "status": 0,
  "createdDate": "2024-11-26T07:50:42.693Z",
  "username": "m"
```



```
  "invoiceId": "3fa85f64-5717-4562-b3fc-2c963f66afc6
  "amount": 150
}'
```

equest URL

```
https://qa-assignment.sortcrm.com/api/Invoices/pay
```

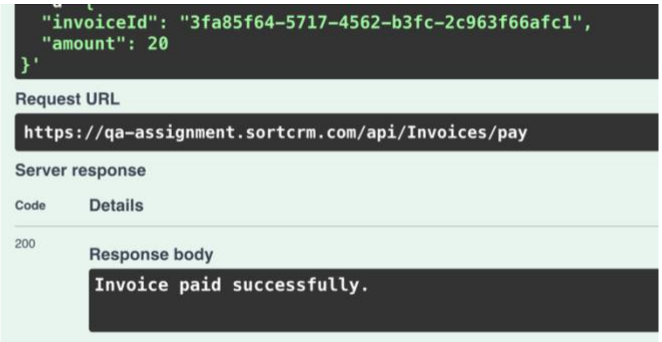
erver response

ode Details

00 Response body

```
Invoice paid successfully.
```

✓



```
  "invoiceId": "3fa85f64-5717-4562-b3fc-2c963f66afc1",
  "amount": 20
}'
```

Request URL

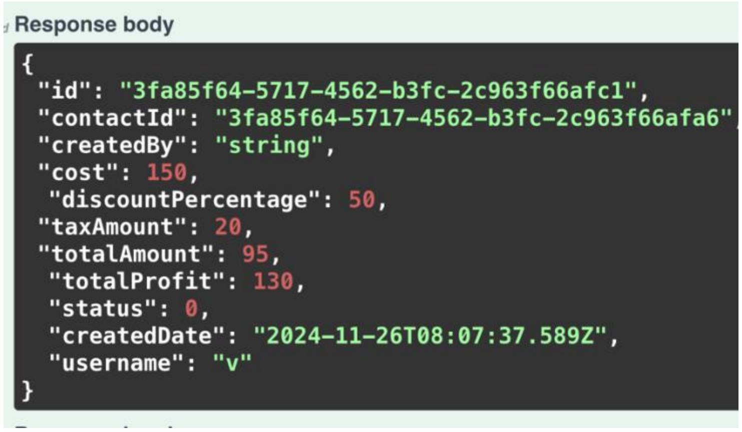
```
https://qa-assignment.sortcrm.com/api/Invoices/pay
```

Server response

Code Details

200 Response body

```
Invoice paid successfully.
```



Response body

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afc1",
  "contactId": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
  "createdBy": "string",
  "cost": 150,
   "discountPercentage": 50,
  "taxAmount": 20,
  "totalAmount": 95,
   "totalProfit": 130,
   "status": 0,
   "createdDate": "2024-11-26T08:07:37.589Z",
   "username": "v"
}
```

- Bug ID:BUG_08
- Edited Invoice Information Not Reflected in GET Command
- Title: The invoice system doesn’t reflect the edited value in GET Command

Description:
When updating an invoice by its ID using a POST command, the subsequent GET command to retrieve the invoice information returns outdated (old) data instead of the updated information.

Steps to Reproduce:

1. POST Command:
- Use a POST request to edit an invoice with a specific ID
- Provide updated information in the payload (e.g., Cost, Total amount)

"cost": 10
"Total amount": 19

2. GET Command:
- Use a GET request to retrieve the updated invoice by its ID.

3. Check the response from the GET command.

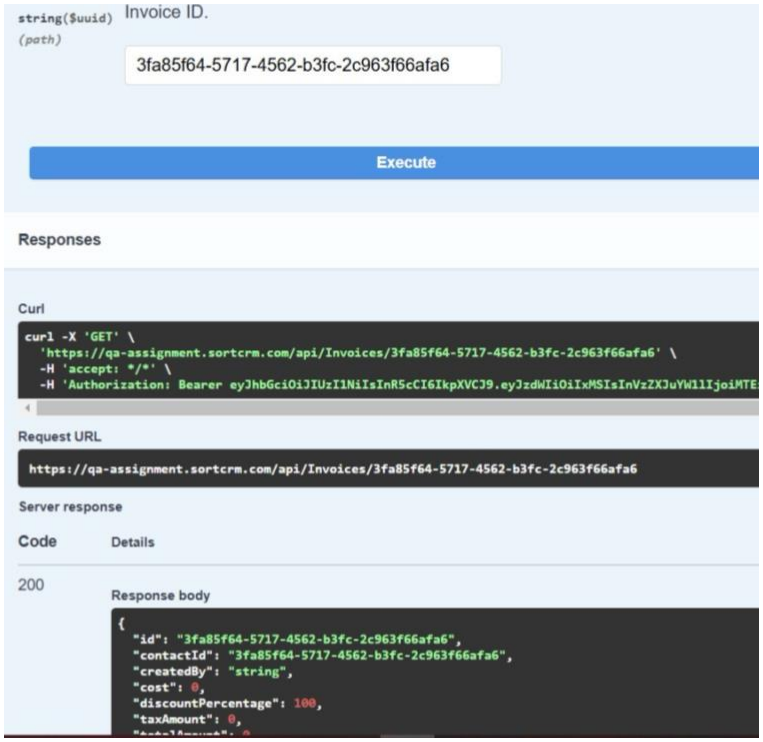
Expected Result:
The GET command should return the updated information for the invoice (e.g., cost = 10, Total amount = 19).

Actual Result:
The GET command returns the old invoice information (e.g., Cost =0, amount = 0), instead of the updated values.

Environment:
Browser: Google Chrome
Version: [130.00]
Operating System: Windows 10
Platform: Web

Priority: High
Severity: High

Screenshots/Attachments:



string($uuid) (path) Invoice ID.

3fa85f64-5717-4562-b3fc-2c963f66afa6

Responses

Curl

```
curl -X 'GET' \
  'https://qa-assignment.sortcrm.com/api/Invoices/3fa85f64-5717-4562-b3fc-2c963f66afa6' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMSIsInVzZXJuYW1lIjoiMTE
```

Request URL

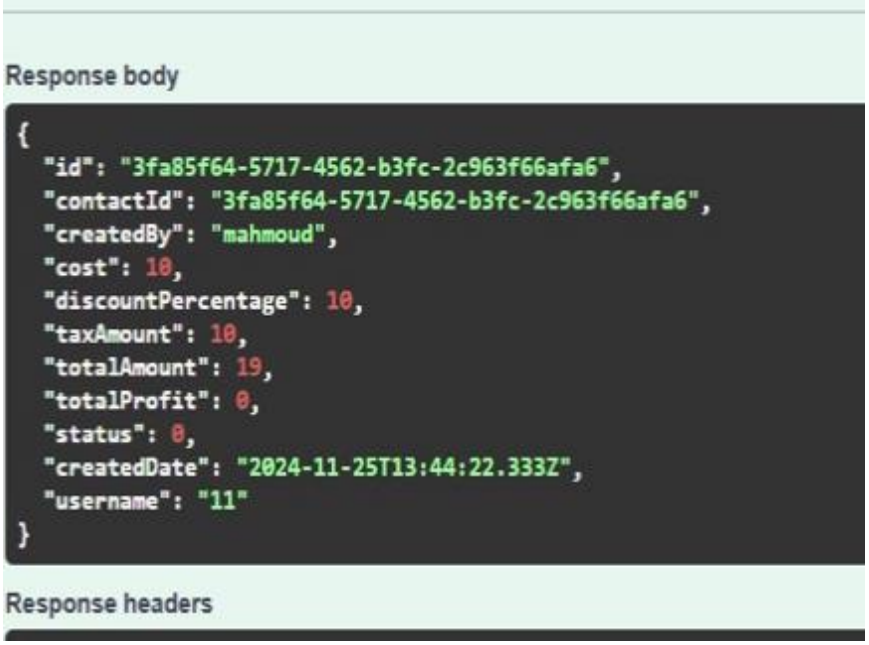
```
https://qa-assignment.sortcrm.com/api/Invoices/3fa85f64-5717-4562-b3fc-2c963f66afa6
```

Server response

Code Details

200 Response body

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "contactId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "createdBy": "string",
  "cost": 0,
  "discountPercentage": 100,
  "taxAmount": 0,
  "totalAmount": 0
```



Response body

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "contactId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "createdBy": "mahmoud",
  "cost": 10,
  "discountPercentage": 10,
  "taxAmount": 10,
  "totalAmount": 19,
  "totalProfit": 0,
  "status": 0,
  "createdDate": "2024-11-25T13:44:22.333Z",
  "username": "11"
}
```

Response headers