Freelance Project: API Testing for CRM Module

1. Project Overview:

- Project Type: API Testing for CRM Module
- Objective: Perform functional and non-functional testing of a CRM module based on provided User Stories. This task involves creating a comprehensive test plan, conducting API testing, and submitting bug reports with technical documentation.

2. Scope of Work:

- **Test Plan Creation:**
    - Create a detailed test plan for the provided CRM module user stories, covering both functional and non-functional aspects.
    - Define the test cases, testing methods, and expected outcomes for all functionalities, including contact and invoice management.
- **API Testing:**
    - Perform API testing using the provided API endpoints, ensuring thorough coverage of the functionalities listed in the user stories.
    - Use the "Bearer" token authentication process to interact with the API endpoints.
    - Test the following functionalities:
        - Manage Contacts: Test the CRUD (Create, Read, Update, Delete) operations for contacts and balance adjustments.
        - Manage Invoices: Test the creation, return, and payment processes for invoices, ensuring the system appropriately adjusts balances and handles edge cases (e.g., returning or paying an invoice).
- **Bug Reporting:**
    - Identify and report hidden bugs left intentionally in the CRM module by the development team.
    - Submit structured bug reports with reproduction steps, expected vs. actual results, and severity levels.
- **Technical Documentation:**
    - Provide detailed documentation of your findings, including:
        - Test Plan: A structured document outlining your testing approach.
        - Bug Reports: A well-organized list of bugs found, including screenshots or other relevant evidence.
        - Recommendations for improvement or additional test cases.

3. Deliverables:

- Comprehensive Test Plan covering all the user stories (functional and non-functional).
- Bug Reports documented with clear descriptions, reproduction steps, and severity levels.
- Technical Documentation including test cases, test results, and recommendations.
- Final Report summarizing your approach and overall findings.

4. Tools & Technologies:

- API testing tools like Postman, Swagger, or any preferred API testing platform.
- Documentation tools like Google Docs, Excel, or Google Sheets for test plans and bug reports.
- JIRA, Trello, or similar tools for managing bug tracking (if preferred).

5. Timeline & Hours:

- Total Hours: 32 hours
- Estimated Timeline: 10 days (including time for initial test plan creation, API testing, bug reporting, and documentation).

6. Freelance Terms:

- Rate: $10 per hour
- Total Cost: $320 (Paid in Egyptian Pound)
- Payment Schedule: Payment upon submission of deliverables or as per agreed milestones.
- Communication: Regular updates through email or Calendy platform, ensuring transparency throughout the project.

Main Points:

1. Adherence to the requirements, specifications, and instructions.
2. Test Coverage.
3. Bug Report.
4. Technical Documentation quality, readability, and maintainability.

TASK:

In this Job, you will be provided with a set of simplified User Stories that outline the features and functionalities of CRM Module. Your primary tasks will be:

1. Test Plan Creation: Based on the User Stories, create a comprehensive test plan that outlines the test cases, testing methods, and the expected outcomes. This should cover both functional and non-functional requirements.

2. Bug Reporting: You will be given access to the API endpoints of the CRM Module, which contain **hidden bugs** that the Dev team left for you on purpose to hunt. Your task is to identify these bugs and report them in a structured manner.

Expectations:

1. Attention to Detail: Ensure that all edge cases are covered in your test plans.
2. Documentation: All findings should be well-documented. This includes the test plans, bug reports, and any recommendations for improvement.
3. Perform API testing.

CRM Module User Stories:

A- Manage Contacts:

As an Employee I want to manage my contacts by creating, viewing, updating, and deleting them, as well as adjusting their balance, so that I can maintain accurate and up-to-date information for my business interactions.

Acceptance Criteria:

- **View all my contacts with optional pagination and search** parameters.

- Request a contact using its unique ID.
- Create new contacts
- Update information for an existing contact.
- Remove contacts.
- Add a specific amount to a contact's balance.

B- Manage Invoices:

As an Employee I want to manage my invoices by creating, viewing, returning, and paying them, so that I can maintain accurate financial records and handle billing efficiently.

Acceptance Criteria:

- **View a list of invoices associated with my account.**
- **I should receive the detailed information of any invoice if it exists.**
- **When I provide valid invoice details and submit a creation request for a specific contact, then a new invoice should be created, associated with the specified contact, and my contact's balance should be appropriately adjusted, and the total profit of the invoice should be calculated.**
- **When I submit a request to return an existing invoice, Then the invoice should be marked as returned, and the contact's balance should be adjusted accordingly. Additionally, I should no longer be able to pay the returned invoice.**
- **When I submit a payment request for a specific invoice with a valid amount, Then the invoice should be marked as paid, and the payment amount should be added to the contact's balance.**

Testing Process:

1. Navigate to API Link.
2. Each endpoint is well documented, make sure to read the API documentation carefully.
3. Register an account using your own email Under Account/Register endpoint.
4. Use the login endpoint to login and obtain a token so you can start testing.
5. Use the token to authorize swagger, before entering the value to swagger add a prefix to it like this: Bearer $APIKEYVALUE
    a. For example, if the token value was: 123435564664664
        i. add it like this in the blue value box: Bearer 123435564664664