A. All Identified Issues:

Functionality:

1. Expiration Date Fields:
    - The year selection uses a dropdown format, which is outdated and makes it harder for users compared to a simpler and quicker method like a calendar or direct input.
    - Is the entire expiration date section mandatory to fill out, or is it only the year selection that is required?
2. Required Fields:
    - Apart from the asterisk (*), it is not clear which fields are mandatory.
    - The unclear guidance can cause customers to make typing mistakes and waste time.
    - The “Credit card billing street address” field includes two text fields, one of which has an asterisk. This makes it unclear why both are needed and how to use them.
3. State/Province Field:
    - This field appears limited to specific countries and does not cater to users from different regions .Even if the dropdown is functional, it involves scrolling through a very long list, making it even more inconvenient for users.

Usability:

1. Middle Initial (MI) Field:
    - This field is irrelevant for most users globally and might confuse international users.
2. Instructions for the Card Number Field:
    - The note "No dashes or spaces" might confuse users. Adding an input mask to automatically prevent invalid formats would improve usability.
3. Error Message:
    - It is unclear if validation occurs while entering the information. If validation only happens upon submitting the form, any errors might require the user to start filling out the form again.
    - There is no visible evidence of where error messages would appear when fields are filled incorrectly or missing.
4. Postal Code Field:
    - The form does not provide guidance on the required format, and it might only support specific ZIP code styles, e.g., UK postal codes with letters or Israeli postal codes with seven digits.

Design:

1. Color Contrast:
    - The light blue background and black text do not provide enough contrast, making it harder to read for users with visual impairments.
2. Lack of Visual Hierarchy:

- The form appears visually uniform, with no clear separation between different sections like "Payment Information" and "Billing Address".
- The text size of the text field label is the same as the header size, which can be confusing. e.g., "Cardholder Name" and "First Name".

Security:

1. Card Number Field:
    - There is no visible indication that the entered data is encrypted. e.g., masking the number except for the last four digits.
2. No Secure Feedback:
    - The form does not display any visual indicator (e.g., a lock icon or PCI-DSS standard certification) to assure users that their data is being transmitted securely.
    - The form is designed in a way that makes it look fake and unreliable, giving the impression that it might be a scam.

Content:

1. Lack of Detailed Instructions:
    - The message "Please fill in the fields below" is too generic and does not explain the purpose of the form.
2. Incomplete Guidance:
    - The form does not clarify whether it is meant for personal, business, or other use.
    - It is unclear if the form is a payment form or account information form based on the header.

Main Issue:

The form faces challenges across multiple areas, including functionality, usability, design, content, security, and global adaptation. A key concern is its lack of global readiness, which makes it challenging for international users to complete the form efficiently. In addition to enhancing its global adaptability and dynamic features, ensuring the form complies with global security standards, such as PCI-DSS regulations, is critical. This will safeguard sensitive payment information, build user trust, and meet regulatory requirements for a SaaS company serving an international customer base.

B. Test Cases:

Test Case 1: Validate Error Message for Empty Required Fields

Description: Check that the form displays appropriate error messages when required fields are left empty.

Steps:

1.Leave all required fields blank.
2.Attempt to submit the form.
3.Verify the error messages for each empty field.

Expected Result: Each required field should display a clear and specific error message, such as “This field is required.”

Test Case 2: Verify Input Mask for “Card Number” Field
Description: Check that the “Card Number” field restricts users from entering invalid characters (e.g., letters, symbols, or spaces).
Steps:
1. Try entering valid numbers like 4111111111111111.
2. Try entering invalid input such as abcd!@#$%^&* (with no dashes or spaces).
3. Attempt to submit the form after entering invalid input.

Expected Result: The field should only accept valid numeric input. Invalid input should be blocked immediately, and an error message should appear if the input is incorrect.

Test Case 3: Validate “Postal Code” Format by Country
Description: Ensure that the “Postal Code” field validates the format based on the selected country.
Steps:
1. Select “United States” as the country and enter a valid numeric ZIP code .
2. Select “United Kingdom” as the country and enter a valid UK postal code (including letters)
3.Enter an invalid postal code for both countries.
4.Attempt to submit the form.

Expected Result: The form should accept valid postal codes for the selected country and reject invalid formats with a clear error message, such as “Invalid postal code format for the selected country.”

C. Suggested Product Solution:

A practical solution for addressing the form’s issues is to use a well-known, secure payment gateway that complies with global PCI-DSS standards such as "Sripe" or "PayPal". Such solutions often provide customizable forms that can be tailored to the company’s specific needs, ensuring security, adaptability, and compliance with international regulations.

If the company decides to build the form in-house, React is a highly recommended choice. Its dynamic and flexible nature makes it ideal for resolving the main challenges of the form, such as its lack of global readiness and unclear usability. With React, the form can dynamically adjust fields like “State/Province” based on the user’s location, improving its adaptability for international users. React also facilitates the implementation of real-time validation, which ensures errors are caught immediately, reducing user frustration and preventing the need to refill the entire form. Additionally, React simplifies the integration of essential security measures, such as encryption and visible trust indicators, building user confidence and meeting global standards. By leveraging React, the company can create a secure, globally adaptable, and user-friendly form that addresses its most critical weaknesses.