TraceLabs Task

Testing the Etherscan Registration form:

1.Create TEST SCENARIOS for the registration form on Etherscan website (https://etherscan.io/register)

2.Explain how Playwight with TypeScript tools would be utilized to enable an automatic testing system

3.Create E2E browser tests that will cover all test scenarios

Note: You can ignore the CAPTCHA when creating your test scenarios.

***TEST SCENARIOS***

***Positive Test Scenarios***

1. Register with valid username, email, password, and confirmed password - ***(script name – valid\_credentials.ts)***
2. Register using a valid email format (e.g. test@example.com) - ***(script name – valid\_email\_format.spec.ts)***
3. Password and confirm password fields match exactly - ***(script name – matching\_passwords.spec.ts)***
4. Password meets minimum complexity (e.g. at least 8 characters, including numbers/symbols if required) - ***(script name – password\_complexity.spec.ts)***
5. Successful form submission redirects to login or confirmation screen - ***(script name – successful\_registration\_redirect.spec.ts)***

***Negative Test Scenarios***

1. Leave all fields empty and click “Create Account” expecting validation messages - ***(script name – empty\_form\_validation.spec.ts)***
2. Enter invalid email format (e.g. test@.com, test.com) expecting email validation error - ***(script name – invalid\_email\_format.spec.ts)***
3. Use different values in password and confirm password expecting mismatch error - ***(script name – password\_mismatch.spec.ts)***
4. Use already registered email – expect “email already exists” error - ***(script name – email\_already\_exists.spec.ts)***
5. Use username with invalid characters (e.g. <>!) expecting validation error - ***(script name – invalid\_username\_characters.spec.ts)***
6. Use overly long values in fields (e.g. 100+ characters) expecting truncation or error - ***(script name – long\_values\_in\_fields.spec.ts)***

***Edge/Usability Test Scenarios***

1. Test form with leading/trailing spaces in fields - ***(script name – leading\_trailing\_spaces.spec.ts)***
2. Use keyboard only (Tab navigation) to complete and submit form - ***(script name – keyboard\_only\_navigation.spec.ts)***
3. Try pasting password into confirm password field - ***(script name – paste\_password\_confirm\_field.spec.ts)***
4. Test field-specific error messages disappear once corrected - ***(script name – error\_messages\_disappear\_on\_correction.spec.ts)***
5. Verify proper label and focus behavior for screen reader accessibility - ***(script name – screen\_reader\_accessibility.spec.ts)***