Cypress+Allure project review
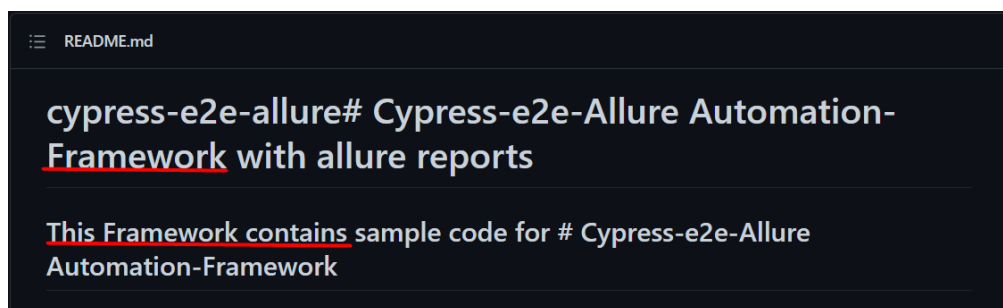
Link to GitHub repository - https://github.com/AZANIR/cypressAllure

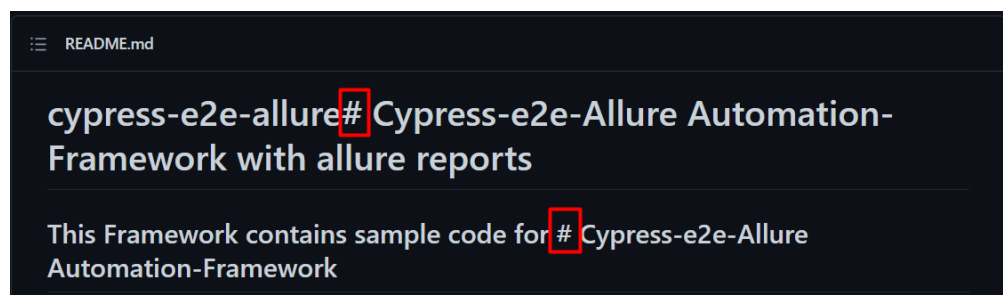Link to GitHub Pages - https://azanir.github.io/cypressAllure/

Reviewer – Chuiasova Tetiana
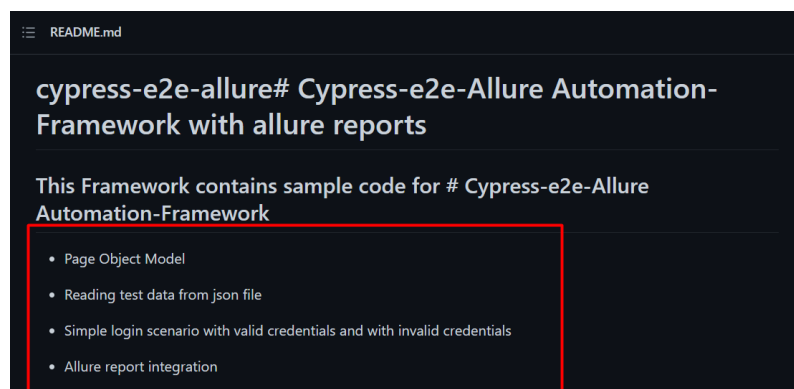
Project issues

1. Readme.md. Incorrect project name and description. A framework is a structure on which software can be built, but this repository represents a project, not a framework.
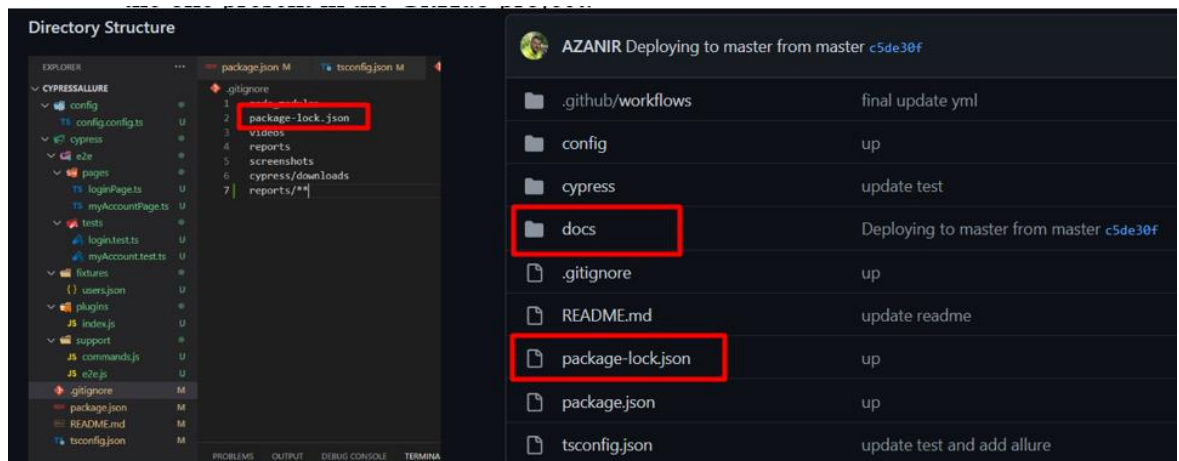

2. Readme.md. The hash sign should not be present in the project name and description.
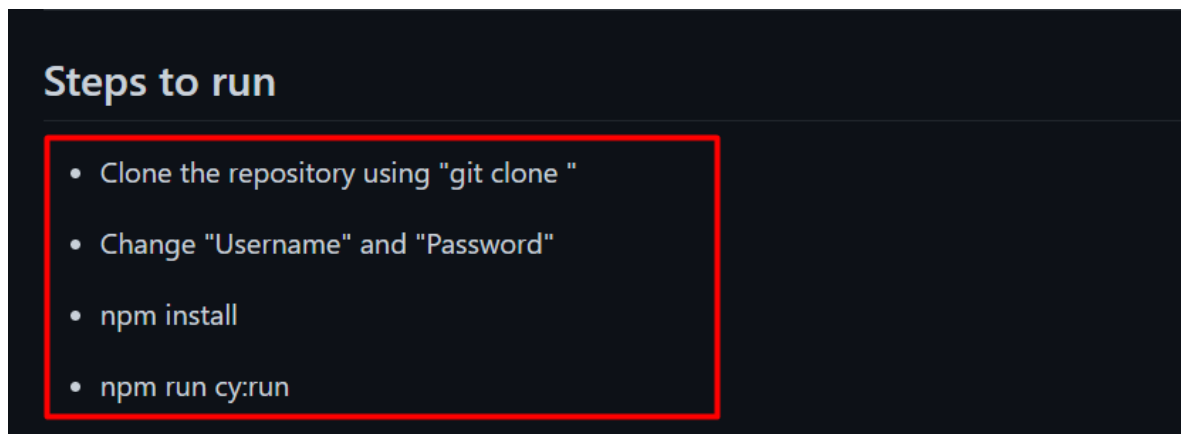

3. Readme.md. There is no title for this area (screenshot below).

4. Readme.md. The directory structure present in the readme does not match the one present in the GitHub project. Also package-lock.json file should not be in GitHub repo.
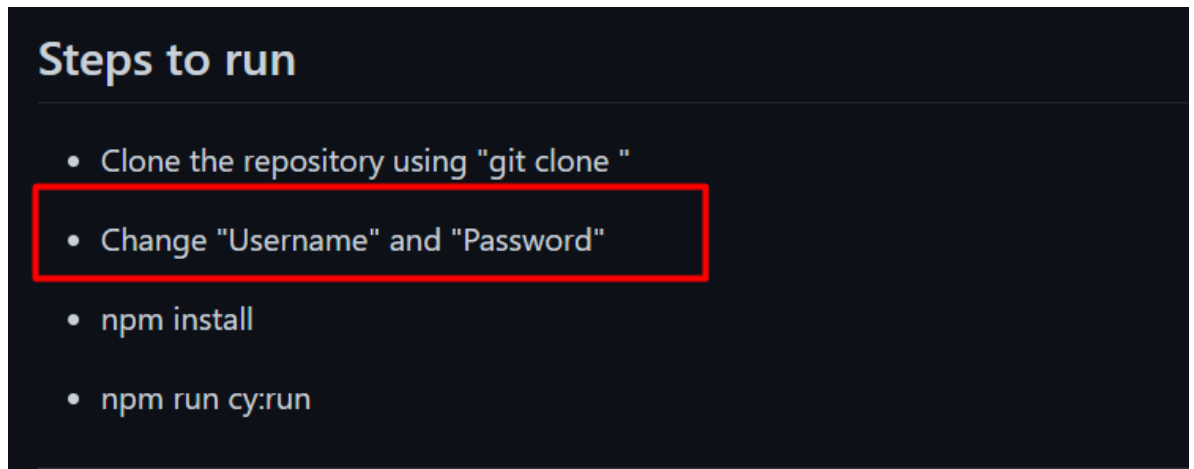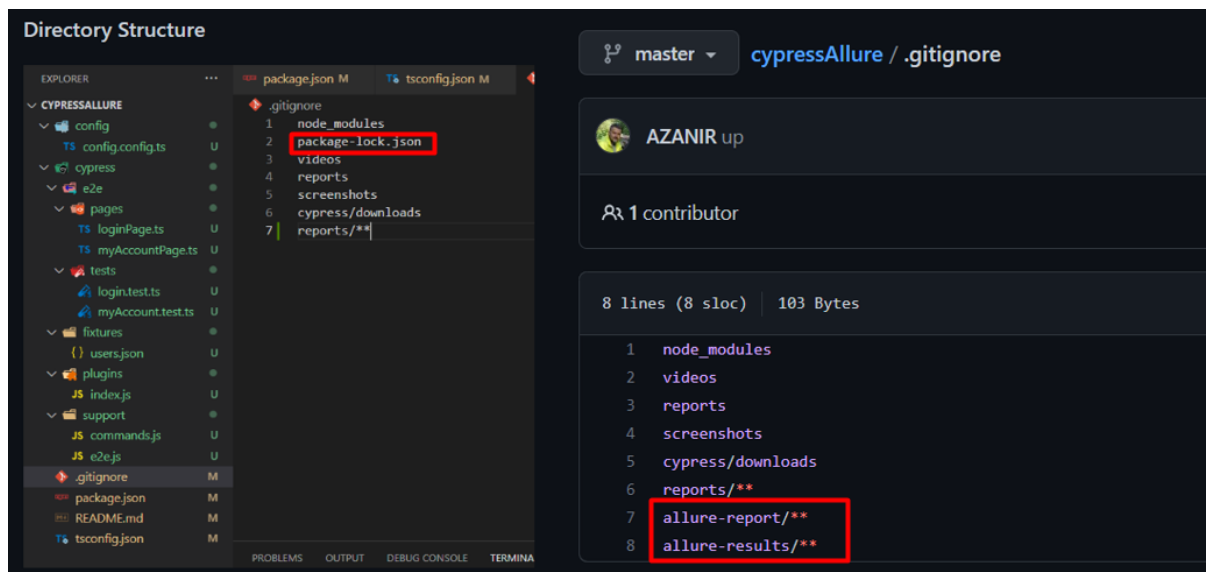

5. Readme.md. Steps to run should be in code block.

6. Readme.md. The whole git clone command should be described, including repository name. For example "git clone https://github.com/AZANIR/cypressAllure.git".
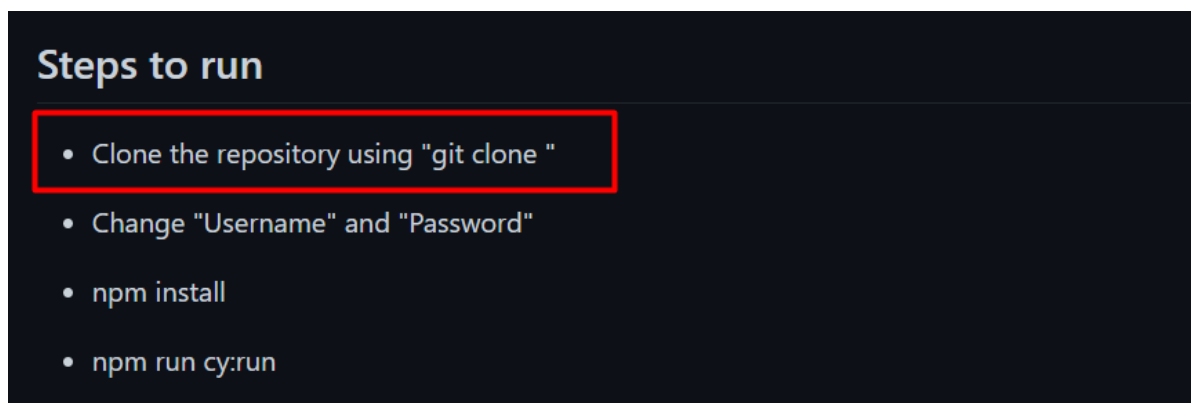
7. Readme.md. Unclear step to run. Who and why should change "Username" and "Password" fields?

8. Readme.md. No requirements section. Some system requirements should be described in the readme file, for example node version, cypress version, Visual code version and etc.

9. Readme.md. gitignore file that present on the screenshot in Readme.md does not match with .gitignore file in GitHub repo.

10. Project structure. The docs directory should be in a separate branch.

11. Tests. It is better to use the "require" method instead of "import".

12. login.test.ts. It is better to use the constants for input data.

13. Tests. It is better to start test cases title with the word "Should." For example "Should login with valid credentials".

14. myAccount.ts. What is the use of "beforeEach" if there is only one test?

```
⑂ master ▾        cypressAllure / cypress / e2e / tests / myAccount.test.ts / <> Jump to ▾

   ▣ AZANIR update test

   ⋈ 1 contributor

11 lines (10 sloc)  │  288 Bytes
   1    import { loginPage } from "../pages/loginPage";
   2
   3    describe('My Account Functionality', () => {
   4        beforeEach(() => {
   5            cy.visit('https://google.com');
   6            //loginPage.launchApplication()
   7        })
   8        it('Sample Test', () => {
   9            console.log("This is a sample test")
  10        })
  11    })
```
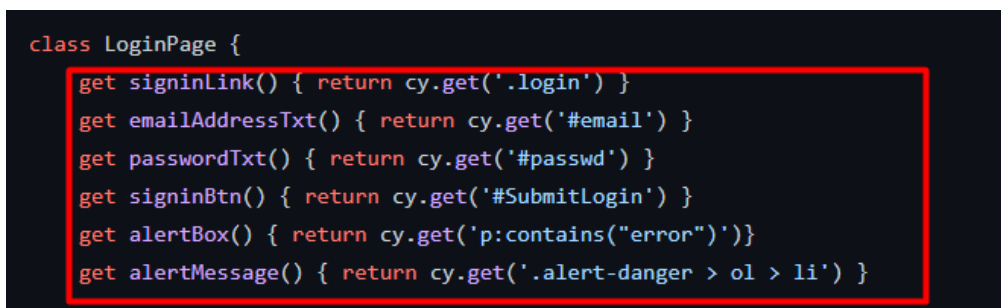
15. Page objects. It is better to use the constants instead of "get" method.

```
class LoginPage {
    get signinLink() { return cy.get('.login') }
    get emailAddressTxt() { return cy.get('#email') }
    get passwordTxt() { return cy.get('#passwd') }
    get signinBtn() { return cy.get('#SubmitLogin') }
    get alertBox() { return cy.get('p:contains("error")')}
    get alertMessage() { return cy.get('.alert-danger > ol > li') }
```

16. Page objects. In the method name it is better to describe its work. For example "clickLoginButton".

```
14
15    public login(emailId: string, password: string) {
16        this.signinLink.click()
17        this.emailAddressTxt.type(emailId)
18        this.passwordTxt.type(password)
19        this.signinBtn.click()
20    }
21
22    public validateLoginError(errorMessage: string) {
23        this.alertBox.should('be.visible')
24        this.alertMessage.should('have.text', errorMessage)
25
26    }
27  }
28  export const loginPage: LoginPage = new LoginPage()
```

17. Page objects. It is better to use "module.exports" for creating class instances.


```
22          public validateLoginError(errorMessage: string) {
23              this.alertBox.should('be.visible')
24              this.alertMessage.should('have.text', errorMessage)
25
26          }
27      }
28  export const loginPage: LoginPage = new LoginPage()
```

18. Page objects. It is better to use one general page where all browser methods (like click, type etc.) described. And then just call them from another pages.

Report issues

1. The screenshot of report in Readme.md doesn't match with report in GitHub Pages.
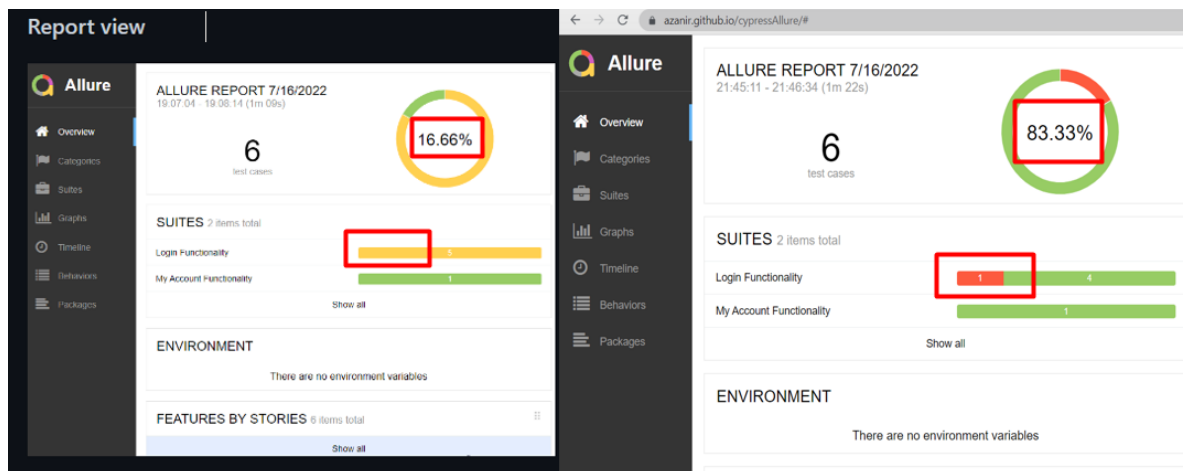
2. All report files should be pushed on the separate branch.