

# 1. Test Case Dashboard Sign Up Page

N o.	Action	inputs	Test Case	Expected output	Result
1	Launch Sign Up Page	https://brick-qa-assignment.herokuapp.com/;jsessionid=0D4FA430979F956478D7F42451A18E57	Verify the URL & verify the main page	Page launched and validate the URL	Pass
2	Click Register	All Form is Empty	verify that error occurred	Error : Please enter a firstname Error: Please enter a last name Error: Please provide an email Error: Please enter a password	Pass
3	Fill Name	Filbert	verify that error occurred	Error: Please enter a last name Error: Please provide an email Error: Please enter a password	Pass
4	Fill both first and last name	Filbert Wijaya	Verify the form is filled with the same as input, and error	Error: Please provide an email Error: Please enter a password	Pass
5	Fill Email only	abc@gmail.com	Verify the email and error validation	Error: Please enter a password Error : Please enter a firstname Error: Please enter a last name	Pass

6	Input Name, and Email with Different matching password	Filbert Wijaya <a href="mailto:abc@gmail.com">abc@gmail.com</a> pass: 123456 pass2: 111111	Verify the error	Password need to match	Pass
7	Fill All Data		Verify the date data input	Succesfully registered	Pass
8	Enter Password Below 6 Characters	11111	verify the password error	Please enter at least 6 characters.	Pass
9	Enter invalid email	asd	verify error message	Please enter a valid email address.	Pass
10	Filling Up all Required Form except Number	phone number is empty	verify that phone number form showing any error message	no error	Fail
11	Enter the Same Email Registered	abc@gmail.com	verify that the email taken message	email has been taken	Pass

## Login Test Case

N o.	Action	inputs	Test Case	Expected output	Result
1	Click Log In	https://brick-qa-assignment.herokuapp.com/login	Verify the URL & verify the main page	Page launched and validate the URL	Pass
2	Click Login	All Form is Empty	verify that error occurred	Error : Harap isi bidang ini	Pass
3	Fill Password only, Click Login	Email is Empty	verify that error occurred	Error : Harap isi bidang ini	Pass
4	Fill Email only, Click Login	Password Is Empty	Verify the form is filled with the same as input, and error	Error : Harap isi bidang ini	Pass
5	Fill false password, valid email	abc@gmail.com	Verify the email and error validation	Error: Wrong Email or Password	Pass
6	Fill Valid Password With False Email	123456	Verify the error	Error: Wrong Pass or Email	Fail
7	Fill All Data	<a href="mailto:abc@gmail.com">abc@gmail.com</a> 111111	Verify data input	Succesfully login	Pass
8	Logout		validate logout session	back to login page	Pass
9	Click Forgot Password		Verify the action	move to forgot password page	Fail
10	Click Sign Up		verify the page navigation to sign up page	move to sign up page	Pass

## 2. Bug Reports

Category	Label	Value
Bug ID 1	Id Number	sign up #1
	Name	signup
	Reporter	Filbert Wijaya

	Submit Date	2022
Bug Overview	Summary	When user does not fill the phone number while this form is required (*) it does not return error from the system.
	URL	https://brick-qa-assignment.herokuapp.com/
Environment	Platform	Windows
	Operating System	Windows 8
	Browser	Chrome
Bug Details	Step to Reproduce	going to the sign up page – fill all the data – do not fill the phone number – click register
	Expected	Error within the phone number form (this form is required)
	Actual	register succesful without filling the phone number
	Description	new product is shown (bugs)
Bug Tracking	Severity	Major
	Assigned To	abcd
	Priority	High
Notes		Fix

Category	Label	Value
Bug ID 2	Id Number	login # 1
	Name	login
	Reporter	Filbert Wijaya
	Submit Date	2022
Bug Overview	Summary	When user fill the password, and does not fill the username and then click the login button, message pop up for fill the email is shown
	URL	https://brick-qa-assignment.herokuapp.com/login
Environment	Platform	Windows

	Operating System	Windows 8
	Browser	Chrome
Bug Details	Step to Reproduce	going to the sign up page – click login – fill password – click login
	Expected	Shows error about username or password is error
	Actual	email form is required
	Description	new product is shown (bugs)
Bug Tracking	Severity	low
	Assigned To	abcd
	Priority	High
Notes		Fix

Category	Label	Value
Bug ID 3	Id Number	login # 2
	Name	forgot password
	Reporter	Filbert Wijaya
	Submit Date	2022
Bug Overview	Summary	When user clicks forgot password, error is shown
	URL	<a href="https://brick-qa-assignment.herokuapp.com/forgot">https://brick-qa-assignment.herokuapp.com/forgot</a>
Environment	Platform	Windows
	Operating System	Windows 8
	Browser	Chrome
Bug Details	Step to Reproduce	going to the sign up page – click login – click forgot password
	Expected	shows forgot password page
	Actual	no page is shown (error)
	Description	new product is shown (bugs)
Bug Tracking	Severity	Major

	Assigned To	abcd
	Priority	High
Notes		Fix

Category	Label	Value
Bug ID 4	Id Number	login # 3
	Name	username
	Reporter	Filbert Wijaya
	Submit Date	2022
Bug Overview	Summary	When user successfully login, the username is generated randomly without user agreement (user can not decide what username they want)
	URL	https://brick-qa-assignment.herokuapp.com/dashboard?sessionId=ccd08f35-cbba-415e-a200-29222d77530a
Environment	Platform	Windows
	Operating System	Windows 8
	Browser	Chrome
Bug Details	Step to Reproduce	going to the sign up page – click login – enter valid email and password – click login
	Expected	user can decide what username they want
	Actual	username generate randomly at login page
	Description	new product is shown (bugs)
Bug Tracking	Severity	Major
	Assigned To	abcd
	Priority	High
Notes		Fix

Category	Label	Value
----------	-------	-------

Bug ID 5	Id Number	register # 2
	Name	verify email
	Reporter	Filbert Wijaya
	Submit Date	2022
Bug Overview	Summary	When user successfully registered , verify message is shown. The user did not receive the verify email and User can not verify the email, result in the is_verified in database will always false
	URL	https://brick-qa-assignment.herokuapp.com/h2/login.do?jsessionid=2afa06293dc573611e06eab5e3b36816
Environment	Platform	Windows
	Operating System	Windows 8
	Browser	Chrome
Bug Details	Step to Reproduce	going to the sign up page – register all the required form – register – received the message
	Expected	user get the verify email
	Actual	user did not receive an email for verify, is_verify remain false in database
	Description	new product is shown (bugs)
Bug Tracking	Severity	Major
	Assigned To	abcd
	Priority	High
Notes		Fix

### 3. Automation test

There are 3 files,

- generalFunction.js, contain the script that used to open the app, validate text, user input, page navigation, etc.
- Login.js and register.js, contain the functions of tests , button ids, user input
- registerLogin.js, contain the execute file to do the test scenarios

#### generalFunctions.js

```
module.exports = class GeneralFunction {
  open() {
    return browser.url("https://brick-qa-assignment.herokuapp.com/");
  }
  async inputValueElement(element, value) {
    return (await element).setValue(value);
  }
  async clickElement(element) {
    return (await element).click();
  }
  async toHaveValue(element, value) {
    return (await expect(element)).toHaveValue(value);
  }
  async toHaveText(element, value) {
    return (await expect(element)).toHaveText(value);
  }
  async toHaveId(element, value) {
    return (await expect(element)).toHaveId(value);
  }

  async clearValue(element, value) {
    for (var i = 0; i < value.length; i++) {
      await this.inputValueElement(element, ["\u0008"]);
    }
  }
};
```

#### register.js

```
const GeneralFunction = require("../GeneralFunction");

class register extends GeneralFunction {

  async register() {
```



```
    await browser.pause(2000);

    await (await expect(browser)).toHaveUrl("https://brick-qa-assignment.herokuapp.com/");
  }

  async firstName() {
    const firstName = $("#firstName");
    //first name element
    await this.inputValueElement(firstName,"Filbert");
  }

  async lastName() {
    const lastName = $("#lastName") //last name element

    await this.inputValueElement(lastName,"Wijaya");
  }

  async email() {

    const email = $("#email")

    await this.inputValueElement(email,"abc@gmail.com");
  }

  async phone() {

    const phone = $("#phoneNumber")
    await this.inputValueElement(phone,"12345");
  }

  async address() {

    const address= $("#address")
    await this.inputValueElement(address,"road");
  }

  async password() {

    const password= $("#password")
    await this.inputValueElement(password,"111111");
  }

  async confirmPassword() {

    const confirmPass = $("#confirm_password")
    await this.inputValueElement(confirmPass,"111111");
  }
}
```

```
    await this.browserPause();

  }
  async registerButton() {
    const register = $("#myform > div.form-row-last > input")
    await this.clickElement(register)
  }
}
```

### Login.js

```
class login extends GeneralFunction {

  async login() {

const login = $("#/html/body/div/div/form/div[7]/p/a")

    await this.clickElement(login)

    await browser.pause(2000);

    await (await expect(browser)).toHaveUrl("https://brick-qa-assignment.herokuapp.com/login")
  }
  async email() {
const email = $("#your_email");
    await this.inputValueElement(email,"abc@gmail.com");

  }
  async password() {
    const lastName = $("#password")

    await this.inputValueElement(password,"111111");
  }
  async loginBtn() {

    const login = $("#myform > div.form-row-last > input")

    await this.clickElement(login)

  }
  async verifyLogin() {
```

```
    await browser.pause(2000);

    await (await expect(browser)).toHaveUrl("https://brick-qa-assignment.herokuapp.com/dashboard?sessionId=ae2e8634-6608-41c4-ae61-d168acb6cee9")
  }
  async logoutBtn() {

    const logout = $("body > div.centered > a")

    await this.clickElement(logout)

    await (await expect(browser)).toHaveUrl("https://brick-qa-assignment.herokuapp.com/login")
  }
```

#### registerLogin.js (e2e)

```
const register = require("../pageobjects/register.js");
const login = require("../pageobjects/login.js");

describe("Register and Login", () => {
  it("Open App", async () => {
    await register.open();
  });
  it("Fill First Name", async () => {
    await register.firstName();
  });
  it("Fill Last Name", async () => {
    await register.lastName();
  });
  it("Provide Email", async () => {
    await register.email();
  });
  it("Enter Phone Number", async () => {
    await register.phone();
  });
});
```

```
it("Enter Address", async () => {
  await register.address();
});
it("Enter Password", async () => {
  await register.password();
});
it("Enter Confirm Password", async () => {
  await register.confirmPassword();
});
it("Click Register", async () => {
  await register.registerButton();
});
it("Enter Email to Login", async () => {
  await login.login();
});
it("Enter Password to Login", async () => {
  await login.password();
});
it("Click Login ", async () => {
  await login.loginBtn();
});
it("Verify Login", async () => {
  await login.verifyLogin();
});
it("Log Out", async () => {
  await logout.logoutBtn();
});
});
```