verifymy

Test Resport for Bugbank Site

Summary

This plan will describe the process to test the site called Bugbanck and your features described in this document.

- Features to be tested:
  - User registration
  - Financial transactions

Testing Strategy

- Manual testing

The site under testing will be checked manually, executing all predicted tests.
It could be necessary to use exploratory testing.

- Automated testing

To deliver the quality assurement is predicted automated tests from selected test cases to take part in the regressive tests.
It will be developed with Cypres.io

All automated tests will run before each code delivery in the branch Main in the repo site in Github.

Table 1

Requirement
No major issues in main functions
It should be possible to transfer money between accounts.
It should be possible to register and access user account

Tests to be executed

1. Acceptance Test

- Register new user account

1. Register new user account		
Objective: Make sure that it is possible to register a new user account.		
Preconditions: Access to the site		
Steps	**Expected Result**	**Result: Pass**

1. Click in ‘Registrar’ 2. Fill all fields: a. teste@gmail.com b. Fulano da Silva c. teste123 3. Click in ‘Cadastrar’	- It should show a popup informing that the account has been created - The fields should be validated (email, name, password)	- Pass

<table>
<tr><td colspan="3"><b>2. Test the fields’ validations</b></td></tr>
<tr><td colspan="3"><b>Objective:</b> Make sure that it has validation when input</td></tr>
<tr><td colspan="3"><b>Preconditions:</b> Access to the site</td></tr>
<tr><td><b>Steps</b></td><td><b>Expected Result</b></td><td><b>Result: Pass</b></td></tr>
<tr><td>1. Click in ‘Registrar’
2. Fill out the email’s field: ‘test.com’</td><td>- It should show a message: “Invalid Format”</td><td>- Pass</td></tr>
</table>

<table>
<tr><td colspan="3"><b>3. Test the hide/show function to password field</b></td></tr>
<tr><td colspan="3"><b>Objective:</b> Make sure that it is possible to show the password in the field</td></tr>
<tr><td colspan="3"><b>Preconditions:</b> Access to the site</td></tr>
<tr><td><b>Steps</b></td><td><b>Expected Result</b></td><td><b>Result: Pass</b></td></tr>
<tr><td>3. Click in ‘Registrar’
4. Fill out the password
5. Click in the hide icon button</td><td>- It should appear the password</td><td>- The password hasn’t been shown</td></tr>
</table>

- Transfer money between accounts

<table>
<tr><td colspan="3"><b>4. Test the transfer money between accounts</b></td></tr>
<tr><td colspan="3"><b>Objective:</b> Make sure that it is possible to transfer money.</td></tr>
<tr><td colspan="3"><b>Preconditions:</b> It has created two accounts, A and B, both with balance</td></tr>
<tr><td><b>Steps</b></td><td><b>Expected Result</b></td><td><b>Result: Pass</b></td></tr>
<tr><td>1. Logging in account A
2. Click in ‘Transferencia’
3. Fill account B
4. Fill the value to transfer: “100.23”
5. Fill the description
6. Click in “Transferir agora”</td><td>- The account B should have:1100.23
- The account A should have:899,77</td><td>- Pass</td></tr>
</table>

5. Test transfering money with a value greater than the accounts has		
Objective: Make sure that it is impossible to transfer the value.		
Preconditions: It has created two accounts, A and B, both with balance. And account A balance is about 1000		
Steps	**Expected Result**	**Result: Pass**
7. Logging in account A 8. Click in 'Transferencia' 9. Fill account B 10. Fill the value to transfer: "100000" 11. Fill the description 12. Click in "Transferir agora"	- It should be not transferred - Account A should have the same balance - Account B should have the same balance - The transaction should be registered in the statement	- Pass

2. Exploratory Test

Table 2

Tested Feature: Register Page	Notes
1. Validate the register page in Firefox Browser	**The register page has a fail when the animation changes the fields. Picture 1 appendix A**
2. Validate limit text for fields name, email and password	**This field hasn't limit to input characters. This issue could be exploited to malicious behavior**
3. Validate password pattern	**The password hasn't a basic pattern to improve the security.**
4. Validate the changing field to register	**When changing to register forms the fields don't get cleaned.**
5. Validate transfer money to other account	**After transfer, while showing success popup, the transfer button is still enabled to press and send more money.**
6. Validate Bank statement	**The register of Bank statements haven't time, just a date. It should have a date and**

	time.

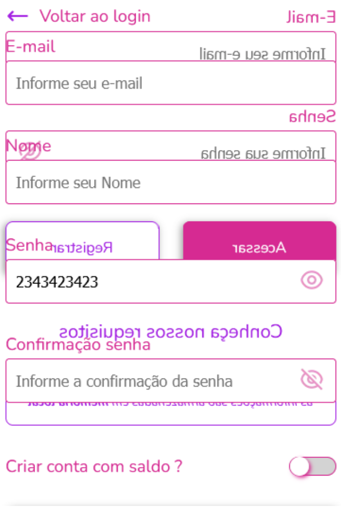
3. Conclusion

All acceptance tests have passed.
However in exploratory testing there are some issues.

- Security (issues 2,3) table 2
  - Issue 2: It should has limit to insert characters in the fields
  - Issue 3: The password should has pattern such as, as least one number and special characters
- UI (issue 1)
  - Issue 1: It should be fixed in the browser Firefox. In Chrome the view is correctly
- Functional (4,5)
  - Issue 4: It Should clean up all form fields when access the page
  - Issue 5: It should disable all buttons when it is showing popup
- Readiness (6)
  - Issue 6: The list of transactions should have date and time. It's important to create readiness in a audit process

Appendix A

Picture 1 - View register form from Firefox Browser



2343423423