iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** X **Month of publication:** October 2022

DOI: https://doi.org/10.22214/ijraset.2022.46682

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue X Oct 2022- Available at www.ijraset.com

Functional Testing on A software Model For Creating A Stable Cryptocurrency Using Fiat Currency For Global Electronic Commerce

Alex Kibet[1], Simon Karume[2], Nelson Masese[3]
Kabarak University, 13 P.O. Box Private Bag, Kabarak, 20157, Kenya

Abstract: ***Software testing is part of a set of activities that ensure high-quality software. It primarily aims at revealing defects that have been inserted into software at various stages of its development. In functional testing, test requirements are derived from software specifications. This paper proposes a functional testing/ evaluation that was performed using the functional specification provided during a Model for Creating a Stable Cryptocurrency Using Fiat Currency for Global Electronic Commerce design and verifies the model against the functional requirements. The functional approach of model effectiveness was used to establish model functions first, and then to build criterion measures to assess how well the objectives were met. The functional requirements of the evaluation model were then transformed into test cases. Furthermore, the test case reports were presented as final findings.***

Keywords: software testing; testing techniques and criteria; functional testing;

I. INTRODUCTION

Software testing/ evaluation is a knowledge area within the field of software engineering and system development, which strives for quality and continually contributes to the process and product improvement. The test's main objective is to reveal defects in the software artifact so these may be solved before any damage. Ideally, the testing activity must be systematic, and the techniques used must balance cost reduction and increase the levels of defect detection, should any exist. Each technique has a set of test criteria, which may be used during the conception, selection, and evaluation of a test set.

The developed model for creating a stable cryptocurrency using fiat currency for online trading used in this illustration was evaluated using functional testing. Functional testing/ evaluation was performed using the functional specification provided during the model design and verifies the model against the functional requirements (Chung et al, 2012). The functional evaluation deals with attaining the defined functional requirements. The purpose of Functional tests is to test model functionality, by providing appropriate input and verifying the output against the functional requirements. This testing checked the User Interface, APIs, Database, Client/Server communication, and other functionality of the model. The evaluation mainly concentrated on;

1) Mainline functions- testing the main functions of the developed model
2) The model's basic Usability – involved the basic usability testing of the model. It checked whether a user (expert user in this case) can freely navigate through the screens without any difficulties.
3) Accessibility- Checked accessibility of the model functionalities
4) Error Conditions: Usage of testing techniques to check for error conditions and to check whether suitable error messages are displayed.

II. DESCRIPTION OF EVALUATION CRITERIA

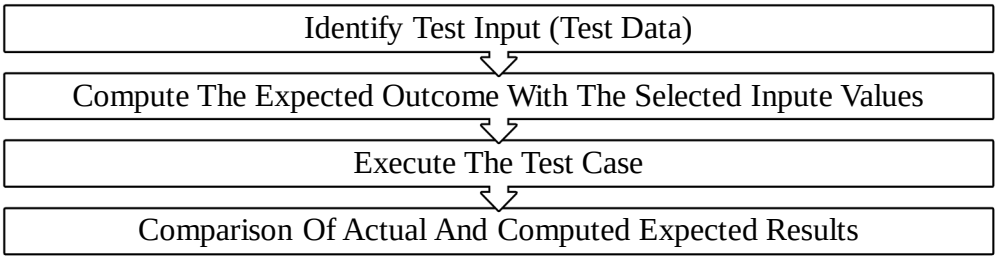


Figure 1 Description of model Evaluation Criteria source: (Khanom & Miah, 2004)

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue X Oct 2022- Available at www.ijraset.com

The model evaluation process involved the definition of the model functional requirements. This was in line with the model design (which takes into account the scenarios and the focus group discussion output). The model requirements were defined according to the following criteria: "functionality, completeness, consistency, accuracy, performance, usability, fit with the model, and other relevant quality attributes" as pointed in (Khanom & Miah, 2004). After the entire model requirements had been specified, a test input or test data was identified based on the requirements and grouped into three test cases. An expected outcome was specified with each selected test input value for each test case before computing or running the test case. Thereafter, the test cases were executed and the outcome was recorded per input. In the end, a comparison of the actual outcome and expected results was done and a conclusion was drawn.

III. FUNCTIONAL TESTING PROCESS

The functional testing aimed to address the core purpose of the model and to prove the overall concept of this study. To test and validate the model, this study adopted a guideline described by Shas (2018).

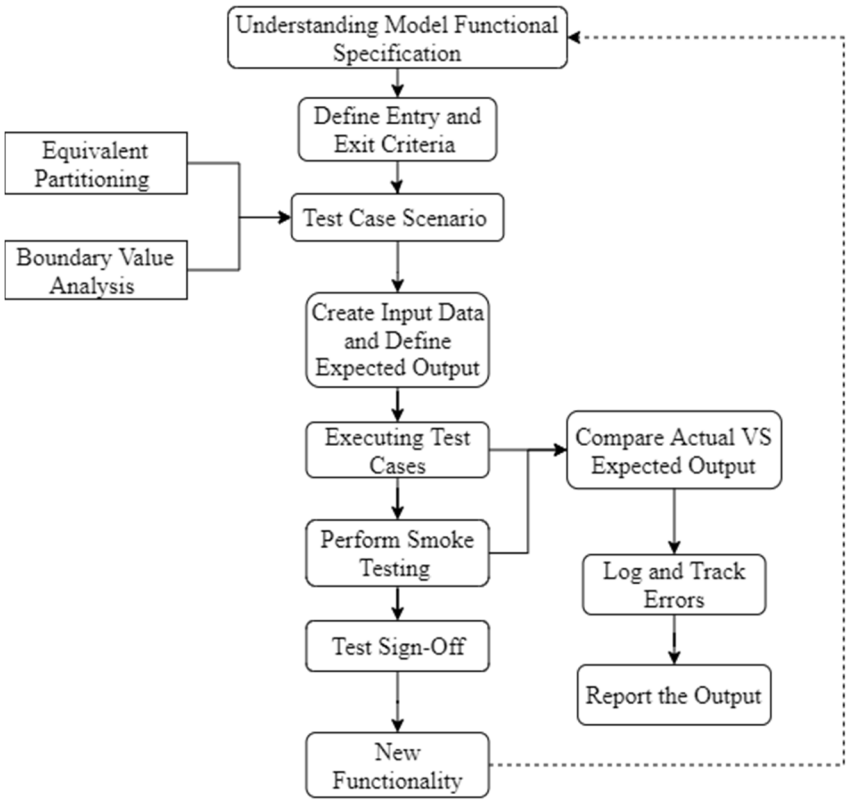


Figure 1 Shas process for conducting functional testing source: Shas (2018)

The entire steps were, however, not used but the steps were summarized but still kept the process viable. The functional testing process is divided the evaluation process into the following steps.

1) Step 1. Test Goal definition

The main goal of functional testing was to check how closely the model feature/specification is working as per the specifications. The functional testing goals mainly focused on validation and defect testing. I.e. to demonstrate that the model meets the requirements and to discover the defects in the functionality.

2) Step 2. Entry Criteria and Exit Criteria specification

The entry criteria (The beginning) involve the requirement specification, Test Cases preparation, Test data creation, and model setup ready for evaluation. While the exit criteria are when the Execution of all the functional test cases had been completed and no critical P2 bugs are open.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue X Oct 2022- Available at www.ijraset.com

3) Step 3. Listing the scenarios to create functional test cases

This step designed the model test scenarios for the design specification. A `test scenario' is the summary of the model's functionality. Based on these scenarios test cases were prepared.

Here is the list of possible scenarios for our payment gateway example.

a) Users create an account with the model
b) Users authenticated based on the information provided
c) Buy stable counts
d) Transfer stable counts
e) Users can dissolve their crypto accounts
f) Spend the stablecoin
g) Convert the coin into other cryptocurrencies

4) Step 4. Definition of Input data and the expected output

Input data for the functionality testing as per the requirement specification was specified. Later from the requirement specification, the output was determined for the functionality under test. The functionalities targeted the following features

a) Payment gateway
b) Debit/Credit Card Options
c) API
d) Swap operation

5) Step 5. Executing test cases

The prepared test cases were executed and the outcome was recorded for comparison as shown in table 11.

IV. FUNCTIONAL TESTING

As previously stated, the goal of this methodology was to evaluate the created model against the functional requirements/specifications. Each functional requirement in the model's minimum viable product was tested by giving acceptable input and comparing the output to the functional requirements as shown below.

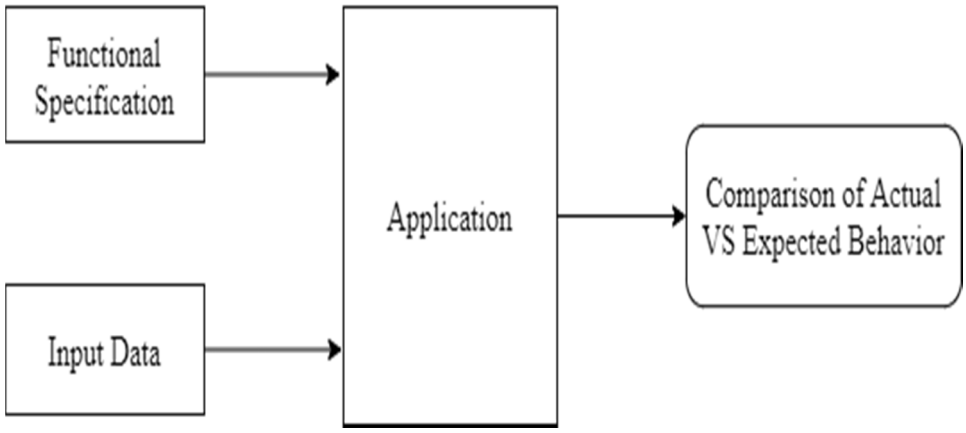


Figure 2 Functional testing source: (author)

The functional approach of model effectiveness was used to establish model functions first, and then to build criterion measures to a ssess how well the objectives were met. The functional requirements of the evaluation model were then transformed into test cases. The test case specifications, steps for execution, and the expected results were defined as shown below. Each test case was run and the results were kept track of.

Table 1 the model Test cases to compare the output source: (author)

TC-ID	Specifications	Steps To Execute		Expected Results	Status
TC-01	Check if the model can allow users to create profiles or accounts with the model using the user's name, phone number, and any other information necessary for KYC. (key creation)	1	Have a potentially new user	It should save the user details and allow the user to access the model. Also, the user's data should be protected using encryption.	
		2	Generate a biodata for the user		
		3	Fill in the user information		
		4	Click on the register/submit button		
TC-02	Check if the user can access model services upon account creation.	1	Try to access it on a different phone without the model setup	It should allow the user to access the model services only after account creation and for the otherwise case, the model shouldn't allow access	
		2	Reload the mobile after successful registration in the TC-01		
		3	Check if the user accesses the model services		
TC-03	Check if a user can buy cryptocurrency based on the fiat-token operation	1	Raise a request for crypto	It should allow debiting of the user's crypto account with an equivalent fiat currency.	
		2	Permit fiat currency transfer from the selected digital account.		
		3	submit an equivalent digital fiat currency		
		4	Wait for response		
TC-04	Check if a user can liquidate his/her account (convert his crypto to fiat currency)	1	Initiate the conversion process	It should allow conversion or liquidation service of crypto to fiat currency. in the event of withdrawing more than the balance then it should display an error message.	
		2	Specify the amount of crypto to convert more than the current balance.		
		3	Repeat the same process 2 with a valid amount		
		4	Click on submit button		
TC-05	Check if a user can use his/her cryptocurrency to pay for goods and services	1	Check for goods and services to buy	It should allow a user to pay for items by crediting their accounts and showing the ownership of the item purchased	
		2	Select the item for purchase		
		3	Click on the buy button		
TC-06	Check if the stablecoin can be converted into another cryptocurrency	1	Specify the user account details	The model should allow users to convert their cryptocurrencies from having Kenya shillings as the base currency to another	
		2	Specify the base currency		
		3	Then run		
TC-07	Check if the created cryptocurrency can be transferred from one account to the another	1	Access the transfer module	The model should only allow the transfer of cryptocurrencies less than the current balance.	
		2	Initiate transfer with negative and positive data		
		3	Specify the recipient's details		
		4	Then submit		
TC-08	Check if the distributed data can be accessed by the network crews	1	Open the host network	The transaction records must have been recorded in the blockchain platform. The entire blockchain-based information about the transactions must be transparent	
		2	Open the ropsten etherscan		
		3	Search the details with the user account address		

	The Test Case status definitions are	
	Status	Meaning
	Passed (P):	Test run-result matches the expected result
	Failed (F):	In some cases: i) Test run-result did not match the expected result ii) The result did match as per expectation but caused another problem.
	Not Run (NR):	The test has not yet been executed either because of the module malfunction or action needed

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue X Oct 2022- Available at www.ijraset.com

V. TEST STATUS REPORTING

The test cases were run and the results were kept track of. To assess whether the functionality is performing as expected, the actual result after executing the test case was compared to the expected output (derived from the requirement specification). The system test revealed that it was indeed feasible to deliver a model for creating stable cryptocurrency using fiat currency. The overall test was a pass. The test results are presented in the table below.

Table 2 Test Status Reporting source: (author)

TC ID	TC-01	TC-02	TC-03	TC-04	TC-05	TC-06	TC-07	TC-08
Status	P	P	P	P	P	P	P	P
FT	A	B	C	D	E	F	E	H and G

Key

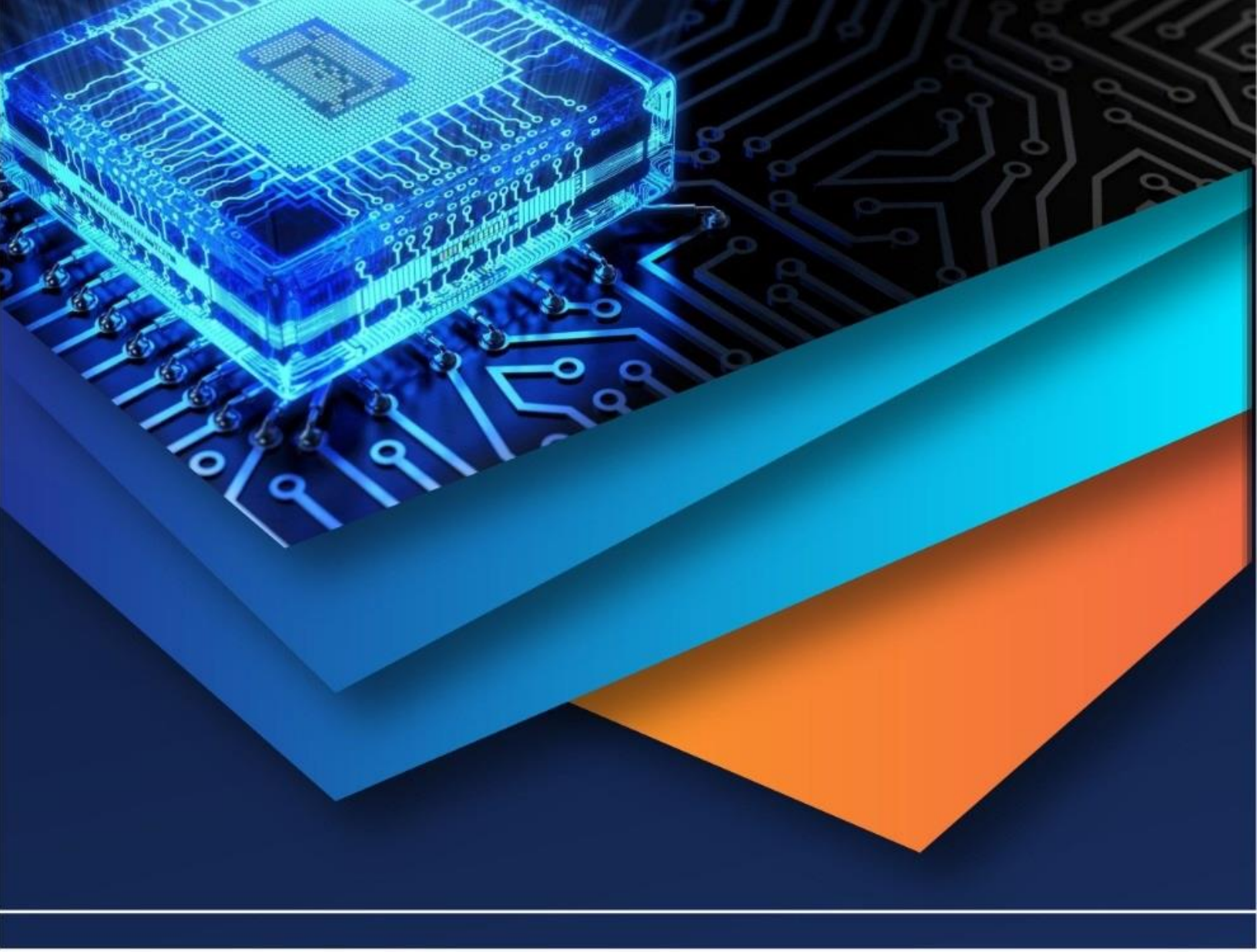
- FT -Functionality Traceability
- P-Passed

VI. CONCLUSION

The viability of the concept and demonstration of workability was done in this demonstration to ascertain the model's practical potential. Model prototyping was used as a valuable exercise to allow visualization of product functioning by providing an interactive model of end product design, navigation, and layout. This paper presents an evaluation of the model performance. The model evaluation checked the user Interface, APIs, Database, Security, communication, and other functionalities.

REFERENCES

[1] Chung, L., Nixon, B. A., Yu, E., & Mylopoulos, J. (2012). Non-functional requirements in software engineering (Vol. 5). Springer Science & Business Media.
[2] Khanom, N., & Miah, S. J. (2020). On-Cloud Motherhood Clinic: A Healthcare Management Solution for Rural Communities in Developing Countries. Pacific Asia Journal of the Association for Information Systems, 12(1), 3.
[3] Shas, H. (2108, 2 17). SIMFORM. Retrieved from Functional Testing: https://www.simform.com/blog/functional-testing/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY