Adeyeye Martins

Adeyeye079@gmail.com

Carbon Android app  
mobile APP testing

Carbon is a simple, mobile platform for Payments, Bills, Smart Investments, Credit Reports and Instant Loans to help cover unexpected expenses or urgent cash needs. You can apply for a Carbon loan 24 hours a day, 7 days a week with our quick application process that lets you know your status within minutes.

**Revision and Signoff Sheet**

**Document History**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Description of Change |
| 1.0 | 22nd Sept, 2019 | Martins Adeyeye | Creation of initial draft version |
| 1.01 | 23rd Sept, 2019 | Martins Adeyeye | Update of initial draft version |
|  |  |  |  |

Table of Contents

1. Introduction  
   1.1. Purpose  
   1.2. Project Overview  
   1.3. Audience
2. Test Plans   
   2.1. Test Objectives  
   2.2. Test Assumptions  
   2.3. Test Principles  
   2.4. Test Cases  
   2.5. Data Approach  
   2.6. Scope and Type of Testing  
    2.6.1. Exploratory Testing  
    2.6.2. System Integration Testing  
    2.6.3. Performance Testing  
    2.6.4. Security Testing  
    2.6.5. Regression Testing  
    2.6.6. User Acceptance Test (UAT)
3. Execution Strategy  
   3.1. Entry and Exit Criteria  
   3.2. Test cycles  
   3.3. Validation and Defect Management  
   3.4. Test Metrics  
   3.5. Defect tracking & Reporting
4. Test Management Process  
   4.1. Test Management Tool  
   4.2. Test Design Process  
   4.3. Test Execution Process  
   4.4. Risk Analysis: Test Risks and Mitigation Factors  
   4.5. Communication Plan and Team Roster  
   4.6. Role Expectation
5. Test Environment
6. Approvals
7. Test case Screen shoots
8. Summary Report

# INTRODUCTION

## Purpose

This test plan describes the testing approach and overall framework that will drive the testing of the Carbon Android Mobile Application. This document includes:

* **Test Strategy**: This outlines the rules governing test execution, including the givens of the project (e.g.: start / end dates, objectives, assumptions); description of the process to set up a valid test (e.g.: test scope, entry / exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).
* **Execution Strategy**: Describes how the test will be performed and clearly lay out the process which will be used to identify and report defects. Processes to also implement defects; resolution are outlined.
* **Test Management**: Processes to handle the logistics of the testing and all the events that occur during execution (e.g.: communications, escalation procedures, risk and mitigation, team roster) are described as well.

## Project Overview

The purpose of this application, known as Paylater now Carbon is for Payments, Bills, Smart Investments, Credit Reports and Instant Loans to help cover unexpected expenses or urgent cash needs.

## Audience

* **The Test Lead/Manager:** Organizes the test activities in the overall project schedule and condenses these activities into a document known as the Test Plan, reviews the document, tracks the performance of the test according to the tasks herein specified, ensures the document is approved by relevant stakeholders and is accountable for the results.
* **Test team members:** They will perform the tasks specified in this document, and provide input and recommendations as applicable.
* **The Technical Team:** Also known as the Development team, they will ensure that the test plan and deliverables are in line with the software/platform design, provide the test environment, bear the sole responsibility for its uptime & availability and follow agreed procedures for resolving defects identified during tests.
* **The stakeholders’ representatives and participants:** (individuals as identified by the PMO Leads) may take part in the UAT test to ensure the business is aligned with the results of the test.
* **Business analysts:** Will participate in approving the test documentation, provide inputs on functional changes (if any) and possibly also take part in User Acceptance Test exercises.

### System Integration Testing

**PURPOSE:**  This will be performed to verify the functions of the application with respect to how its systems work. It is carried out by user input into the application via the test cases/scripts derived from the solution specifications and validating the application’s responses/output against expected results.

**Scope:** This encompasses ONLY the functional elements of the Carbon Android App specifications as outlined below:

|  |  |
| --- | --- |
| 1 | Sign In |
| 2 | Airtime Recharge |
| 3 | Fund Wallet |
| 4 | Wallet Transaction Filter |

**TESTERS**: Test team/QA.

**METHOD**: The test will be performed per the test cases/scripts derived from the solution specifications, some of which will be automated while others are manually executed.

**TIMING**: After Exploratory tests are successfully completed.

**Mobile Application Test Scenarios & Test Cases**

**Test Scenario #1: *Check the Registration Functionality of the app***

**Test case**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | | Car\_001 | Test Case Description | | Test the App Functionality of Carbon Android app | | | | | |
| Created By | | Martins | Reviewed By | | Carbon QA lead | | Version | | 0.1 | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Tester's Name | | Martins | Date Tested | | 21-Sep -2019 | | Test Case (Pass/Fail/Not Executed) | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| S # | Prerequisites app Application: | | |  | S # | Test Data app Application : | | | | |
| 1 | Access to internet | | |  | 1 | Tap the icon button of app | | | | |
| 2 | App downloaded in the mobile | | |  | 2 | Tap the next button in the app | | | | |
| 3 |  | | |  | 3 |  | | | | |
| 4 |  | | |  | 4 |  | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
| Test Scenario | Verify on Clicking on Registration button | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Step # | Step Details | | Expected Results | | Actual Results | | | Pass / Fail / Not executed / Suspended | | |
|
| 1 | Navigate to Registration button of the app | | App page should open | | As Expected | | | Pass | | |
| 2 | Enter the first and surname in the empty text filed | | Fields should take the values | | As expected | | | Pass | | |
| 3 | Enter the email address in text field. | | Enter the values and values enter successfully | | As expected | | | Pass | | |
| 4 | Tap the date of birth button filed and enter the date of birth | | Fields should take the values | | AS expected | | | Pass | | |
| 5 | Enter the phone number in the test filed | | Only numeric values taken | | As expected | | | Pass | | |
| 6 | Accept the term with tapping the agree button | | Pop up menu accept the function | | The account register successfully | | | Fail  account already exist with the same number | | |

**Test Scenarios #2: *Check the Sign in Functionality of the App***1. Check that user can sign in with valid phone number and valid PIN  
2. Check that user is not able to sign in using unregistered phone number and Valid PIN  
3. Check that user is not able to sign in using registered Phone number with Invalid/wrong PIN  
4. Check that user is not able to sign in using incorrect phone number format  
5. Check that user is not able to sign in using only phone number  
6. Check that user is not able to sign in using only pin  
7. Check that user is not able to sign in using empty phone number field  
8. Check that user is not able to sign in using empty PIN field  
9. Check that user is not able to sign in using registered phone number and old PIN after changing the PIN  
10. Check that user can retrieve their PIN using the forgot PIN link  
11. Check that user is not able to login without Internet connect (Wi-Fi or Data connection).  
12. Check that user is not able to login when data is low or exhausted  
13. Check that the user account is deactivated temporarily or block after 3-5 wrong PIN is entered  
14. Check that the user is sent a notification when the account is blocked  
15. Verify if user can delete a digits one by one using backspace key  
16. Verify if the sign in credential entered is retain after an incoming call  
17. Verify that the password is in encrypted form when entered  
18. Verify the password can be copy-pasted  
19. Verify that encrypted characters in “Password” field should not allow deciphering if copied  
20. Verify that User should be able to login with the new PIN after changing the PIN

**Test case No: 2**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Scenario ID | | Car\_002 | Test Case Description | | Test the Sign in Functionality of Carbon App | | | | | |
| Created By | | Martins | Reviewed By | | Carbon QA lead | | Version | | 0.1 | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Tester's Name | | Martins | Date Tested | | 21 -Sep -2019 | | Test Case (Pass/Fail/Not Executed) | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| S # | Prerequisites mobile app Application: | | |  | S # | Test Data mobile app Application: | | | | |
| 1 | Access to internet | | |  | 1 | Tap on sign in button | | | | |
| 2 | App is download in the phone | | |  | 2 |  | | | | |
| 3 |  | | |  | 3 |  | | | | |
| 4 |  | | |  | 4 |  | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
| Test Case ID- 001 | Verify that users can Successfully Sign in with Valid credential | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Step # | Step Details | | Expected Results | | Actual Results | | | Pass / Fail / Not executed / Suspended | | |
|
| 1 | Tap the carbon app icon in the phone | | App should open | | As Expected | | | Pass | | |
| 2 | Tap on the button “Sign in “Button | | Page should be load with text filed | | As Expected | | | Pass | | |
| 3 | Enter the given phone number in the text filed | | Filed should take the numeric value | | As expected | | | Pass | | |
| 4 | Enter the given pin in the text filed | | Filed should take the values | | Page responding perfectly | | | Pass | | |
| 5 | Tap on the sign in ” button | | App should be load | | Page opens successfully | | | pass | | |
| 6 | Enter the given verification code | | App sign in perfectly | | As expected | | | Pass | | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID- 002 | Verify that users cannot Sign in using invalid Phone Number | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Step # | Step Details | | Expected Results | | Actual Results | | | Pass / Fail / Not executed / Suspended | | |
|
| 1 | Tap the carbon app icon in the phone | | App should open | | As Expected | | | Pass | | |
| 2 | Tap on the button “Sign in “Button | | Page should be load with text filed | | As Expected | | | Pass | | |
| 3 | Enter unregistered phone number in the text field | | Field should take the numeric value | | As expected | | | Pass | | |
| 4 | Enter the given pin in the text filed | | Field should take the values | | Page responding perfectly | | | Pass | | |
| 5 | Tap on the sign in ” button | | App should be load | | Page displays expected Error message | | | pass | | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID- 003 | Verify that User cannot Sign in using invalid PIN | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Step # | Step Details | | Expected Results | | Actual Results | | | Pass / Fail / Not executed / Suspended | | |
|
| 1 | Tap the carbon app icon in the phone | | App should open | | As Expected | | | Pass | | |
| 2 | Tap on the button “Sign in “Button | | Page should be load with text filed | | As Expected | | | Pass | | |
| 3 | Enter the given phone number in the text filed | | Field should take the numeric value | | As expected | | | Pass | | |
| 4 | Enter the given pin in the text filed | | Field should take the values | | Page responding perfectly | | | Pass | | |
| 5 | Tap on the sign in ” button | | App should be load | | Page displays expected Error message | | | pass | | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID- 004 | Verify that User cannot Sign in using empty fields | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Step # | Step Details | | Expected Results | | Actual Results | | | Pass / Fail / Not executed / Suspended | | |
|
| 1 | Tap the carbon app icon in the phone | | App should open | | As Expected | | | Pass | | |
| 2 | Tap on the button “Sign in “Button | | Page should be load with text filed | | As Expected | | | Pass | | |
| 3 | Enter the given phone number in the text filed | | Filed should take the numeric value | | As expected | | | Pass | | |
| 4 | Enter the given pin in the text filed | | Filed should take the values | | Page responding perfectly | | | Pass | | |
| 5 | Tap on the sign in ” button | | App should be load | | Page displays expected Error message | | | pass | | |

**Test Scenarios #3: *Test the Buy Airtime functionality in Carbon App***

1. Check that user cannot enter alphanumeric input as phone number
2. Check that user cannot input a phone number less than or greater 11 digits
3. Check that user cannot enter a negative value for airtime amount
4. Check that user cannot enter alphabetical value for airtime amount
5. Check that user cannot enter alphanumeric value for airtime amount
6. Check that the airtime amount doesn't take special characters
7. Check that user cannot enter a value less than 100 naira for airtime amount
8. Check that user cannot enter a value greater than 100,000 for airtime amount
9. Check that the phone number filters the corresponding network
10. Verify if user is able to pay using the new Master Card added
11. Verify if user is able to pay using the new Visa Card added
12. Verify if user is able to pay using the new Debit Card added
13. verify if user is able to pay using Wallet
14. For paying with wallet,
15. Phone number should be matched against network provider before user gets to payment
16. Check that user cannot purchase airtime higher than the amount in wallet
17. For paying with bank, entering new bank details:
18. Check that bank account is exactly 10 digits
19. Check that only numeric digits are inputted as bank account number
20. Check that account number matches selected bank
21. For paying with debit card,
22. Check that card number is exactly 16 digits
23. Check that user cannot enter non-numeric value for card number
24. Check that the expiry date is a valid date
25. Check that user cannot enter value less than 01 and greater than 12 for month
26. Check that user cannot enter a year lesser than the current year
27. Check that user cannot enter more than 3 digits for CVV
28. Check that user can only enter numerical values for CVV

**Test case No: 3**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Scenario | | Car\_003 | Test Case Description | | Test the Buy Airtime functionality in Carbon App | | | | | |
| Created By | | Martins | Reviewed By | | Carbon QA lead | | Version | | 0.1 | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Tester's Name | | Martins | Date Tested | | 12-SEP -2019 | | Test Case (Pass/Fail/Not Executed) | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| S # | Prerequisites mobile app Application: | | |  | S # | Test Data mobile app Application: | | | | |
| 1 | Access to internet | | |  | 1 | Tap on buy air time button on the home screen of the app | | | | |
| 2 | App installed in the mobile phone | | |  | 2 |  | | | | |
| 3 |  | | |  | 3 |  | | | | |
| 4 |  | | |  | 4 |  | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
| Test Scenario | Verify on Tapping the Buy Air time button | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Step # | Step Details | | Expected Results | | Actual Results | | | Pass / Fail / Not executed / Suspended | | |
|
| 1 | Login in to the Android app by entering the phone number and pin code | | app should open | | As Expected | | | Pass | | |
| 2 | Tap on the “Buy Air Time “Button | | Page should be load in next tab and show information | | As Expected | | | Pass | | |
| 3 | Select the number from the already given numbers ” | | Page should be load | | As expected | | | Pass | | |
| 4 | Enter the values in the text filed | | Numeric values should be taken | | As expected | | | Pass | | |
| 5 | Select the mobile network **MTN** from the given networks | | Action performed successfully | | As expected | | | pass | | |
| 6 | Chose the payment method pay as Atm card | | Action performed successfully | | As expected | | | Pass | | |
| 7 | Tap on the “Securely pay ”button | | Action performed successfully | | As expected | | | Pass | | |
| 8 | Enter the Given PIN | | Air time purchase successful | | As expected | | | pass | | |
| 9 | Select the mobile network **AIRTEL** from the given networks | | Only allowed the MTN user | | Not taken values only taken the values of MTN | | | FAIL | | |

**Test Scenario #4: *Test the functionality of Fund the wallet in the Dashboard in carbon App***

1. Verify that user can successfully fund wallet using bank transfer menu
2. Verify that user can successfully fund wallet using Master card
3. Verify that user can successfully fund wallet using VISA card
4. Verify that user can successfully fund wallet using Verve Card
5. Verify that user can successfully fund wallet using Debit card / others
6. Check that user cannot enter a non-numeric value
7. Check that user cannot enter a negative value
8. Check that funding amount is greater than or equal to N500
9. Check that user cannot copy an alphanumeric or decimal number into the value field

**TESTCASE NO 4**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Scenario ID | | Car\_004 | Test Case Description | | Test the functionality ofFund the wallet in the Dashboard in carbon App  / | | | | | |
| Created By | | Martins | Reviewed By | | Carbon QA lead | | Version | | 0.1 | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Tester's Name | | Martins | Date Tested | | 12-SEP -2019 | | Test Case (Pass/Fail/Not Executed) | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| S # | Prerequisites Mobile App Application: | | |  | S # | Test Data Mobile App Application: | | | | |
| 1 | Access to Internet | | |  | 1 | Tap on the fund wallet button | | | | |
| 2 | Carbon app installed in the mobile phone | | |  | 2 |  | | | | |
| 3 |  | | |  | 3 |  | | | | |
| 4 |  | | |  | 4 |  | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
| Test Scenario | Verify on Tapping the Fund wallet Functionality | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Step # | Step Details | | Expected Results | | Actual Results | | | Pass / Fail / Not executed / Suspended | | |
|
| 1 | Login in to the Android app by entering the phone number and pin code | | App should open | | As Expected | | | Pass | | |
| 2 | Tap on the “Fund wallet “Button | | Page should be load and show information | | As Expected | | | Pass | | |
| 3 | Chose the Fund with ATM CARD | | Page should be load in next tab and show information | | As expected | | | Pass | | |
| 4 | Enter the amount of 1000 for fund in the test filed | | numeric values taken perfectly and show the next page | | As expected | | | Pass | | |
| 5 | Chose the payment method pay vis ATM CARD | | Page should be load in next tab and show information | | As expected | | | Pass | | |
| 6 | Enter the given PIN in the text filed | | Wallet top –up don successfully | | As expected | | | Pass | | |

**Test Scenario #5:** ***Test the functionality of transactions with filters***

1. Check that all types displays all transactions  
2. Check that wallet transactions is filtered and displays only   
3. Check that fund transfers only displays transfer transactions  
4. Check that airtime displays only airtime purchases  
5. Check that bill payment only displays the corresponding transactions  
6. Check that loan only displays its type of transactions

**TEST CASE NO 5**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | | CAR\_005 | Test Case Description | | Test the functionality of transactions with filters | | | | | |
| Created By | | Martins | Reviewed By | | Carbon QA lead | | Version | | 0.1 | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Tester's Name | | Martins | Date Tested | | 21-SEP -2019 | | Test Case (Pass/Fail/Not Executed) | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| S # | Prerequisites Mobile App Application: | | |  | S # | Test Data Mobile App Application: | | | | |
| 1 | Access to internet | | |  | 1 | Tap on the Transaction button | | | | |
| 2 | Carbon app installed in the mobile phone | | |  | 2 |  | | | | |
| 3 |  | | |  | 3 |  | | | | |
| 4 |  | | |  | 4 |  | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
| Test Scenario | Verify on Tapping the Transaction button | | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Step # | Step Details | | Expected Results | | Actual Results | | | Pass / Fail / Not executed / Suspended | | |
|
| 1 | Login in to the Android app by entering the phone number and pin code | | App should open | | As Expected | | | Pass | | |
| 2 | Tap on Transaction button | | Page should be load in same tab and show information | | As Expected | | | Pass | | |
| 3 | Select the filter with “WALLET” OPTION | | Page should be load in next tab and show information | | As expected | | | Pass | | |
| 4 | Select the transition wallet amount | | Pop up menu opened and shoed information | | As expected | | | Pass | | |

**Fundamental Functional Test Scenarios for a Mobile Application**

1. To validate whether all the required mandatory fields are working as required.

2. To validate that the mandatory fields are displayed in the screen in a distinctive way than the non-mandatory fields.

3. To validate whether the application works as per as requirement whenever the application starts/stops.

4. To validate whether the application goes into minimized mode whenever there is an incoming phone call. In order to validate the same we need to use a second phone, to call the device.

5. To validate whether the phone is able to store, process and receive SMS whenever the app is running. In order to validate the same we need to use a second phone to send sms to the device which is being tested and where the application under test is currently running.

6. To validate that the device is able to perform required multitasking requirements whenever it is necessary to do so.

7. To validate that the application allows necessary social network options such as sharing, posting and navigation etc.

8. To validate that the application supports any payment gateway transaction such as Visa, Mastercard, Wallet etc as required by the application.

9. To validate that the page scrolling scenarios are being enabled in the application as necessary.

10. To validate that the navigation between relevant modules in the application are as per the requirement.

11. To validate that the truncation errors are absolutely to an affordable limit.

12. To validate that the user receives an appropriate error message like “Network error. Please try after some time” whenever there is any network error.

13. To validate that the installed application enables other applications to perform satisfactorily, and it does not eat into the memory of the other applications.

14. To validate that the application resumes at the last operation in case of a hard reboot or system crash.

15. To validate whether the installation of the application can be done smoothly provided the user has the necessary resources and it does not lead to any significant errors.

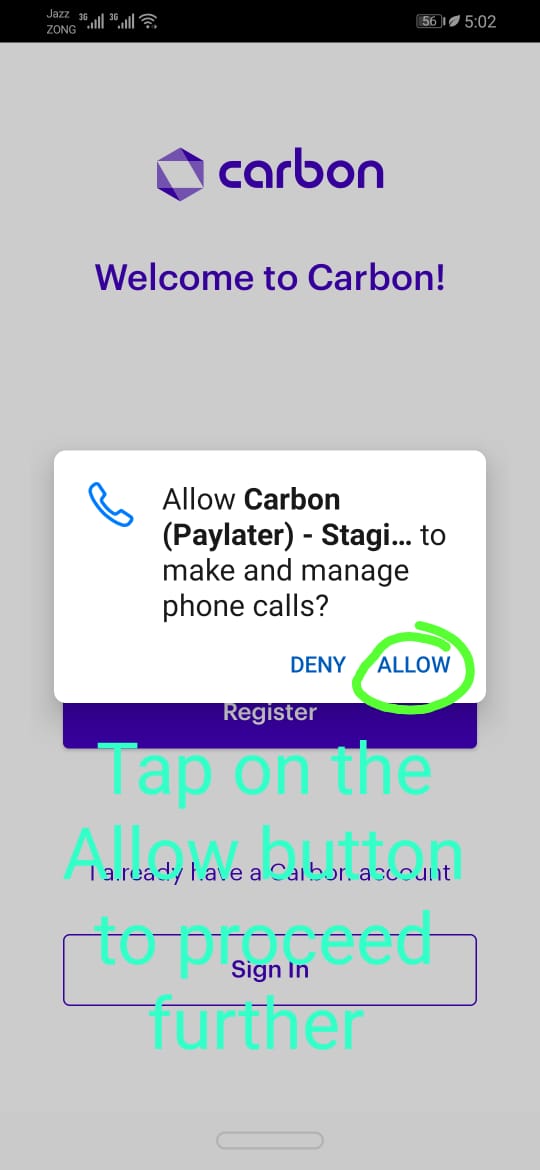
16. To validate that the application performs auto start facility according to the requirements.

17. To validate whether the application performs according to the requirement in all versions of Mobile that is 2g, 3g and 4g.

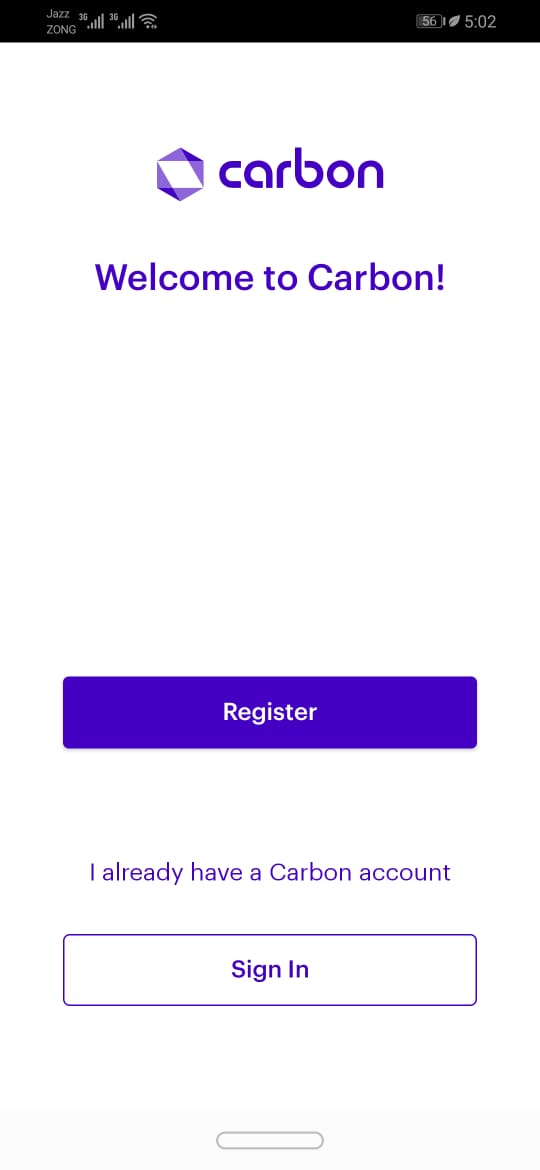
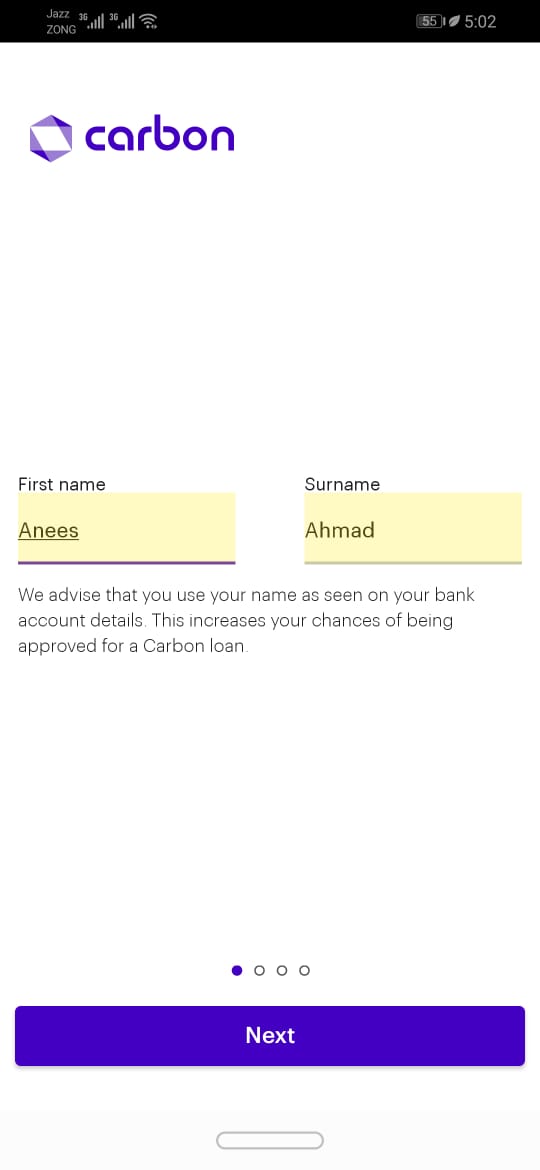
18. To validate whether the application provides an available user guide for those who are not familiar to the app

**Testing Screen SHOOTS**

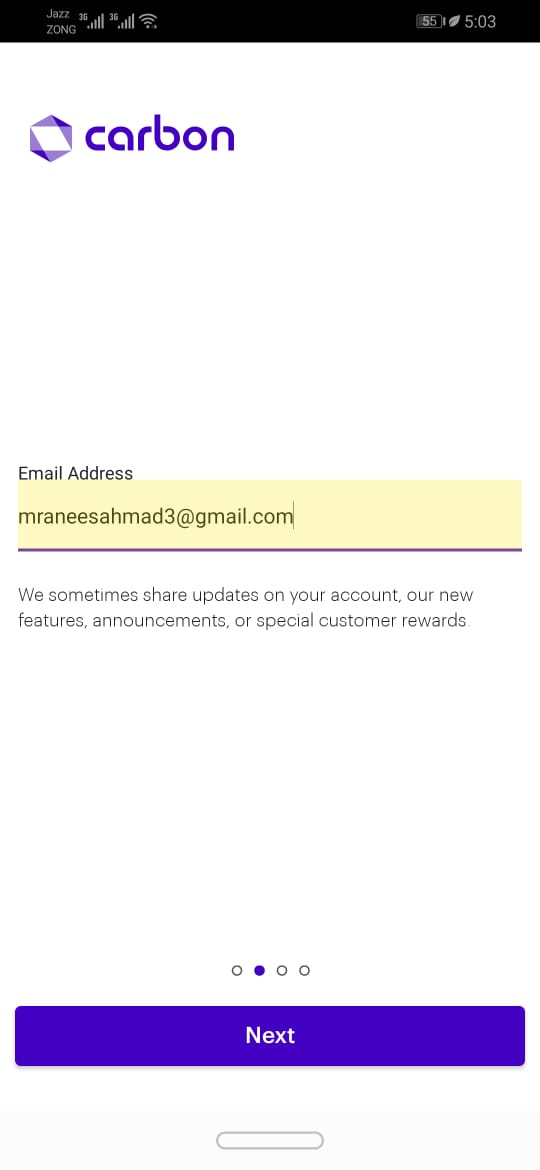
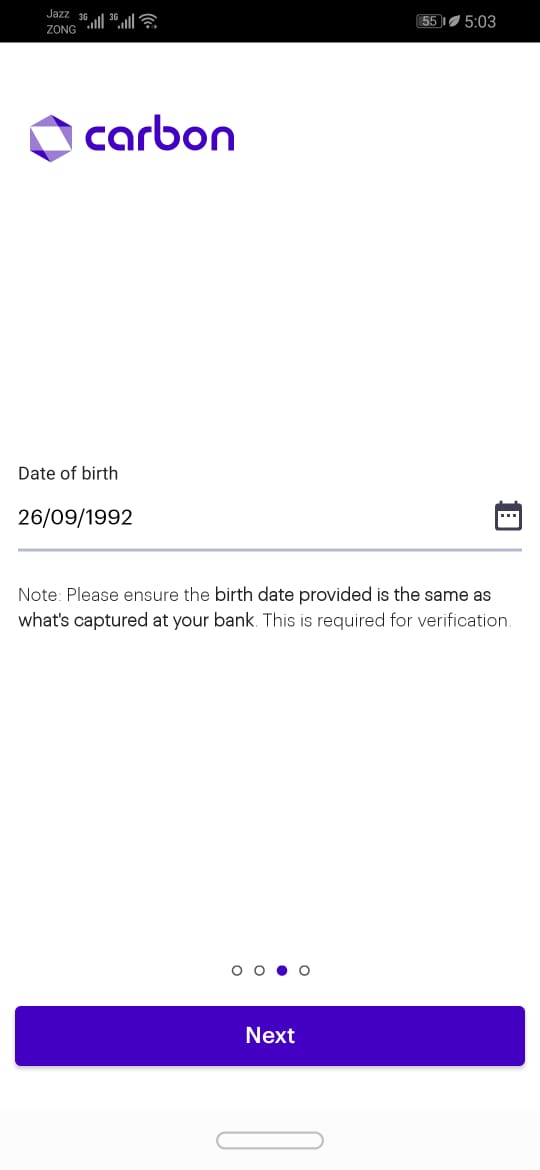
**Test case No1: Screen shoot**

****

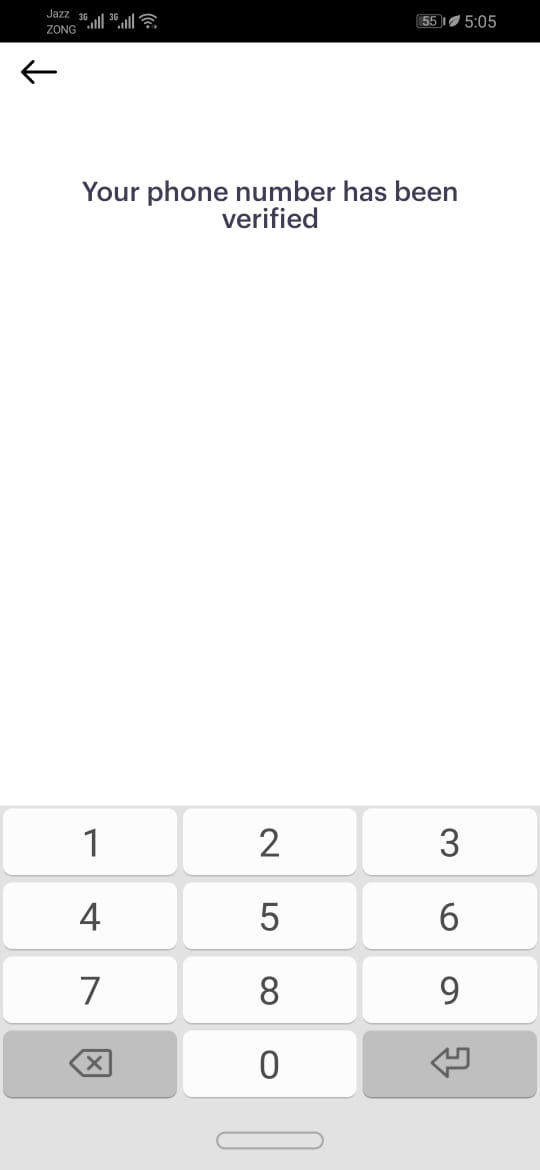
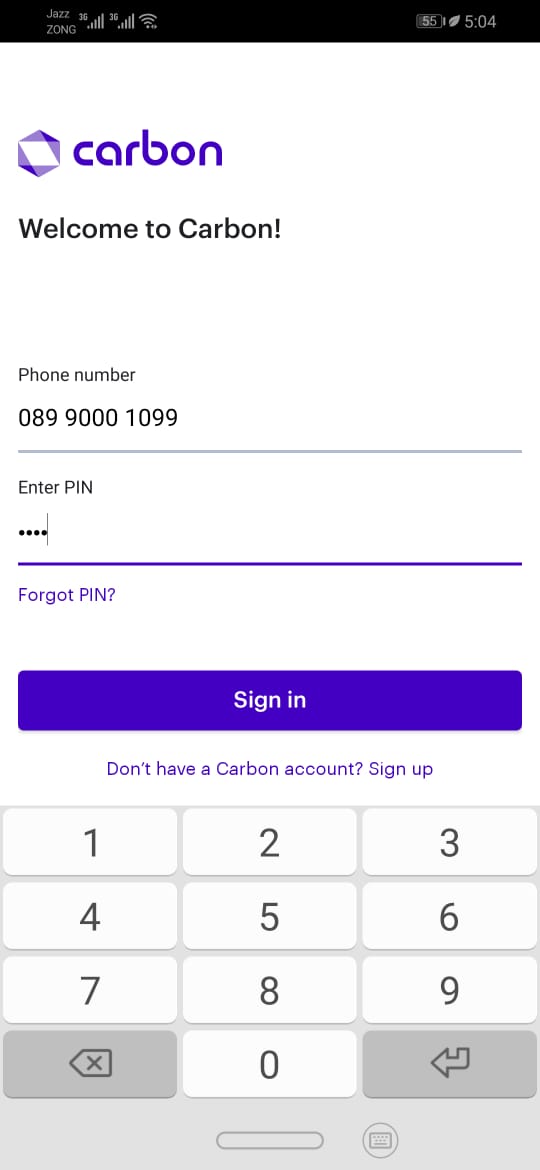
**Test case No1: Screen shoot**

****

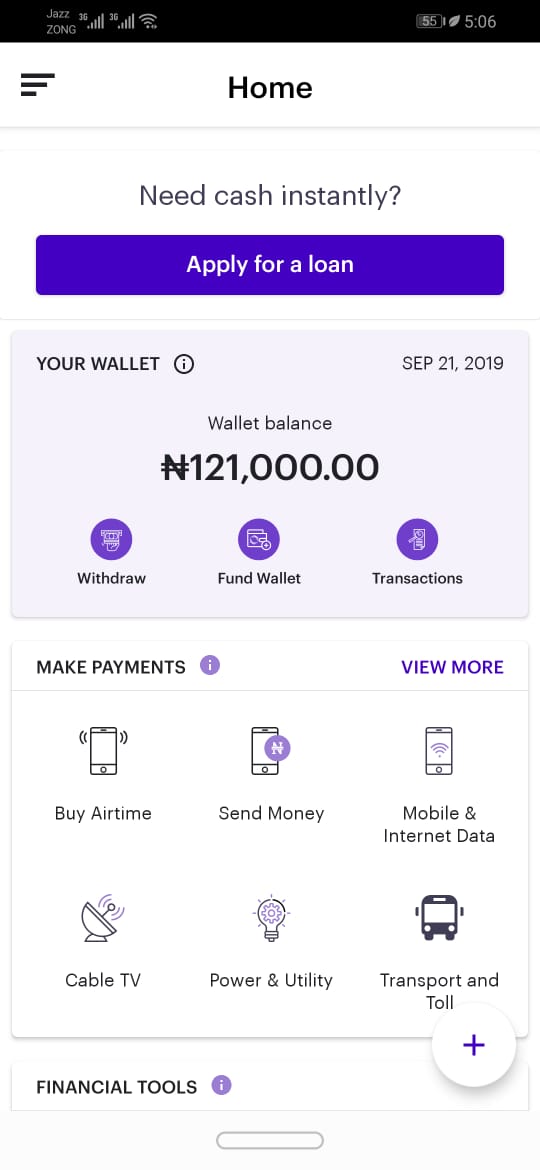
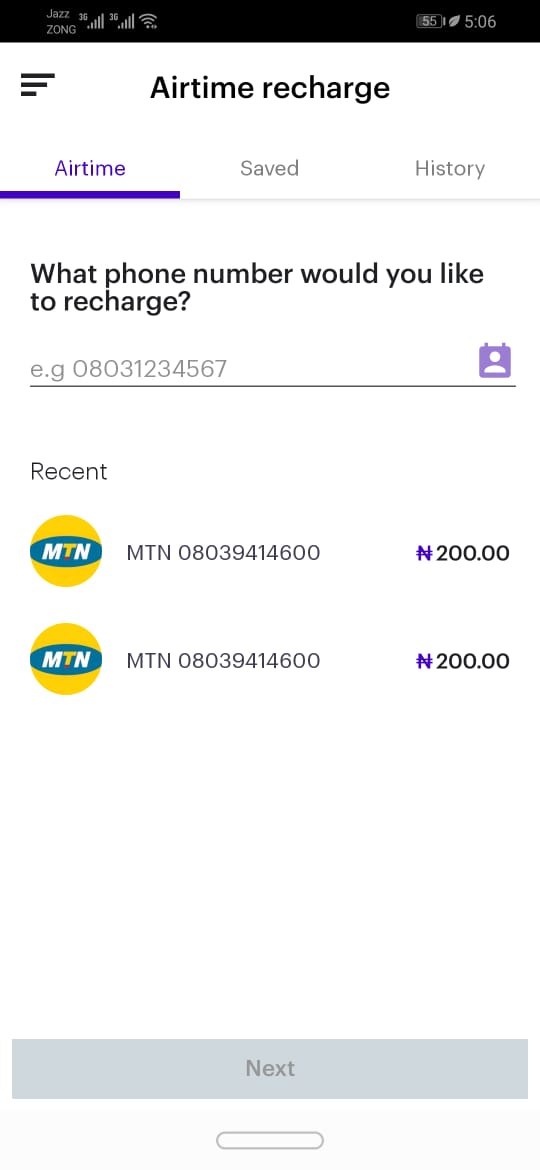
**Test case No1: Screen shoot**

****

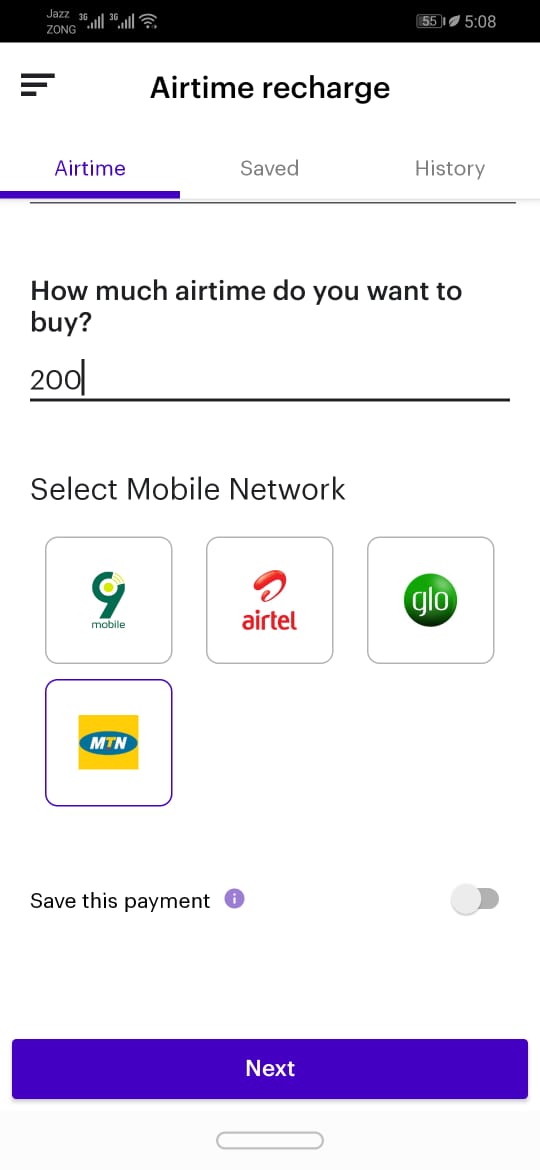
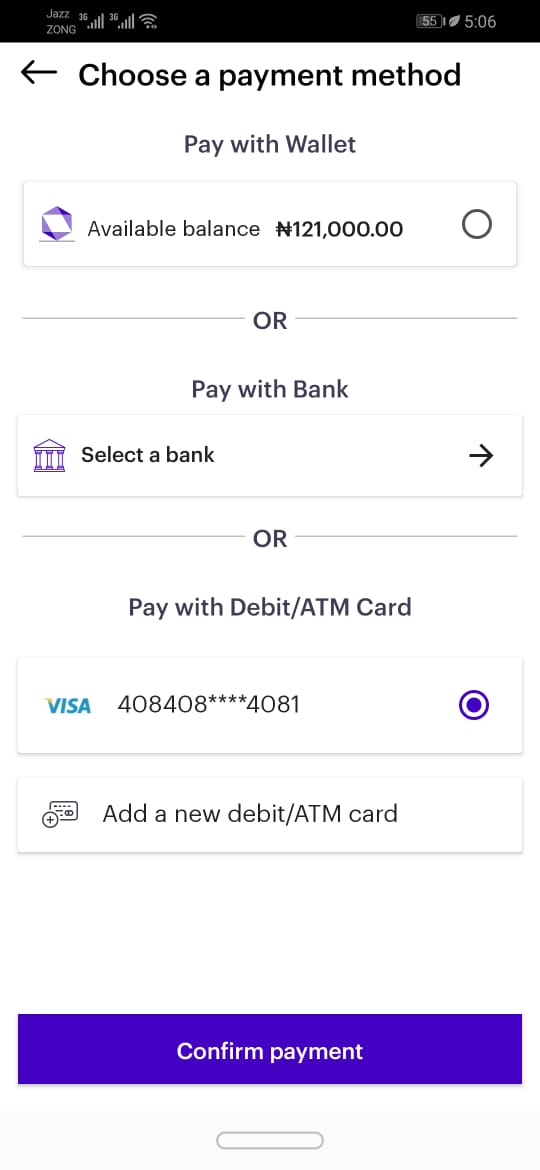
**Test case No2: Screen shoot**

****

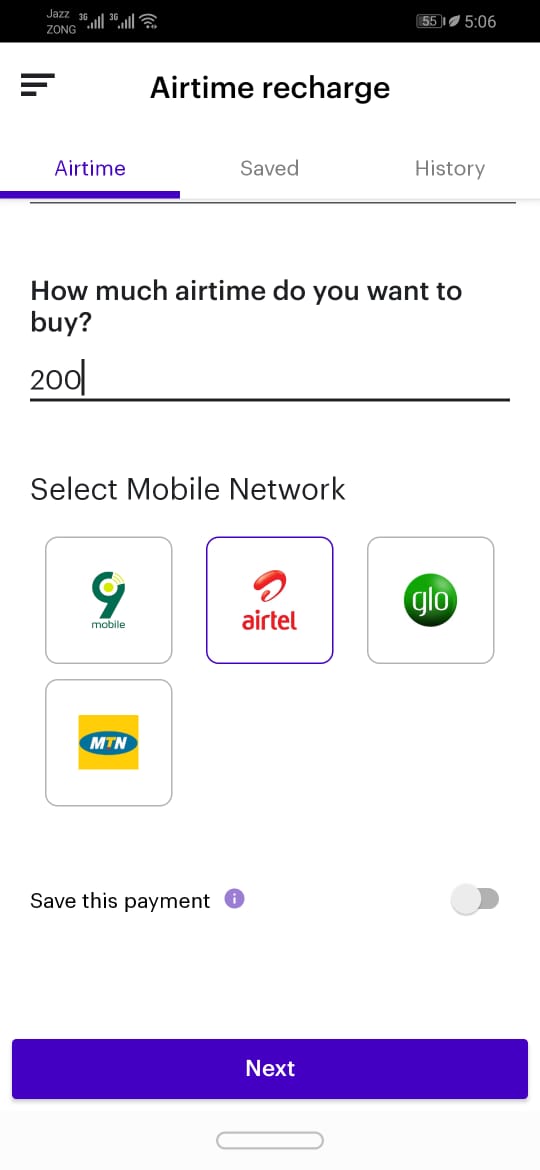
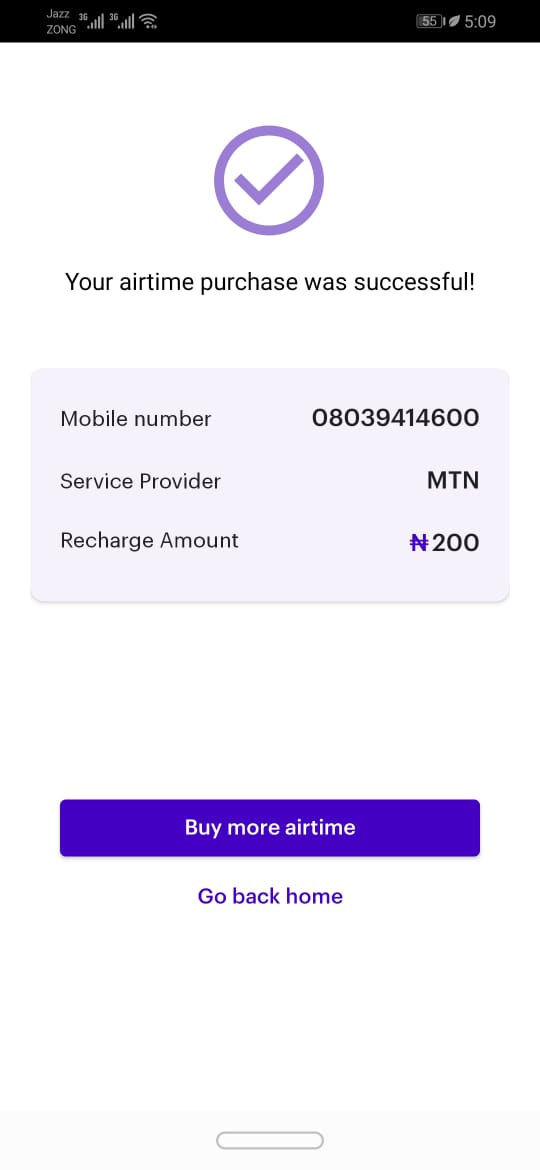
**Test case No3: Screen shoot**

****

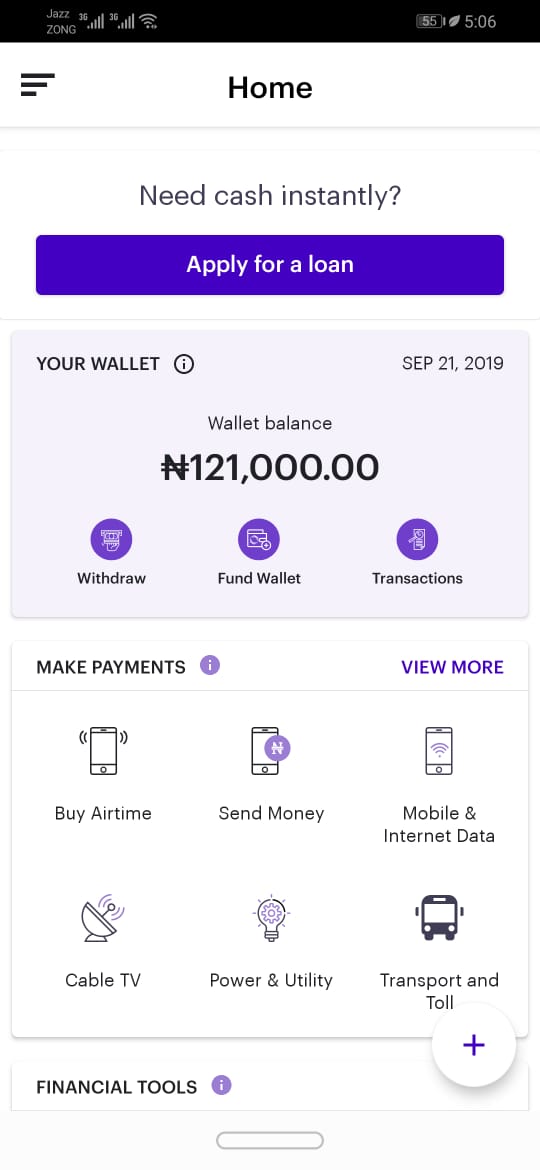
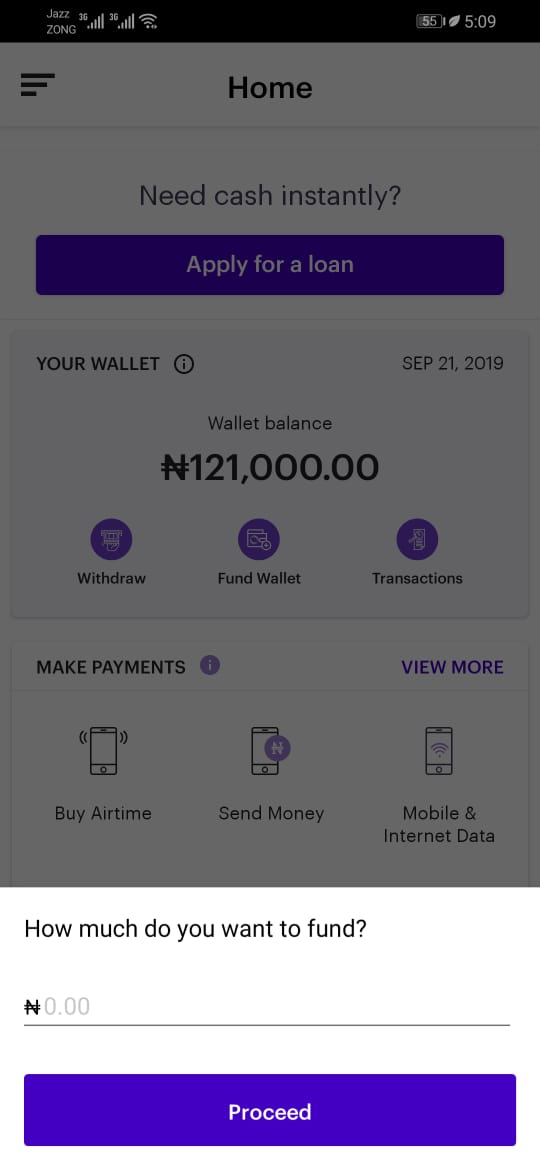
**Test case No3: Screen shoot**

****

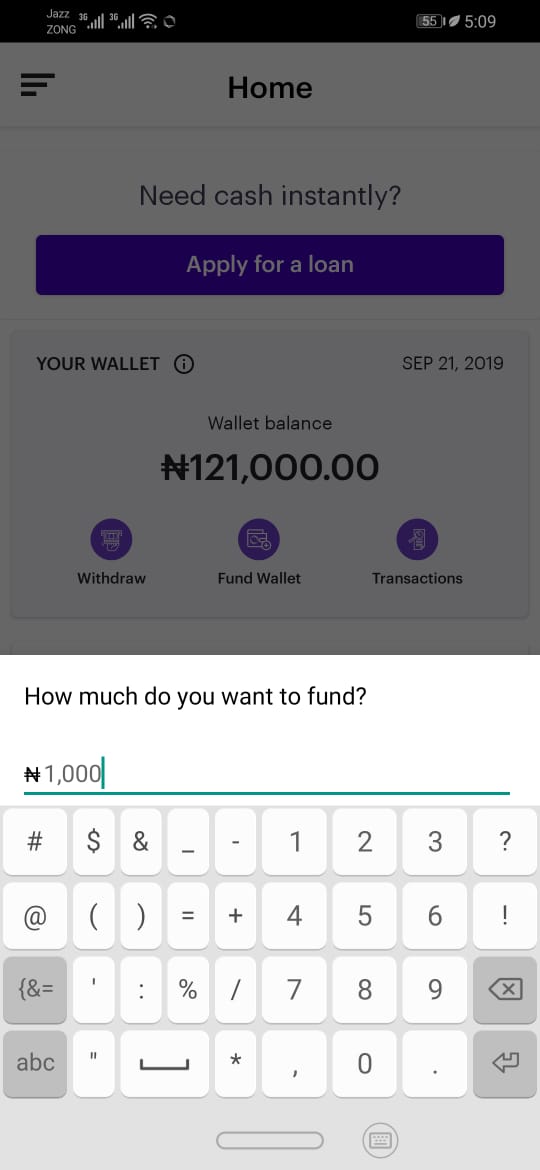
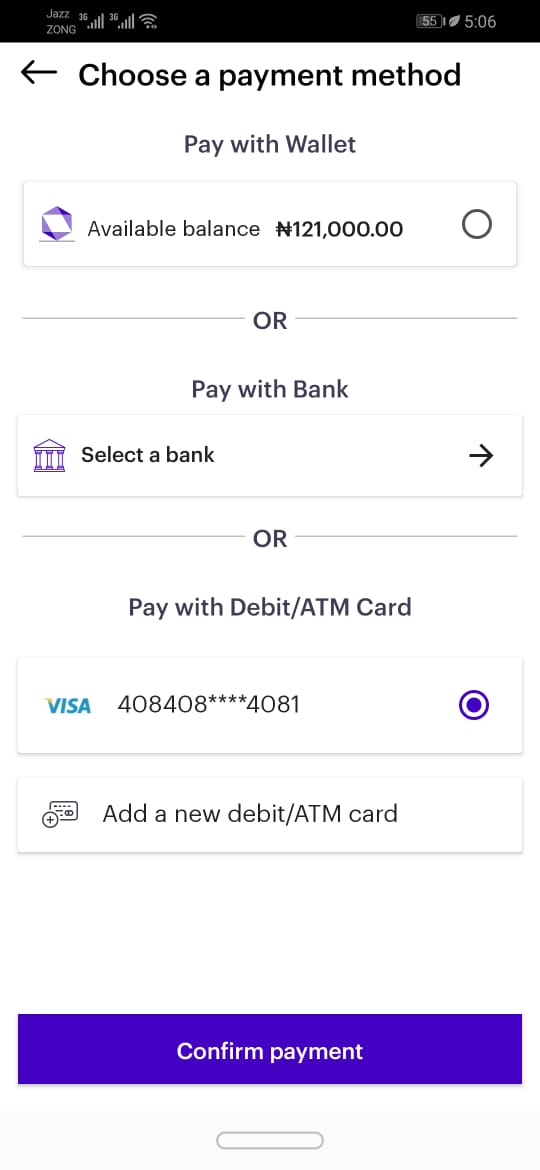
**Test case No3: Screen shoot**

****

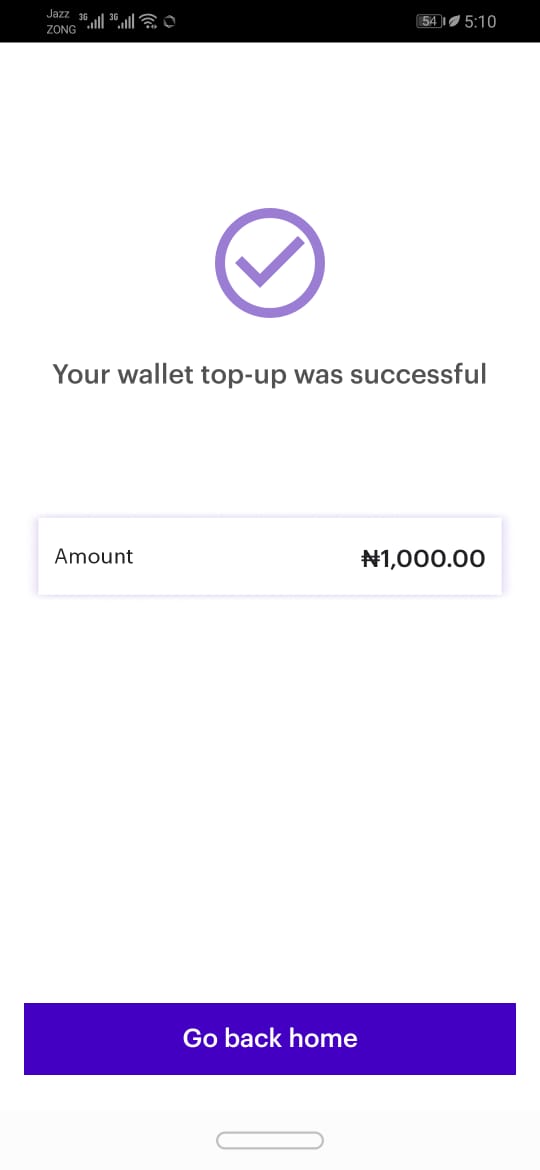
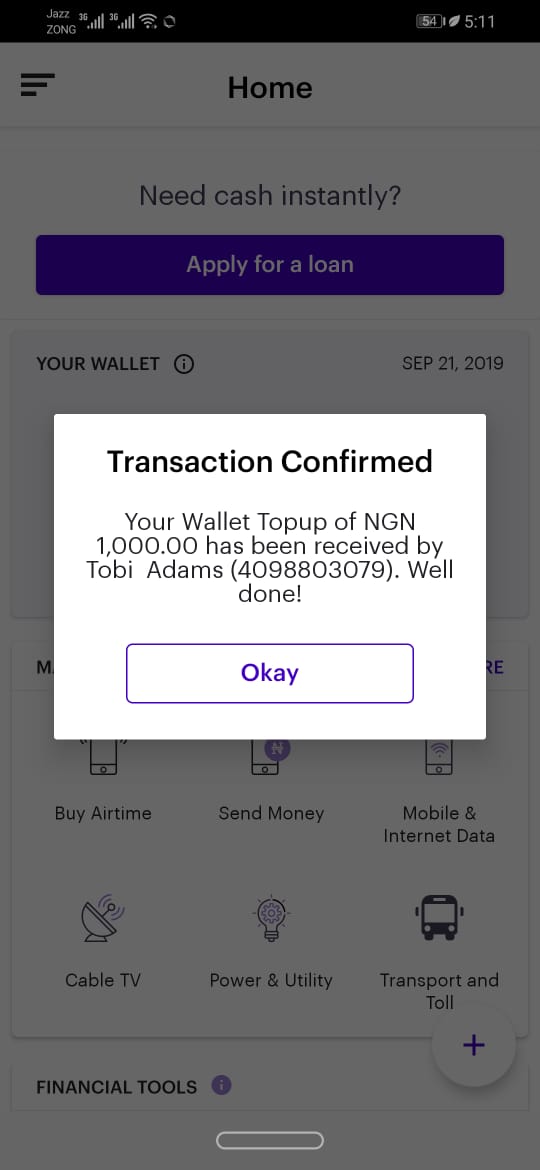
**Test case No4: Screen shoot**

****

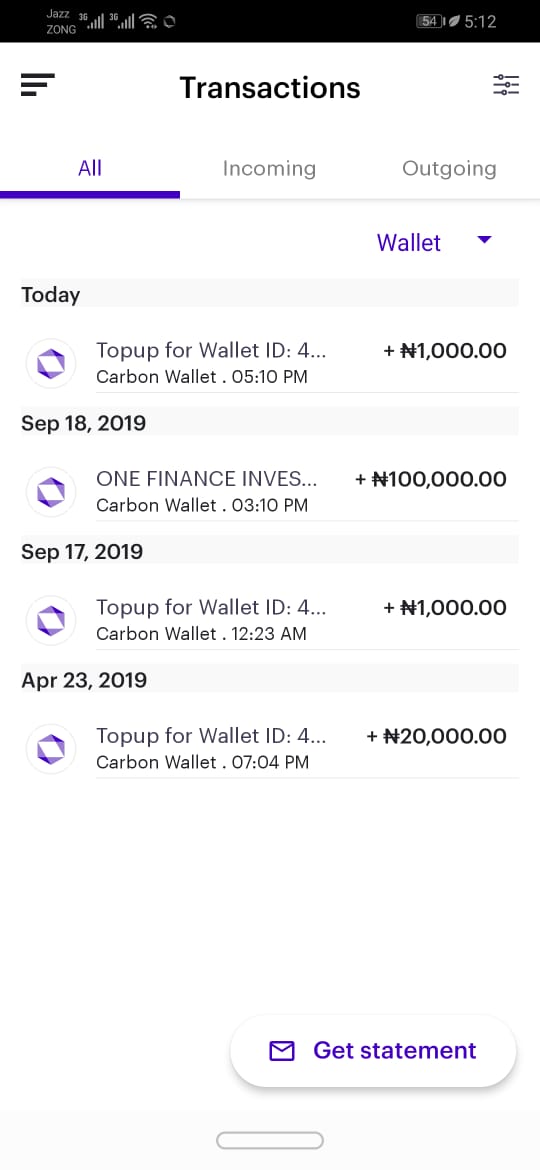
**Test case No4: Screen shoot**

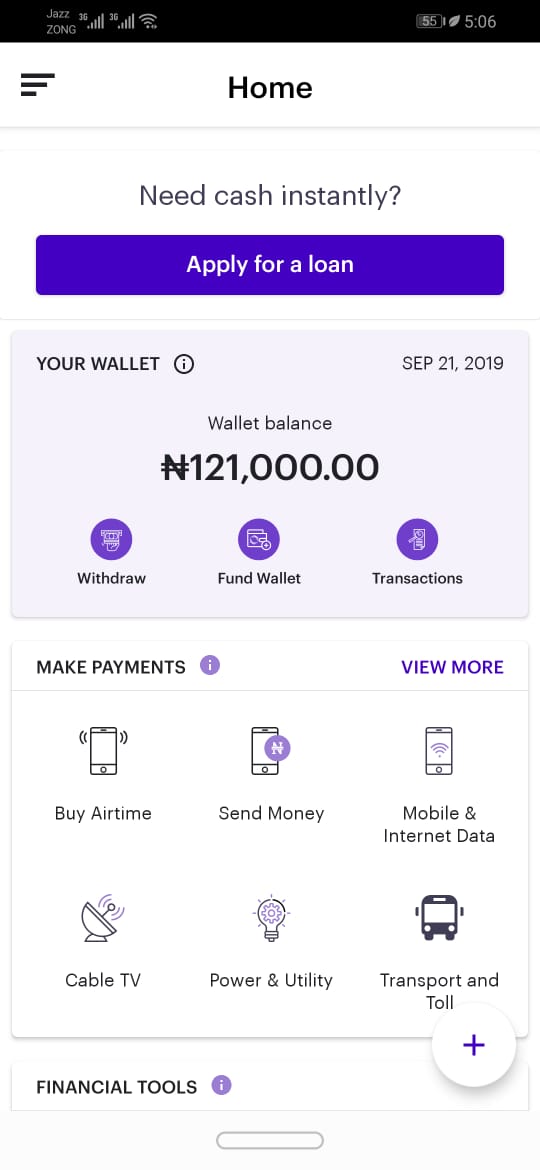
****

**Test case No4: Screen shoot**

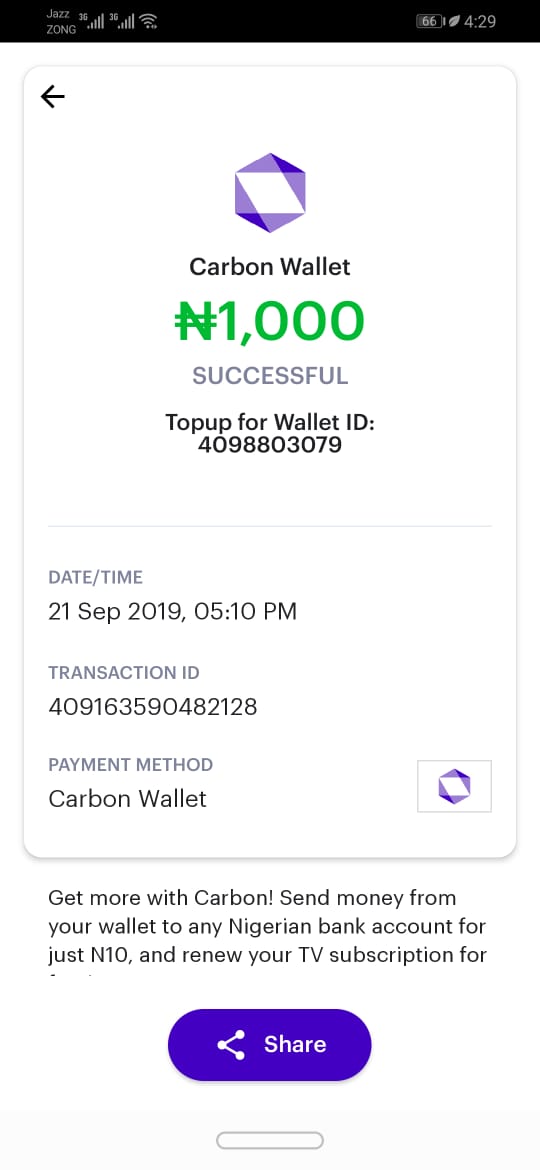
****

**Test case No5: Screen shoot**

****

****

**Test case No5: Screen shoot**

****

**Summary Report**

The Test Cases based on the given information https**:** **Carbon Android App**

* The working of App is perfect.
* The-Loading time of APP perfect and not facing any time issues.
* THE testing OF APP is done on android version PIE 9.0
* The testing data is based on 21 September 2019