

Pizza Price Calculator Test Report

Contents

[1. Overview 3](#_Toc104129577)

[2. Test plan 3](#_Toc104129578)

[3. Test method 5](#_Toc104129579)

[4. Test results 6](#_Toc104129580)

[5. Analysis and conclusions 10](#_Toc104129581)

[6. Software approval 11](#_Toc104129582)

[7. Document authority 11](#_Toc104129583)

1. Overview

This document is the Software Test Report of the testing phase of the PizzasOnly price calculator software development project.

The test plan is to ensure the price calculator code works correctly, which include data and actions to test all code, as well as the expected outcomes. Then the test plan is followed to test the code using suitable testing and debugging tools, the test results are recorded in the test plan, including the actual outcomes and screenshots or log files of the test. The code execution is traced to detect and correct any errors in the code, screenshots of the files within the debugger and the process of tracing the code execution, are also included in this document.

Part 4: Test and debug code

You must now verify the price calculator application works correctly by testing and debugging the code you developed in Part 3.

1. Design a test plan to ensure your price calculator code works correctly. Your test plan should include data and actions to test all code, as well as the expected outcomes. Refer to the provided scenario and specifications to ensure that you test your code meets all the requirements.
2. Test plan

The following table is a plan for testing the price calculator.

Table 1 – Test plan

| Test ID | Test name | Summary | Expected result |
| --- | --- | --- | --- |
| 01 | Negative or 0 input (Boundary conditions) | The customers enter a negative umber of pizzas. For example, -10, -1. | A message must be displayed to inform the customers that they have entered an invalid number. The cost must not be calculated. |
| 02 | Invalid data | A customer enters a character or a word intentionally or otherwise, for example A, Ten, four. | A message must be displayed to inform the customers that they have entered an invalid number. The cost must not be calculated at this point. |
| 03 | Out of bound input (Boundary conditions) | A customer wants to order more than the maximum number of pizzas allowed, for example PizzasOnly can deliver a maximum of 15 pizzas but a customer needs to order 20 pizzas. | A message must be displayed to inform that the PizzasOnly cannot accept more than a maximum number of pizzas. The cost must not be calculated. |
| 04 | No data | The use tries to place order without entering any input | A message must be displayed to inform that the customer must enter the number of pizzas. The cost must not be calculated. |
| 05 | Promotion period | The promotion can be applied only during a promotional period. | A message must be displayed to inform that It is not promotional period if the customer want to order when it is not promotional period. |
| 06 | One large pizza | One large pizza will cost $6.45. | The price for one pizzas is calculated and the cost for one large pizzas is displayed. |
| 07 | Two large pizzas (Valid data) | Two large pizzas will cost $12.00. | The price for two pizzas is calculated and the cost for two large pizzas is displayed. |
| 08 | Three large pizzas (Valid data) | Three large pizzas will cost $14.00. | The price for three pizzas is calculated and the cost for three large pizzas is displayed. |
| 09 | Four or more pizzas (Valid data) | Four or more pizzas will use a combination of the above prices to ensure the best  price for the customer. For example, the best price for five pizzas would be two  pizzas ($12.00) + three pizzas ($14.00). | The best price for entered number of pizzas is calculated and the cost is displayed. |

1. Test method

The test will be performed on Monday, 23rd of May 2022.

ICT technician Basir Sultani and the software development team leader, Christina will be involved in testing.

The Visual studio code (IDE), VS live server, /VS code debugger, chrome browser will be used to perform the tests.

1. Test results

The Test ID in this table is correspondent to (as listed in the Test planning), it includes the actual results from the testing, screenshots of the testing and the relevant comments or information, and the outcome of the test.

1. Follow your test plan to test your code using suitable testing and debugging tools. Record the test results in the test plan, including the actual outcomes and screenshots or log files of the test.
2. Use a debugging tool and trace the code execution to detect and correct any errors in your code. Include screenshots in your testing and debugging document of:
   1. your files within the debugger
   2. the process of tracing the code execution.

Table 1 – Test results

| Test ID | Test results | Screenshots of testing/ relevant comments | | Test outcome  (pass/fail) |
| --- | --- | --- | --- | --- |
| 01 |  |  | Pass | |
| 02 |  |  | Pass | |
| 03 |  |  | pass | |
| 04 |  |  | Pass | |
| 05 |  |  | Pass | |
| 06 |  |  | Pass | |
| 07 |  |  | Pass | |
| 08 |  |  | Pass | |
| 09 |  |  | Pass | |

1. Analysis and conclusions

All of the test cases in test plan have successfully passed the tests, however there is an error in the code, where the form on submit gives and error “page not found”.

Software very well meets the specifications in the requirements document including:

Table 3 – Software evaluation

|  |  |
| --- | --- |
| Software specification | Evaluation |
| Promotional period. | The promotional period can only be applied during a promotional period. |
| One pizza | One large pizza must cost $6.45. |
| Two pizzas | Two large pizzas will cost $12.00. |
| Three pizzas | Three large pizzas will cost $14.00. |
| Four or more | Four or more pizzas will use a combination of the above prices to ensure the best price for the customer. |

1. Software approval

Software is approved for installation by:

Name

Christina Kaiser

Role

Software Development Team Leader

Signature

CK

Date

22/05/2022

1. Document authority

This Pizza Price Calculator Test Report has been authorised by Catherine Dunn the CEO of Gelos Enterprises and is available to all staff. It has been developed in line with all relevant legislation, in consultation with committee representatives and will be revised on a regular basis.

Approval date: 30/05/2022