

PIZZA PARLOR.COM

TESTING

PLAN

Business Owner: Marti Baker

Development Team “Aqua”:

Robert Permenter

Kathleen O’Leary

Rikki Nelson

June 10, 2012

Table of Contents

Introduction	3
Document Description	
Related Documents	
Resource Requirements	4
Hardware	
Software	
Test Tools	
Staffing	
Responsibilities	
Training	
Dependencies/Risks	5
Dependencies	
Risks	
Features to be Tested	6
Features Not to be Tested	7
Approach	7
Testing Timeline	
Testing Levels	
Schedule and Milestones	
Test Case Format	
Item Pass/Fail Criteria	8
Appendix A: Test Cases	9

INTRODUCTION

Document Description

This document is a Test Plan for the Pizza Parlor project, produced by Team Aqua. It describes the testing strategy and approach to testing that Team Aqua will use to validate the quality of this product prior to release. It also contains various resources required for the successful completion of this project.

The focus of the Pizza Parlor project is to support those new features that will allow the employees to access an Administrative Site in order to manage customers and orders. Those features include:

- Administrative Log-In Page: entry of a user name and password.
- Administrative Menu Page: selection of specific report
- Administrative Report Pages: display of selected Customer Listing, Open Orders, or All Orders.
- Creation of the Pizza Parlor database to store all data received from customers and orders.

This release of the Pizza Parlor project will also include legacy bug fixing, and redesigning or including missing functionality from previous release as follows:

- Updated Web pages using PHP.
- Added an Order Confirmation Page for documenting order selections and user-friendliness.
- Added PHP code to insert new order data into a Pizza Parlor Database.
- Added PHP code to send customer a confirmation email upon confirmation of order.

The following implementations were made:

- Installed XAMP in order to process PHP script.
- Installed MySQL in order to create and store the database as well as process queries.

Related Documents

The following documents have been utilized in the design, development, and implementation of the new Pizza Parlor Customer and Order Management System:

- Functional Requirements and Technical Specifications Report, which includes:
 - Project Plan
 - Website Wire Frame
 - System Story Board
- System Design Document
- Implementation Plan

RESOURCE REQUIREMENTS

Hardware

The testing machine should meet the following minimum specifications:

- Operating System: Windows 2007 or better
- Hard drive: 100 GB
- RAM: 8 GB
- Processor: 64-bit, 4 cores

Software

The following software is required to be installed and operating on the testing machine:

- XAMP – Local host for PHP processing.
- MySQL – Database processing.
- Firefox – Internet browser.
- Microsoft Excel – Test Case file format.
- Microsoft Expressions – Enable editing of HTML/PHP code for immediate bug resolution.
- Toad for MySQL – Enable editing of database queries for immediate bug resolution.

Test Tools

Apart from manual tests, the following tools will be used:

- No other tools required.

Staffing

The following individuals comprise the testing staff:

- Robert Permenter – Testing Coordinator
- Kathleen O’Leary – Test Technician
- Rikki Nelson – Test Documenter

Responsibilities

The specific responsibilities of each testing function are as follows:

- Testing Coordinator:
 - Verifies compliance with the Test Plan
 - Verifies initial conditions of Test Cases
 - Directs response actions required to immediately resolve bugs reported by testing
- Test Technician:
 - Conducts testing in accordance with Test Cases
- Test Documenter:
 - Documents completion of each Test Case step
 - Annotates details of any bugs found within the applicable Test Case's notes section.

Training

Each testing staff member must meet the following training requirements:

- Student in an EdCC Computer Information Sciences or Computer Sciences academic program.
- Demonstrated familiarity with required hardware and software.

DEPENDENCIES/RISKS

Dependencies

In addition to resource requirements previously discussed, testing of the Pizza Parlor System depends on the following items:

- Availability of testing machine (due to being in a public testing location).
- Availability of the local server for the testing machine (a semi-private server under client's control).
- Availability of client to review bugs for resolution priority or acceptance.
- "Debug Coding" should be active to allow tester to view results of queries and data manipulations.

Risks

There are several parts of the project that may not be within the control of the Pizza Parlor System development but have direct impacts on the process and should be checked as well:

- Location of System Files (They may not all be located in proper file folders)
 - Resolve by placing all system files within proper folder for local host.
- System Corruption (files for the system may become corrupted upon installing on test machine).
 - Resolve by making a "testing copy" of files separate from system's master files.

FEATURES TO BE TESTED

The following is a list of the features to be focused on during testing of the application:

- Navigational Link functionality
- Placing an Order:
 - Size selection
 - Crust selection
 - Specialty selection
 - Topping selection (if Build-Your-Own specialty selected)
- Input of Customer Information:
 - Required Entries (entry and validation):
 - First and Last names
 - Street Address
 - City, State, and Postal Zip Code
 - Phone Number
 - Email Address
 - Optional Entries (entry only):
 - Apartment or Unit Number
- Order Summarization:
 - Dynamic update on Place Order Page
 - Correct transfer and update on Order Confirmation Page
 - Insertion of customer and order data into database after Confirm Order selected
 - Transmission of confirmation email to customer after Confirm Order selected
 - Correct transfer and update on Order Summary Page
- Admin Log-in
 - Username entry
 - Encrypted password entry
 - Login validation
- Reports
 - Retrieval of the following reports from the database:
 - Customer Listing (list of all customers with personal data)
 - Open Orders (list of orders awaiting completion in kitchen)
 - All Orders (list of all orders filled)
 - Updating the following from Reports Results pages:
 - Open Orders (select “completed” status of order once finished in kitchen)

FEATURES NOT TO BE TESTED

All testing in the following areas will be indirect as a result of other testing efforts, as these areas will not be specifically addressed:

- Web Page compilation
 - Pizza Parlor Ordering website
 - Admin website
- Admin site Logoff

APPROACH

Testing Timeline

The estimated testing time line for this project is very aggressive (one (1) day). As such, any delays in the development process, test planning process, or installation and verification of the third party software could have significant effects on the test plan. The acceptance testing is expected to take one (1) day from the date of application delivery from system test and is to be done in parallel with the current application process.

Testing Levels

The project will have two levels of testing, System and Acceptance. The details for each level are further defined as follows:

SYSTEM TESTING will consist of performing control flow testing of each portion (orders and administration) so as to cover testing of all individual features in the process of browsing the website pages and using the individual features on each page.

ACCEPTANCE TESTING will be performed after correction of any bugs reported from System testing. If there are no bugs reported from System testing, than the System testing results will be submitted for Acceptance testing approval. If Acceptance testing is required, it will consist of regression testing of those test cases that resulted in Bug Reports.

Schedule and Milestones

The following table indicates the phases and deliverables required as milestones for each phase:

Test Phase	Days	Milestone/Deliverable
Planning	5	Test Plan with Test Cases
System Testing	0.5	ST Testing Report
Acceptance Testing	0.5	Acceptance Document (with approval signature)

Test Case Format

Test Cases shall be written in the following format:

[Insert Test Case #]	[Insert Test Case Title]	Pass/Fail:	Bug Notes:
Step 1:	[Insert action step]		
Step 2:			
Step 3:			
Expected Result:	[Insert expected result]		

Test Case Numbers will indicate TC for “Test Case”, a code for the level of testing, and the sequential number of case being tested during this test plan. Testing level codes are as follows:

- ST: System Testing
- AT: Acceptance Testing

For example, TC-UT-02 indicates the second test within Unit Testing phase.

If a step is completed without indication of failure, the Pass/Fail column shall be marked with an “X”. If a step fails, then the attempt shall be noted with the word “FAIL”. If the step is not applicable, then the attempt shall be noted using “N/A”.

ITEM PASS/FAIL CRITERIA

Each step of a test case must pass, unless it is determined to not be applicable (in which case a note shall be made within the test case explaining how the step is not applicable). Should a test fail any single step, the test case ends and a Bug Report must be written. Write the word “FAIL” in the “Pass/Fail” column for the step or result that failed and write detailed notes describing the bug that caused failure in the right hand column titled “Bug Notes”.

Upon successful completion of all steps, indicate the test completion with the word “PASS” in the “Pass/Fail” column for the “Expected Result” row.

APPENDIX A: TEST CASES

TC-ST-01	<u>Specialty Pizza Link</u>	<u>Pass/Fail</u>	<u>Bug Notes:</u>
Step 1	Click link		
Step 2			
Step 3			
Step 4			
Step 5			
Result	Page content changes to show specialty pizza descriptions		
TC-ST-02	<u>Fair Usage Link</u>	<u>Pass/Fail</u>	<u>Bug Notes:</u>
Step 1	Click link		
Step 2			
Step 3			
Step 4			
Step 5			
Result	Page content changes to show fair usage statement		
TC-ST-03	<u>Place Order Link</u>	<u>Pass/Fail</u>	<u>Bug Notes:</u>
Step 1			
Step 2			
Step 3			
Step 4			
Step 5			
Result	Page content changes to show Place Order selections		
TC-ST-04	<u>Change Pizza Size</u>	<u>Pass/Fail</u>	<u>Bug Notes:</u>
Step 1	Click "Large" radio button		
Step 2			
Step 3			
Step 4			
Step 5			
Result	Order Summary should update dynamic summary and price change.		

TC-ST-05	<u>Change Crust Selection</u>	<u>Pass/Fail</u>	<u>Bug Notes:</u>
Step 1			
Step 2			
Step 3			
Step 4			
Step 5			
Result	Order Summary should update dynamic summary and price change.		

TC-ST-06	<u>Change Specialty Selection</u>	<u>Pass/Fail</u>	<u>Bug Notes:</u>
Step 1			
Step 2			
Step 3			
Step 4			
Step 5			
Result	Order Summary should update dynamic summary and price change.		

TC-ST-07	<u>Select Toppings</u>	<u>Pass/Fail</u>	<u>Bug Notes:</u>
Step 1			
Step 2			
Step 3			
Step 4			
Step 5			
Result	Order Summary should update dynamic summary and price change.		

TC-ST-08	<u>Invalid-to-Valid Customer Info</u>	<u>Pass/Fail</u>	<u>Bug Notes:</u>
Step 1	Click order (no entries)		
Step 2	Type in First Name		
Step 3	Click order (no entries)		
Step 4	Type in Last Name		
Step 5	Click order (no entries)		
Step 6	Type in street Address		
Step 7	Click order (no entries)		
Step 8	Type in City		
Step 9	Click order (no entries)		
Step 10	Type in State		
Step 11	Click order (no entries)		
Step 12	Type in Zip Code		
Step 13	Click order (no entries)		
Step 14	Type in Phone Number		
Step 15	Click order (no entries)		
Step 16	Type in Email Address		
Step 17	Click order (no entries)		
Result	After each blank or invalid entry, focus should return to input control with "Required" in red prompt. Final click of "Order your Pizza" should link to Order Confirmation Page.		

TC-ST-09	Data FROM Place Order Page	Pass/Fail	Bug Notes:
Step 1	Perform TC-ST-08		
Step 2	View customer information variables		
Step 3	View pizza order variables		
Step 4			
Step 5			
Result	All variables should be grabbed as posted by prior web page		
TC-ST-10	Correct Order Button	Pass/Fail	Bug Notes:
Step 1	Perform TC-ST-09		
Step 2	Click Correct Order Button		
Step 3			
Step 4			
Step 5			
Result	Should return to Place Order Page		
TC-ST-11	Confirm Order Button	Pass/Fail	Bug Notes:
Step 1	Perform TC-ST-10		
Step 2	Click Confirm Order Button		
Step 3			
Step 4			
Step 5			
Result	Should go to Order Summary Page		
TC-ST-12	Data FROM Order Confirmation Page	Pass/Fail	Bug Notes:
Step 1	Perform TC-ST-11		
Step 2	View customer information variables		
Step 3	View pizza order variables		
Step 4	View hidden email verification		
Step 5			
Result	All variables should be grabbed as posted by prior web page and web page showing proper order summary and "thank you" message.		

TC-ST-13	Invalid-to-Valid Admin Login	Pass/Fail	Bug Notes:
Step 1	Open Admin Login Webpage		
Step 2	Click "Login" (no entries)		
Step 3	Type in valid User Name		
Step 4	Click "Login" (no entries)		
Step 5	Type in Invalid Password		
Step 6	Click "Login" (no entries)		
Step 7	Type in Valid Password		
Step 8	Click "Login" (no entries)		
Result	After each blank or invalid entry, focus should return to input control with "Please enter valid User Name and Password" in red prompt. Final click of "Login" should send to Admin Menu Page.		

TC-ST-14	Customer Listing Report	Pass/Fail	Bug Notes:
Step 1	Perform TC-ST-13		
Step 2	Click Customer Listing Report		
Step 3			
Step 4			
Step 5			
Result	Reports Result Page displays tabular listing of all customer information by customer.		

TC-ST-15	All Orders Report	Pass/Fail	Bug Notes:
Step 1	Perform TC-ST-13		
Step 2	Click All Orders Report		
Step 3			
Step 4			
Step 5			
Result	Reports Result Page displays tabular listing of all orders by order number.		

TC-ST-16	Open Orders Report	Pass/Fail	Bug Notes:
Step 1	Perform TC-ST-13		
Step 2	Click Open Orders Report		
Step 3			
Step 4			
Step 5			
Result	Reports Result Page displays tabular listing of all open orders by order number.		

TC-ST-17	Completing an Order in Admin	Pass/Fail	Bug Notes:
Step 1	Perform TC-ST-16		
Step 2	Verify order listing includes "Completed" checkboxes.		
Step 3	Click a to "complete an order"		
Step 4	Click "Update Orders" button		
Step 5			
Result	Open Orders Report should refresh (new query for report), and completed report should not be included in report.		