Senior test plan – master test plan  
photo castle 1.0

**Student Name: Brian Aguilar Diaz**

**Project Advisor Name: Dr. Sean Hayes**

**Expected Graduation Date: Spring 2022**

**Test plan identifier**

Photo Castle 1.0 release – Master Test Plan v1.0 The structure of this document follows and is based on the IEEE standard for software test documentation and provides a summary of what a test plan can/should contain.

**Introduction**

* The objective of Photo Castle 1.0 is to be the new competitor in the photo editing software market. Photo Castle wants to offer everything Photoshop offers with a neat and simple user interface at no cost for the customer. The project goal is then to offer a photo editing software free of cost that can offer photoshop tools with a much simpler UI. The first release of this software will be named and known as Photo Castle 1.0
* The audience that is targeted are the people who want to edit their photos professionally however they do not want to pay the premium prices photoshop currently offers. The audience is also not looking find a new learning curve in a program that is like photoshop. The program that is going to be produced will be guided to the ease of use for the customer and will provide many great tools that photoshop currently offers.
* The issue is that the Photo Castle 1.0 software needs to be tested in a proper manner in order for it to work for its customers. The testing should be organized into levels to deeply find any errors and to correct them before any release is sent out. These levels are listed as following:
  + Code Quality and Dependency
  + Main System Interface
  + System Quality
  + Tools Implementation and Use

In these levels they are all going to be tested for the same aspects. The Code quality is to make sure that testing on a small set of code will work and integrate correctly. The main system will mostly provide performance, ease of use, reliability, and functionality. The system quality is then the acceptance of the systems incorporations. The final test is the use of all tools and its functionality and performance on the photos.

**References**

* Related Documents include the following:
  + Project Plan
  + Project Proposal
  + Requirements Document

**Test Items**

* The Items that will be tested are:
  + Photo Castle 1.0 Software and Infrastructure
  + Other supporting software

**Features to be Tested**

* The testing will include many levels as stated in the introduction, in every level there will be testing of the following for the Photo Castle software.
  + When you click on the drawing tools, you will be able to use them on the image.
  + The filters on the left should apply entire filters to the image.
  + The batch tool should allow multiple images to be selected and applied filters, rotations, and even size reductions.
  + The crop, resize, rotate tool should work and edit the photo chosen.
  + The color palette will have a variety of colors to choose from.
  + The undo and redo button.
  + The open, save, and copy tools.
  + Tool customizability.
  + Brightness, Contrast, and Saturation tools.
  + All tools shall be tested.

The Usability testing of all tools can be inserted in this table if any errors occur:

|  |  |  |  |
| --- | --- | --- | --- |
| Tool | Result | Issues | Feedback |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Features Not to be Tested**

* All features will be tested at every level following the aspects mentioned before however if the level is not of high importance and is of low importance than its possible it may be skipped.

**Approach**

* The testing on Photo Castle will be of priority testing. Each level will be tested however each level will have its importance such as high, medium, and low testing. This rule shall be followed unless the low testing is a pre-requisite of a higher level. The testing will be of all manual testing done by the tester or professor. The tester will go to each tool and test its accessibility and functional-use first. If the tool works well the reliability is marked off and its performance level as well. The tester will then look for any vulnerabilities in the tool and test for the IEEE coding standards. If all aspects are checked off then the level is complete and the tester can continue to another art tool. The main tools are the high priority while the subcategories of the tool will be of medium priority. The low priority are the tools that will be added as an extra and not a requirement. The end approach will be the time it took to test Photo Castle tools, the amount of test cases produced and incidents. The performance will be timed and the organization of the tool will be criticized.

**Item Pass/Fail Criteria**

* The Item Pass Criteria for each level of testing will need to be passed in order to continue to the next level. The approval of this level being passed will be of the Professor when he is checking off every aspect. The Item Fail Criteria will also be of the Professors decision. If the level cannot meet all of the aspects to be tested then the project Photo Castle 1.0 will be delayed or restarted.

**Suspension Criteria and Resumption Requirements**

* If the testing is being done on Photo Castle 1.0 and the software crashes then the testing must be stopped. If the testing is stopped because of a crash, then the software must be restarted. Once the program starts up again then the testing may resume. Testing can also be skipped for a subcategory tool if the main tool previously failed.

**Test Deliverables**

* The Test Deliverables include the following:
  + Master Test Plan (this document itself)
  + Test Cases
  + Test Scripts
  + Defect/Enhancement logs
  + Test Reports

**Test Environment**

* Photo Castle 1.0 will have the test environment hardware of their own personal computer to test. At first release the Photo Castle program will only be an executable and will not have any network connections needed. The hardware requirements for a laptop/desktop are i5, 8gb ram, colored screen, running windows 10, and an HD display is preferred but not needed. The tools needed to test Photo Castle 1.0 are simple manual calculations and automated tools will not be needed.

**Estimate**

* The effort is put by the professor testing Photo Castle 1.0 and making sure all the levels are passed. The cost is the professors time.

**Schedule**

* The following schedule will be met:

|  |  |  |  |
| --- | --- | --- | --- |
| Description | Start Date | End Date | Duration |
| Project Proposal and Requirements Document | 08/20/2021 | 08/30/2021 | 10 days |
| Draft Test Plan & Project Update | 08/20/2021 | 10/08/2021 | 18 days |
| Updating the Test Plan by its feedback | 10/09/2021 | 11/21/2021 | 44 days |
| Continuing the Project Construction | 10/09/2021 | 11/21/2021 | 44 days |
| Final Test Plan & Project Repository | 11/21/2021 | 11/22/2021 | 1 day |

**Staffing and Training Needs**

* Summary: The Photo Castle Lead manager will make sure that staff that are going to work on the project have experience with:
  + General testing techniques
  + Knowledge of other photo editing programs
  + Knowledge of testing tools that will be used
  + Photo Castles requirements document and project proposal

**Responsibilities**

* The following individuals will be responsible for:
  + Brian Diaz (CEO of Photo Castle) – Project Lead and Tester
  + Dr. Sean Hayes (Director/Project Overseer) – Senior Project Approval and Tester

**Risks**

* The following identifies all the likely risks the project may undergo:
  + The software becomes unusable after a crash. The testing is then delayed until the program can be fixed by the lead programmer.
  + The main tools to edit photos are not working. The testing is delayed again until the programmer can fix the tools.
  + The tools of the program cause errors in other categories to occur. The testing is delayed once again and the program may need to be restarted.
  + Not enough time, the project may undergo many errors and test cases and may not meet the deadline to turn in the project.
  + Communication between tester and programmer may not understand each other. This causes the program to run even more time wasted without being tested.

**Assumptions and Dependencies**

* Not sure what is to be included here.

**Approvals**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Approval Signatures** |  |  |  |  |
| Name of Professor, Role |  | Professor Sean Hayes, Project Sponsor |  | Brian Diaz, Project Manager |