Software Test Plan

for

Prom Sign-in project



Falcon Computer Company:

Parth Agarwal, Project Manager

Grey Kienzle, Lead Programmer

Gene Yu, Systems Analyst

Poolesville High School

As of 5/28/2019

Version 1.0.0

|  |  |  |  |
| --- | --- | --- | --- |
| Version release | Major party | Major changes | Date |
| 0.1.0 | Gene Yu | Table of contents, references (section 2) | 5/7 |
| 0.1.1 | Parth Agarwal | Introduction (section 3) | 5/7 |
| 0.1.2 | Grey Kienzle | Changed document structure, edited section 4 | 5/25/2019 |
| 1.0.0 | Parth Agarwal, Grey Kienzle, Gene Yu | Sections 5-18 | 5/27-28/2019 |
|  |  |  |  |

Table of contents

1. References
2. Introduction
   1. Definitions and acronyms
3. Test Items (Functions)
4. Software Risk Issues
5. Features to be Tested
6. Features not to be Tested
7. Approach (Strategy)
8. Item Pass/Fail Criteria
9. Suspension Criteria and Resumption Requirements
10. Test Deliverables
11. Remaining Test Tasks
12. Environmental Needs
13. Staff and Training Needs
14. Responsibilities
15. Schedule
16. Planning Risks and Contingencies
17. Approvals
18. Appendices

# 1. References

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Title of document | Date | Author | Publishing organization | Status or report number or ISBN or version |
| Prom Sign-in Software Requirements Specification | 2019 | Falcon Computer Company | Falcon Computer Company | 1.0.0 |
| IEEE 829-2008 - IEEE Standard for Software and System Test Documentation | 2008 | 829\_WG - Std for Software Test Documentation Working Group | Institute of Electrical and Electronics Engineers | [829-2008](https://standards.ieee.org/standard/829-2008.html) |
| TEST PLAN OUTLINE  (IEEE 829 Format) | 2014 | Systeme Evolutif Limited | Systeme Evolutif Limited | [RS-MTP01.3](https://jmpovedar.files.wordpress.com/2014/03/ieee-829.pdf) |
| Prom Sign-in Software Project Management Plan | 2019 | Falcon Computer Company | Falcon Computer Company | 1.1.1 |
| Prom Sign-in List of Tests | 2019 | Falcon Computer Company | Falcon Computer Company | Incomplete |

# 2. Introduction

The purpose of this document is to outline the objects and processes relating to testing the Prom Sign-in program being developed by FCC. The document provides specific procedures for ensuring that the Prom Sign-in program meets all requirements listed in section 4 of the SRS. As stated in the SPMP, the project manager, systems analyst, and main programmer all contribute to the creation and maintenance of this document. This project is being made for Ms. Alexandra Hicks and is under the supervision of Estep Corporations.

2.1 Definitions and acronyms

* SPMP: Software Project Management Plan
* SRS: Software Requirements Specification
* SDD: Software Design Description
* FCC: Falcon Computer Company
* Estep Corporations: The parent corporation of FCC and many other companies involving computer programming
* Corp: corporation
* Head of Estep Corp: Chief Executive Officer Mr. Mark Estep
* CEO: Chief Executive Officer
* API: Application programming interface
* User interface: the component of a program which the user and program interact with each other. The user controls the program and the program provides feedback.

3. Test items (functions)

FCC will test all features below.

* Correct sign in/out mechanics
* Functional ID entry using both keyboard and scanning tool
* Functional sorting mechanisms that connect the app with the spreadsheet
* Information displays that access data from the spreadsheet

4. Software risk issues

The software being tested is being developed by the party producing this document, so the risks present will consist of past unfixed bugs and general misunderstanding of the project requirements.

5. Features to be tested

|  |  |  |
| --- | --- | --- |
| Feature | Specification met | Importance level (Essential, High, Medium, Low) |
| 1. Manual student ID entering | 4.2 | E |
| 2. Student ID scanning | 4.2 | E |
| 3. Student picture display | 4.3 | M |
| 4. Student name displayed | 4.4 | E |
| 5. Student ID displayed | 4.4 | H |
| 6. Student ticket number displayed | 4.4 | E |
| 7. Student guest list displayed | 4.4 | E |
| 8. Guest information displayed | 4.4 | E |
| 9. Data sorting | 4.6 | H |
| 10. Sign In time recorded | 4.7 | H |
| 11. Sign out time recorded | 4.7 | H |
| 12. Final verification | 4.8 | E |

6. Features not to be tested

The following features will not be tested directly because testing of features listed in section 5 of this document will fulfill all necessary checks for the features listed here.

|  |  |
| --- | --- |
| Feature | Specification met |
| Vegas gold background | 4.1 |
| Poolesville logos in corner | 4.1 |
| Black text | 4.1 |
| Student info recorded | 4.5 |
| Guest info recorded | 4.5 |
| Access to data | 4.5 |

7. Approach (strategy)

The strategy for testing will stress the importance of the different type of tests that are possible, and will require testing for all possible scenarios that are needed. The prom event deals with lots of diversity of students and the program has to be able to effectively deal with the diversity with no issue. The testing will also be heavily focused towards causing no errors as the event needs to be run smoothly, and an error in the program can be very detrimental to the course of the event. The high priority features that are listed above will be tested the most, and must fulfill its desired functions stated in the SRS. The scanner tool will be a key part in implementing our testing as it’s a crucial device used along with the program, and our strategy will require us to consider all the ways a student is able to be signed in. FCC also will spend a substantial amount of time testing with the Techleon programs, and testing the effectiveness of our programs working together. The same type of approach described above should be used when working with Techleon. Any hardware or software error which occurs in the programs relating to our higher priorities will be become crucial in future plans to fix. Anything that involves hindering the success of the prom event becomes of highest importance in the testing process.

Testing of features must be documented in the format specified in section 18.1 of this plan. Tests be added in chronological order to the document called List of Tests, which is in the Deliverables directory of the project folder.

8. Item pass/fail criteria

For an item to pass testing, every test case below and in the Software Test Cases document for that item must succeed. A test case succeeds when the expected outcome for that test case occurs. When the program begins, the screen shows a text field for entering IDs, a submit button for the text field, a sign in button, and a sign out button. A drop down menu contains sorting options, which are information fields which correspond to columns in the spreadsheet. There is also a sort button for the drop down menu.

|  |  |  |  |
| --- | --- | --- | --- |
| Test description | Steps and edge cases for performing test | Item Number (specified in section 5 of this plan) | Pass Criteria |
| App does not crash when student ID present in sheet is entered. | 1. Enter a valid student ID into the text field on the sign in screen. This ID must be present in the data sheet.  1a. Edge cases: First and last ID listed in data sheet of known students who go to Poolesville High School.  1b. Edge cases: first and last ID listed in data sheet of known students who go to any MCPS school other than Poolesville High School.  2. Click the submit button. | 1 | App refreshes. The text field shows up blank. The submit button, sign in button, and sign out button appear on screen. All student data in the spreadsheet is not modified in any way. (Other changes in the app may occur.) |
| App does not crash when student ID not present in sheet is entered. | 1. Enter a six digit valid MCPS ID number into the ID text field on the sign in screen. This number must not match any student ID in the data sheet.  2. Click the submit button. | 1 | App refreshes. The text field shows up blank. The submit button, sign in button, and sign out button appear on screen. All student data in the spreadsheet is not modified in any way. Some indication shows up informing the user that the student ID is not present in the data sheet. |
| App does not crash when no student ID is entered. | 1. Enter a value into the ID text field on the sign in screen. This number must not match any student ID in the data sheet.  1a. Edge case: Enter a negative number or zero number into the text field.  1b. Edge case: Enter a value that includes numbers and non-numeric characters into the text field.  1c. Edge case: Leave the text field blank.  2. Click the submit button. | 1 | App refreshes. The text field shows up blank. The submit button, sign in button, and sign out button appear on screen. All student data in the spreadsheet is not modified in any way. If a non-empty value was entered into the text field, some indication shows up informing the user that the value does not match up with any form of student identification in the data sheet. If the text field was blank, an indication is optional. |
| App does not crash when student ID that corresponds to an ID number in the spreadsheet is scanned. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. This ID must be present in the data sheet.  1a. OPTIONAL edge case: scan an MCPS ID card barcode of a student that does not attend Poolesville High School.  2. Click the submit button. | 2 | Before the submit button is clicked, the student ID scanned appears in the ID text field. After the submit button is clicked, the app refreshes. The text field shows up blank. The submit button, sign in button, and sign out button appear on screen. All student data in the spreadsheet is not modified in any way. (Other changes in the app may occur.) |
| App does not crash when student ID that corresponds to an ID number not present in the spreadsheet is scanned. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. The ID must not match with a student ID in the spreadsheet.  1a. OPTIONAL edge case: scan a dummy barcode which represents a six digit number not present in the spreadsheet.  1Xa. If the test is infeasible due to all possible student IDs being listed in the spreadsheet, this test may be ignored.  2. Click the submit button. | 2 | Before the submit button is clicked, the student ID scanned appears in the ID text field. After the submit button is clicked, the app refreshes. The text field shows up blank. The submit button, sign in button, and sign out button appear on screen. All student data in the spreadsheet is not modified in any way. Some indication shows up informing the user that the student ID is not present in the data sheet. |
| Student information (name, ID, picture, ticket number, guests) is not displayed when student ID not present in sheet is entered. | 1. Enter a six digit valid MCPS ID number into the ID text field on the sign in screen. This number must not match any student ID in the data sheet.  2. Click the submit button. | 3-8 | After the submit button is clicked, no student information (name, ID, picture, ticket number, guests) appears. All student data in the spreadsheet is not modified in any way. |
| Student information (name, ID, picture, ticket number, guests) is not displayed when no student ID is entered. | 1. Enter a value into the ID text field on the sign in screen. This number must not match any student ID in the data sheet.  1a. Edge case: Enter a negative number or zero number into the text field.  1b. Edge case: Enter a value that includes both numbers and non-numeric characters into the text field.  1c. Edge case: Leave the text field blank.  2. Click the submit button. | 3-8 | After the submit button is clicked, no student information (name, ID, picture, ticket number, guests) appears. All student data in the spreadsheet is not modified in any way. |
| Student information (name, ID, picture, ticket number, guests) is not displayed when student ID that corresponds to an ID number not present in the spreadsheet is scanned. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. The ID must not match with a student ID in the spreadsheet.  1a. OPTIONAL edge case: scan a dummy barcode which represents a six digit number not present in the spreadsheet.  1Xa. If the test is infeasible due to all possible student IDs being listed in the spreadsheet, this test may be ignored.  2. Click the submit button. | 3-8 | After the submit button is clicked, no student information (name, ID, picture, ticket number, guests) appears. All student data in the spreadsheet is not modified in any way. |
| Student picture is displayed when a student ID present in the spreadsheet is entered. | 1. Enter a valid student ID into the text field on the sign in screen. This ID must be present in the data sheet.  1a. Edge cases: First and last ID listed in data sheet of known students who go to Poolesville High School.  1b. Edge cases: first and last ID listed in data sheet of known students who go to any MCPS school other than Poolesville High School.  2. Click the submit button. | 3 | After the submit button is clicked, correct student picture appears. All student data in the spreadsheet is not modified in any way. |
| Student picture is displayed when student ID that corresponds to an ID number in the spreadsheet is scanned. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. This ID must be present in the data sheet.  1a. OPTIONAL edge case: scan an MCPS ID card barcode of a student that does not attend Poolesville High School.  2. Click the submit button. | 3 | After the submit button is clicked, correct student picture appears. All student data in the spreadsheet is not modified in any way. |
| Student name is displayed when a student ID present in the spreadsheet is entered. | 1. Enter a valid student ID into the text field on the sign in screen. This ID must be present in the data sheet.  1a. Edge cases: First and last ID listed in data sheet of known students who go to Poolesville High School.  1b. Edge cases: first and last ID listed in data sheet of known students who go to any MCPS school other than Poolesville High School.  2. Click the submit button. | 4 | After the submit button is clicked, correct student name appears. All student data in the spreadsheet is not modified in any way. |
| Student name is displayed when student ID that corresponds to an ID number in the spreadsheet is scanned. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. This ID must be present in the data sheet.  1a. OPTIONAL edge case: scan an MCPS ID card barcode of a student that does not attend Poolesville High School.  2. Click the submit button. | 4 | After the submit button is clicked, correct student name appears. All student data in the spreadsheet is not modified in any way. |
| Student ID is displayed when a student ID present in the spreadsheet is entered. | 1. Enter a valid student ID into the text field on the sign in screen. This ID must be present in the data sheet.  1a. Edge cases: First and last ID listed in data sheet of known students who go to Poolesville High School.  1b. Edge cases: first and last ID listed in data sheet of known students who go to any MCPS school other than Poolesville High School.  2. Click the submit button. | 5 | After the submit button is clicked, correct student ID number appears. All student data in the spreadsheet is not modified in any way. |
| Student ID is displayed when student ID that corresponds to an ID number in the spreadsheet is scanned. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. This ID must be present in the data sheet.  1a. OPTIONAL edge case: scan an MCPS ID card barcode of a student that does not attend Poolesville High School.  2. Click the submit button. | 5 | After the submit button is clicked, correct student ID number appears. All student data in the spreadsheet is not modified in any way. |
| Student ticket number is displayed when a student ID present in the spreadsheet is entered. | 1. Enter a valid student ID into the text field on the sign in screen. This ID must be present in the data sheet.  1a. Edge cases: First and last ID listed in data sheet of known students who go to Poolesville High School.  1b. Edge cases: first and last ID listed in data sheet of known students who go to any MCPS school other than Poolesville High School.  2. Click the submit button. | 6 | After the submit button is clicked, correct student ticket number appears. All student data in the spreadsheet is not modified in any way. |
| Student ticket number is displayed when student ID that corresponds to an ID number in the spreadsheet is scanned. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. This ID must be present in the data sheet.  1a. OPTIONAL edge case: scan an MCPS ID card barcode of a student that does not attend Poolesville High School.  2. Click the submit button. | 6 | After the submit button is clicked, correct student ticket number appears. All student data in the spreadsheet is not modified in any way. |
| A student’s guests are displayed when a student ID present in the spreadsheet is entered, and the student who that ID corresponds to has guests. | 1. Enter a valid student ID into the text field on the sign in screen. This ID must be present in the data sheet. The student must at least one guest.  1Xa. If no students have guests, this test may be ignored.  2. Click the submit button. | 7 | After the submit button is clicked, correct student guests appear. All student data in the spreadsheet is not modified in any way. |
| A student’s guests are displayed when student ID that corresponds to an ID number in the spreadsheet is scanned, and the student who that ID corresponds to has guests. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. This ID must be present in the data sheet. The student must have at least one guest.  1Xa. If no students have guests, this test may be ignored.  2. Click the submit button. | 7 | After the submit button is clicked, correct student guests appear. All student data in the spreadsheet is not modified in any way. |
| A student’s guests are not displayed when a student ID present in the spreadsheet is entered, and the student who that ID corresponds to does not have guests. | 1. Enter a valid student ID into the text field on the sign in screen. This ID must be present in the data sheet. The student must not have any guests.  1Xa. If no students have guests, this test may be ignored.  2. Click the submit button. | 7 | After the submit button is clicked, no student guests appear. All student data in the spreadsheet is not modified in any way. |
| A student’s guests are not displayed when student ID that corresponds to an ID number in the spreadsheet is scanned, and the student who that ID corresponds to does not have guests. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. This ID must be present in the data sheet. The student must not have any guests.  1Xa. If no students have guests, this test may be ignored.  2. Click the submit button. | 7 | After the submit button is clicked, no student guests appear. All student data in the spreadsheet is not modified in any way. |
| A student’s guests’ information (name, ID [if applicable], picture[if applicable], ticket number, person who brought them) is displayed when a student ID present in the spreadsheet is entered, and the student who that ID corresponds to has guests. | 1. Enter a valid student ID into the text field on the sign in screen. This ID must be present in the data sheet. The student must at least one guest.  1Xa. If no students have guests, this test may be ignored.  2. Click the submit button. | 8 | After the submit button is clicked, correct student guest information appears. All student data in the spreadsheet is not modified in any way. |
| A student’s guests’ information (name, ID [if applicable], picture[if applicable], ticket number, person who brought them) is displayed when student ID that corresponds to an ID number in the spreadsheet is scanned, and the student who that ID corresponds to has guests. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. This ID must be present in the data sheet. The student must have at least one guest.  1Xa. If no students have guests, this test may be ignored.  2. Click the submit button. | 8 | After the submit button is clicked, correct student guest information appears. All student data in the spreadsheet is not modified in any way. |
| A student’s guests’ information (name, ID [if applicable], picture[if applicable], ticket number, person who brought them) is not displayed when a student ID present in the spreadsheet is entered, and the student who that ID corresponds to does not have guests. | 1. Enter a valid student ID into the text field on the sign in screen. This ID must be present in the data sheet. The student must not have any guests.  1Xa. If no students have guests, this test may be ignored.  2. Click the submit button. | 8 | After the submit button is clicked, no student guest information appears. All student data in the spreadsheet is not modified in any way. |
| A student’s guests’ information (name, ID [if applicable], picture[if applicable], ticket number, person who brought them) is not displayed when student ID that corresponds to an ID number in the spreadsheet is scanned, and the student who that ID corresponds to does not have guests. | 1. With the text cursor active in the ID text field, scan an MCPS ID card barcode. This ID must be present in the data sheet. The student must not have any guests.  1Xa. If no students have guests, this test may be ignored.  2. Click the submit button. | 8 | After the submit button is clicked, no student guest information appears. All student data in the spreadsheet is not modified in any way. |
| Sorts spreadsheet correctly by first name. | 1. Choose the first name option from the drop down menu.  2. Click the sort button. | 9 | After the sort button is clicked, the spreadsheet is sorted by row by first name. |
| Sorts spreadsheet correctly by last name. | 1. Choose the last name option from the drop down menu.  2. Click the sort button. | 9 | After the sort button is clicked, the spreadsheet is sorted by row by last name. |
| Sorts spreadsheet correctly by student ID number. | 1. Choose the student ID option from the drop down menu.  2. Click the sort button. | 9 | After the sort button is clicked, the spreadsheet is sorted by row by ID number, and if there are students without ID numbers, they are moved to the bottom. |
| Sorts spreadsheet correctly by school. | 1. Choose the school option from the drop down menu.  2. Click the sort button. | 9 | After the sort button is clicked, the spreadsheet is sorted by row by school. |
| Sorts spreadsheet correctly by sign in time. | 1. Choose the sign in time option from the drop down menu.  2. Click the sort button. | 9 | After the sort button is clicked, the spreadsheet is sorted by row by sign in time, and if there are students who have not signed in, they are moved to the bottom |
| Sorts spreadsheet correctly by sign out time. | 1. Choose the sign out time option from the drop down menu.  2. Click the sort button. | 9 | After the sort button is clicked, the spreadsheet is sorted by row by sign out time, and if there are students who have not signed out, they are moved to the bottom |
| Sign in time is recorded in the spreadsheet when the sign in button is pressed when a student is displayed. | 1.ID of student should be entered for a display to be shown  2.Sign In button shall be clicked  3.Check if the current time was stored in the correct cell in the spreadsheet | 10 | A time is written to the correct cell of the spreadsheet in the row that corresponds to the sign in status of the student whose information is displayed. |
| Sign in time is not recorded in the spreadsheet when the sign in button is pressed when a student is not displayed. | 1. Sign In button will be pressed, with no entering of a student ID  2.Check if the spreadsheet remains unchanged | 10 | A time is not written to the spreadsheet. All student data in the spreadsheet is not modified in any way. |
| Sign out time is recorded in the spreadsheet when the sign out button is pressed when a student is displayed. | 1.ID of student should be entered for a display to be shown  2.Sign Out button shall be clicked  3.Check if the current time was stored in the correct cell in the spreadsheet | 11 | A time is written to the correct cell of the spreadsheet in the row that corresponds to the sign out status of the student whose information is displayed. |
| Sign out time is not recorded in the spreadsheet when the sign out button is pressed when a student is not displayed. | 1. Sign Out button will be pressed, with no entering of a student ID  2.Check if the spreadsheet remains unchanged | 11 | A time is not written to the spreadsheet. All student data in the spreadsheet is not modified in any way. |
| Error is displayed when an invalid ID is entered or scanned. | 1. Enter or scan an ID number which does not correspond to a student ID in the spreadsheet.  1a. Edge case: Enter a negative or zero number.  1b. Edge case: Enter a value with both numbers and non-numeric characters.  2. Click the submit button. | 12 | Error message occurs. All student data in the spreadsheet is not modified in any way. |
| Error is not displayed when an valid ID is entered or scanned. | 1. Enter a valid MCPS student ID number or scan a valid MCPS ID card barcode. The ID must match with a student ID in the spreadsheet.  1a. Edge cases: Enter the first/last IDs of students who attend Poolesville High School.  1b. Edge cases: Enter the first/last IDs of students who attend any MCPS school other than Poolesville High School.  2. Click the submit button. | 12 | Error message does not occur. All student data in the spreadsheet is not modified in any way. |
| Error is displayed when sign in/out is clicked and no student information was displayed. | 1. After performing steps necessary for preventing student information from displaying, click a sign in/out button.  1a. Edge case: Start the program and click a sign in/out button without performing any intermediate steps. | 12 | Error message occurs informing the user that the informing the user that there was no valid student to sign in. All student data in the spreadsheet is not modified in any way. |
| Error is not displayed when sign in/out is clicked and student information was displayed. | 1. After performing steps necessary for displaying student information, click the submit button. | 12 | Error message does not occur. |

9. Suspension criteria and resumption requirements

A series of test cases for an item will cease when one case in a series fails. When this happens, the programmer will be notified about the test in the format described in section 18.1. Testing will resume when the programmer notifies all members of the testing committee about the error that has been fixed through a report in a format aligning with the methods described in section 18.1. Resumption will consist of restarting the series of tests for an item, including retesting cases of the item that have already been tested.

10. Test Deliverables

Along with the Software Test Plan, the test deliverables include the Software Test Cases as well as their results (when finished).

11. Remaining test tasks

FCC still needs to test the code in the methods described in section 8, fix it, and repeat the process until the code is without bugs.

12. Environmental Needs

All testing must be done on a Chromebook device, where the program will be running at the event. Some features are able to be tested on any functional device with Chrome, but all final and official tests will be performed on a Chromebook. The only special hardware needed for the testing process is the external scanner for the Sign In process. Only features #4-#7 will be require the use of the scanner to properly test.

13. Staffing and Training Needs

No special training is needed for the testing of the program. Testers should be able to handle a scanner and know how to use one. Testers should also have a basic understanding of the project and be familiar with its code. The testers will only include members of FCC, who have worked on building the program, and members of Techleon, who will need to work closely with the project to see if both programs work with another effectively.

14. Responsibilities

There is no individual person with the lead tester position in FCC. The responsibilities of the tests will be evenly distributed among the Project Manager, System Analyst, and Lead Programmer( names listed on cover page). There are no specific features that either of the members will be responsible for. The Project manager will be in charge of making sure all tests are done properly and on schedule. The manager will also be in charge of working with Techleon and mantainting the connection between FCC and Techleon. The manager is also responsible for the delivery of the testing document to the client and parent corporation. The System Analyst will make sure the tests are accurate and are in accordance to the project’s goals and client’s needs, and will evaluate the test’s effectiveness. The analyst will also be responsible for communicating with the client for the specifics of the tests. The Lead Programmer will be in charge of making the sure the problems detected by the tests are solved.

15. Schedule

The programmer ensures that the code is completed by June 3, and testing is to be culminated by June 5. This will ensure that testing finishes on time. All schedule plans will be monitored by the manager.

16. Planning Risks and Contingencies

The first risk involved with the project is the absence of a lead tester in FCC. The absence of this role means that all members of FCC must contribute to testing. If a member is not able to perform testing beforehand

As a main programmer normally would not have access to the testing document, the project manager and systems analyst must take on a larger volume of testing responsibility to ensure that the programmer does not perform tests insufficiently due to discretion. The systems analyst must make certain that all non-optional edge cases are completed.

17. Approvals

The product cannot be considered complete until the following agree that the project can continue to its next stages. The following parties must sign their names.

|  |  |
| --- | --- |
| Parth Agarwal, FCC project manager |  |
| Gene Yu, FCC systems analyst |  |
| Grey Kienzle, main programmer |  |
| Liz Attumalil, project manager of Techleon |  |

18. Appendices

18.1 Test format

Date: <Day of week, mm/dd/yy>

Name of person performing test: <Name of person>

Role of person performing test: <Role of person>

Feature tested: <Feature shorthand in ten words or less>

Description: <Purpose and significance of feature, one to three sentences>

Section of SRS fulfilled by feature: <“4.X”, according to the SRS, where X is a number>

Fulfills SRS requirement: <“True” or “False”>

Screenshot of program window before <action performed>: <PNG image>

Screenshot of program window after <action performed>: <PNG image, or “N/A if not applicable”>

Screenshot of spreadsheet before <action performed>: <PNG image>

Screenshot of spreadsheet after <action performed>: <PNG image, or “N/A” if not applicable>

Failure description: <If necessary, for each obstacle, provide description of why the program does not fulfill its SRS requirement. Describe each errors/alert in one to three sentences.>

Additional comments to programmer: <Optional>