Software Requirements Specification

for

Prom Sign-in project



Falcon Computer Company: Parth Agarwal, Grey Kienzle, Gene Yu

Poolesville High School

As of 5/30/2019

Version 1.1.0

|  |  |  |  |
| --- | --- | --- | --- |
| Version release | Major party | Major changes | Date |
| 0.1.0 | Grey Kienzle, Parth Agarwal | Sections 1-2 | 4/28/2019 |
| 0.1.1 | Gene Yu | Added sections 2.6 and 2.7 | 4/28/2019 |
| 0.2.0 | Gene Yu | Added sections 1.5 and 1.6  Wrote section 3.1 (incomplete) | 4/29/2019 |
| 0.3.0 | Gene Yu  Grey Kienzle, Parth Agarwal | Sections 3.1, 3.2  Section 4, adding features | 4/30/2019 |
| 1.0.0 | Gene Yu  Grey Kienzle, Parth Agarwal | Sections 3.3, 3.4  Section 4, adding features,  Section 5 | 5/1/2019 |
| 1.0.1 | Grey Kienzle | Updated figure 4-1, removed section 4.8 | 5/30/2019 |
| 1.1.0 | Gene Yu | Made sections 2.1 and 4 more specific | 5/30/2019 |
|  |  |  |  |
|  |  |  |  |

Table of contents

1. Introduction
   1. Purpose……………….3
   2. Document Conventions……………….3
   3. Intended Audience and Reading Suggestions……………….3
   4. Product Scope……………….4
   5. References……………….4
   6. Definitions and acronyms……………….4
2. Overall Description
   1. Product Perspective……………….6
   2. Product Functions……………….6
   3. User Classes and Characteristics……………….6
   4. Operating Environment……………….7
   5. Design and Implementation Constraints……………….7
   6. User Documentation……………….7
   7. Assumptions and Dependencies……………….7
3. External Interface Requirements
   1. User Interfaces……………….7
   2. Hardware Interfaces……………….9
   3. Software Interfaces……………….9
   4. Communications Interfaces……………….10
4. System Features
   1. Color and Appearance……………….10
   2. ID Entrance and Scanner……………….10
   3. Student Picture Displayed……………….11
   4. Student Information Displayed……………….11
   5. Data Table……………….11
   6. Sorting Options……………….12
   7. Sign-In Status……………….12
5. Other Nonfunctional Requirements
   1. Performance Requirements…………….12
   2. Safety Requirements……………..12
   3. Security Requirements…………...12
   4. Software Quality Requirements…………...12

# 1. Introduction

1.1 Purpose

The purpose of this document is to specify all details and features of the Poolesville High School Prom Sign-In computer program in development by FCC for client Mrs. Alexandra Hicks. The document provides information regarding the project’s purpose, requirements, and desired functionality.

1.2 Document Conventions

This Document was created based on the IEEE template for System Requirement Specification Documents.

1.3 Intended Audience and Reading Suggestions

This document is intended for the client, Mrs. Hicks, but is appropriate for any high school student, any high school teacher, or any programmer who is interested in the product’s internal components. Programmers, read sections 1.5, 3.1, 3.3, 4, 5. All others, read 1.6, 3.1, 4.

1.4 Product Scope

The Prom Sign In program is a tool to increase the convenience of checking in students and other attendants of Poolesville High School’s annual prom night. Standard users, which include Mrs. Hicks and Poolesville High School’s security staff, will use a scanner tool to verify attending students on prom night while entering. The program provides individual data on attendants and is used for verifying students who purchase prom tickets as well as for maintaining the security of Poolesville High School on prom night. Users will be able to sort the students based on information fields. The program also makes the student information accessible to all staff involved in managing lists of attendants. The current solution does not allow Mrs. Hicks to access these lists on prom night.

1.5 References

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Title of document | Date | Author | Publishing organization | Status or report number or ISBN or version |
| IEEE Software requirements specification Template | 1999 | Karl E. Wiegers | Microsoft Press | [0-7356-0631-5](https://users.csc.calpoly.edu/~csturner/courses/308w09/Ch9_DocReqts.pdf) |
| IEEE Standard for Software User Documentation | 2001 | 1063\_WG - Std for Software User Documentation Working Group | Institute of Electrical and Electronics Engineers | [1063-2001](https://nciphub.org/files/Y2ZhYTQ4OGY5NzJkODBhN2QwOGUxNGZlOWRjNjkzYTFjYzIwMTkzZjA4ZmEwNTM0ODM5YjQ1ZTZiYWQyZDdiNzovd3d3L25jaXBodWIvYXBwL3NpdGUvY29sbGVjdGlvbnMvMTEzMi9JRUVFX1N0YW5kYXJkMTA2My5wZGY=) |
| Prom sign-in program user manual | 2019 | Falcon Computer Company | Falcon Computer Company | 1.0 |
|  |  |  |  |  |

1.6 Definitions and acronyms

* FCC: Falcon Computer Company
* SRS: software requirements specification
* UI: user interface, the features, hardware, and virtual space that facilitate the direct exchange of information between a user and a computer
* GUI: graphical user interface, a type of user interface that provides direct visual feedback to the user
* Computer software: electronically stored data or computer instructions
* Linux: a name that describes various open source operating systems built around the Linux operating system kernel
* Kernel: the computer program that manages the rest of an operating system
* Open source software: software that can be copied, modified, and shared without restrictions
* Browser: web browser
* Frame: also known as a group box, a visual window in a GUI that holds one or multiple graphical elements
  + E.g. a browser window
* Data encryption: the process of translating data to a form that is harder to interpret. Reversing encryption requires a pattern, password, or key.
  + Key: information that specifies how to encrypt/decrypt data
* HTTPS: Hypertext Transfer Protocol Secure AKA Hypertext Transfer Protocol over TLS, is a communication protocol which encrypts information sent between clients and web services. The only public information available is the user’s IP address and the domain name of the website.
  + TLS: Transport Layer Security, the successor of Secure Sockets Layer (SSL), is a protocol of data encryption. Before sending data, the sender and receiver agree on a shared key to, respectively, encrypt and decrypt the information so that only the sender and receiver know what the information is.
* FTP: File Transfer Protocol, a standard protocol for transferring computer files over the Internet
* Cloud computing: a term that refers to storing, reading, and writing data stored on the Internet (the cloud) rather than on a local hard drive. In this way, the data can be accessed from any computer.
* File system: a computer’s hard drive cataloging system which contains references of all directories and files in the drive, usually in a hierarchical manner.
  + Directory: also known as a folder, a file system structure that contains relative references to itself, all of its subdirectories (children), all of its files it stores, and the directory it is contained in (parent).
  + Root directory: hierarchically, the top-most directory in a file system. The root directory has no parent directory.
* RAM: random-access memory, also known as system memory or just memory, a physical part of a computer which stores volatile (erased when computer power is interrupted) data. Writing and reading data in RAM takes almost the same amount of time.
* Hard drive: also known as a hard disk or hard disk drive, a physical data storage device which uses magnets to store/retrieve/modify persistent (does not depend on computer’s power state) data on one or multiple rotating platters.
* Cookie: a piece of data sent by a website and stored in a temporary file directory on the user’s computer. This allows websites to remember information specific to the user and web browser.

# 2. Overall Description

2.1 Product Perspective

The goal of the project is to make student information accessible to our client as the client signs in or signs out students for the prom. The current old program for sign in does not allow the user to view student information. Mrs. Hicks is the main person who will be using this Prom Sign-in program. The development of the project will be centered around the needs and convenience of the client herself. The product is to be run on a Chromebook or a Windows 10 PC, replacing the old program which runs on an old Linux-based Dell laptop.

2.2 Product Functions



Figure 2-1: Flowchart describing general program function

2.3 User Classes and Characteristics

Our main user will be Mrs.Hicks, and she will use the tool for prom night to be check in students with MCPS student IDs using her scanner. She is a teacher that is in charge of prom and other student activities. Any other teacher involved in managing who attends will also use the program. Other users will include teachers or staff members that need to access information of the students attending prom, such as Mrs. Levine, the principal of the school, the secretary of the school, and the security team. Other programmers or students that require a solution for checking in prom attendants can use the program in other schools with their respective staff members, as well.

2.4 Operating Environment

The program will run on Google Chrome, and the program will require at least standard Google Chrome version 70 or above. The program runs on any operating system capable of running Google Chrome as specified before. See the last paragraph of section 3.3.

2.5 Design and Implementation Constraints

There are several constraints that limit the way the program is written. First, because the program must function on an MCPS-order Chromebook, it must run on said Chromebook without violating MCPS website constraints. In addition, the unpredictability of MCPS-provided Chromebooks makes them more of a challenge to test on.

2.6 User Documentation

The client, Mrs. Hicks, will receive a hard copy and a persistent online copy of a user manual created based on the 1603-2001 IEEE Standard for Software User Documentation. (See section 1.5).

2.7 Assumptions and Dependencies

Creation of the product depends on unbroken access to the internet, as all spreadsheet data will be stored on Google Drive and the program itself will be stored in the Google Cloud service.

Guests from schools other than Poolesville High School are linked with students who bring them. Guests and their proxies sign in and sign out at the same time.

# 3. External Interface Requirements

3.1 User Interfaces

The frame of the program GUI is contained in a vertically scrollable browser tab. This frame’s size can be altered by dragging the edges of the browser window or by toggling the browser’s fullscreen function. Unless otherwise mentioned, all of the following elements are separate from components of the browser’s GUI. Every screen of the program has a help button at the bottom right corner accompanied by an about button. The about button triggers a pop-up menu that provides background information on the program and displays contact information of FCC. The help button detects the current screen and opens a new browser window that contains the relevant section in a help catalog (see section 4). For further help, the user can click a button located in the lower right corner of the help and about screens to open the user manual. Every screen, if applicable, has a back button at the top left corner and a forward button directly to the right of the back button. To refresh a screen, the user can use the browser’s refresh button.

Text is printed directly on the frame or on panels on the frame. Images can appear directly on the frame or on panels. Sections of spreadsheets can appear on frames and panels. Frames and panels are scrollable.

All warnings and error messages appear as pop-up windows. The bottom of each such pop-up window always contains an OK button and/or Cancel button center-justified on the bottom of the window, which closes the window. When there are multiple buttons in a pop-up window, buttons line up along the bottom of the window in a center-justified manner. Clicking the X button in the upper left of the pop-up window performs the same action as clicking an OK or Cancel button.

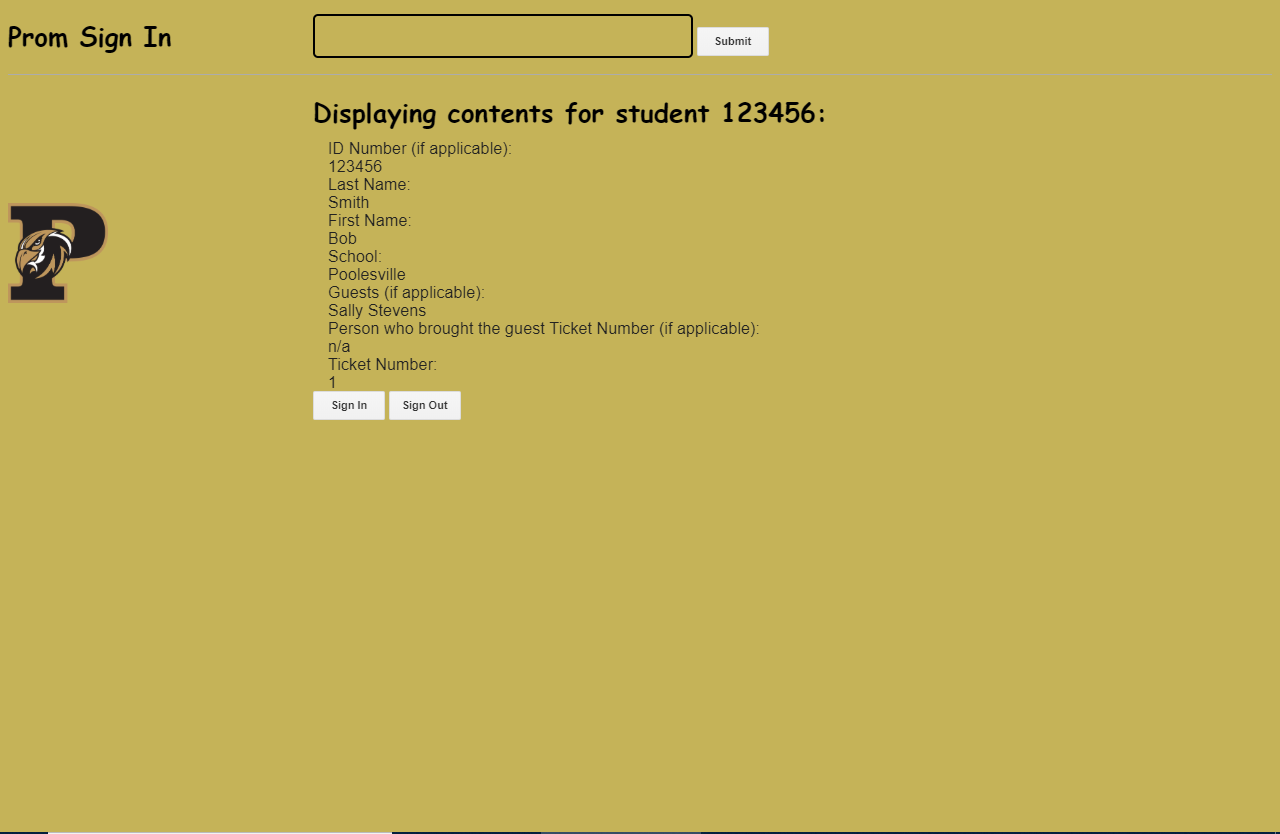


Figure 3-1: An example window of the Prom Sign-in program. Pictures of students would be shown on the right side of the program, and guest information would also be shown.

Keyboard shortcuts

Hyphens (-) indicate pressing multiple keys at the same time.

|  |  |  |
| --- | --- | --- |
| Shorthand | Longhand | Function |
| ctrl-C or ⌘-C | control-C or command-C | Copy text, images, table cells, spreadsheet cells to computer’s clipboard |
| ctrl-V or ⌘-V | control-V or command-V | Paste contents from computer’s clipboard |
| ctrl-W or ⌘-W | control-W or command-W | close current browser tab |

3.2 Hardware interfaces

The program is compatible with any physical built-in/USB/Bluetooth alphanumeric English keyboard with an enter key, cardinal directional arrow keys, escape key, and all typewriter keys (with their respective standard secondary shift symbols): all 26 English letters, digits from 0 through 9, control/command key, shift key, caps lock key, tab key, alt/option key, all English punctuation marks, backslash key, backspace/delete key, spacebar. The user may require function keys (F1 through F12), system keys, and application keys to make use of all keyboard shortcuts.

The program is compatible with any built-in/USB/Bluetooth trackpad/touchpad and any USB/Bluetooth mouse. The program is compatible with any computer/TV monitor as well as any type of speaker that is compatible to the user’s computer. The user’s computer must have a physical power button and at least two USB 2.0 compatible ports not used by the computer’s keyboard, mouse/trackpad, monitor, or speakers.

3.3 Software Interfaces

The Prom Sign-in product is written in Google Apps Script (file extension .gs), a programming language based on JavaScript. The HTML and CSS components of the product are also written in the Google Apps Script project development environment. Apps Script API has methods that can create/modify/execute Apps Script projects and remotely execute Apps Script functions. Applications using the Apps Script API to call Google script functions must be stored in the same project as those scripts. The Prom Sign-in product is a web application which creates, reads, and manipulates data in Google Sheets files stored in the user’s Google Drive.

Google Apps Script is part of G Suite, a package of cloud computing tools and various apps, including Google Drive. Google Drive is an online file storage service. G Suite is part of the Google Cloud service. Google scripts run only on Google servers. For information on how data gets transferred over the internet, see section 3.4.

The user’s web browser application can be executed from any directory on the active hard drive’s file system. Web browsers store cookies on the user’s computer but otherwise take space in RAM. Separate browser tabs act as separate tasks in memory.

3.4 Communications Interfaces

The program and all of its files are accessible by users and by the program itself on Google-owned domains over HTTPS connections. For communications by the program, HTTPS is strictly preferred unless it is unavailable. See section 1.6 for information on the above network protocols. The program runs on Google Chrome version 70 or above.

# 4. System Features

4.1 Color and appearance

The background of the application must be black and gold and contain a Poolesville Falcon logo on the left hand side of the program. When the program is run, the background of the user interface will be Vegas gold with a black text overlay, with a Falcon in the corner. This is of low priority because the program's functionality is not affected by appearance.

4.2 ID entrance and scanner

The student search system must be located at the top of the program. There will be a box for text entry for the student’s ID or the student’s ticket number at the opening display of the program. The user must be able to toggle between searching by student ID or ticket number by the use of an HTML radio input. The text field must have a submit button. The student will either enter his/her 6-digit number or scan his/her MCPS student ID card, then click the submit button. It is required to work with an external scanning device and a standard keyboard, but only one will be required for the program to function. This feature is essential for the program, as it is the only means for student identification in the program itself.

4.3 Student picture displayed

When a student ID/ticket number is scanned or entered, an image of the student with that ID or ticket number will appear as another method to ensure that the student whose ID was scanned or entered is the same student who scanned or entered the ID/ticket number. This is moderately important as it provides visual confirmation that the ticket holder is the person who bought the ticket. This picture shows up on the right hand side of the program.

4.4 Student information displayed

The user will be able to view a student’s basic information and data directly after the sign in. Alongside the student picture, there will the following information which will be pulled from a Google Sheet and displayed in the following order: student ID number, last name, first name, school, guests the student has, the information of the guests, ticket number of the student, and whether or not they are signed in/out. For each guest, the same information mentioned in the previous sentence must also be displayed in the same order specified. The information will show up in a pop-up box on the display for the user to view, and the box can be open and closed. This is a high priority due to it providing a manner for the user to view student info on the spot, the feature that is missing in the current sign-in program.

4.5 Data table

Student data will be stored in a Google Sheet, giving the user the ability to see all student information in one place. This data will be accessible via a direct hyperlink on the program's opening page. The data table is high priority as it is a necessary source for information that keeps the prom attendance system organized and secure. The data table has the following information headers for each student in columns from left to right:

* ID Number (if applicable)
* Last name
* First name
* School
* Guests (if applicable)
* Person who brought the guest Ticket Number (if applicable)
* Ticket number
* Sign In Time
* Sign Out Time
* Picture access ID

The data table would also store the piece of information typed in the program’s text field in cell Q1. This cell must be cleared if the program is closed.

4.6 Sorting options

A user will be able to access a seperate student table with all the previous mentioned information, along with sign in and sign out times. There will be multiple ways to be able to categorize and group the students together with specific sorting options provided. The user will be able to sort the students by all the data fields listed in 4.5, allowing the user to determine statistics such as who is signed in/out. This is of high priority because it allows external users such as the security team to view the statistics they want.

4.7 Sign-in status

When a student ID is scanned or entered, buttons will be displayed allowing the user to sign in/out the student whose ID was scanned or entered. It will also write to the data table the time at which the student signed in/out. This is high priority as it keeps a record of where students are. The sign in/out buttons must not be shown before a student ID is typed in.

# 5. Other Nonfunctional Requirements

5.1 Performance requirements

The program must take up no more than 50 megabytes of RAM according to the user's computer's task manager application. Slowing shall be unnoticeable or nonexistent, and any buttons clicked will fulfill their task within one second on a device unaffected by major errors.

5.2 Safety requirements

The user should not place the scanning tool in a manner that emits a laser into any person’s eyes.

5.3 Security requirements

The Google Sheet of student data must only be accessible to FCC, Techleon, and Poolesville High School staff. Students not involved in the prom attendance management are not allowed to view student information.

5.4 Software quality requirements

The program works on all devices with Chrome version 70 and above. The program is expected to be used for future years at PHS also.