Software Requirements Specification

for

Coronavirus Vaccination Integrated System (CVIS)

Version 1.0 approved

Prepared by Kaleb Miller, Robert Silvey, Carter Smith, Anthony Telerico

Software Engineering, Kent State University

March 4th, 2021

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Robert Silvey	3/4/21	Starting work on document	1.0
All Group	3/15/21	Filling in info from part 2	1.1
Kaleb Miller	4/17/21	Preliminary work on part 4	1.2
Carter Smith	4/17/21	Preliminary work on part 4	1.2
Anthony Telerico	4/18/21	Preliminary work on part 4	1.2
All Group	5/1/21	A lot of progress towards part 4	1.3
All Group	5/3/21	Finished part 4	1.4

1. Introduction

1.1 Purpose

The purpose of the software is to optimize the distribution, scheduling, and shipment of vaccinations to minimize the Coronavirus spread. (v1.0). The intention of this software is to hasten the ability of Kent and all of its remote campuses in spreading the vaccine for the Covid-19 virus. This includes vaccinating all students and facilities through precise and non-overlapping scheduling, constant refreshes of vaccine stock, and dedicated vaccine brands through specific campuses.

1.2 Document Conventions

When writing this SRS, no specific subsection when highlighted but specific sections were held towards the end to ensure that correct documentation was performed within the SRS document. All aspects revolving around the general code not; not overviewing the general purpose and requirements were saved towards the end in order to solidify peak accuracy in the SRS documentation.

1.3 Intended Audience and Reading Suggestions

This SRS is intended for developers, project managers, and users who wish to receive further documentation on the purpose, planning, and execution of the software meant to hasten the herd immunity of the Covid-19 pandemic. Those who have concerns with the potency of the software can read this to ensure that they are safely and securely receiving the vaccine along with thousands of other students around Kent. The SRS contains documentation that specifics on how the software is run, as well as documentation about personal information security, future plans for the software, and any issues that may currently exist in this Alpha-build. It is suggested to read the software in its entirety, but those who want reassurance in the validity of the software can skip over sections two and three. Sections two and three specify the functionality of the software and are not necessarily understandable to the general public.

1.4 Product Scope

The purpose of the software is to optimize the distribution, scheduling, and shipment of vaccinations to minimize the Coronavirus spread. The goal is to vaccinate as many people as possible.

1.5 References

All references were provided by Kent State University facility in order to ensure a sound software development process. All information and questions regarding the software engineering process and software itself can be directed to the Computer Science branch of Kent State University. All styles, standards, and system requirements were approved and followed through by Kent State University standards and user no outside resources.

2. *2. Overall Description*

2.1 Product Perspective

This software is a new self-contained product. This product uses software such as PHPMyAdmin to ensure its functionality but does not replace any existing software. This product's intent is to provide an innovative way for users among the Kent campuses to schedule appointments and hasten the vaccination process in order to create herd immunity among the masses of the city of Kent.

2.2 Product Functions

- Book a vaccination appointment
- Check eligibility for vaccination
- Prompt rescheduling of appointments
- Order more vaccines
- Send a FlashAlert and post a message
- Prepare and print vaccination appointment schedule
- Compute money spent and earned
- Visualize data

2.3 User Classes and Characteristics

Class Specifications

Class name:

Patient

Documentation:

Definition:

Either a student registered at any KSU campus, or a faculty or an employee, either part time or full time, that works for any Kent State University campus.

Constraints:

No additional appointment is permitted within the year, when the KSU-HS patient has completed the vaccination treatment (i.e., no more than 2 doses).

Class name:

Insurance

Documentation:

Definition:

Does the patient have insurance.

Constraints:

Class name:

Campus

Documentation:

Definition:

A campus is a Kent State Campus that holds multiple stations.

Constraints:

The campus must specify whether it is the main campus or a regional campus in order to check for eligibility of vaccine orders.

Class name:

Appointment

Documentation:

Definition:

An appointment is a scheduled date and time for any patient to receive a vaccination of a certain brand at a certain campus.

Constraints:

Appointments must inherit data from a patient such as their student ID and the integer of what vaccine they are on (0,1,2) and contain a date as well. Each station can deliver up to 50 vaccine doses per day. A vaccination appointment consists of 10-minute sessions between 8:00am and 18:00pm, Monday through Friday.

Class name:

Vaccine

Documentation:

Definition:

An injection to boost immunity to Coronavirus.

Constraints:

Vaccines are supplied from pharmaceutical companies in limited quantities upon request. Requests are only valid when the station’s stock of vaccines is below 50.

Class name:

DataBase

Documentation:

Definition:

Accesses the SQL database.

The SQL database has 4 tables: appointment, campus, logins, and users.

Appointment has columns: AppointmentID (auto-increment), UserID, Campus, AppointmentDate, Appointment Time, VaccineBrand, Complete.

Campus has columns: CampusName, isRegional, VaccinesOnHand, VaccinesGiven, Revenue, VaccineBrand, deliveryDate.

Logins has columns: Email, Password, ID (auto-increment), IsAdmin.

Users has columns: ID (auto-increment), Name, Vaccinations, Insurance, Brand, Completed

Constraints:

Class name:

User

Documentation:

Definition:

A parent class of patient for logging in that adds flexibility and future-proofing for any non-patient users to be added in the future.

Constraints:

Class name:

LogInMenu

Documentation:

Definition:

Opens up a GUI to let the user login with email and password.

Constraints:

Class name:

PatientMenu

Documentation:

Definition:

Allows a user to select an option to use the program

Constraints:

Class name:

ViewApptsMenu

Documentation:

Definition:

A menu that allows a user to cancel and reschedule appointments.

Constraints:

Class name:

AppointmentMenu

Documentation:

Definition:

Allows the user to create an appointment.

Constraints:

2.4 Operating Environment

The software will operate via PHPMyAdmin database software which will be used to imitate integration with the Kent Flashline platform. Windows, Mac, and Linux using a 32 or 64 bit operating system will be able to run this program. Users must have downloaded PHPMyAdmin as well as software such as Apache in order to run the software. A simple GUI will appear and will allow the users to perform all actions that the software offers currently. No GPU is required for execution of this program.

2.5 Design and Implementation Constraints

Limitations in this project will include how much traffic the database will be able to handle. The program must account for how many people will use the program, If the database cannot handle the traffic the program could run into issues and fail.

2.6 User Documentation

An online guide will be made in the future to provide proper installation methods for PHPMyAdmin and other necessary software requirements in order to run the software. A visual guide in the form of a YouTube video will be linked with the project in order to allow users to have a seamless experience getting the program up and running. In the future, a link to Kent Flashline will be made, invalidating all of the previous guides and allowing users to have a simplified entry experience in order to use the software. All notifications and notices involving the software will be sent via email once the user has registered with the software. Notifications such as successful scheduling date/time, upcoming vaccination, and vaccine restock will be sent to users that register with the software.

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

2.7 Assumptions and Dependencies

The software operates under the dependency of a supply of vaccines from pharmaceutical companies.

3. External Interface Requirements

3.1 User Interfaces

A GUI interface will be available for ease-of-use aspects for the users. Each user interface will allow users to interact with simple pages that allow them to view all aspects of the software without limitation. There is currently no means of returning to a previous tab in the software without closing and re-opening the interface. In the future, this feature will be provided along with extra Quality-of-life improvements to allow the user experience to be more seamless.

3.2 Hardware Interfaces

Our local sql database simulates how users would log on to a website like flashline via their own hardware.

3.3 Software Interfaces

Our simulated flashline website will connect to an sql database. The simulated website and the database go back and forth with data to determine what appointments are available as well as recording the appointments in the database.

3.4 Communications Interfaces

The alert class will send out emails with an SMTP that are SSL certified.

4. System Features

4.1 Log In

4.1.1 Description and Priority

Lets the user log into the system to perform actions.

High priority.

4.1.2 Stimulus/Response Sequences

The user calls the LogInMenu.

The LogInMenu gives the user a GUI to enter an email and password into.

The User enters their credentials.

If the credentials entered match a login of the database, the user is logged in. If the credentials do not find a match, then an invalid login message is displayed.

4.1.3 Functional Requirements

REQ-1: Display a GUI
REQ-2: Accept credentials
REQ-3: Check credentials vs the database
REQ-4: Successful login
REQ-5: Return invalid login error

4.2 Schedule Appointment

4.2.1 Description and Priority

Users are able to schedule an appointment.
High priority.

4.2.2 Stimulus/Response Sequences

User logs in via 4.1.
GUI displaying 4 options including schedule an appointment.
User selects Schedule an Appointment.
System determines available appointment dates.
GUI displays the calendar of available appointment dates.
User picks a date.
System determines available appointment times.
GUI displays available times.
User picks a time.
GUI provides confirmation.
User oks.

System enters appointment into database.

4.2.3 Functional Requirements

REQ-1: Display GUI with an option to schedule appointments.
REQ-2: Determine available appointment dates by parsing the database.
REQ-3: Display GUI with appointment dates.
REQ-4: System determines available appointment times.
REQ-5: GUI displays available times.
REQ-6: GUI provides confirmation.
REQ-7: System enters appointment into database.
REQ-8: Cancel out of scheduling an appointment.

4.3 Reschedule Appointment

4.3.1 Description and Priority

Users are able to reschedule appointments. Medium priority.

4.3.2 Stimulus/Response Sequences

User logs in via 4.1.

GUI displaying 4 options including reschedule an appointment.

System fetches and displays user’s appointments.

User selects an appointment to reschedule.

Appointment is canceled.

GUI displays the calendar of available dates for appointments.

User has to pick a date.

GUI displays available times.

User picks a time.

GUI provides confirmation.

User oks.

System enters appointment into database.

4.3.3 Functional Requirements

REQ-1: Display GUI with all 4 options
REQ-2: Fetch and display user’s appointments.
REQ-3: Cancel an appointment
REQ-4: GUI displays the calendar of available dates for appointments.
REQ-5: GUI displays available times.
REQ-6: GUI provides confirmation.
REQ-7: System enters appointment into database.

4.4 Cancel Appointment

4.4.1 Description and Priority

Users are able to cancel appointments.

High Priority.

4.4.2 Stimulus/Response Sequences

User logs in via 4.1.

GUI displaying 4 options including cancel an appointment.

System fetches and displays user’s appointments.

User selects an appointment to reschedule.

Appointment is canceled.

GUI displays the calendar of available dates for appointments.

User has to pick a date.

GUI displays available times.

User picks a time.

GUI provides confirmation.

User oks.

System enters appointment into database.

4.4.3 Functional Requirements

REQ-1: GUI displaying 4 options
REQ-2: System fetches and displays user’s appointments.
REQ-3: Cancel an appointment.
REQ-4: GUI displays the calendar of available dates for appointments.
REQ-5: GUI displays available times.
REQ-6: GUI provides confirmation.
REQ-7: System enters appointment into database.

4.5 Order More Vaccines

4.7.1 Description and Priority

Campuses order more vaccines when their stock is below 50 or 150 depending on if the campus is regional or the main campus.

4.7.2 Stimulus/Response Sequences

Campus requests more vaccines in a quantity of their choosing.

Pharmaceutical companies accept or deny the request.

If campuses receive more vaccines then an alert goes out.

4.7.3 Functional Requirements

REQ-1: Request more vaccines
REQ-2: Send alert if

4.6 View Station Data

4.7.1 Description and Priority

Users view data on how many vaccines have been administered.

4.7.2 Stimulus/Response Sequences

Sign in with 4.1

GUI displays 4 options

Users select View Station Data

System shows a bar graph.

4.7.3 Functional Requirements

REQ-1: GUI displaying 4 choices
REQ-2: Generate a bar graph

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Built to be lightweight and suitable for any 32- or 64-bit machine. Will run on windows, mac and Linux machines, as well as any other operation system. All machines must require an internet connection in order to operate successfully. A loss of connection while inputting personal data will result in a loss of all data without exposure to personal information online.

5.2 Safety Requirements

Loss of data could result in exposure of personal data such as Email, name, and location/time data of the person who scheduled the appointment. if the system were to go down across the board, the user would need to reschedule an appointment at a later date as the failure of the software might foreclose the scheduling date of the user who had been vaccinated once. For users that have not been vaccinated at all, users will need to reschedule a vaccination date at a later time. If a full-scale outage were to occur and all scheduling were lost, estimated time before the person could get vaccinated might take up to a month or more past their original schedule date due to mass schedules from all previous people scheduled. For people who might have lost their date for their second appointment, this might mean that their second vaccination might occur potentially months after the first, preventing full immunity to the Covid-19 virus.

A back-up cloud-based server COULD be implemented to store all scheduling datas and user data so that a full-scale loss of data would not occur. If a breach in security were to occur, personal data would be lost without the use of encryption of user data, which is not currently implemented. User email, name, and location time data could be exposed with a potential breach of security so means would need to be taken to encrypt said data.

5.3 Security Requirements

Using this product implies that one is at risk of having their email and name exposed to potential hackers without the use of encryption methods not currently available or implemented. In the future, users must verify their email and a cloud based secure storage will become available. Users will also be offered to ensure the integrity of the account through two-factor authentication. Users are fully accountable for ensuring their account safety through two-factor auth when it becomes available.

5.4 Software Quality Attributes

Program will be able to be deployed on Windows, Mac, and Linux devices. It has been designed to be lightweight and able to run on any system. Updates and patches will be available on the web for download when they come around. Database will be up 24/7 and maintained on a dedicated host.

5.5 Business Rules

This product is exclusively used to schedule appointments for users and ensure a swift and efficient means of procuring 2 doses of the Covid-19 vaccine. Administrators led by the campuses have automatic systems in place to keep the vaccine in stock at all times to ensure a thorough and swift vaccination service. Vaccine brands are specified per campus so those who chose a vaccine brand will have to choose the vaccine to get vaccinated at. All campuses only offer one vaccine type; those who desire one vaccine brand over the other will need to schedule an appointment with the designated campus that offers the brand of vaccine that is desired.
<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

6. Other Requirements

SQL database must be maintained for users to be able to connect and schedule appointments.

Appendix A: Glossary

Patient	linked to Vaccine/Appointment	StudentID Name # of doses brand date vaccinated campus vaccinated at
Insurance	part of Patient	StudentID Company
Campus		Stations Regional/Main
Station	part of Campus	Total_Vaccines_Available Total Vaccinations Students Vaccinated Revenue Generated
Appointment	linked to Patient/Date/Vaccine/Station	ID Date Brand Dose Doctor
Doctor	part of Station Linked to Appointment	Name Vaccines_Administered Salary Station
Date	linked to Appointment/Vaccine order	Day Month year Time
Vaccine	linked to Patient/Appointment/Vaccine Order	Brand Price ID Date does_number
Data	Linked to Station	Total_revenue Total_Students_vacinated
Vaccine Order	linked to Vaccine/Date/Dependent on station	Vaccine Station Total_Price Number_of_Vaccines Elegibility_for_order Delivery_Date

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>