**Vaccine Clinic Information and Scheduling System (VCISS)  
Software Requirements Specification**

Ian Parish (ip), Jiacheng Wei (jw),

Thomas Renn (tr), Xing Qian (xq) - 4-12-2021 - v1.01

**Table of Contents**

1. SRS Revision History 2

2. The Concept of Operations (ConOps) 2

2.1. Current System or Situation 2

2.2. Justification for a New System 2

2.3. Operational Features of the Proposed System 2

2.4. User Classes 3

2.5. Modes of Operation 3

2.6. Operational Scenarios (Also Known as “Use Cases”) 3

3. Specific Requirements 5

3.1. External Interfaces (Inputs and Outputs) 5

3.2. Functions 6

3.3. Usability Requirements 8

3.4. Performance Requirements 9

3.5. Software System Attributes 9

4. References 9

5. Acknowledgements 9

# 1. SRS Revision History

**Date Author Description**

**————— ——— ————————————————————————————**

4-11-2021 ip, jw, tr, xq Created the initial document.

# 2. The Concept of Operations (ConOps)

## 2.1. Current System or Situation

Our world is currently suffering under the effects of a global pandemic. Fortunately, various vaccinations have been developed and are ready for mass distribution. However, this presents a new problem - systems must be created to coordinate the distribution of the vaccines in the most expedient, efficient, and appropriate manner possible. We propose a Vaccine Clinic Information and Scheduling System (VCISS) in order to tackle this problem that will offer clinic administrators a platform to manage traffic and appointments and will offer the general public a simple system to interface with for making appointments with local clinics or vaccination centers.

## 2.2. Justification for a New System

The website vaccinefinder.org is partnered with the CDC to provide locations of clinics or pharmacies that are offering vaccines and list the available stock at this location. This website is not always consistent with the actual availability of vaccine stock and available appointments, which can be seen from following the secondary links to get a scheduled appointment. The VCISS should provide vaccine appointment information based on actual user data and clinic data to provide a more useful user-experience to limit the time wasted finding an available provider. The VCISS website will provide user registration and login systems to facilitate the provision of records for users. It will keep actual user records to ensure that user information is up to date along with clinic information.

## 2.3. Operational Features of the Proposed System

The main core feature of the VCISS system is the appointment scheduling functionality. The VCISS will have users create an account or someway to interact with the system, so that they are able to interface with the scheduling system. This system will contain records of available time slots at clinics where the user can request to see available appointments in a certain area. The users will then be able to schedule an appointment at these open time slots. This scheduled appointment will be viewable by the account that made the request and will update the entire system to remove that appointment time slot from the clinics availability.

## 2.4. User Classes

There are a variety of different user classes for people seeking to schedule an appointment due to the mandate of different groups being eligible to receive the vaccine at specified times.

* Patients – This is a general term for someone who already has an appointment scheduled.
* Elderly – A user who is above the age of 65.
* Front-line workers & teachers – Workers that interact with the general public on a frequent basis.
* Under 65 with high-risk medical conditions – People who are at risk for complications with Covid-19 due to underlying medical conditions.
* General users – People not associated with any of the categories above.

## 2.5. Modes of Operation

There are 2 major user modes of operating:

1. User: The user already has an account and can access the scheduling system interface.
2. Non-user: The non-user has not set up an account or is not logged in and is therefore not able to access the scheduling system.

## 2.6. Operational Scenarios (Also Known as “Use Cases”)

There are 3 major use cases for the system:

1. **Non-user creates an account:**
   1. **Brief description:** This non-user case describes this user class would be able to create an account, in order to login into the system to set up a future appointment to be vaccinated.
   2. **Actors:** Non-users.
   3. **Preconditions:** 
      1. A non-user must have access to a valid email address.
   4. **Steps to Complete the Task:** 
      1. The non-user clicks the create new account by entering in the required information.
      2. User selects “Create Account” once all required information has been entered
      3. User receives verification email and follows the activation link
      4. User is served a confirmation page verifying that the account has been created
   5. **Postconditions:**
      1. The patient now has access to the appointment generation system and can book an appointment for receiving a vaccine.
2. **User schedules an appointment:**
   1. **Brief description:** This use case describes how a patient would use the VCISS to schedule an appointment to be vaccinated
   2. **Actors:** Users with access to a valid, verified account
   3. **Preconditions:**
      1. User has navigated to the VCISS main page and logged in with valid a valid account
   4. **Steps to Complete the Task**:
      1. User selects an onscreen button to navigate to the scheduler
      2. User selects relevant filters (such as vaccine type and distance from their saved address to possible clinics)
      3. User selects an onscreen button to initiate search and is served a results page
      4. Of the available options, the user clicks on a link to their preferred clinic’s info page
      5. User is presented with available times and can select one of them to schedule an appointment
      6. User receives confirmation via a confirmation page, as well as an email confirmation (or to their phone, depending on their preference)
   5. **Postconditions:**
      1. Users are successfully scheduled for a vaccination appointment at a clinic of their choosing.
      2. Users can log in to view or cancel their appointment information
3. **Patient wants to view or cancel their appointment:**
   1. **Brief description:** This use case describes how a patient would use the VCISS to view information for an already scheduled appointment, or cancel the appointment
   2. **Actors:** A patient with access to a valid, verified account that has a confirmed appointment scheduled.
   3. **Preconditions:**
      1. User has navigated to the VCISS main page and logged in with valid a valid account
   4. **Steps to Complete the Task:**
      1. User selects an onscreen button to navigate to their personal information
      2. Among the various account personalization options, the user can select on onscreen button to view their upcoming appointment(s)
      3. The user is served a page that shows their scheduled appointments, along with basic clinic information, and appointment time/date and a cancellation button
      4. If the user selects the cancellation button, they are served with a warning pop up message indicating that their appointment will indeed be cancelled, with accept or reject buttons
      5. If the user selects the reject button, their cancellation request will not be carried out and the pop up is dismissed
      6. If the use selects the accept button, they are served a cancellation confirmation page, and sent an email confirming their cancellation
   5. **Postconditions:**
      1. User’s scheduled appointment has been cancelled
      2. The appointment time is freed for the clinic

# 3. Specific Requirements

Unless otherwise specified, these requirements are absolutely necessary.

## 3.1. External Interfaces (Inputs and Outputs)

1. **Patient account (input/output):**
   1. The patient account is both an output of the system and used as an input to access the scheduling system.
   2. The source of the patient account is created the first time a user interacts with the system. The patient account can then be used with an appointment request.
   3. Valid ranges for this interface are verified accounts.
   4. The patient account can be verified and measured as a valid account by checking the system records.
   5. The patient account will be associated with an email address in the form of a string.
2. **Schedule request (input)**
   1. A schedule request is a request to see a list of available time slots and locations for creating an appointment request.
   2. The source of the schedule request will be sent from a user account to the system records.
   3. This input will contain a description of a geographical area, such as a state or city, along with other filterable options that the system can processes.
   4. The units of measure for a schedule request will be if the request when sent, generates response from the database.
   5. The data formats will be in the form of an API request.
3. **Schedule response (output)**
   1. A schedule response is the product of a request to see a list of available time slots for a specific clinic location or multiple clinic locations in a geographical area.
   2. The destination of the output will the user account that sent the request. The appointment time slots will be available from a database to maintain system records.
   3. This output will contain the addresses and time slots associated with the specific filtered parameters that the user entered in the schedule request to generate the schedule response.
   4. The units of measure for a schedule response are whether a response is generated for every request.
   5. The data format of a schedule response will be in the form of an API response.
4. **Appointment request (input)**
   1. An appointment request will be used to schedule a single appointment for a user account that contains the date, time and location of a possible appointment.
   2. The source of the appointment request will be generated by a user account.
   3. Valid ranges of inputs will be from a verified time slot, at a verified clinic location within the system by a verified user account.
   4. The appointment request can be measured by if the request was approved by the system.
   5. The date portion of this input will be in the format mm/dd. The time portion will be formatted as hh:mm in military time. The location portion will be formatted as an entity of 5 components:
      1. The numbered address
      2. The street name
      3. Town/city
      4. State
      5. Zip code
5. **Appointment Response (output)**
   1. The appointment response can have either an approved status or denied status. If approved, the response contains the date, time and location of the appointment. If denied, gives a short description of why the request was denied.
   2. Destination of output: The user account will receive a message, where they can view it later.
   3. The valid range of outputs are approved or denied, which are determined by the processing of an appointment request.
   4. The appointment can be measured by either being approved or denied.
   5. The appointment response can have either an approved status or denied status. If approved, the response contains the date, time and location of the appointment. If denied, gives a short description of why the request was denied.

## 3.2. Functions

1. **Account creation**
   1. Emails can be validated by using specified string formatting.
   2. Sequence for processing inputs:
      1. The user enters in all the required information they need to create an account.
   3. Error handling/response:
      1. In the case where the account creation does not work, they user will be prompted to try entering the information again.
   4. Relationship of outputs to inputs:
      1. The non-user enters in the required user information to identify distinct accounts. Once that is complete
      2. The user then has received the output of an account that they are able to use to access the scheduling system.
2. **Generate a schedule response**
   1. This user request will be verified that it has all the required fields to pass the schedule request.
   2. Sequence of operations:
      1. User sends the request to the database via an API request.
      2. The database queries for information given the parameters of the schedule request.
   3. Response:
      1. If there is an error in the request, there will be a response from the database describing the error. If it is with the request, the user must fix their schedule request. If it is the database has an error, then the user must wait for the problem to be resolved.
   4. Relationship of outputs to inputs:
      1. The schedule request is processed by the database.
      2. The database responds to the user account with a schedule response.
3. **Generate appointment response:**
   1. The inputs for this function are a valid appointment request, which can be validated via the system records to make sure there is an availability at the requested location and time.
   2. Sequence of operations:
      1. The user must generate a schedule request to see available time slots.
      2. From the available time slots, the patient must generate an appointment request that they are able to attend.
      3. The request is then validated that there is an open availability for the appointment request.
   3. Error handling:
      1. In the case where multiple requests are made simultaneously by different users, the user making the request with the highest priority will be given the appointment. If there is still a conflict, the user with the oldest age will be given an appointment.
   4. Relationship of outputs to inputs:
      1. The appointment request is validated at above (b).
      2. If the request is approved, this time slot is removed from the system records and given to the patient in the form of an appointment response.
4. **View appointments:**
   1. The input for this is a valid account that can be verified against system records, and the user must be able to log in for security purposes.
   2. The sequence for processing inputs is as follows:
      1. The user gains access to their account.
      2. The user then sends a request to the record system to view their current appointment.
   3. Error handling:
      1. In the case where a user does not revive a response, they may have to wait a certain amount of time before retrying their response as a possibility that there are to many users attempting to access this service.
   4. Relationship of outputs to inputs:
      1. The inputs are validated as above in section b.
      2. The record system then responds with the user accounts appointment time and location, if the user has set up and appointment.
5. **Cancel previously scheduled appointment:**
   1. The input for this can be validated by checking the record system to see if a user has an appointment to cancel.
   2. Sequence of processing inputs:
      1. The user’s account appointment if verified by the system, to make sure it exists.
   3. Error handling:
      1. The user may not have an appointment to cancel, so if the user attempts this, they will receive an error that there is no appointment to cancel.
   4. Relationship of output to inputs:
      1. The input is a request to cancel an appointment of a specific user account.
      2. The output is a response from the record system that notifies the user of a canceled appointment and adds the time slot back into the record system.
6. **Filter schedule response (not absolutely required):**
   1. This is an intermediate step to sending a schedule response, where the filter items are already validated prior to the request entering the filter.
   2. Sequence of processing inputs:
      1. The user sends a schedule request and adds the appropriate filter categories to the schedule request.
      2. The record system then applies the filter categories from the schedule request and generates the response.
   3. Error handling:
      1. The user may apply to many filters that removes all the available appointments from the schedule response. The response will let the user know that there are no available schedules for the given request.
   4. Relationship of outputs to inputs:
      1. The input is a schedule request with filters.
      2. The response is then filtered and outputs a scheduled response with only the specified filter categories.

## 3.3. Usability Requirements

1. 90% of the users should be able to create an account and schedule an appointment in under than 15 minutes if there are appointments available to be scheduled.
2. 95% of users should be able to cancel a scheduled appointment in under 5 minutes.
3. There should be at least 2 filterable categories for a schedule request.
4. Users should be able to view their appointment details within 5 minutes of placing an appointment request.
5. The scheduling response system should always be available during the week to see if there are available appointments (not absolutely necessary, but should be as close to this as possible).

## 3.4. Performance Requirements

1. Schedule requests for a small geographical location (under 100 clinic locations) should take no longer than 5 seconds to pull up available clinics within the area.
2. Schedule requests for a medium geographical location (100 to 1000 clinic locations) should take no longer than 1 minutes to pull up available clinics within the area.
3. When a user requests a filtered schedule request, the response should only include scheduling information relevant to the user’s request.
4. An appointment response should take less than 1 minutes to notify the user if they received the time slot for the location or not.

## 3.5. Software System Attributes

1. **Account security**
   1. It is necessary that a user be confident that their personal information is not accessible or obtainable by unauthorized persons or entities, and that authorized personnel are able to obtain and modify private personal information. We must strive for 0 security vulnerabilities.
2. **Reliability**
   1. The VCISS must be able to properly and flawlessly schedule appointments for any user. If the system fails, it is due to external inputs (clinic information) and is handled elegantly
3. **Usability**
   1. All demographics of people will be interacting with our system, and as such, must be simple and approachable for a user with as little as no technical background. A simple and straightforward approach is required

# 5. Acknowledgements

**Possible Minor Future Revision**