Final Project Documentation

**1.0 Project Overview**

My project is an application that tracks vaccinations in a pharmacy setting. The users for this would be pharmacy technicians and Pharmacist. They are used to using computer software to manage daily task, they would be able to understand simple programs after few uses. Overall goal would be to simplify and streamlined as possible, to make it work efficiently and quickly. The pharmacy setting is a fast paced environment.

The problem I was trying to solve was in direct relation to my work experience. I currently work as a pharmacy technician. On a daily basis we deal with vaccinations, and have to input them in a system that is slow and not always effective. I thought I would use this opportunity to create an application in a simplified form that would work as a solution to this problem. Hardware limitation for this project is mostly related to maintaining the code language in python and limited data structure options. It also is limited by the allowed time to work on this project.

My project allows a user to login, and then brings them to a main home screen. The main home screen display a ready queue for patients waiting to be vaccinated in a list format. It also provides buttons to logout, enter new patients, verify patients, and view completed vaccinations. The logout button takes the user out of the application and back to the login screen. The new patient button takes the user to where they can input new patient information and select what type of vaccination they want. When they submit information on this screen, it adds the new patient to the database, sends the user back to the main home screen, and displays the patient as ready for vaccination. This allows the pharmacist to view the list and see what shot to give each patient. The verify patient button is used after the vaccination is completed. It allows the pharmacist to document information regarding the vaccination. This button sends the user to a new screen where it displays the current selected patient information. This screen also displays new entry options for the remaining information needed for the vaccination to be completed. Once the user hits complete, it is updated in database, and they are sent back to home screen. That verified patient is no longer seen on the main home page, they are now moved to the completed vaccinations. They can be view by clicking the completed vaccinations button.

**2.0 Data Plan**

The data that needs to be manipulated is the patient information. The program should be able to handle new entries, edit entries, and organize it as well. I wanted to use an external file to handle the data which would allow me to work with python’s file management system. I, however, don’t want a full blown database as it seems unnecessary for the current project. Most information would be displayed in by an array and be able to access each individual patient through index. Each patient would need a name, date of birth, sex, city, state, status, vaccine, location of vaccination, date of vaccination, who administered it, and their title. On entry they would input information for name, date of birth, sex, city, state, status, and vaccine. Status would need a default value to display it is ready to be vaccinated. The remaining values would need placeholders till it is updated in verify patient.

I eventually settled with a comma-separated value file that would work for what I wanted. I used Microsoft Excel to edit the file PatientDatabase.csv. Using python read and write, I can then draw on whole rows or individual columns to edit or insert new data. It also works with file modes, which would limit me from overriding a file with the incorrect input.

**3.0 Visual Layout**

Entry

Home

Login

Verification

Completed

There are a five states of the application that interact. The five states are login, home, entry, verify, and completed. The application loads up in login state. This prompts to the user to login, once done it sends them to home. Home can branch off to entry, verify, completed. Entry, verify, completed, or back to login. Verify, completed, and Entry can send you back to home or back to login.

The login state has one main window that centers the login screen. There are three labels, two entry, and one button. The entries have default values, so you can log in because there is no user database. The submit button, when clicked, draws data from the entry fields and checks them against a list of authorized users. If valid sets user status to login is true, and passes it to the home state. If false, it stays on the same state, and uses the final label to display the error.

The home state loads up a top display and a bottom display. The top and bottom dThe top display has two labels and three buttons. It can also only two labels and two buttons depending on if the status is home or entry, verify, and completed. The first label is the username. The second label is the status of the application. The first button on the home state is patient entry. This will send the user to patient entry state and change status to patient entry. The second button on home state will send to completed vaccinations and change status as well. The final button on home state will log the user out and send them back to login sate. For all other states besides home, it removes the first and second button and replaces it with a return to home state button. All states besides login maintain the top display. The bottom display is the second part and it shows a list box of all info the csv file with a status field of zero. The info displayed is the first name, last name, and the status of the vaccination. The patients can be selected and sent to the verify state by clicking the verify button below it with them highlighted.

The entry state bottom display has seven labels, seven entry, and one button below them. They are all displayed using a simple pack method with padding. The seven labels and seven entry are for the seven fields needed to input a new patient: first name, last name, date of birth, sex, city, state, and vaccination type. The button is for submission of the patient. When clicked it gathers all the info and sends it to patient entry method. The patient entry method puts all info into a list and adds a default value of zero to an extra eighth field. It then writes to the csv file with a write command. After it is done, it sends back to the home state where you can see the new patient in the list box

The completed state bottom display has one list box. It is displayed with a pack method. The list box reads the csv file and outputs the first name, last name, status, and date of completion for all entries with a status of one.

The verify state bottom display has eleven labels, four entry, and one button. All are displayed with pack method and padding. The first seven labels are populated by the patient passed to the state. They are the first name, last name, date of birth, sex, city, state, and vaccine. The last four and their corresponding entry fields are for the remaining input required to finalize the vaccinations: location of vaccination, date of vaccination, who administered, and title. It also changes the status from zero to one. When the button is clicked it gathers all the data, verifies there is data, and submits it the csv file via write command.

The whole structure is controlled by an if statement. When statuses are changed in the application, the user is sent back to the beginning of the if statement. It then displays the correct state depending on that status.

**4.0 The Project**

All files included:

Richart\_Final\_Project.py

PatientDatabase.csv

Final\_Project\_Documentation.doc

Login Information:

Username = user1

Password = password

Extra Info:

All fields require data entry for submission, don’t leave any empty.