User Functionalities:

Firstly, we created a global vector for Users and initialized it with all the data from the user\_people.csv in it. After that, we used the login function in int main() to loop through all three of the vectors for Admin, Management, and User to match the person logging in with their respective functionalities. For defining the user functionalities, we created a user class where a user could change their password, change their vehicle information, and view their vehicle information. We also created the Ticket class for the ticket functions and used global vector as the data structure for the ticket information.

User Function menu:

Once we find a match in the User vector for the user logging in, we call this function for their allowed functionalities. Inside this function we made a do while loop and call the menu display function from the User class to output the options for the user. Then we ask the user for their option choice. We use if and else statements to use the user input to carry out the particular function. For example, if the user inputs 1, it calls the change password function from the User class. If user inputs 15, it breaks the do while loop and logs out. We also have input validation to make sure the user input matches one of the options.

Changing user password:

We created this function in the User class and call it from the user function in int main(). Once the user logs in, we store their user id and pass it to user function and then to this function as an argument. In this function we loop through the User vector and find the user with the ID of the logged in user. Then we get the new password as user input and call the set password function in User class to change the password.

Changing Vehicle Information:

We basically follow the same format for this function as the change password function. We search the user info in the vector using the ID passed to this this function from the login. We then get the user input for the new vehicle and use the set method in User class to change the vehicle information.

View Vehicle Information:

To view the vehicle information, we pass the ID of logged in user to this function just like the previous functions and use that to search through the user vector. Once found, we output the information for just that user.

View Ticket:

For viewing the ticket, we use the Ticket class and the Ticket vector data structure. We first call the search plate function in the User class and pass it the logged in user’s ID. Search plate function returns the plate number for this user which we then pass to the view ticket function in the Ticket class. This function then loops through the Ticket vector and displays all the ticket instances for this particular plate number.

Pay Ticket:

We follow the same procedure for this as the View ticket and search for the plate number and pass it to this function along with the user ID. We then display the ticket information for this user and ask the ticket number for the ticket the user would like to pay. After user inputs a valid ticket number, loop through the ticket vector and find the ticket. We then ask the user if they would like to pay now for this ticket. If user inputs yes, we change the status for this ticket in the ticket vector.

Appeal Ticket:

We ask the user for the ticket number they would like to appeal and loop through the ticket vector to find the ticket. Once found, we ask the user for the appeal information and reason and store it in the ticket vector using the set methods of the Ticket class.

View Appeal:

In this function, we ask the user for the ticket number for which they would like to see the appeal information for and use that ticket number to find the ticket in the Ticket vector. Once found, simply display the appeal information for that ticket using the get methods of the Ticket class.

Log out:

When user chooses this option in the function menu, we beak the while loop and log out as this user.

Register User:

For this function, we ask the user for the information and use that information to create another instance of the User class and store it at the end of the User vector.