**1. public static void main() <Main menu>**

BEGIN

DISPLAY “Covid-19 Vaccine Registration System”

DECLARE int variable inp, set to 69

DO WHILE inp<0 OR inp>2

DISPLAY “Choose an Option : 1.Login, 2.Register, 0.Exit”

READ User\_Input as input

END WHILE

set inp to 69

IF input == 1

set inp to 69

DO WHILE inp<0 OR inp>3

DISPLAY “Choose an Account: 1. Citizen, 2. non-Citizen, 3. Admin, 0. Exit”

CASE based on inp

CASE ‘1’

citizenLogin()

CASE ‘2’

nonCitizenLogin()

CASE ‘3’

adminLogin()

CASE ‘0’

DISPLAY “Exiting Program”

END PROGRAM ENDCASE

ENDWHILE

ELSE IF input == 2

set inp to 69

DOWHILE inp<0 OR inp>3

DISPLAY “Choose Register Account Type: 1. Citizen,2. non-Citizen, 3. Admin,0. Exit”

CASE based on inp

CASE ‘1’

citizenRegister()

CASE ‘2’

nonCitizenRegister()

CASE ‘3’

adminRegister()

CASE ‘0’

Display “Exiting Program”

END PROGRAM

END CASE

ELSE IF input == 0

DISPLAY “Exiting Program”

END PROGRAM

ENDIF

END

**2. citizenLogin()**

BEGIN

Citizen[] registeredCitizen = APPEND CITIZEN DATA FROM TXT FILE

DISPLAY “Citizen Account Login”

WHILE TRUE

DISPLAY “Please enter username: “

READ as inputName

DISPLAY “Please enter password: “

READ as inputPw

FOR Admin i in registeredAdmin

IF inputName == registeredCitizen.username AND inputPw == registeredCitizen.password

citizen()

BREAK

END IF

DISPLAY “Incorrect Username or Password!”

ENDWHILE

END

**3. nonCitizenLogin()**

BEGIN

NonCitizen[] registeredNonCitizen = APPEND CITIZEN DATA FROM TXT FILE

DISPLAY “Non-Citizen Account Login”

WHILE TRUE

DISPLAY “Please enter username: “

READ as inputName

DISPLAY “Please enter password: “

READ as inputPw

FOR NonCitizen i in registeredCitizen

IF inputName == registeredNonCitizen.username AND inputPw == registeredNonCitizen.password

nonCitizen()

BREAK

END IF

END FOR

DISPLAY “Incorrect Username or Password”

END WHILE

END

**4. adminLogin()**

BEGIN

Admin[] registeredAdmin = APPEND ADMIN DATA FROM TXT FILE

DISPLAY “Citizen Account Login”

WHILE TRUE

DISPLAY “Please enter username: “

READ as inputName

DISPLAY “Please enter password: “

READ as inputPw

FOR Admin i in registeredAdmin

IF inputName == registeredAdmin.username AND inputPw == registeredAdmin.password

admin()

BREAK

END IF

DISPLAY “Incorrect Username or Password!”

ENDWHILE

END

**5. citizenRegister()**

BEGIN

DECLARE String variable username, password, password2

DECLARE char variable gender

DECLARE int variable age, citID

DISPLAY “Citizen Account Register”

DO WHILE TRUE

DISPLAY “Enter username : “

READ as username

DISPLAY “Enter password : “

READ as password

DISPLAY “Gender (M/F/O(Male, Female, Others)) : “

READ as gender

DISPLAY “Enter age: “

READ as age

DISPLAY “Citizen ID : “

READ as citID

DISPLAY “Please Re-enter your password: “

IF password == password2

BREAK

ENDIF

DISPLAY “Please refill information, password was not matched”

END DO

Citizen citizen = new Citizen (username, password, gender, age, citID)

APPEND Citizen to Txt File

citizenLogin()

END

**6. nonCitizenRegister()**

BEGIN

DECLARE String variable username, password, password2

DECLARE char variable gender

DECLARE int variable age, passportNum

DISPLAY “non-Citizen Register”

DO WHILE TRUE

DISPLAY “Enter username: “

READ as username

DISPLAY “Enter password: “

READ as password

DISPLAY “Gender (M/F/O(Male,Female,Others)): “

DISPLAY “Enter age: “

READ as age

DISPLAY “Enter Passport Number: “

READ as passportNum

DISPLAY “Please Re-enter your password: “

IF password == password2

BREAK

ENDIF

DISPLAY “Please refill information, password didn’t match”

END DO

nonCitizen noncitizen = new nonCitizen(username, password, age, passportNum)

APPEND nonCitizen to Txt File

nonCitizenLogin()

END

**7. adminRegister()**

BEGIN

DECLARE String variable username, password, password2

DECLARE char variable gender

DECLARE int variable age, govPw

DISPLAY “Citizen Account Register”

DO WHILE TRUE

DISPLAY “Enter username : “

READ as username

DISPLAY “Enter password : “

READ as password

DISPLAY “Enter gender (M/F/O(Male/Female/Others)): “

READ as gender

DISPLAY “Enter age: “

READ as age

DISPLAY “Government Password : “

READ as govPw

DISPLAY “Please Re-enter your password: “

IF password == password2

BREAK

ENDIF

DISPLAY “Please refill information, password was not matched”

END DO

Admin admin = new Admin (username, password, gender, age)

APPEND admin to Txt File

adminLogin()

END

**8. citizen()**

BEGIN

DECLARE int variable inp set to 69

DISPLAY “non-Citizen Menu”

DO WHILE inp < 0 OR inp > 4

DISPLAY “Choose an Option: 1.Register for Vaccination 2. Modify Profile 3.View Vaccination Status 4.View Appointment 0. Exit”

READ User\_input as inp

END DO

CASE based on inp

CASE ‘1’

citizenVacReg()

CASE ‘2’

citizenModify()

CASE ‘3’

citizenView()

CASE ‘4’

citizenAppoint()

CASE ‘0’

DISPLAY “Exiting Program”

EXIT PROGRAM

ENDCASE

END

**9. nonCitizen()**

BEGIN

DECLARE int variable inp set to 69

DISPLAY “non-Citizen Menu”

DO WHILE inp < 0 OR inp > 4

DISPLAY “Choose an Option: 1.Register for Vaccination 2. Modify Profile 3.View Vaccination Status 4.View Appointment 0. Exit”

END DO

CASE based on inp

CASE ‘1’

nonCitizenVacReg()

CASE ‘2’

nonCitizenModify()

CASE ‘3’

nonCitizenView()

CASE ‘4’

nonCitizenAppoint()

CASE ‘0’

DISPLAY “Exiting Program”

EXIT PROGRAM

ENDCASE

END

**10. admin()**

BEGIN

DECLARE int variable inp set to 69

DISPLAY “Admin Menu”

DO WHILE inp<0 OR inp>4

DISPLAY “Choose an option: 1.User Account Management

2. Manage Vaccination Appointment for Regisitered People 3.Vaccine Management 4.Generate Report 0.Exit”

END DO

CASE based on inp

CASE ‘1’

adminUser()

CASE ‘2’

adminAppoint()

CASE ‘3’

adminVaccine()

CASE ‘4’

adminReport()

CASE ‘0’

DISPLAY “Exiting Program”

EXIT PROGRAM

ENDCASE

END

**11. citizenVacReg()**

BEGIN

DISPLAY “Citizen Vaccination Registration”

\*APPLY VACCINE\*

VaccineApplication[] = [(object1, vacName, vacDate, vacPlace),

END

**12. citizenModify()**

BEGIN

DECLARE int variable inp set to 69

DISPLAY “Your Profile”

DISPLAY “Citizen ID: “,

DISPLAY “1. Username: “,

DISPLAY “2. Password: “,

DISPLAY “3. Gender: “,

DISPLAY “4. Age: “,

DISPLAY “0. Back”

DOWHILE inp<0 OR inp>4

DISPLAY “Choose record to update: “

READ User\_input as inp

ENDDO

CASE based on inp

CASE ‘1’

DISPLAY “Enter new Username: “

CASE ‘2’

DISPLAY “Enter new Password: “

CASE ‘3’

DISPLAY “Enter new Gender (M/F/O(Male, Female, Others)): “

CASE ‘4’

DISPLAY “Enter new Age: “

CASE ‘0’

citizen()

END

**13. citizenView()**

BEGIN

END

**14. citizenAppoint()**

BEGIN

END

**15. nonCitizenVacReg()**

BEGIN

END

**16. nonCitizenModify()**

BEGIN

END

**17. nonCitizenView()**

BEGIN

END

**18. nonCitizenAppoint()**

BEGIN

END

**19. adminUser()**

BEGIN

DECLARE inp, set to 69

DISPLAY “Admin User Management”

DO WHILE inp<0 OR inp>4

DISPLAY “Choose an option : 1. Register New Users 2. Modify Existing Users 3. View User 4. Search User 0. Exit Program”

END DO

CASE based on inp

CASE ‘1’

adminUserRegister()

CASE ‘2’

adminUserModify()

CASE ‘3’

adminUserView()

CASE ‘4’

adminUserSearch()

CASE ‘0’

DISPLAY “Exiting Program”

EXIT PROGRAM

ENDCASE

END

**20. adminAppoint()**

BEGIN

END

**21. adminVaccine()**

BEGIN

END

**22. adminReport()**

BEGIN

END

**23. adminUserRegister()**

BEGIN

DECLARE inp set to 69

DOWHILE inp<0 or inp>2

DISPLAY “Select which account to register to : 1. Citizen 2. non-Citizen 0. To go back”

END DO

CASE based on inp

CASE ‘1’:

citizen()

CASE ‘2’:

nonCitizen()

CASE ‘0’:

adminUser()

END

**24. adminUserModify()**

BEGIN

DECLARE inp, found, newUsername, newPassword, newGender, newAge

DISPLAY “Search by:” SearchBarButton ListButton

IF source == SearchBarButton //Group Layout

DISPLAY “Choose account type to modify:” CitizenButton nonCitizenButton Back

IF source == CitizenButton

DISPLAY “Enter username: “

DataIO.citizenChecking(inp);

IF (found == null)

DISPLAY “User is not registered!”

ELSE

DISPLAY username

DISPLAY password

DISPLAY gender

DISPLAY age

DISPLAY citID

ENDIF

DISPLAY “Choose data to update: UsernameButton PasswordButton GenderButton AgeButton”

IF source == UsernameButton

DISPLAY “Enter new username:”

Citizen.setName

ELSE IF source == PasswordButton

DISPLAY “Enter new password:”

Citizen.setPassword

ELSE IF source == GenderButton

DISPLAY “Enter new Gender:”

Citizen.setGender

ELSE IF source == AgeButton

ELSE IF source == nonCitizenButton

DISPLAY “Enter username: “

DataIO.nonCitizenChecking(inp);

ELSE IF source == Back

adminMenu()

ENDIF

ELSE IF source == ListButton //Box Layout \*Not Done Yet\*

BoxLayout();

ENDIF

END

**25. adminUserView()**

BEGIN

END

**26. adminUserSearch()**

BEGIN

DECLARE inp, set to 69

DOWHILE inp<0 or inp>2

DISPLAY “User Search, Choose an account to search: 1. Citizen 2. non-Citizen 0. To go Back”

END DO

IF inp == 1

RegisteredCitizen[] = READ DATA FROM citizen.txt

DOWHILE true

DISPLAY “Citizen User Search Enter 0 to go back, Enter username : “

READ UserInput as String username

IF Integer.parseInt(username) == 0

Break

ENDIF

IF contains(RegisteredCitizen, username)

FOR (String registered in RegisteredCitizen)

IF username equals RegisteredCitizen

DISPLAY “User Found”

DISPLAY “registered.getName()”

DISPLAY “registered.getAge()”

DISPLAY “registered.getGender()”

DISPLAY “registered.getCitID()”

DISPLAY “Repeating Program”

BREAK

END IF

END FOR

ELSE

DISPLAY “Username doesn’t match”

END IF

END DO

DISPLAY “Redirecting to User Management Menu”

adminUser()

ELSE IF inp == 2

NonRegisteredCitizen[] = READ DATA FROM noncitizen.txt

DOWHILE true

DISPLAY “NonCitizen User Search Enter 0 to go back, Enter username : “

READ UserInput as String username

IF Integer.parseInt(username) == 0

Break

ENDIF

IF contains(NonRegisteredCitizen, username)

FOR (String registered in NonRegisteredCitizen)

IF username equals NonRegisteredCitizen

DISPLAY “User Found”

DISPLAY “registered.getName()”

DISPLAY “registered.getAge()e”

DISPLAY “registered.getGender()”

DISPLAY “registered.getCitID()”

DISPLAY “Repeating Program”

END IF

END FOR

ELSE

DISPLAY “Username doesn’t match”

END IF

END DO

DISPLAY “Redirecting to User Management Menu”

adminUser()

ELSE

adminUser()

END IF

END

**27. DataIO Admin**

DECLARE ArrayList<Admin> allAdmins

READ FUNCTION:

Try:

READ from admins.txt

WHILE !(EOF) //End Of File

READ String username

READ String password

READ Gender gender

READ Int age

CREATE new Admin objcect (username, password, gender, age) set to a

ADD a to Admin ArrayList

END WHILE

Catch Exception Error:

DISPLAY “Error while reading file”

END TRY

WRITE FUNCTION:

Try:

DECLARE PrintWriter p to write into “Admins.txt”

FOR (Admins a in allAdmins):

PRINT Admin.getName

PRINT Admin.getPassword

PRINT Admin.getGender

PRINT Admin.getAge

PRINT LINE

END FOR

CLOSE FILE

Catch Exception Error:

DISPLAY “Error while reading file”

END TRY

Admin checking FUNCTION (parameter String username)

Declare Admin variable found set to null

FOR (Admin a in allAdmins):

IF (username equals to a.getName():

Return a

END IF

END FOR

Return null

**28. DataIO People**