**PROJECT TITLE**

**Covid Management System**

**Group Members**

Ananthakrishnan (AM.EN.U4AIE21113)

Gautham Krishnan (AM.EN.U4AIE21129)

Mridul Sharma (AM.EN.U4AIE21173)

K Bharadwaj (AM.EN.U4AIE21135)

Kiran P (AM.EN.U4AIE21175)

Surya Vamsi Vema (AM.EN.U4AIE21162)

**ABSTRACT**

This Project is based on a management System for COVID-19. In this software we can store and organize data of different users and use it for different purposes. The program consists of three main parts that are:

* Quarantine Management- This feature of the software is used to manage the quarantining of multiple individuals. The software stores the data such as quarantine start date and end date, vaccination status, etc. The individuals can be sorted based on any of the fields which can be used to manage the quarantine.
* User Management- this feature is used to store the basic details of an individual and manage it.
* Donation- It allows the users to donate to the Covid-19 relief fund.

The software stores all the data in multiple files which are loaded on the starting of the program. The program has two actors, Admin and User. Only Admin has the privileges to add new users, modify the details of existing users and withdraw the donated funds. The Admin/User should login every time the program starts to access the features of the software.

***Keywords:*** UML, Use case, Disease management, Government, Vaccination, Covid-19, User Profile, Quarantine, Health/Fitness, Pandemic

**Functional Requirements of the System**

List the functional requirements of your system here.

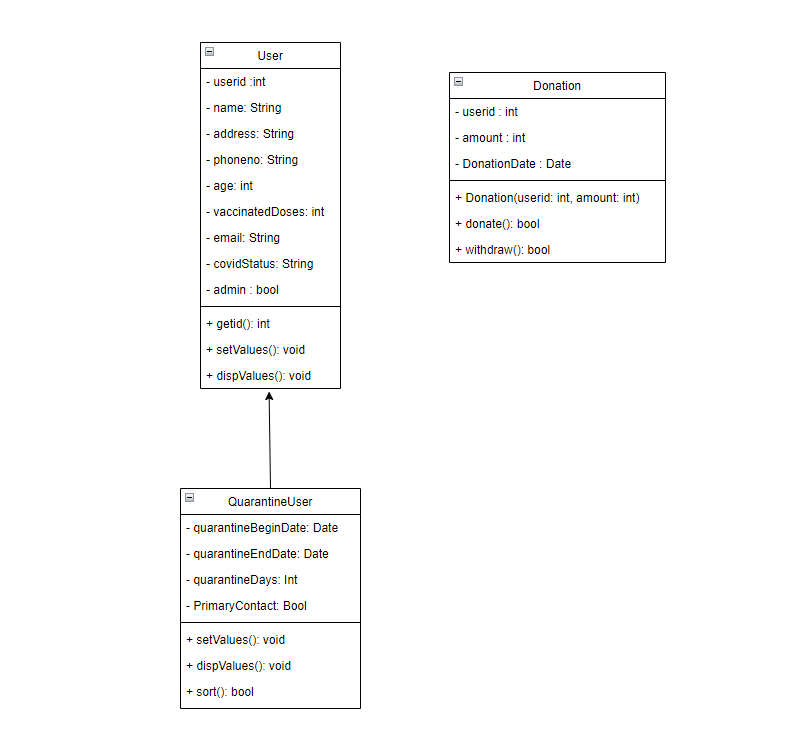
1. User/Admin need to be registered with the system before using the application.
2. User/Admin should login to the system to access the main menu.
3. User/Admin can choose one of the available options.
4. Java runtime environment should be installed.
5. Hardware requirements:

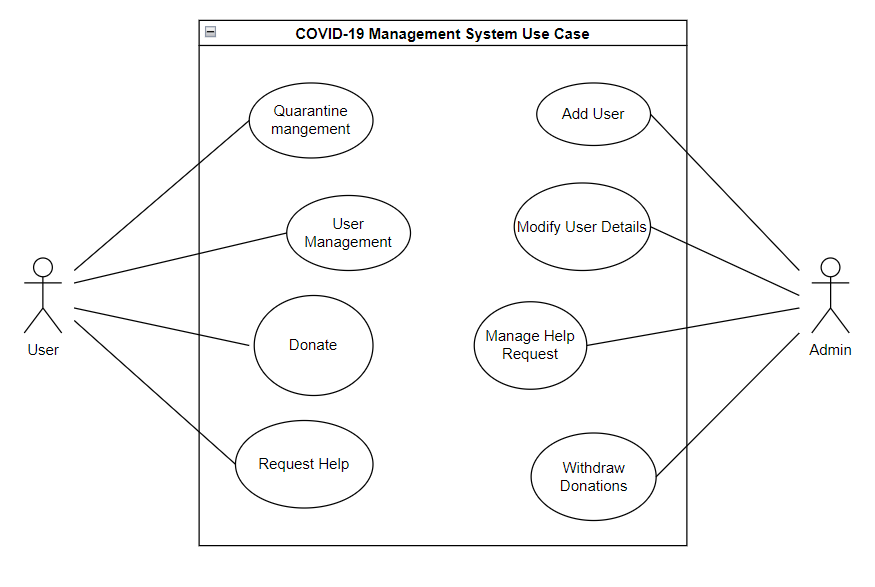
* RAM: At least 1 GB.
* Disk Space: 124 MB for JRE, 2 MB for Java Update.
* Processor: Minimum Pentium 2 266 MHz processor.

**Use Case**

|  |  |
| --- | --- |
| **Actors** | **Use case** |
| User | Views User Details, View quarantined Users, Makes Donation, Modify own details, Request Help |
| Admin | Adds User, Removes User, Modify Users Details, Withdraws Donations, Manage Help Request |

**Class Diagram**



**Use case diagram**

**Use Case Description**

|  |  |
| --- | --- |
| Use case ID | Menu |
| Brief description | The User is able to access the menu based on their details like user-ID and password |
| Primary actors | System User and Admin. |
| Precondition | User must be registered and successfully login. |
| Post condition | The admin / user will be able to access their respective menus |
| Main success Scenario | If the chosen option is valid redirect the user/admin to their particular menus. |
| Actor Action | System Response |
| 1. The user is able to choose one of the options among donate, user management, quarantine management etc. The Admin is also able to choose certain functionalities of the program such as Manage help request. | 2. Checks if the User/Admin chose a valid option. If valid option, go to corresponding screen. |
| 3. The User/Admins gets displayed the requested sub menus for the choice they selected. |  |
| Alternate flow |  |
|  | 2a. If not a valid option, system returns error message.  “Please Enter A Valid Input” |

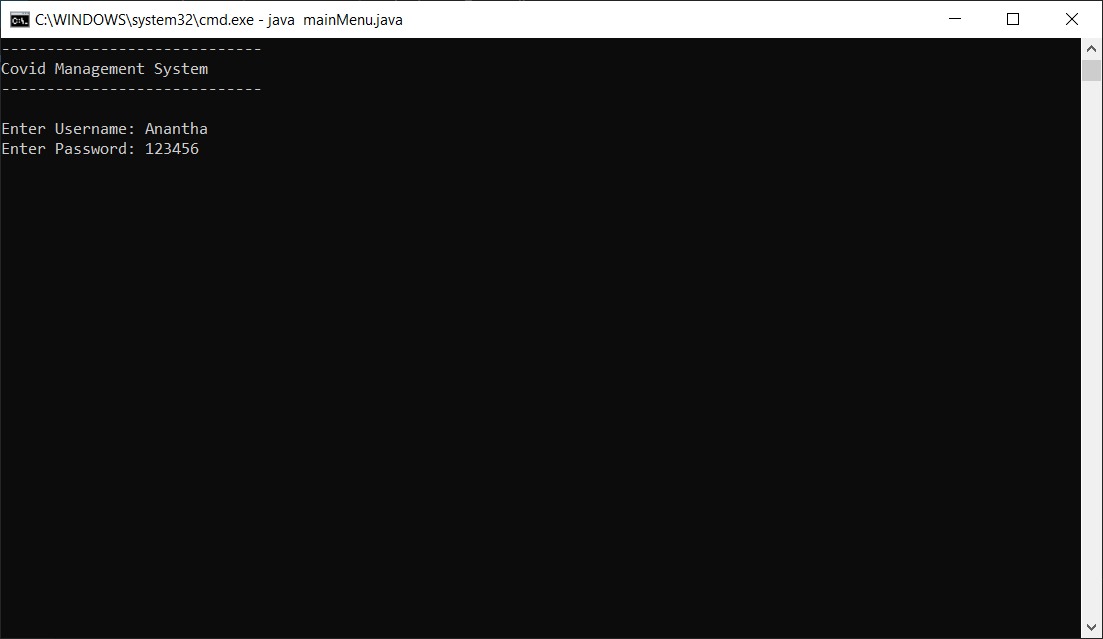
|  |  |
| --- | --- |
| Use case ID | Quarantine Management |
| Brief description | The User can see other User’s and own details and can also sort the registered Quarantine users based on fields such as age, Quarantine begin/end date etc. Admin can modify, add, delete and sort the Users. |
| Primary actors | System User and Admin. |
| Precondition | User must be registered and successfully login. |
| Post condition | If the Admin adds/deletes users or modifies details, update the database. |
| Main success Scenario | If the chosen option is valid perform the necessary operation/ display the requested data. |
| Actor Action | System Response |
| 1. The user chooses one of the options to view quarantined users details and modify own details sort users based on different fields. Admins also have the option to add/delete Users or modify existing users details. | 2. Checks if the User/Admin chose a valid option. If valid option go to corresponding screen. |
| 3. The User/Admins gets displayed the requested details or gets the option to modify own details as per the choice. Admins gets the option to add/modify Users if that option was chosen | 4. Checks if a user with the given User-ID exists and displays the corresponding information. |
| Alternate flow |  |
|  | 2a. If not a valid option, system returns error message. |
|  | 4a. If a User with the ID doesn’t exist, display error message.  “User not Found” |

|  |  |
| --- | --- |
| Use case ID | User Management |
| Brief Description | The User can see their details, others details. Admin can modify the current User details and add/delete Users. |
| Primary actors | System User and Admin |
| Precondition | Users must be registered and successfully log in. |
| Post-condition | If admin adds a new user or modifies user details, update the database |
| Actor Action | Response |
| 1. After completion of login User can see his/her details, other users list. Admin has the option to modify existing users details and add/delete users | 1. System checks if Admin/User chooses a valid option. If option is valid it shows the corresponding details. |
| 1. The User/Admin gets displayed the requested details or Admin gets the option to modify the user details as per choice. |  |
| Alternative Flow |  |
|  | 2a. If not a valid option system displays an error message. |
|  |  |

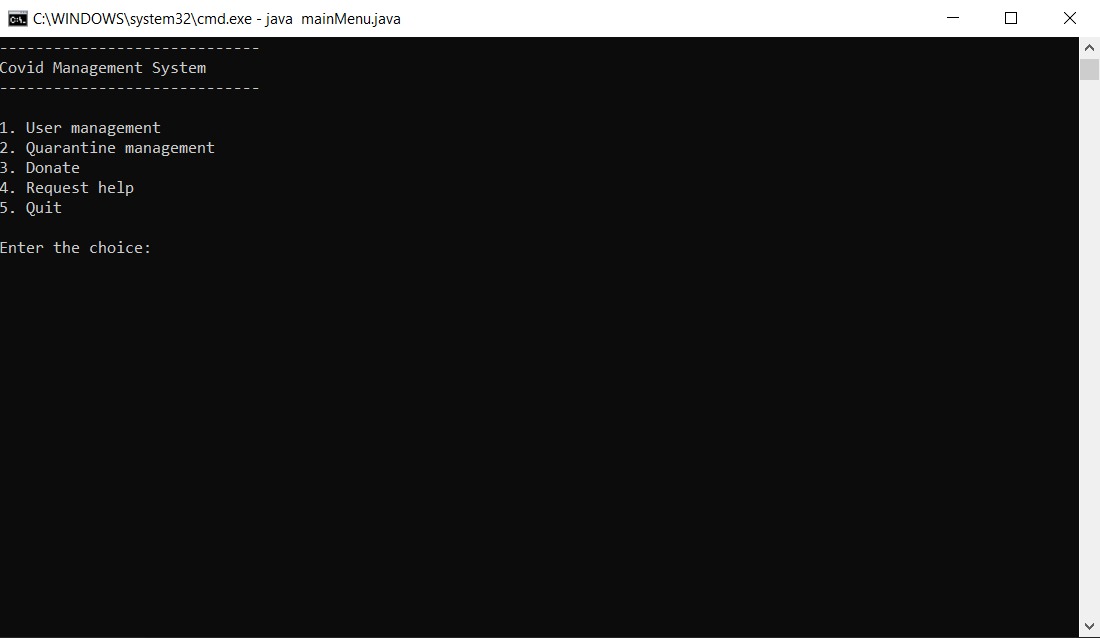
|  |  |
| --- | --- |
| Use case ID | Donation |
| Brief description | The User can make donations and view others donations and Admin can withdraw the donated Amount. Admin can also get the donations summary as total amount donated, withdrawn etc. |
| Primary actors | System User and Admin |
| Precondition | User must register with the system and should login successfully. |
| Post condition | The database is updated after every successful donation and withdrawal. |
| Main success Scenario | If donation is successful, show a success message thanking the user. |
| Actor Action | System Response |
| 1. The user selects donation and enter an amount or requests other details. | 2. If the User is an admin show the option to make donations on behalf of other users else ask the user to enter the amount. |
| 3. User can enter the amount they wish to donate. | 4. System shows a success and thanking message after successful transaction. |
| Alternate flow |  |
| 1a. If the entered option is not available, display error message. |  |

**Screen Shots**

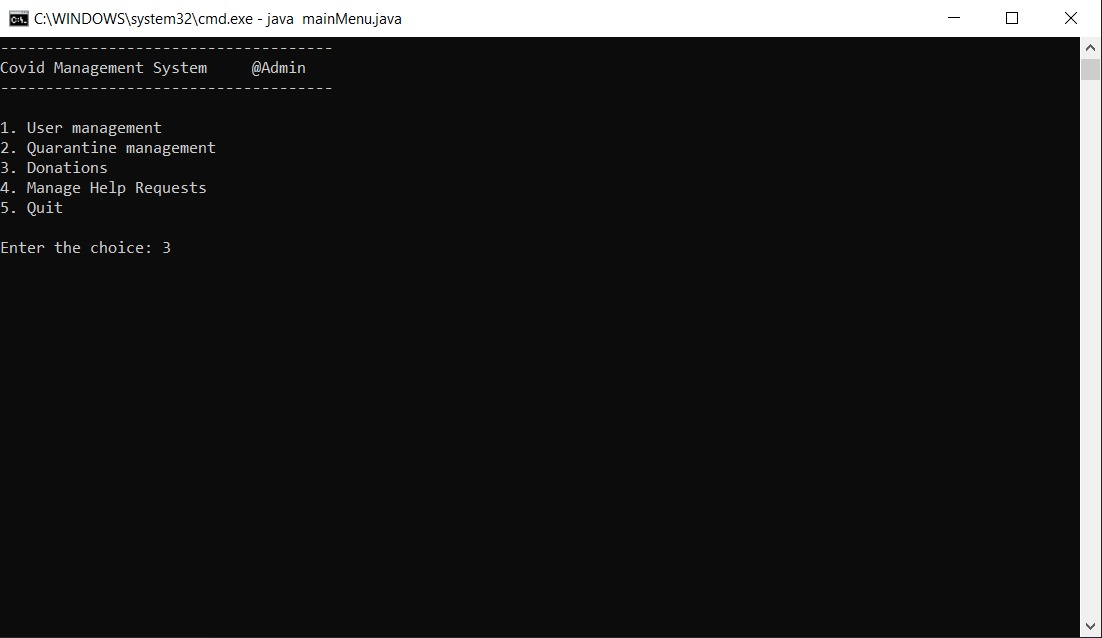
* **Login Screen**



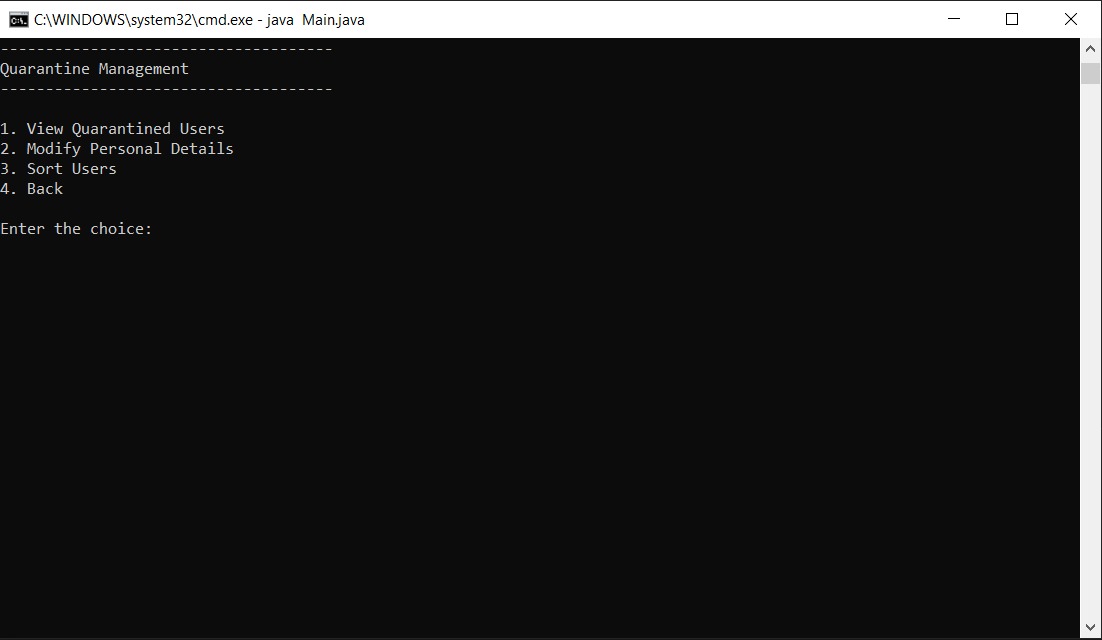
* **User Main Menu**



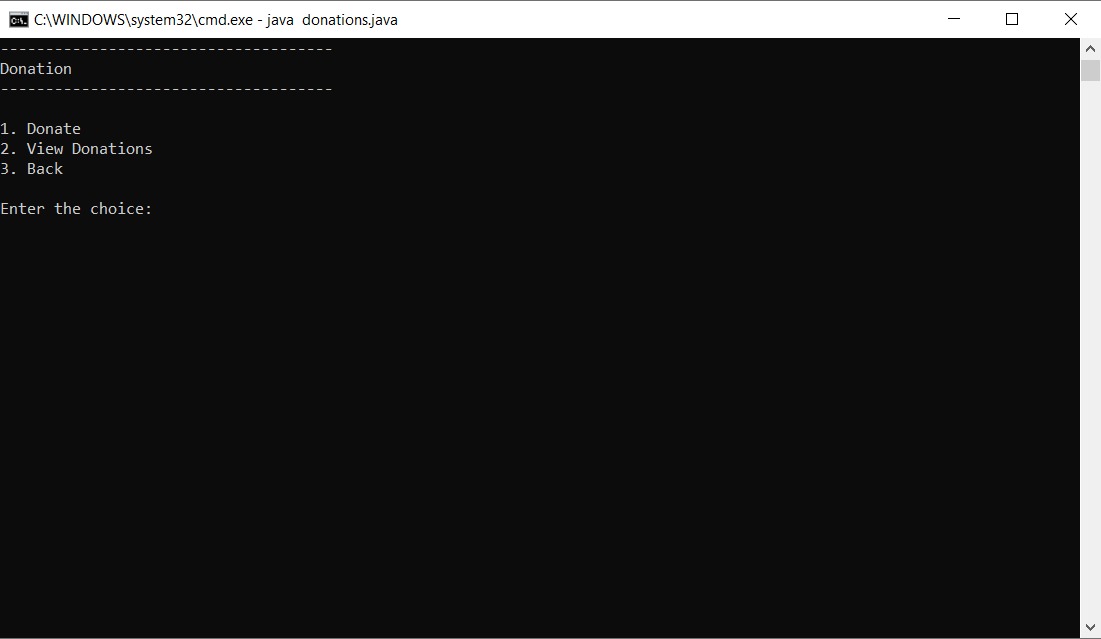
* **Admin Main Menu**



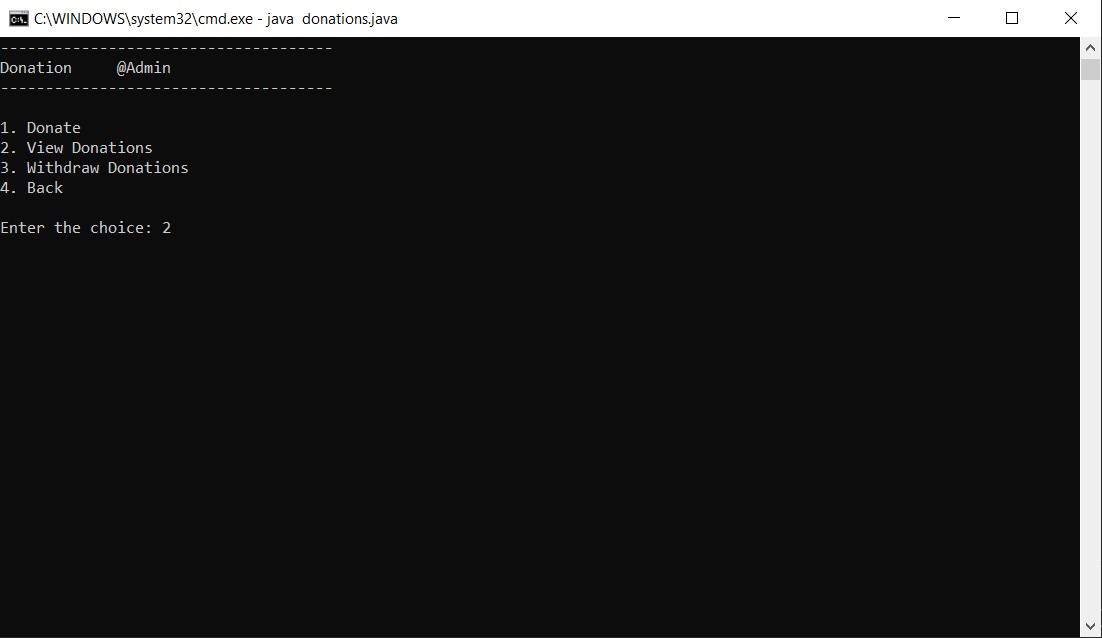
* **Quarantine Management Menu**



* **User Donation Menu**



* **Admin Donation Menu**



* **View User Details**

