SOFTWARE DEVELOPMENT PRACTICES
SRS DOCUMENT FOR COVID-19 TRACKER APPLICATION

DATE:1/12/2022

CREATED BY:
SRI SIVA SWATHIKA S T
NAAZIRA SHAREEFA R
SRIJHHA R M

OBJECTIVE:

Covid 19 tracker application is an application that helps in tracking the vaccinated, non vaccinated and the affected groups of Covid 19 in India.

USERS OF THE SYSTEM:

- Government
- Public

FUNCTIONAL REQUIREMENT:

- Build an application that customers can access by getting the vaccination certificates and further notifications just by entering the aadhar number or the phone number.
- Application should have signup, login, mobile number or aadhar number, dashboard page and customer status page.
- This application maintains the database of affected, recovered and deceased numbers across India.
- Displays the percentage of population vaccinated, not vaccinated and the other details of the population.
- People above the age of 50 should reappear for the booster dose would be notified.
- While the above ones are the basic functional feature expected, the below ones can be nice to have add-on features:
- Multi factor authentication for the sign-in process.
- Location can be enabled during the vaccination to ensure the location of vaccination for future problem, if any.

OUTPUT/POST CONDITION:

- Standalone application/ public application.
- Records the overall status/ position of each state across India.

NON- FUNCTIONAL REQUIREMENTS:

SECURITY

- Secured connection for transmission of data.
- App platform: Phone number/ aadhar number- based credentials
- OTP everytime during login for safer data retrieval.

PERFORMANCE

- Peak Load Performance(during important occasions and festival days, national holidays etc.)
    1. Verification process< 5 secs.
    2. Re-Captcha process< 2secs.
    3. Vaccine certificates<3secs.

MAP DESIGN:

- COVID affected people more than 50% of the state should be orange in colour.
- COVID affected people less than 50% in each state should be yellow in colour.
- COVID affected people are more than 90% in a state should be red in colour.

AVAILABILITY:

- 100% availability

STANDARD FEATURES:

- Maintainability
- Usability
- Availability
- Failover
- Scalability

LOGGING:

- The system should support logging(app/web)

MONITORING:

- Government should be able to monitor the update of every individual for statistical purpose.

BROWSER COMPATIBLE:

- All latest browsers.

TECHNOLOGY STACK:

- HTML
- CSS
- Java Script.

APPLICATION ASUMPTIONS:

- The login page should be the first page rendered when the application loads.
- Unless logged in to the system, the user cannot navigate any other pages.
- Logging out must again redirect to the login page.
- To navigate to the admin side, you can store a user type as admin in the database with a username and password as admin.

VALIDATIONS

- Basic mobile number validation should be performed.
- Password validation should be performed.

FRONTEND:

Customer:

1. **Auth:** Design an auth component whereas auth for tracker app. Once the component is created in tracker app, name the jsx file as same as component name where the user can authenticate login and sign in credentials.
2. **Signup:** Design a signup page component named where user can signup by providing their basic details like
    - **Phone number(mandatory)**
    - **Email (all email apps are allowed)**
    - **Password(optional)**
    - **Sign in link**
    - **Signup box**
3. **Login:** Design a login page component named where the existing customer can log in using the registered mobile number and password.

4. **Dashboard/ Home:** Design a homepage component named that has the navigation bar and list of issued documents with appropriate filter options.
    - **Home button**
    - **Documents issued button**
    - **Nearby vaccination camp/centre button**
    - **Logout button**
5. **Map desing:** Design an Indian Map named COVID MAP where user can see the percentage of covid affected people and recovered by representing the colours.

Admin:

6. **Admin Dashboard:** Design a dashboard page named where the list of percentage of vaccinated people can be displayed on admin side.

BACKEND :

1. **UserModel:** This class stores the user type (admin or the customer) and all user information.

a. **Attributes:**

i. Aadhar: String

ii. password: String

iii. mobileNumber: String

iv. active: Boolean

v. role: String

2. **LoginModel:** This class contains the email and password of the user.

a. **Attributes:**

i. Aadhar: String

ii. password: String

3. Covid 19 Model: This class stores the details of the patient.

a.Attributes:

i. StateId: String

ii. imageUrl: String

iii. patient name: String

iv. Status: String

Controller Layer:

6. SignupController: This class control the user signup

a. Methods:

i. saveUser(UserModel user): This method helps to store users in the database and return true or false based on the database transaction.

7. LoginController: This class controls the user login.

a.Methods:

i. checkUser(LoginModel data): This method helps the user to sign up for the application and must return true or false.

8. Patient Controller: This class controls the add/edit/update/view number of person affected by Covid 19.

a. Methods:

i. List<state> getstate(): This method helps the admin to fetch all datas

from the database.

ii. List<District> getDistric (): This method helps to retrieve all the datas

from the database.

iii. Patient Details EditData(String id): This method helps to retrieve a

affected details from the database based on the Aadhar id.

iv. Patient Details Edit Save(Patient Details data): This method helps to edit a Patient Details and save it to the database.

v. Patient Details Save(Patient Details data): This method helps to add a new Patient Details to the database.

vi. Patient Details Delete (String id): This method helps to delete a Patient