INFORMATION

ABOUT COVID 19

OBJECTIVE

Building an UI for covid 19

USERS OF THE SYSTEM

* Government
* Public

FUNCTIONAL REQUIREMENTS

The application should have sign up, login, profile, dashboard page and individual records.

1. State Name
2. Cases confirmed
3. Deaths
4. Fatality rate
5. Latitude
6. Longitude
7. Last Update

DATA

List of recorded data

* Identification data : CIP, ID or Passport
* Contact details: Telephone numbers or Address
* Health data: The users answers about the symptoms
* Aadhar integration for intimating individual reports to the public.
* Apps that provide medical, treatment, vaccine, testing, or other related information for COVID-19.

POST CONDITION

* Records Persisted in Success & Failure Collections
* Standalone application/Deployed in an app Container

NONFUNCTIONAL REQUIREMENTS

* Apps created specifically for the COVID-19 response may not access personal and sensitive data that is not required to directly support the public health emergency, and epidemiological research
* Apps must handle all personal or sensitive user data securely, including transmitting it using modern cryptography (for example, over HTTPS).
* The system should be available all the times.
* A commercial database in used for maintaining the database & the application server takes care of site.

STANDARD FEATURES

* Number of tested
* Confirmed cases
* Deaths in the country
* A heatmap of the largest concentrations of confirmed covid-19 cases
* Locations of public testing centres in each state
* News updates from major health organisation

KEY POINTS TO REMEMBER

1. The ID and attributes mentioned in the SRS should not be modified at any cost.
2. Strictly adhere to ID mapping and attribute mapping. Failing to do may fail test cases.
3. Adhere strictly to the end points given below.

APP ASSUMPTIONS

* The login page should be the first page rendered when the app works.
* Unless logged into the system, the user cannot navigate to any other pages.
* Logging out must be redirected to the login page.
* Use admin as the username and password to navigate to the admin dashboard.

VALIDATIONS

Basic mobile validations should be performed.

Aadhar password validations should be performed.

PROJECT TASKS

* Complete the covid 19 contact tracing and status apps section in the App content page.
* Privacy requirements
* API requirements
* App review and visibility

CUSTOMER

1. SIGNUP

Design a signup page component where the new customer has options to sign up by providing their basic details

1. Ids:

* User name
* Aadhar number
* Mobile number
* Password
* Confirm password
* Submit button
* Sign in link
* Sign up box

1. Output Screenshot

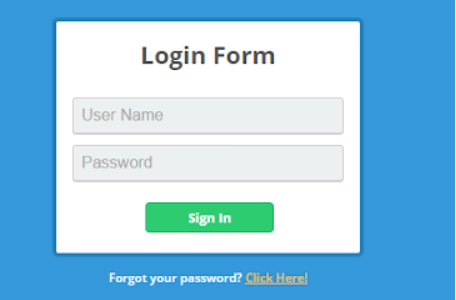
LOGIN:

Design a login page component name where the existing customer can login using the registered email id and password.

1. Ids

* User name
* Password
* Submit button
* Signup link
* Login box

1. Output Screenshot



HOME PAGE

Design a home page component named that has the navigation bar.

1. Ids

1.Username

2.Home button

3.Personnel data

4.Over all the State data

5.Logout button

b. Screen shot



Backend

Class and Method description

Model Layer

1. User Model

This class stores the user type and all user information

1. Attributes

* Aadhar: string
* Password: string
* Mobile Number: string
* Active: Boolean
* Role: string

1. Login Model

This class contains the email and password of the user.

1. Attributes

* Aadhar: string
* Password: string

1. Covid 19 Model

The class stores the details of the patient.

1. Attributes

* State id: String
* Image Url: String
* Patient name: String

CONTROLLER LAYER

SIGNUP CONTROLLER

This class control the user signup

1. Methods

* Save User:

This method helps to store users in the database and return true or false based on the database transaction.

LOGIN CONTROLLER

This class controls the user login.

1. Methods

* Check User:

This method helps the user to sign up for the application and must return true or false.

PATIENT CONTROLLER

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

1. Methods:

* List<state> getstate():

This method helps the admin to fetch all data from the database.

* List<District> getDistric():

This method helps to retrieve all the data from the database.

* Patient Details Edit Data (String id):

This method helps to retrieve a affected details from the database based on the Aadhar id.

* Patient Details Edit Save (Patient Details data):

This method helps to edit a Patient Details and save it to the database.

* Patient Details Save (Patient Details data):

This method helps to add a new Patient Details to the database.

* Patient Details Save (Patient Details data):

This method helps to add a new Patient Details to the data base.

* Patient Details Delete (String id):

This method helps to delete a Patient Details from the database.

THANK YOU

BY

KALPANA K

SOVYA SRI M

BOOMIKA E