



Backend Developer Assignment

Tech Stack: NodeJS, MongoDB

Task Description:

Develop APIs to access/manage data on MongoDB (should be deployed on [Mongodb Atlas](#)) for the given use case.

Use case : Imagine there is an app created for vaccine registration (similar to that of Arogyasetu Cowin part).

App has the functionality to
(for user)

- Register user (Mandatory fields: Name, PhoneNumber, Age, Pincode, Aadhar No)
- User can login through his PhoneNumber and password (set during registration)
- User should be able to see the available time slots on a given day for vaccine registration (first/second dose based on his vaccination status)
- Users can register a slot for the first/second dose of the vaccine (example: register for **1st dose on 1st June 11 AM**).
- Users should be able to register for the second dose, only after completing their first dose of vaccine. Once the registered time slot is lapsed, the user should be considered as vaccinated for that registered dose (first/second).
- User can update/change his registered slot, till 24 hours prior to his registered slot time

(for admin)

- Login using admin credentials (There won't be any api for registering the admin. His credentials should be manually created in the database)
- Check the total users registered and **[filter them by Age/Pincode/Vaccination status (none/First dose completed/All completed)]** - Optional

- Check the registered slots for the vaccine (first dose /second dose/total) on a given day

Vaccine slot details

- Assume that vaccination drive is happening only from 1st June '21 to 30th June '21
- Timings of the vaccine : 10 AM to 5 PM everyday
- Each vaccine slot will be of duration 30 minutes. (So slots will be like 10:00 AM to 10:30 AM, 10:30 AM to 11:00 AM etc)
- In each vaccine slot there will be 10 vaccine doses available (vaccine dose is same for first/second doses. So both users with first dose or second dose can register).
- So total available vaccine doses => $30 \times 14 \times 10 \Rightarrow 4,200$
- Once 10 vaccine doses in a slot is registered, that time slot shouldn't be available for further registrations (unless the registered user modifies his time slot to a different slot)

Deliverables:

- Create APIs for all the use cases mentioned above (both user and admin).
- Create an atlas mongodb free tier instance and use it as the database.
- Share the APIs **private repository** (add **pvsvamsi** as the collaborator), Readme file (with apis and their functionality listed, models) and the atlas mongodb credentials at the end of the assignment deadline

Evaluation Criteria:

- API schema and database models
- Performance optimizations of the APIs and data validations
- Use cases covered
- Explanation/comments for the designed schema and models