# **Flexmoney Yoga Admission**

#### **Problem Statement**

Assume that you are the CTO of an outsourcing firm which has been chosen to build an admission form for the Yoga Classes which happen every month. Requirements for the admission form are:

- Only people within the age limit of 18-65 can enroll for the monthly classes and they will be paying the fees on a month on month basis. I.e. an individual will have to pay the fees every month and he can pay it any time of the month.
- They can enroll any day but they will have to pay for the entire month. The monthly fee is 500/- Rs INR.
- There are a total of 4 batches a day namely 6-7AM, 7-8AM, 8-9AM and 5-6PM. The participants can choose any batch in a month and can move to any other batch next month. I.e. participants can shift from one batch to another in different months but in the same month they need to be in the same batch.

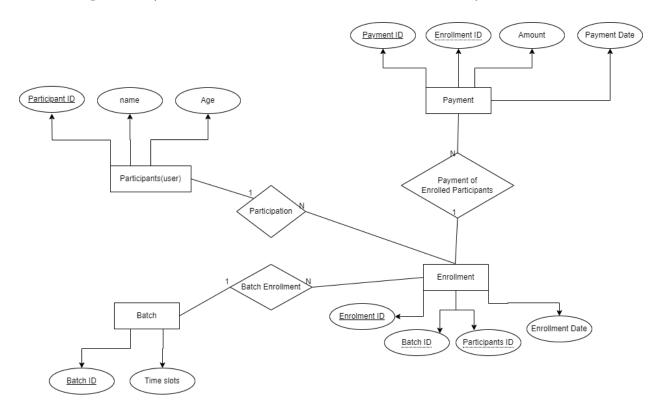
#### **Tech Stake Details**

I first built a backend service to accept requests from the frontend and store them in the database. I have used Node and Express js for it. For the frontend, I am using React js. And MongoDB as our database.

### **Assumptions**

Here for the database, I have used MongoDB although it is a no-SQL and non-Relational database. Since there was no constraint on it. Also, I have attached the ER diagram that I think would work for this problem if we SQL or Traditional Relation Database. I am using MongoDB because it suites well with the Frontend (React) and Backend (Node and Express).

### **ER Diagram (If we use Relational Database)**



## **Implementation Details**

The user would open the form and fill in their details. Frontend will perform basic validations and then it will do a POST request to the backend service with the appropriate header and body.

In the backend, it will receive the payload from React App in the form of JSON. We will again perform basic validation checks and if details in the payload pass these validation checks we will call the CompletePayment() Mock Function and will throw a response message as Payment Successful or else it will throw a message as Payment failed.

## **Repository Details**

Frontend Source Code: MukundLadani/flexmoney-yoga-frontend (github.com)

Backend Source Code: MukundLadani/flexmoney-yoga-backend (github.com)

## **Hosted URL**

Frontend: Flexmoney App (main--stellar-snickerdoodle-b25151.netlify.app)

Backend: Flexmoney-yoga-backend-b9nz.onrender.com