**FlipFit**

Design a backend system for a new enterprise application that Flipkart is launching, FlipFit.  
Flipkart is partnering up with gyms across Bangalore to enter into the fitness space. For the Beta launch the requirements are as follows:

* There are multiple centers in Bangalore. We can expand to multiple cities across and beyond India.
* Each center has **n slots of an hour each**. For eg the Bellandur center has only 6 slots - 3 in the morning of **an hour each** from **6 am to 9 am** and similarly 3 in the evening from **6 pm to 9 pm.** All the centers might/might not be open **7 days a week.**
* Each slot at a center can have **m possible workout variations** - **Weights**, **Cardio, Yoga, Swimming etc.** There could be newer workouts added at center/slot level in the future.
* The number of seats in each workout at each time-slot for a given center is fixed.
* There can be two personas of the users, FK\_VIP\_USER, FK\_NORMAL\_USERS.
* VIP persona users will have access to premium slots while normal users will not be able to see premium slots. This use case will be extended in future with custom offerings for premium users and normal users. The pricing could be different for premium vs normal users,etc…
* A Person can book a maximum of 3 slots in a day for any center. User can have more slots across the centers
* It should be easy to change the underlying data storage mechanism as data scale increases without changing various parts of the application
* User can perform the following operations:
* Register onto the platform
* View the workouts availability/unavailability for a particular day at a center. VIP customers will have a different set of slots shown while normal customers will be shown different sets of slots. Eg: Early morning slots are dedicated only for the VIP customer. In the future, VIP customers might see different pricing for the same centers.(different pricing logic need not be implemented but code should be extensible for the same).
* Book a workout for a user if seats are available in that time slot at that center
* View his/her plan basis day as input
* Cancel his/her workout and the slot should be accommodated by the immediate person in the queue and while filling the slot VIP customers should be given the priority. Once all the VIP customers in the queue are allocated then the normal customers are allocated.

***Test cases:***

Following functionality will be tested via tests during evaluation :

* Addition of center with appropriate details like name,city,location,lat,long
* Addition of new workout Types in a particular center
* Addition/Deletion of slots given a center,workoutType, startTime, number of seats & waiting list queue size
* Details of workouts available at the center
* registering user
* get all available slots across all workoutTypes for a center on a day
* showSlotsForUser given user persona, workout type and time range. Slots can be sorted based on distance
* Book a slot for a user given a slot, workout & day
* viewUserBooking(userId,date)
* cancel slot for a user given a slot & centerName, VIP customers should be allocated first and then normal users
* Notify appropriate users (one who is canceling, one who is moving from waiting queue)

***Guidelines:***

* Time: 120 mins.
* Write modular, clean and demo-able code (Test cases or runtime execution).
* **A driver program/main class/test case is needed to test out the code by the evaluator with multiple test cases**.
* **Use design patterns wherever applicable**
* **Please handle concurrency wherever applicable.**
* **Evaluation criteria**: Demoable & functionally correct code, Code readability, Proper Entity modeling, Modularity & Extensibility, Separation of concerns, Abstractions, Exception Handling, Code comments.
* You are not allowed to use any external databases like MySQL. Use only in-memory data structures.
* No need to create any UX or any HTTP API. It should be a standalone application.