Sung Woo Park

# 1. As a warm up, write out a feature list for this application by extracting feature bullets from the above.

**Feature List**

Membership

1. Two types of memberships—basic and premium.
2. Premium membership gives you 5 free appointments per month.
3. Membership can be either 6-months or 1-year.
4. Gives access to facility center.
5. Membership can only be switched during the renewal period.

Members

1. Change membership from basic to premium and vice-versa.
2. Request personal appointments.
3. Have access to the facilities.

Fitness Center

1. Sells membership.
2. Checks if the person has a membership.
3. Stores the history of customers and membership.
4. Tracks personal trainer appointments to avoid duplicates.
5. Updates the price of memberships.

Facilities

1. Provide weight room, climbing wall, and pool.

# 2.  Now, add four features you think are useful/important for such a fitness center to function well but are not explicitly mentioned in the above description.

**Extended Feature**

Membership

1. Discounted price during promotions and sales.

Members

1. Make an online payment.

Fitness Center

1. Make trainer recommendations.
2. Provide guest passes to the members.
3. Automatic trainer recommendation.
4. Accept online payment.

Trainer

1. See upcoming appointments.

# 3. One method for finding classes and methods to put in a class diagram is to take an English description (like the above) and find *nouns* which could be useful as classes, and *verbs* which could be useful as methods on classes. This is called "textual analysis". It is in the [design lecture notes](http://pl.cs.jhu.edu/oose/lectures/design.shtml#textual) but the professor forgot to cover it this year. In any case it's a very simple idea.

# For this question make two lists, one of nouns and the other of verbs (or verb phrases) from the above which are reasonable candidate class and methods names. Don't be completely brainless in your lists of nouns and verbs, for example merge synonyms, there may be noun/verb phrases as opposed to single nouns/verbs, etc.

**Nouns**

Person

Member

BasicMember

PremiumMember

Membership

BasicMembership

PremiumMembership

FitnessCenter

Facility

WeightRoom

ClimbingWall

Pool

Appointment

OfficeAppointment

PersonalTrainer

PersonalTrainerAppointment

Guest

TransactionHistory

**Verbs**

getId

getName

getAge

getRegistrationDate

getMemberType

getMembershipType

isMember

hasAccess

requestMembershipChange

requestPersonalTraining

canSwitch

payMembershipFee

getMembershipType

getMembershipCount

getBasicMembershipCount

getPremiumMembershipCount

updateMembershipPrice

recordHistory

4. Draw a UML class diagram which serves as a domain model of the above application (including your new features from question 2). Follow the [lecture guidelines](http://pl.cs.jhu.edu/oose/lectures/design.shtml#uml) on which UML syntax features you should use: associations, multiplicities (0..n, 2, \*, etc), whole-part (diamond) annotations, inheritance and also include important fields and methods in the class box. Make sure to verify your UML class diagram syntax to avoid point losses.

5.  Comment briefly on how helpful the textual analysis of question 3 was for your diagram in question 4. Just give your opinion to harvest the full 5 points.

The textual analysis of question 3 was really helpful for me to divide the action of each object. I think this is a good practice to clear your thoughts.