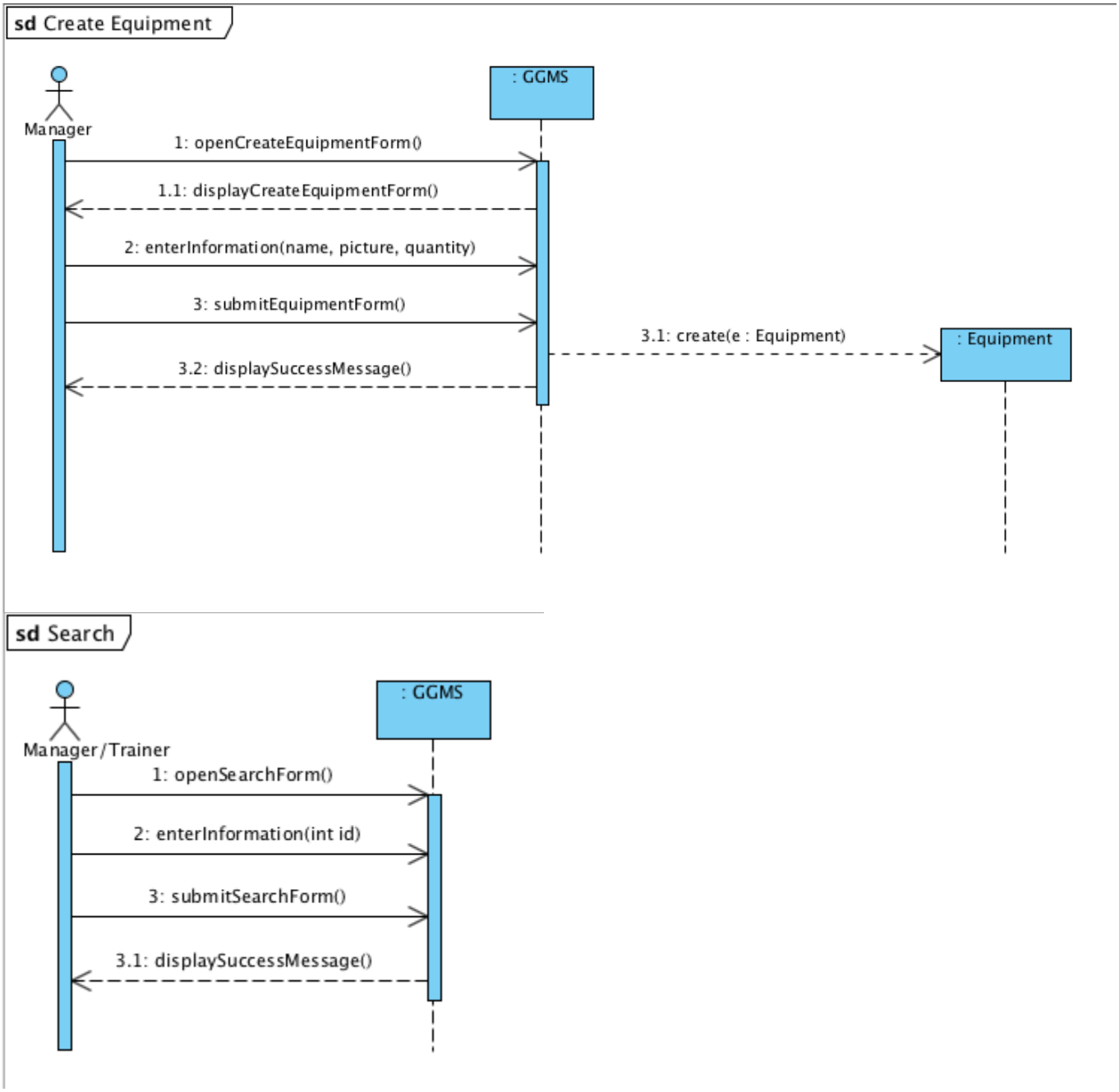
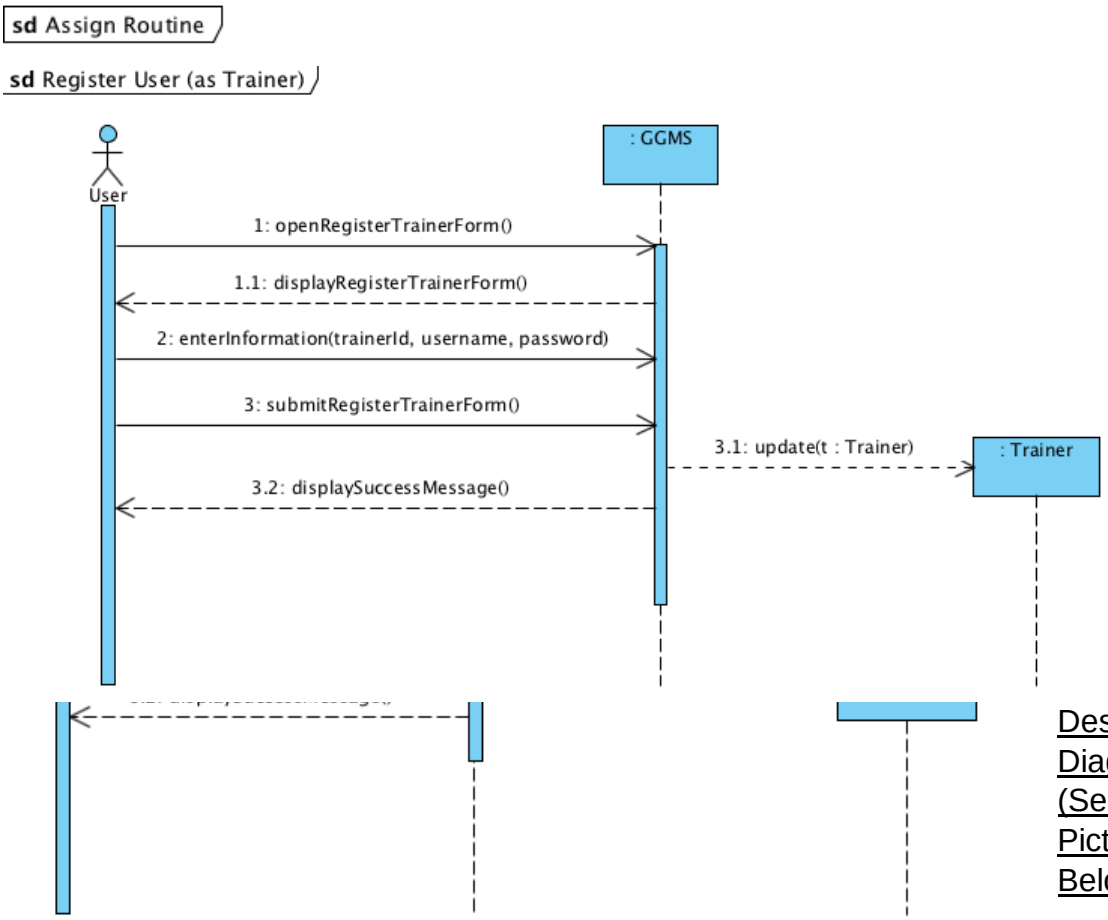
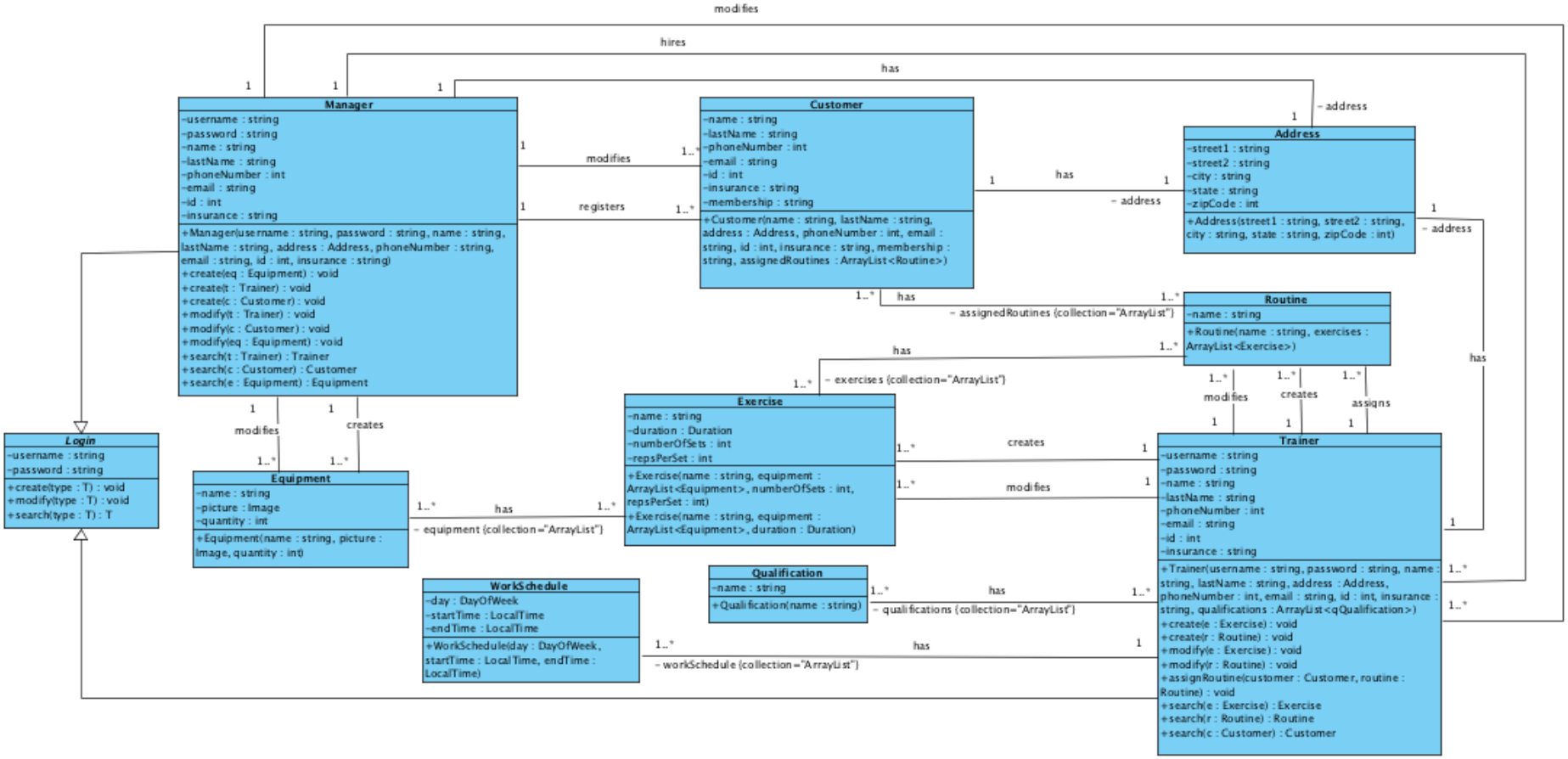
Sequence Diagrams

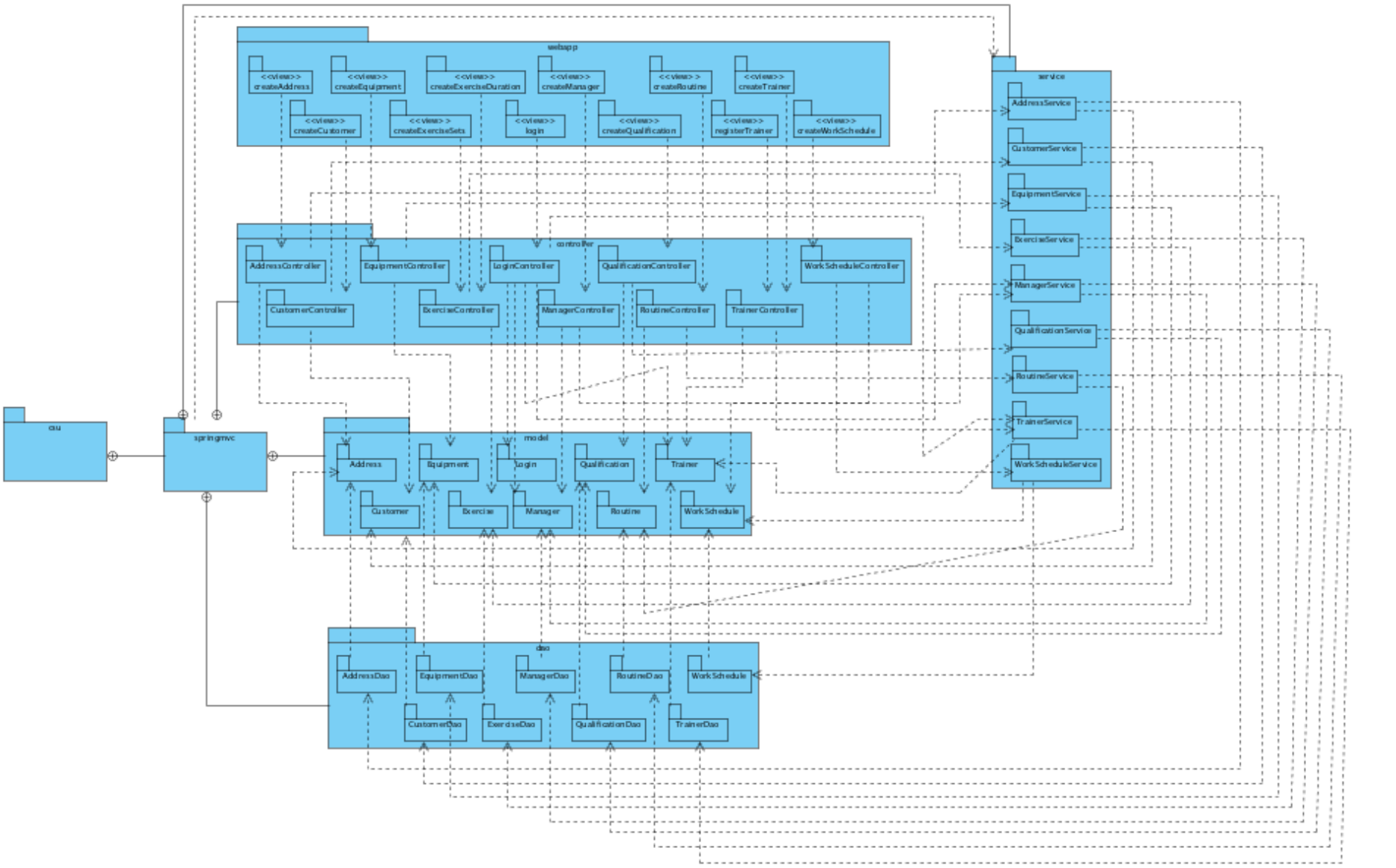




Design Class Diagram (See First Picture Below)

Package Diagram (See Second Picture Below)


modifies
hires
has
Manager
-username : string
-password : string
-name : string
-lastName : string
-phoneNumber : int
-email : string
-id : int
-insurance : string
+Manager(username : string, password : string, name : string, lastName : string, address : Address, phoneNumber : string, email : string, id : int, insurance : string)
+create(eq : Equipment) : void
+create(t : Trainer) : void
+create(c : Customer) : void
+modify(t : Trainer) : void
+modify(c : Customer) : void
+modify(eq : Equipment) : void
+search(t : Trainer) : Trainer
+search(c : Customer) : Customer
+search(e : Equipment) : Equipment
modifies
registers
Customer
-name : string
-lastName : string
-phoneNumber : int
-email : string
-id : int
-insurance : string
-membership : string
+Customer(name : string, lastName : string, address : Address, phoneNumber : int, email : string, id : int, insurance : string, membership : string, assignedRoutines : ArrayList<Routine>)
has
- address
Address
-street1 : string
-street2 : string
-city : string
-state : string
-zipCode : int
+Address(street1 : string, street2 : string, city : string, state : string, zipCode : int)
- address
has
- assignedRoutines (collection="ArrayList")
Routine
-name : string
+Routine(name : string, exercises : ArrayList<Exercise>)
has
- exercises (collection="ArrayList")
modifies
creates
assigns
Login
-username : string
-password : string
+create(type : T) : void
+modify(type : T) : void
+search(type : T) : T
modifies
creates
Equipment
-name : string
-picture : Image
-quantity : int
+Equipment(name : string, picture : Image, quantity : int)
has
- equipment (collection="ArrayList")
Exercise
-name : string
-duration : Duration
-numberOfSets : int
-repsPerSet : int
+Exercise(name : string, equipment : ArrayList<Equipment>, numberOfSets : int, repsPerSet : int)
+Exercise(name : string, equipment : ArrayList<Equipment>, duration : Duration)
creates
modifies
Trainer
-username : string
-password : string
-name : string
-lastName : string
-phoneNumber : int
-email : string
-id : int
-insurance : string
+Trainer(username : string, password : string, name : string, lastName : string, address : Address, phoneNumber : int, email : string, id : int, insurance : string, qualifications : ArrayList<qQualification>)
+create(e : Exercise) : void
+create(r : Routine) : void
+modify(e : Exercise) : void
+modify(r : Routine) : void
+assignRoutine(customer : Customer, routine : Routine) : void
+search(e : Exercise) : Exercise
+search(r : Routine) : Routine
+search(c : Customer) : Customer
WorkSchedule
-day : DayOfWeek
-startTime : LocalTime
-endTime : LocalTime
+WorkSchedule(day : DayOfWeek, startTime : LocalTime, endTime : LocalTime)
Qualification
-name : string
+Qualification(name : string)
has
- qualifications (collection="ArrayList")
has
- workSchedule (collection="ArrayList")


webapp
<<view>>
createAddress
<<view>>
createEquipment
<<view>>
createExerciseDuration
<<view>>
createManager
<<view>>
createRoutine
<<view>>
createTrainer
<<view>>
createCustomer
<<view>>
createExerciseSets
<<view>>
login
<<view>>
createQualification
<<view>>
registerTrainer
<<view>>
createWorkSchedule
service
AddressService
CustomerService
EquipmentService
ExerciseService
ManagerService
QualificationService
RoutineService
TrainerService
WorkScheduleService
controller
AddressController
EquipmentController
LoginController
QualificationController
WorkScheduleController
CustomerController
ExerciseController
ManagerController
RoutineController
TrainerController
cu
springmvc
model
Address
Equipment
Login
Qualification
Trainer
Customer
Exercise
Manager
Routine
WorkSchedule
dao
AddressDao
EquipmentDao
ManagerDao
RoutineDao
WorkSchedule
CustomerDao
ExerciseDao
QualificationDao
TrainerDao