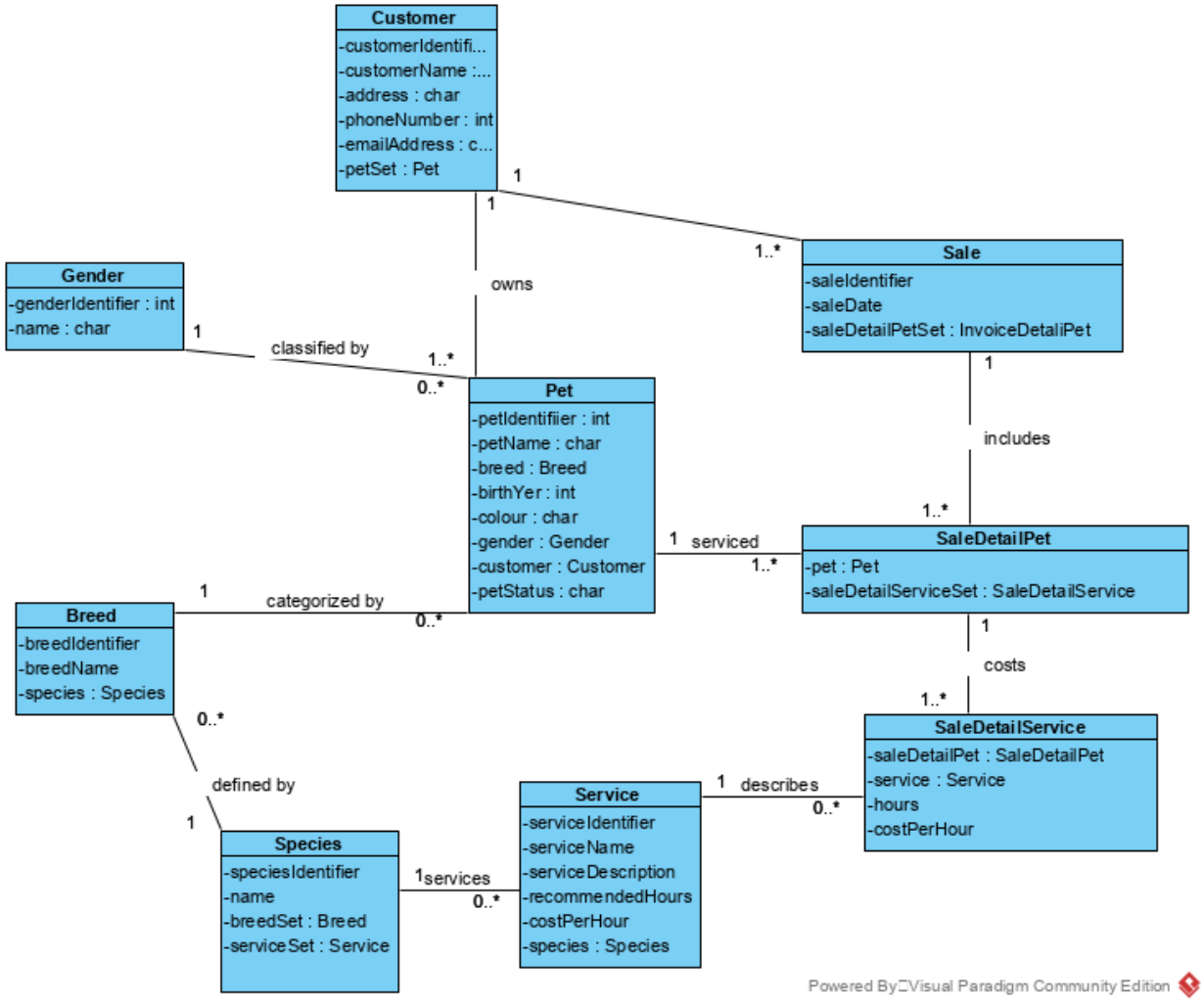
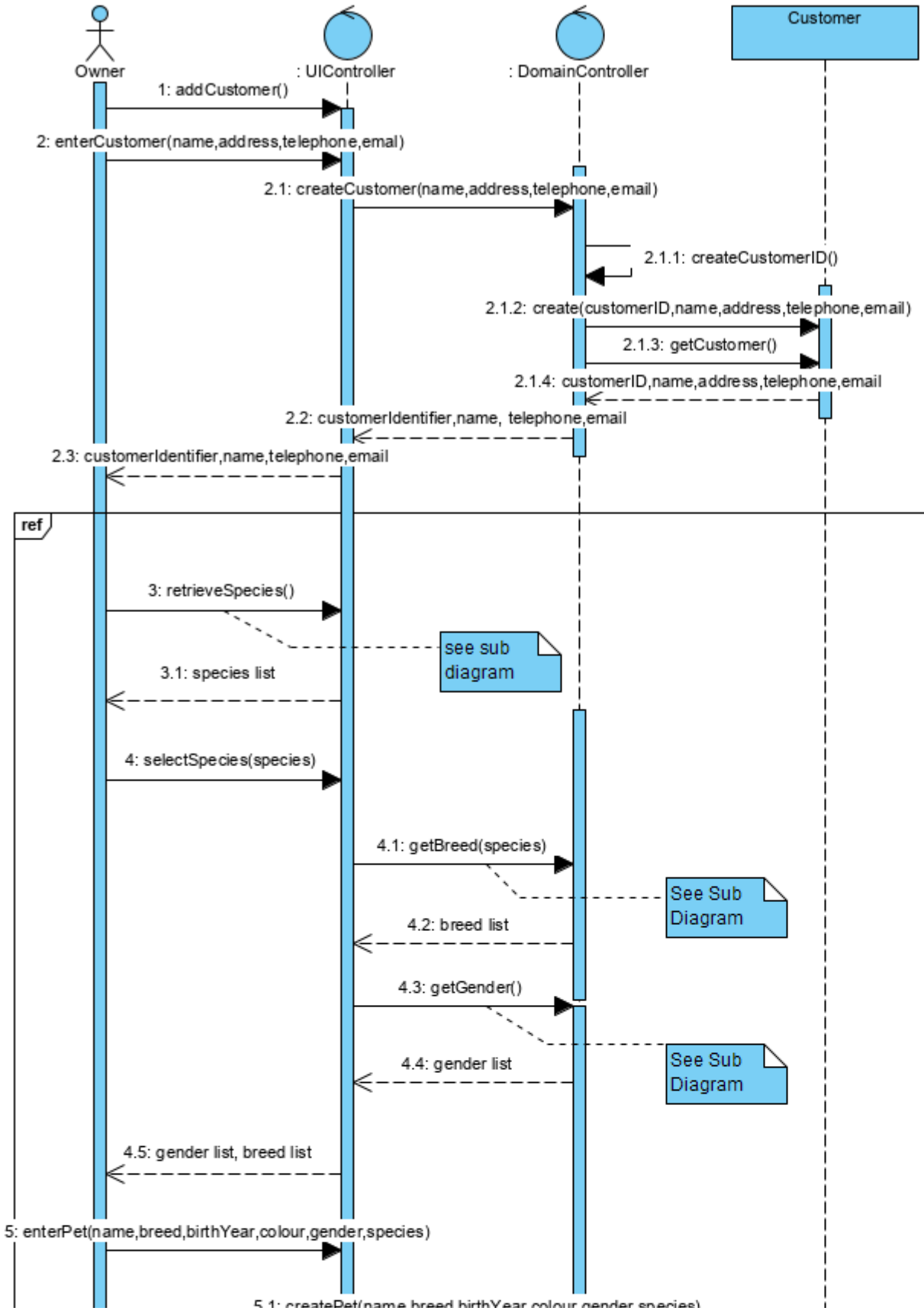
Sale Module

Your tasks:

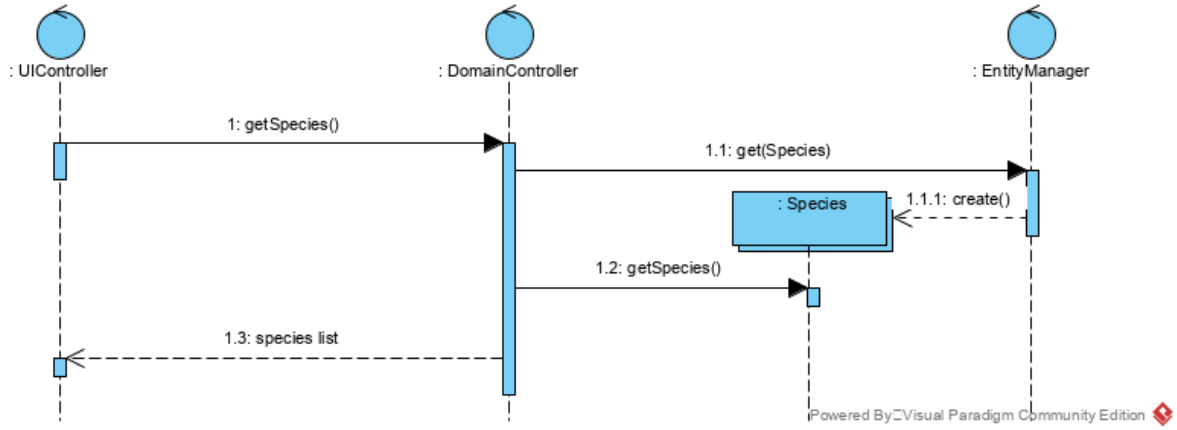
1. Create a class diagram to support the above case study and Systems Use Case Specifications



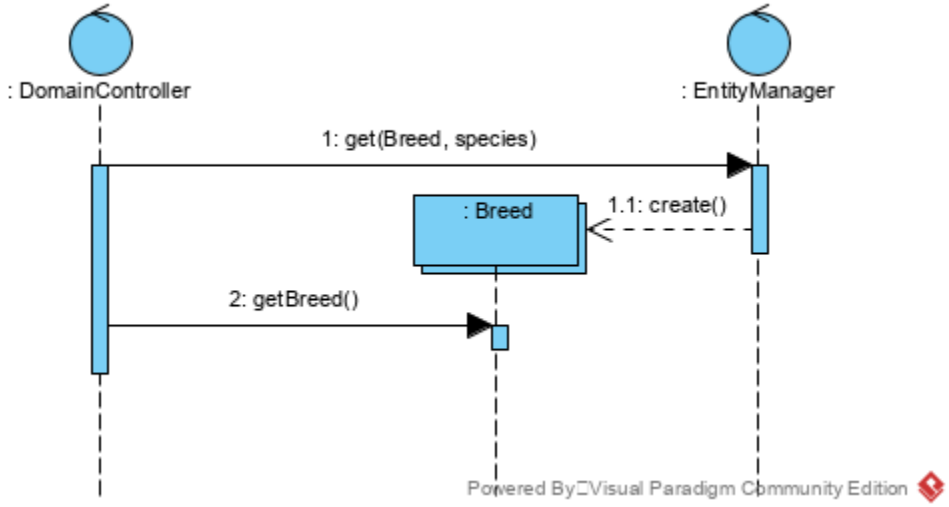
2. Create an object level sequence diagram, detailing the Create Customer systems use case specification


Owner
: UIController
: DomainController
Customer
1: addCustomer()
2: enterCustomer(name,address,telephone,emal)
2.1: createCustomer(name,address,telephone,email)
2.1.1: createCustomerID()
2.1.2: create(customerID,name,address,telephone,email)
2.1.3: getCustomer()
2.1.4: customerID,name,address,telephone,email
2.2: customerIdentifier,name, telephone,email
2.3: customerIdentifier,name,telephone,email
ref
3: retrieveSpecies()
see sub diagram
3.1: species list
4: selectSpecies(species)
4.1: getBreed(species)
See Sub Diagram
4.2: breed list
4.3: getGender()
See Sub Diagram
4.4: gender list
4.5: gender list, breed list
5: enterPet(name,breed,birthYear,colour,gender,species)

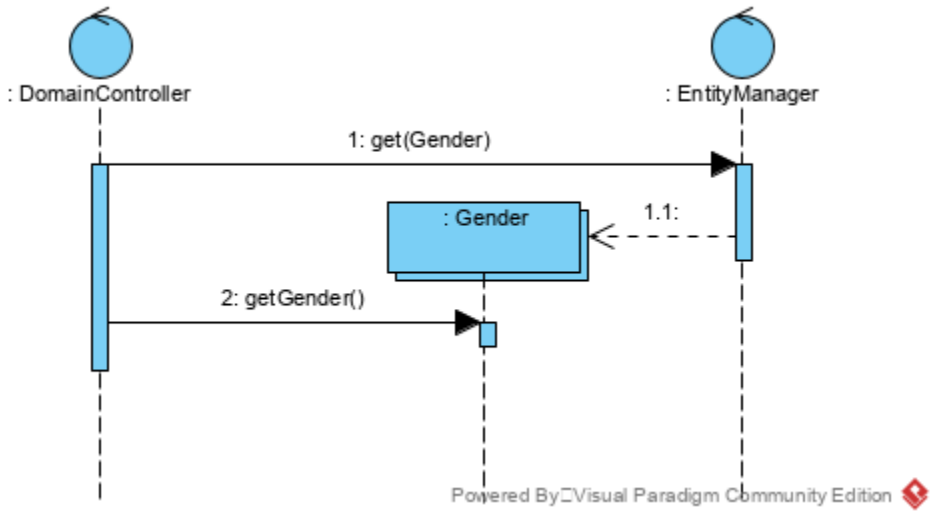
Retrieve Species



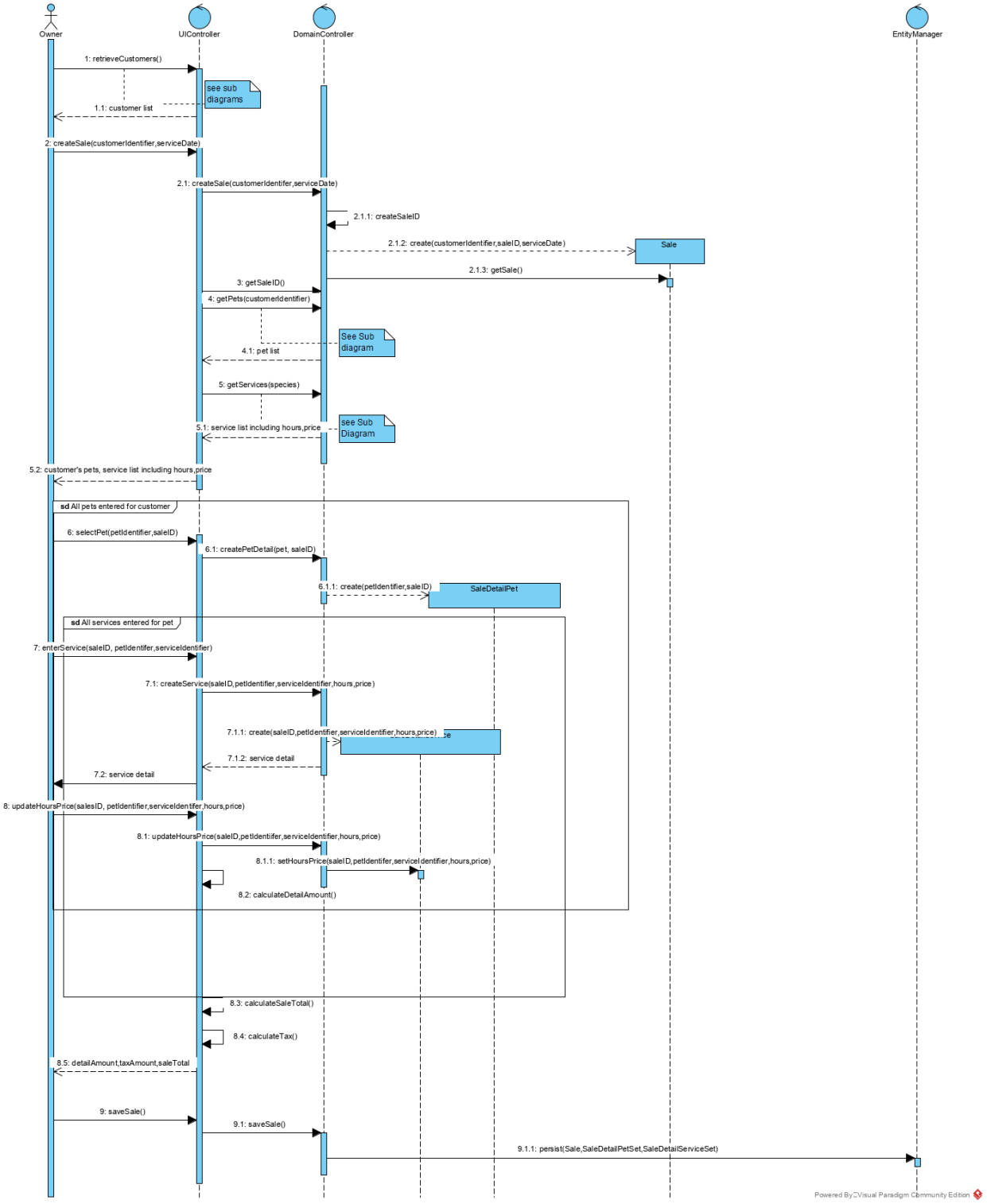
Get Breed

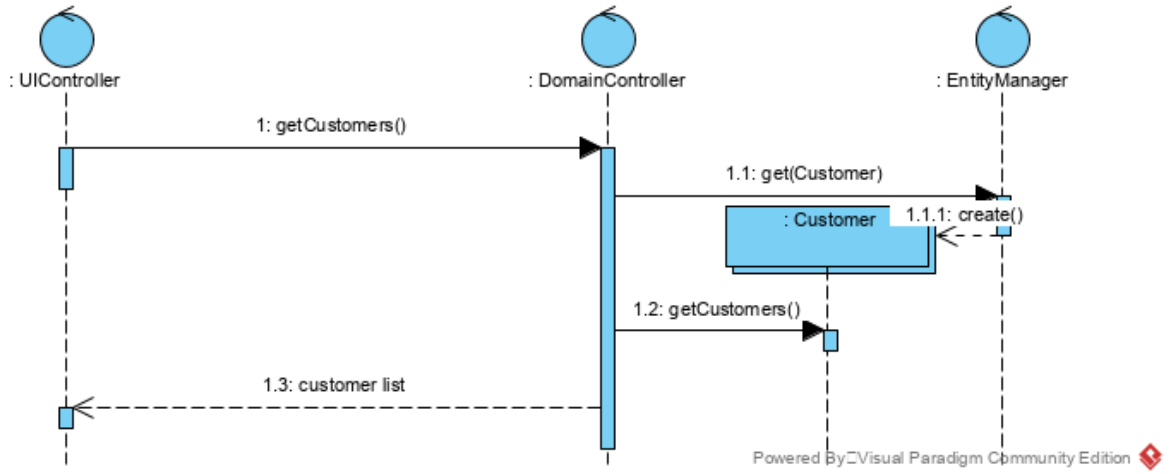


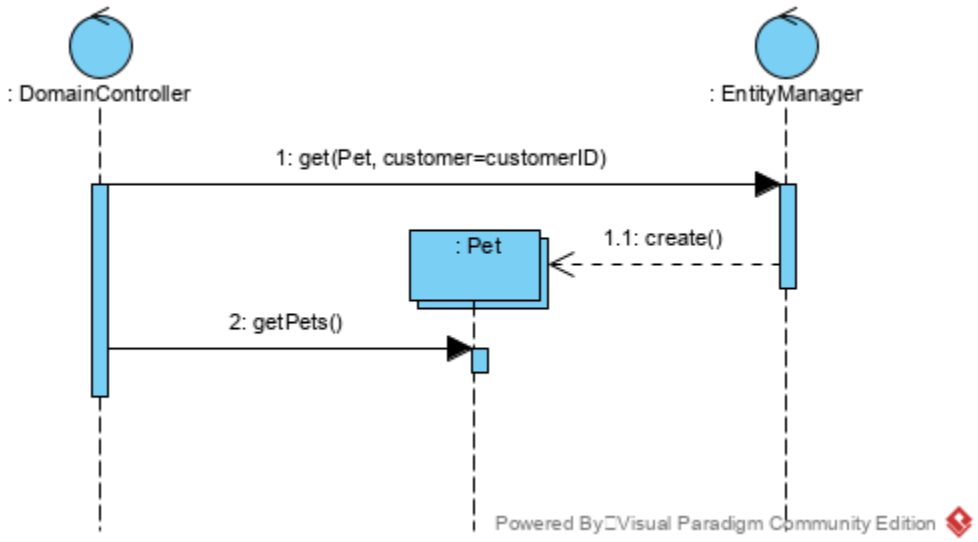
Get Gender

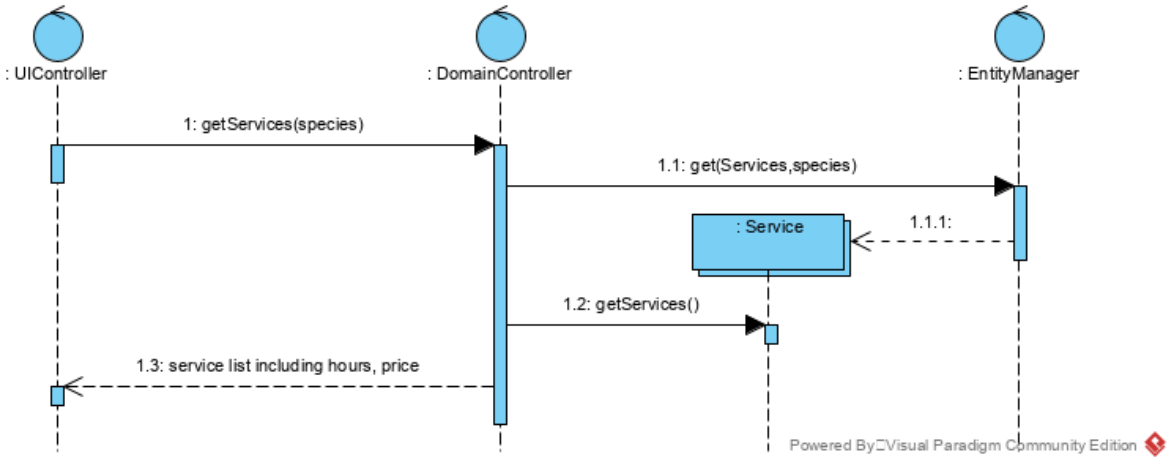

: DomainController
: EntityManager
1: get(Gender)
: Gender
1.1:
2: getGender()
Powered By Visual Paradigm Community Edition

3. Create an object level sequence diagram, detailing the Create Sales use case specification

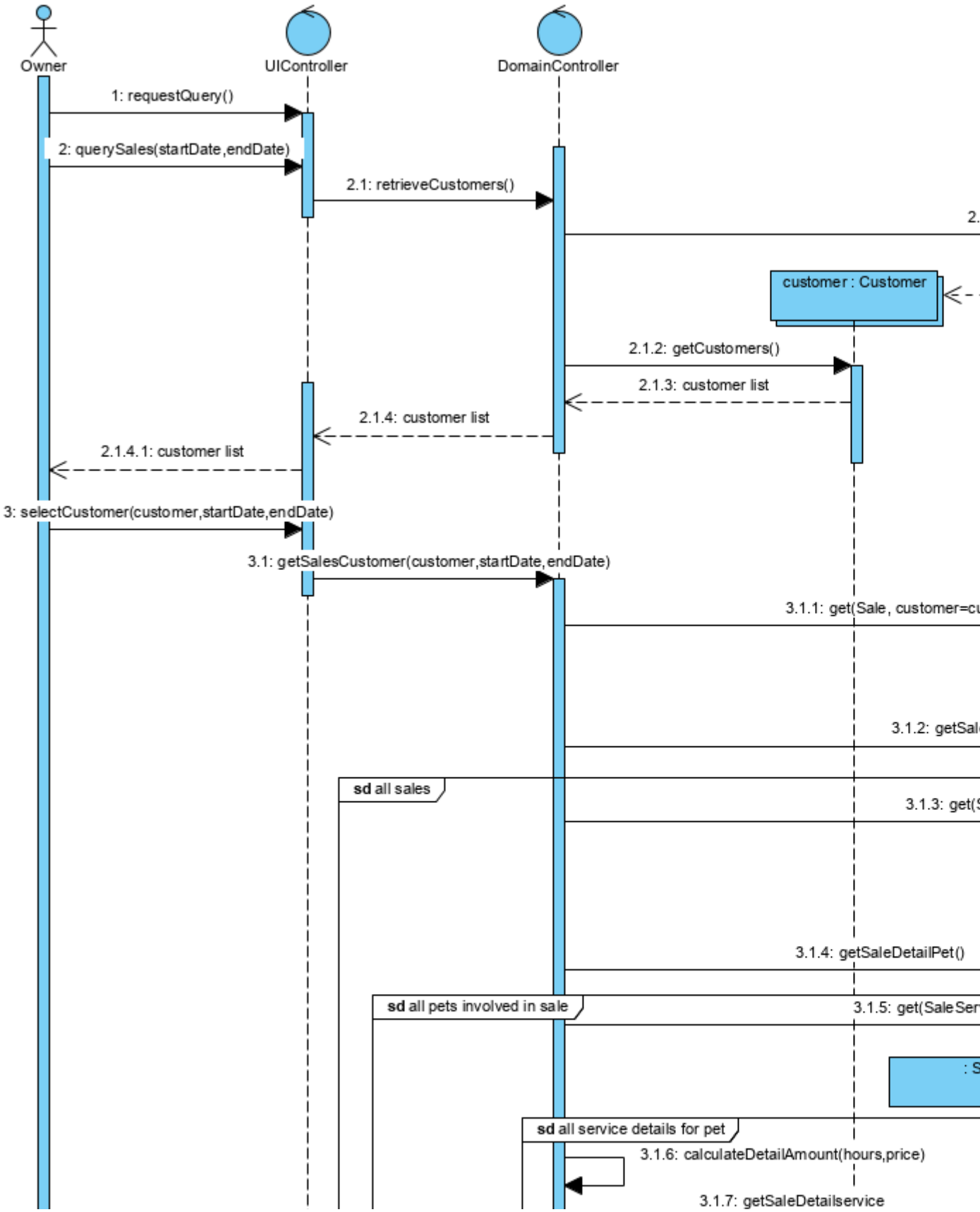



: UIController
: DomainController
: EntityManager
1: getCustomers()
1.1: get(Customer)
: Customer
1.1.1: create()
1.2: getCustomers()
1.3: customer list
Powered By Visual Paradigm Community Edition


: DomainController
: EntityManager
1: get(Pet, customer=customerID)
: Pet
1.1: create()
2: getPets()
Powered By Visual Paradigm Community Edition


: UIController
: DomainController
: EntityManager
1: getServices(species)
1.1: get(Services,species)
: Service
1.1.1:
1.2: getServices()
1.3: service list including hours, price
Powered By Visual Paradigm Community Edition

4. Create an object level sequence diagram, detailing the Query Sales (by customers) system use case specification.


Owner
UIController
DomainController
1: requestQuery()
2: querySales(startDate,endDate)
2.1: retrieveCustomers()
customer : Customer
2.1.2: getCustomers()
2.1.3: customer list
2.1.4: customer list
2.1.4.1: customer list
3: selectCustomer(customer,startDate,endDate)
3.1: getSalesCustomer(customer,startDate,endDate)
3.1.1: get(Sale, customer=cu
3.1.2: getSal
sd all sales
3.1.3: get(
3.1.4: getSaleDetailPet()
sd all pets involved in sale
3.1.5: get(SaleSer
sd all service details for pet
3.1.6: calculateDetailAmount(hours,price)
3.1.7: getSaleDetailservice

Lab Marks:

Submit UML model on time	1
Discussion participation	2
Total	3