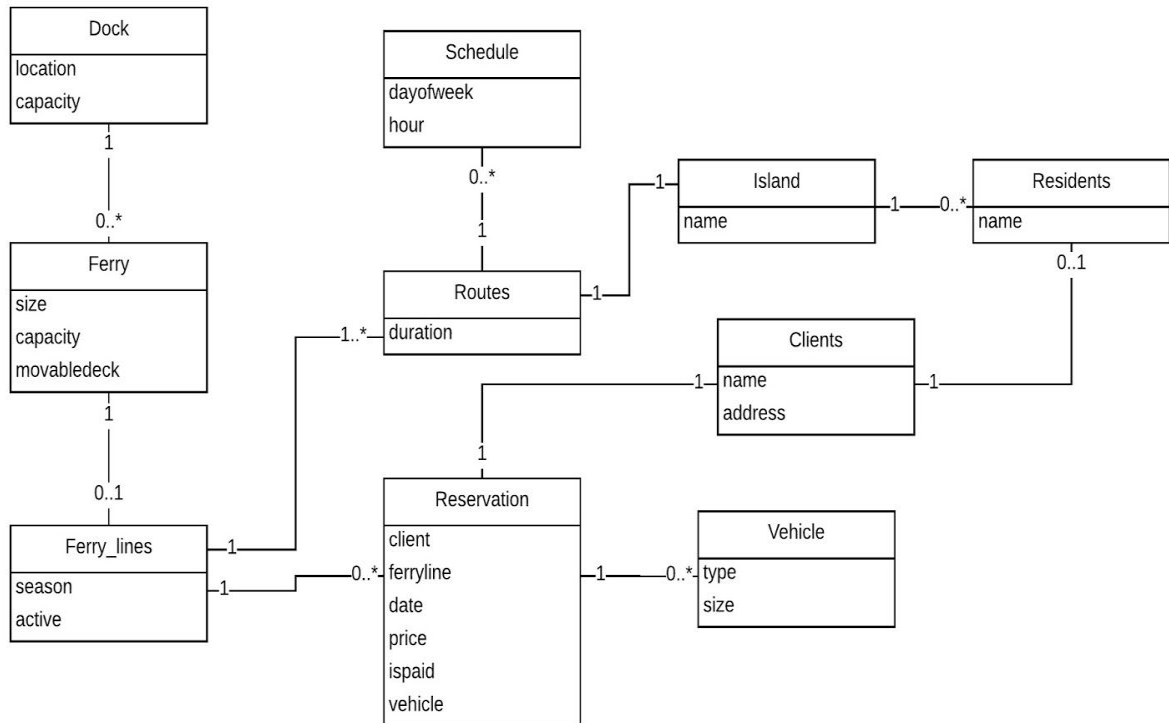# Toolbox, group X

Peter Tomasick, Raul Stelescu, Tobias Mouer, Artur Koziel
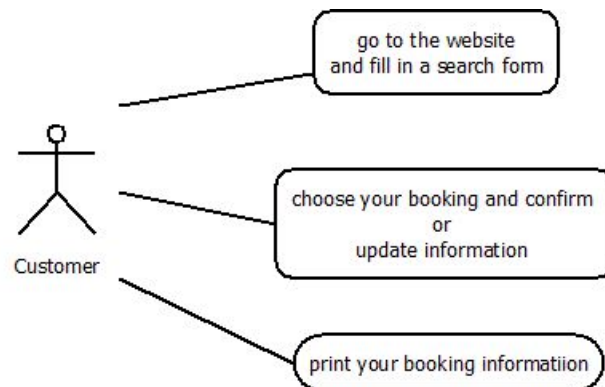Receiving group: What?
Name and email: Cristina Madalina -> cristina.madalina.dragan@gmail.com
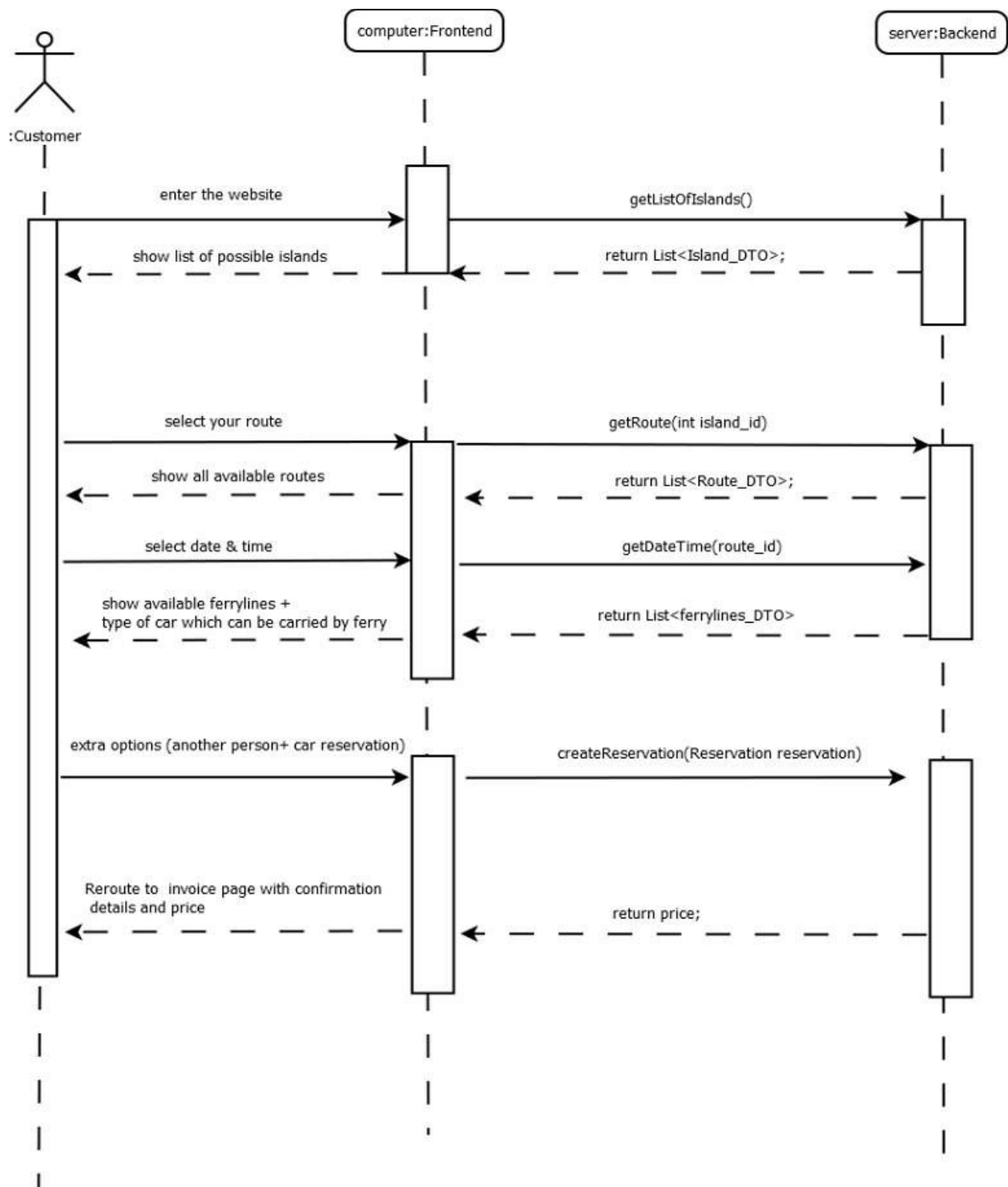
Logical data model

Use Case Model - Use Case Diagram

Use Case Model - System Sequence Diagram

Use Case Model - Detailed Use Case Description

Name   Manage ferry booking

Scope   System under design (SuD)

Level   Goal: Manage to book a ferry

Primary Actor   Customer

Precondition   The customer is on the ferry website

Main success scenario

1. Customer fills in all destination (island) on main page
2. The system will display a list of routes that he would have to chose
3. The user will have to enter the desired date and time of the departure
4. The system will show a list of ferries available for that date and time
5. The user selects the desired ferry and completes the form with details (name, address, number of cars, car type)
6. If there are enough spots on the ferry, the system will create a reservation and will send back the price he has to pay and user is being redirected to 'Print' page

Success guarantees   The customer gets on Print booking website

Extensions   Suggest other dates to book a ferry if result = 0;

Special Requirements   NONE

Communication model : DTO and ETO

```
                    ┌─────────────────────┐
                    │     <<DTO>>         │
                    │  FerryLineIdentifier│
                    ├─────────────────────┤
                    │  id: long           │
                    └─────────────────────┘
                         ▲          ▲
                        /            \
                       /              \
┌──────────────────────────┐   ┌──────────────────────────┐
│        <<DTO>>           │   │        <<DTO>>           │
│     FerryLineRoute       │   │     FerryLineFerry       │
├──────────────────────────┤   ├──────────────────────────┤
│  duration: long          │   │  size: long              │
│                          │   │                          │
│  shedule: ArrayList<Object>│ │  capacity: String        │
│                          │   │                          │
│  island: Object          │   │  movabledeck: boolean    │
└──────────────────────────┘   └──────────────────────────┘
```

```
                        ┌─────────────┐
                        │  Throwable  │
                        └─────────────┘
                          △         △
                          │         │
                  ┌───────────┐  ┌───────┐
                  │ Exception │  │ Error │
                  └───────────┘  └───────┘
                   △        △
                   │        │
                   │    ┌──────────────────┐
                   │    │ RuntimeException │
                   │    └──────────────────┘
                   │        △        △
                   │        │        │
      ┌────────────────────────┐     │
      │ ActionExpiredException  │     │
      └────────────────────────┘     │
                   △         ┌────────────────────────┐
                   │         │  FerryNotFoundException │
                   │         └────────────────────────┘
      ┌────────────────────────┐
      │  TimeExpiredException   │  ┌────────────────────────┐
      └────────────────────────┘  │ ClientNotFoundException │
                                  └────────────────────────┘
```
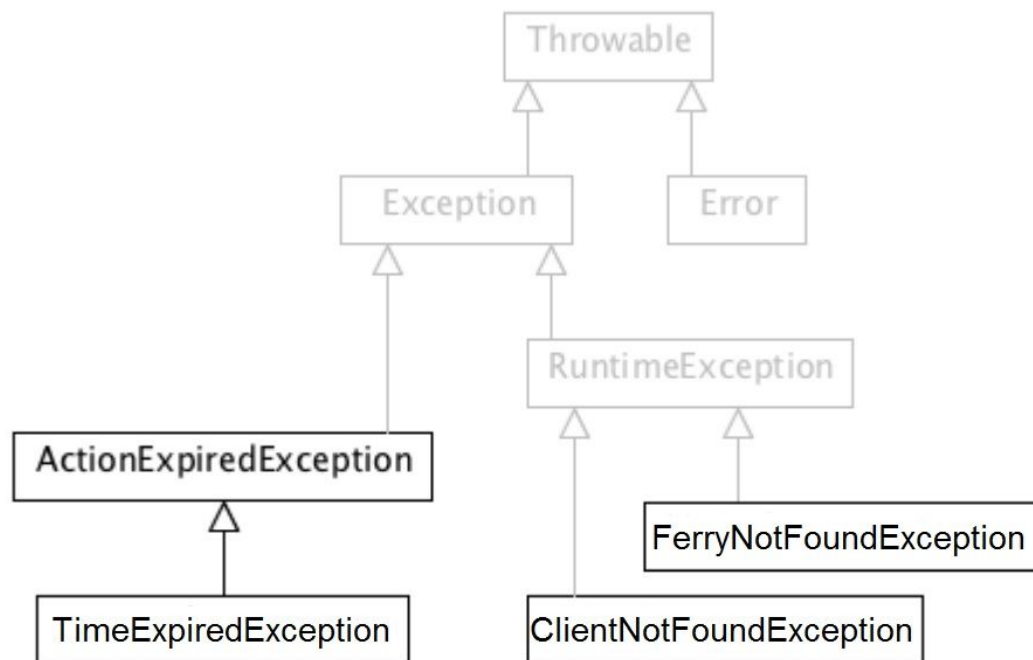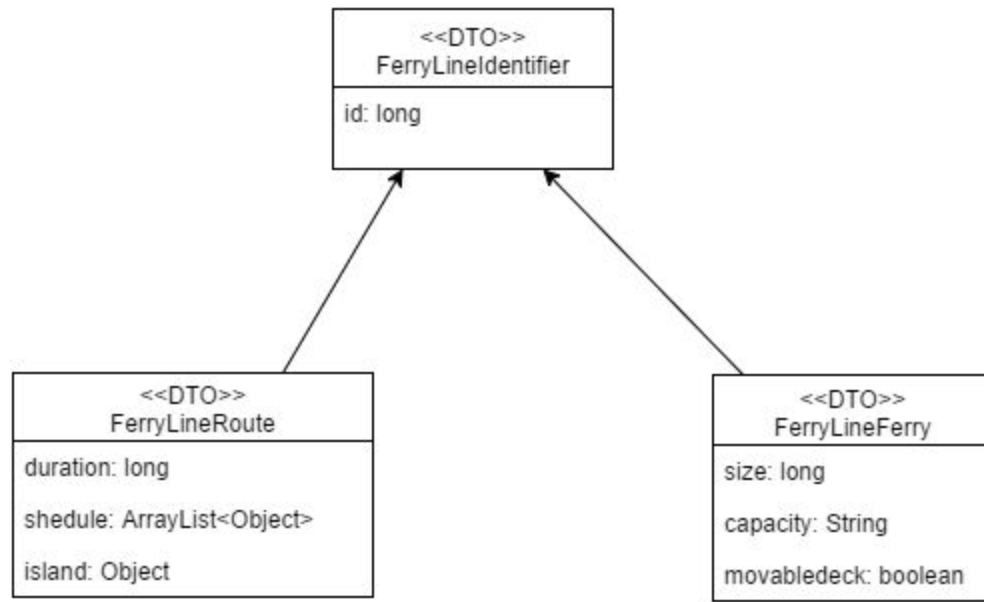
Communication model and verification implementation is on github ->
https://github.com/Zendikar/Th_ferry_lines