

1. Во еден град има одреден број на точаци за изнајмување. Корисникот, за да изнајми точак најпрво треба да се најави на системот со лични податоци (име и презиме) и картичка со која ќе плати за изнајмување на точакот. Корисникот гледа дали моментално има слободни точаци и колку, секој точак означен со посебен реден број. Корисникот може да селектира еден или повеќе точаци означени со реден број што сака да ги изнајми (редниот број на точакот не игра голема улога бидејќи сите точаци се исти). Времето на изнајмување на точакот е фиксно на 1 ден, доколку корисникот сака да го изнајми на повеќе време повторно ја поминува истата процедура. По селектирање на редниот број на точакот/точаците на корисникот му стигнува нотификација за цената која треба да ја плати, при што корисникот има опција да плати или откаже. Со плаќање на корисникот му стигнува нотификација дека точакот е активиран. (Активација на точакот се прави според активирање шифра што точакот ја има системски, секој точак си има посебна шифра според реден број.) Неколку часови пред деактивирање на точакот, на корисникот му пристигнува нотификација за колку време ќе му се исклучи точакот.

2. Ентитети се: `RentItem`, `Bike` и `Rent`, `BikeQuantity`.

Релации се: `contains` (помеѓу `BikeItem` и `Bike`), `contains` (помеѓу `BikeItem` и `Rent`).

3. Атрибути во ентитетот `RentItem` е: `itemPrice` од тип `Money`, `quantity` од тип `int`.

Атрибути во ентитетот `Bike` се: `bikeld` од тип `int` (примарен клуч), `bikeName` од тип `string`, `money` од тип `Money` и `sales` од тип `int`.

Атрибути во ентитетот `Rent` се: `rentId` од тип `int` (примарен клуч), `rentedOn` од тип `Date`, `total` од тип `float`, `rentState` од тип `boolean`.

Атрибути во `BikeQuantity`: `bikeld`.

4. Еден ограничен контекст ќе биде ентитетот `Bike`, втор ограничен контекст ќе биде ентитетот `RentItem` со ентитетот `Rent`.

Кај вториот ограничен контекст `RentItem` и `Rent` не може да се одделат затоа што `Rent` содржи изведен атрибут `total` кој постојано треба да се ажурира и зависи од `RentItem`.

5. Посебни агрегати во ограничените контексти ќе бидат:

прв агрегат: `Bike`

втор агрегат: `Rent` и `RentItem`.

6. Aggregate roots ќе бидат:

прв агрегат: `Bike`

втор агрегат: `Rent`.