

ITTalents Season 15

Test 3

Collections, Exceptions, Concurrency, DB

Да се реализира приложение, симулиращо работата на автосервиз.

Сервизът приема коли за диагностика и сервизни дейности.

В сервиза могат да се извършват различни услуги. Всяка услуга има

- тип – **РЕМОНТ** или **ПОДДРЪЖКА**
- име
- цена
- време за извършване на услугата

Сервизът има предварително дефинирани услуги, които се съхраняват в базата данни.

При стартиране на сервиза, услугите се зареждат в списък и се запазват за последваща употреба. **(5т)**

В сервиза могат да постъпват поръчки за обслужване на кола. Всяка кола има:

- регистрационен номер
- модел
- име на собственик
- телефон за връзка
- година на производство
- дали е в гаранция

При постъпване на кола, тя се обработва от **приемчик**, който проверява дали колата идва за първи път в сервиза.

- Ако автомобил с такъв регистрационен номер не е идвал, то той се добавя в базата данни.
- Ако автомобил с такъв регистрационен номер вече съществува в базата, то се проверява дали телефонът в базата данни е актуален и ако не, се актуализира. **(10т)**

След това **приемчикът** създава поръчка за този автомобил, записва поръчката в базата данни, като за нея подава само ID-то на автомобилът и датата на регистрация. Накрая **приемчикът** добавя поръчката в списък с поръчки за диагностика в сервиза. **(10т)**

Всяка поръчка съдържа

- кола, която се обслужва
- дата на регистриране на поръчката
- диагностик, който е направил диагностиката на колата
- дата на диагностиката
- услуга, която трябва да се извърши
- дата на обслужването
- работник, който е извършил услугата
- дата на затваряне на поръчката (предаване на колата на собственика)

В сервиза работят **диагностици**, които постоянно са в очакване на постъпване на поръчка за диагностика. Ако няма поръчки за диагностика, те чакат да се появи такава. Ако такава е налична, те взимат автомобила и му правят диагностика в рамките на 5 секунди, след което решават от какъв тип услуга се нуждае автомобила (на произволен принцип), редактират поръчката в базата данни (въвеждат дата на диагностика и от кого е направена диагностиката, както и от каква услуга се нуждае автомобила) и преместват поръчката.

- Ако поръчката се нуждае от услуга тип **РЕМОНТ**, тя отива в списък с поръчки „за ремонт“.
- Ако поръчката се нуждае от услуга тип **ПОДДРЪЖКА**, тя отива в списък с поръчки „за поддръжка“.

(15т)

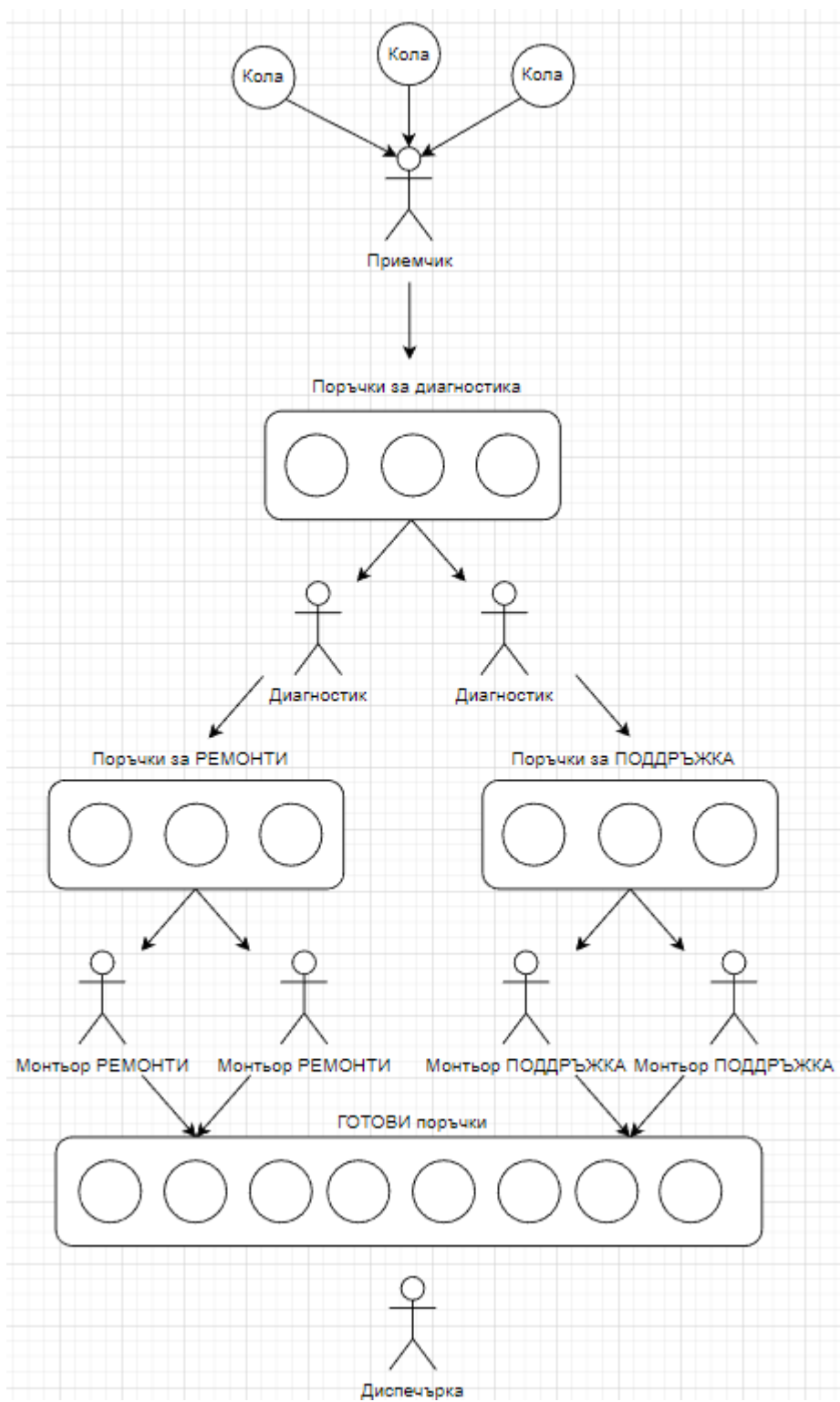
В сервиза също работят **монтъори**. При наемане на монтъор той се зачислява или към поръчки тип **РЕМОНТИ** или към **ПОДДРЪЖКА**.

Монтъорите постоянно са в очакване на постъпване на поръчка за **РЕМОНТИ** или за **ПОДДРЪЖКА**, според това за какво отговарят. Ако няма такава поръчка, те чакат да се появи такава. Ако такава е налична, те взимат автомобила и му прилагат услугата, като това отнема време според вида на услугата. След това редактират поръчката в базата данни (въвеждат дата на поправката и от кого е направена) и преместват поръчката в списък „готови“. **(15т)**

В сервиза работи и една **диспечърка**, която постоянно е в очакване за „готова“ поръчка. Ако такава не съществува, тя чака да се появи. Ако е налична „готова“ поръчка, тя се обажда на телефона за контакт (това отнема 1 секунда), маркира поръчката като затворена в базата данни (подава дата на затваряне на поръчката) и създава текстов файл с данните за поръчката (който принтира и предава на клиента). Текстовия файл е със заглавие - <licence_plate>-<close_date>.txt

(15т)

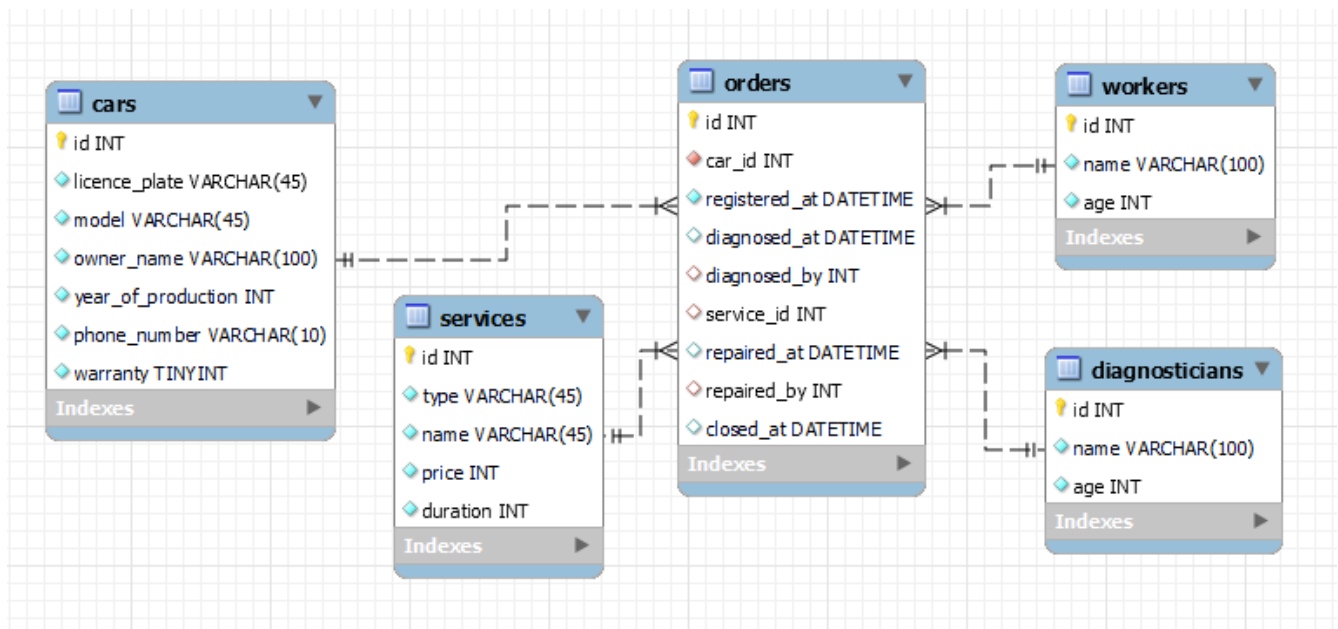
Обща схема на работата на сервиза можете да видите тук:



На всеки месец сервизът прави справки за дейността си, като извлича данни от базата:

- брой поръчки, които са заявени (регистрирани) за последните 30 дни (5т)
- брой поръчки, които са за услуги тип РЕМОНТ и брой поръчки, които са за услуги тип ПОДДРЪЖКА (5т)
- името на диагностика, извършил най-много диагностики на автомобили (5т)
- регистрационният номер на всички автомобили с поне 3 извършени услуги (5т)
- брой поръчки за автомобили в гаранция (5т)
- обща сума на извършените услуги (ако услугата е извършена върху автомобил в гаранция, сумата за услугата е 0) (5т)

Диаграма на базата данни можете да видите тук:



Да се реализира демо, в което:

- Създава се сервиз за автомобили
- Създават се двама **диагностика** с произволни име и възраст, вписват се в базата данни и се пускат „на работа“ в сервиза, в очакване на поръчки за диагностика
- Създават се двама **монтъора** с произволни име и възраст, вписват се в базата данни и се пускат „на работа“ в сервиза, в очакване на поръчки за **РЕМОНТ**
- Създават се двама **монтъора** с произволни име и възраст, вписват се в базата данни и се пускат „на работа“ в сервиза, в очакване на поръчки за **ПОДДРЪЖКА**
- Създава се **приемчик**, който може да регистрира поръчки на автомобили.
- Създава се **диспечърка**, в очакване да завършва готови поръчки
- Създават се 10 автомобили с произволни характеристики. 5 от тях се подават веднъж на приемчика за регистрация. 3 от тях се подават два пъти, 2 от тях се подават 3 пъти.

ЖОКЕР: за да добавите LocalDateTime в базата в колона от тип DATETIME:

```
LocalDateTime dateTime = LocalDateTime.now();  
//... prepared statement logic with first parameter for column of type DATETIME  
ps setObject(1, dateTime)
```