**Функциски зависности и нормализација**

Бидејки табелата на збирни атрибути би била преголема решивме да поделиме по функционалност на апликацијата и тоа.

1. Менаџирање на постоечки продукти (WINE, WINE\_TYPE)

2. Менаџирање на вработени и возила (EMPLOYEE, VEHICLE, VEHICLE\_TYPE, WAREHOUSE, ADDRESS)

3. Менаџирање на достава (SHIPMENT, SHIPMENT\_LOAD, WINE, CUSTOMER, PAYMENT, EXPENSE, SHIPMENT\_HAS\_EXPENSE\_OF\_EXPENSE\_TYPE, EXPENSE\_TYPE)

4. Менаџирање на клиенти (CUSTOMER, CUSTOMER\_TYPE, ADRESS)

**1. Менаџирање на постоечки продукти**

R = { **wine\_id,** wine\_name, base\_price, year\_produced, **wine\_type\_id**, wine\_type\_name, wine\_type\_description, wine\_type\_region }

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| wine\_id | wine\_name | base\_price | year\_produced | wine\_type\_id | wine\_type\_name | wine\_type\_description | wine\_type\_region |
| 1 | Смедеревка | 450 | 2022 | 1 | Смедеревка | Црвено | Кавадарци |
| 2 | Траминец | 650 | 2021 | 2 | Траминец | Бело | Неготино |
| 3 | Смедеревка | 650 | 2021 | 1 | Смедеревка | Црвено | Кавадарци |

Во релацијата постојат следните функциски зависности. Задоволува 1НФ, но не задоволува втора бидејки имаме непримарни атрибути кои не зависат само од кандидат примарниот клуч.

wine\_id → wine\_name, base\_price, year\_produced

wine\_type\_id → wine\_type\_name, wine\_type\_description, wine\_type\_region

**Декомпозиција**

Правиме декомпозиција за да ја доведеме базата до 2НФ, со декомпозиција на релацијата од втората функционална зависност произлегува нова релација и тоа:

R1 = { **wine\_type\_id**, wine\_type\_name, wine\_type\_description, wine\_type\_region }

R2 = { **wine\_id,** wine\_name, base\_price, year\_produced, wine\_type\_id }

Со ова релацијата ги задоволува 1НФ, 2НФ и 3НФ бидејки нема транзитивни зависности.

**2. Менаџирање на вработени и возила**

R = { **employee\_id**, employee\_name, employee\_surname, **vehicle\_id**, registration, **vehicle\_details\_id**, make, model, capacity, **vehicle\_type\_id**,vehicle\_type\_name, **warehouse\_id**, warehouse\_name, **address\_id**, street, city, building\_number, postcode }

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| employee\_id | employee\_name | employee\_surname | vehicle\_id | registration | vehicle\_details\_id | make | model | capacity | vehicle\_type\_id | vhicle\_type\_name | warehouse\_id | warehouse\_name | adress\_id | Street | City | building\_number | postcode |
| 1 | Јован | Јовановски | 1 | КУ111СБ | 1 | Форд | Трансит | 1300 | 1 | Комбе | 1 | Страцин | 1 | Гоце Делчев | Кратово | 12 | 1360 |
| 2 | Коле | Колев | 2 | КУ222СЦ | 1 | Сканиа | Р420 | 20000 | 2 | Камион | 1 | Страцин | 1 | Гоце Делчев | Кратово | 12 | 1360 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

employee\_id → employee\_name, employee\_surname, warehouse\_id

vehicle\_details\_id → make, model, capacity

vehicle\_id → registration

vehicle\_type\_id → vehicle\_type\_name

warehouse\_id → warehouse\_name, address\_id

address\_id → street, city, building\_number, postcode

**Декомпозиција**

Релацијата задоволува 1НФ, но не задоволува 2НФ па според тоа не задоволува ни 3НФ.

Ако почнеме со декомпозиција на фукнционалните завсности произлегуваат нови релации и тоа:

R1 = { employee\_id , employee\_name, employee\_surname, warehouse\_id }

R2 = { vehicle\_details\_id, make, model, capacity, vehicle\_type\_id }

R3 = { vehicle\_id, registration, vehicle\_details\_id }

R4 = { vehicle\_type\_id, vehicle\_type\_name }

R5 = { warehouse\_id, warehouse\_name, address\_id }

R6 = { address\_id, street, city, building\_number, postcode }

R7 = { employee\_id, vehicle\_id } вештачка релација за мапирање employee\_drives\_vehicle

Со овие новодобиени релации покриваме 2НФ, следно останува да провериме дали има транзитивни зависности. Тука ако гледаме имаме транзитивна зависност во

postcode → city, но оваа зависност решивме да ја игнорираме, бидејки нема да имаме посебни кодови по општини, туку 1000 би било за сите општини во Скопје.

Со ова секој непримарен атрибут е зависен само од примарниот клуч во својата релација, па релациите се и во 3НФ.

**3. Менаџирање на достава**

R = { **shipment\_id**, shipment\_date, **wine\_id,** wine\_name, base\_price, year\_produced, **customer\_id**, customer\_name, customer\_email, customer\_phone\_number, wine\_quantity, wine\_base\_price, wine\_agreed\_price, **expense\_type\_id**, expense\_type\_name, expense\_type\_description, amount, **payment\_id**, payment\_status }

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| shipment\_id | shipment\_date | wine\_id | wine\_name | base\_price | year\_produced | customer\_id | custome\_name | customer\_email | customr\_phone\_number | wine\_quantity | wine\_base\_price | wine\_agreed\_price | expense\_type\_id | expense\_type\_name | expense\_type\_description | amount | payment\_id | payment\_status |
| 1 | 12/12/2024 | 1 | Смедеревка | 1000 | 2023 | 1 | Веро | test@test.com | 12345 | 20 | 1000 | 900 | 1 | Гориво | Гориво | 1000 | 1 | draft |
| 2 | 12/11/2024 | 1 | Смедеревка | 1000 | 2023 | 1 | Рамстор | t@t.com | 3214 | 12 | 1000 | 950 | 2 | Патарина | Патарина | 500 | 2 | completed |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Како најголема и најкомплицирана оваа релација има голем број на зависности.

За почеток може да заклучиме дека релацијата е во 1НФ, бидејки нема сложени атрибути,

но не е во 2 НФ, а со тоа ниту во 3НФ.

Релацијата ги има следните функционални зависности.

shipment\_id → shipment\_date, payment\_id, payment\_status

wine\_id → wine\_name, base\_price, year\_produced

customer\_id → customer\_name, customer\_email, customer\_phone\_number

expense\_type\_id → expense\_type\_name

shipment\_id, wine\_id, customer\_id → wine\_quantity, wine\_base\_price, wine\_agreed\_price

shipment\_id, expense\_type\_id → amount

**Декомпозиција**

Од овие зависности со претварање во 2НФ ги добиваме следните релации

R1 = { shipment\_id, shipment\_date, payment\_id, payment\_status }

R2 = { wine\_id, wine\_name, base\_price, year\_produced }

R3 = { customer\_id, customer\_name, customer\_email, customer\_phone\_number }

R4 = { expense\_type\_id, expense\_type\_name }

R5 = { shipment\_id, wine\_id, customer\_id, wine\_quantity, wine\_base\_price, wine\_agreed\_price }

R6 = { shipment\_id, expense\_type\_id, amount }

Овде мора да имаме изведени релации за атрибутите кои произлегуваат од врската m:n:p, врската меѓу релациите shipment, wine и customer и врската m:n меѓу shipment и expense\_type примарниот клуч тука ќе биде комбинација од примарните клучеви во соодветните релации.

Со ова доаѓаме поблизу до нормална форма на релацијата, но имаме транзитивна зависност за типот payment\_status преку payment\_id во релацијата.

R1 = { shipment\_id, shipment\_date, payment\_id, payment\_status }

Oдносно shipment\_id транзитивно го одредува payment\_status преку payment\_id

Продолжуваме со декомпозиција и добиваме две нови релации за shipment и за payment.

Бидејки кардиналноста е 1:1 може да избереме каде ќе го чуваме надворешниот клуч па според тоа во нашиот случај ќе избереме да го чуваме клучот кон payment како надворешен клуч во shipment.

Релациите кои ќе се добијат се следните:

R1\_1 = { shipment\_id, shipment\_date, payment\_id }

R1\_2 = { payment\_id, payment\_status }

Бидејки кардиналност е 1:1 би можеле статусот да го чуваме во истата релација shipment, но намерно решивме да ги одвоиме за да имаме отворени можности за проширување на табелите.

За транзитивната зависност во релацијата customer (R3) ќе биде дообјаснето во делот што следи односно менаџирање на клиенти.

**4. Менаџирање на Клиенти**

R = { **customer\_id**, customer\_name, customer\_email, customer\_phone\_number, **customer\_type\_id**, customer\_type\_name, customer\_type\_description, **address\_id**, street, city, building\_number, postcode }

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| customer\_id | customer\_name | customer\_email | customer\_phone\_number | customer\_type\_id | customer\_type\_name | customer\_type\_description | address\_id | street | city | building\_number | postcode |
| 1 | Веро | test@test.com | 12123456789 | 1 | Правно лице | Претпријатие | 1 | Тестна | Скопје | 24 | 1000 |
| 2 | Веро Џамбо | test@test.com | 121987654 | 2 | Тип 2 | Големо претпријатие | 1 | Тестна | Скопје | 24 | 1000 |
| 3 | Име Имев | test@test.com | 23412421 | 3 | Физичко лице | Поединец | 2 | Улица 16 | Скопје | 25 | 1000 |

Во релацијата постојат следните функциски зависности. Задоволува 1НФ но не задоволува 2НФ бидејки имаме непримарни атрибути кои не зависат само од кандидат примарниот клуч.

customer\_id → customer\_name, customer\_email, customer\_phone\_number, address\_id

customer\_type\_id → customer\_type\_name, customer\_type\_description

address\_id → street, city, building\_number, postcode.

**Декомпозиција:**

Од постоечката релација со декомпозиција ќе се добијат релациите

R1 = { customer\_id, customer\_name, customer\_email, customer\_phone\_number, address\_id, customer\_type\_id }

R2 = { customer\_type\_id, customer\_type\_name, customer\_type\_description }

R3 = { address\_id, street, city, building\_number, postcode }

Во новодобиените релации имаме транзитивна зависност преку

customer\_email → customer\_name, customer\_phone\_number

customer\_phone\_number → customer\_name, customer\_email

Овие зависности ќе бидат оставени бидејки немаме потреба да креираме нова релација за истите. Целта на ова е да намалиме на сложеност на прашалниците кога ќе треба да извлечеме податоци за клиентите. Ќе исфрлиме непотребни JOIN операции.