**Тема22**

**Проект по бази от данни за летище**

**Изготвил: Любомир Русинов, ф.н.72107**

1. **Обхват на модела. Дефиниране на задачата.**

Базата от данни за летището ще съхранява информацията за данните в него. Разработената база от данни ще обслужва вътрешната система на летището, т.е. няма да обслужва сайта. Летището все още има единствен терминал, който има име, и от него се осъществяват всички полети. За всеки един полет се съхранява следната информация: авиокомпания, изход, от който пътниците трябва да се качат, гишета, на които пътниците могат да регистрират своя багаж, начална и крайна точка на полета, номер, който го определя еднозначно, час и дали е пристигащ или заминаващ.

На Летището работят служители и всеки един от тях има име, длъжност, работно място в сградата на летището и служебен номер, който го определя еднозначно. Работните им места може да са на гишета, изходите към полети или паспортен контрол. Тези работни места имат служебно зададени номера, които ги определят еднозначно. За охрана на летището е наета частна фирма, чиято информация не е включена в базата данни на летището.

Летището и неговият персонал работят с пътниците, за които се пази информация. Съхраняват се имена, имейл, дали е заминаващ или пристигащ и билет, с който пътува пътника. Всеки един билет съдържа номер на полета, за който е предназначен и уникален номер, който го определя еднозначно.

1. **Множество от същности и техните атрибути**

* Терминал - име
* Полети - авиокомпания, номер, изход, гишета, начална и крайна точка, вид, час
* Служители – служебен номер, име, длъжност
* Работно място – служител на смяна, вид
* Изход
* Гише
* Паспортен контрол
* Пътници –име, билет, имейл, вид
* Билети – уникален номер, номер на полет, начална и крайна точка

1. **Домейн на атрибутите**

* Терминал – име: низ
* Полети - име на авиокомпанията: низ, номер: низ, начална и крайна точка: низ, вид: низ, час: низ
* Служители – служебен номер: естествено число, име: низ, длъжност: низ
* Работно място – служител на смяна: низ, вид: низ
* Изход- номер: низ
* Гише - номер: низ
* Паспортен контрол - номер: низ
* Пътници – вид: низ, име: низ, имейл: низ,
* Билети – уникален номер: низ, номер полет: низ

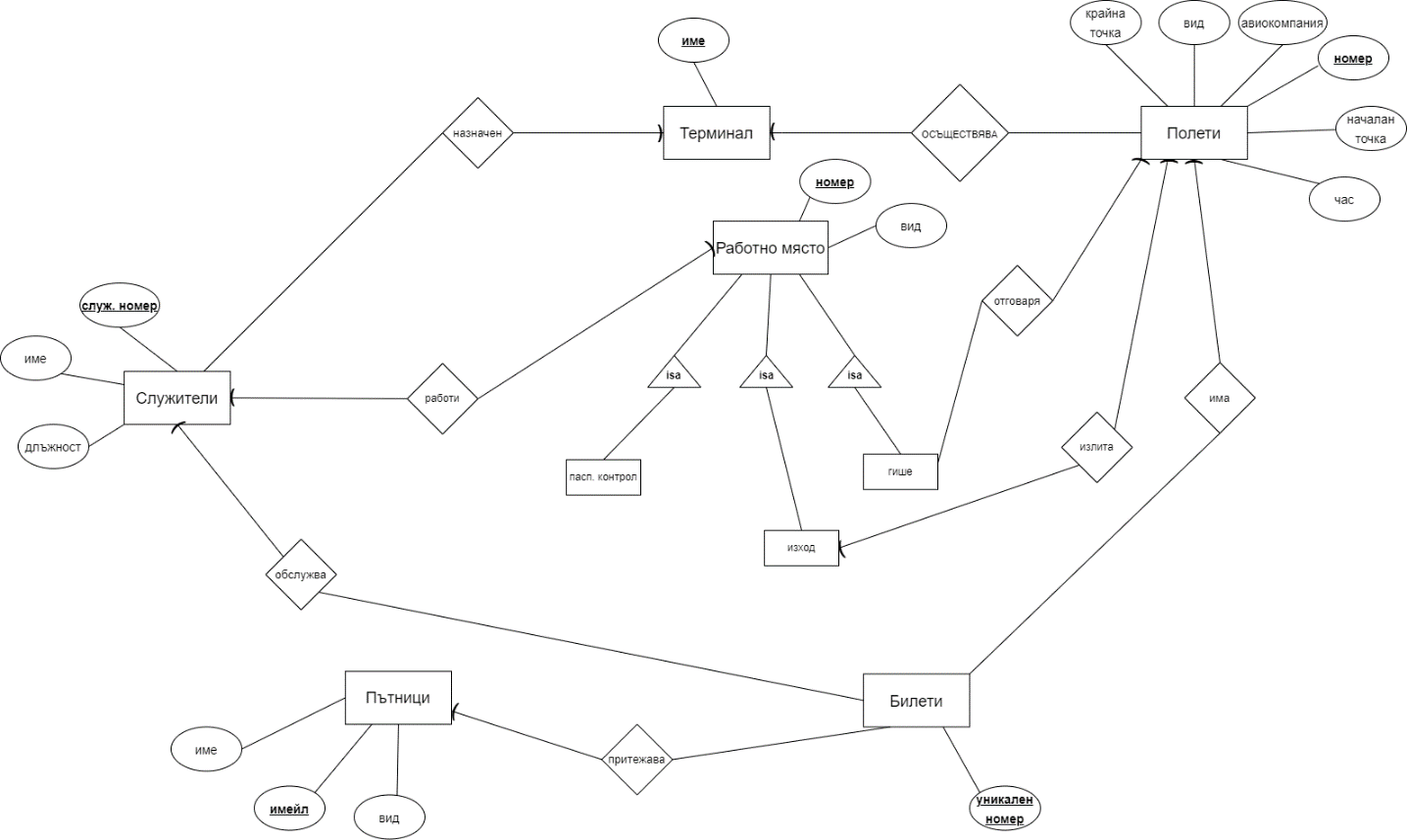
1. **Връзки**

* На един Терминал се осъществяват много Полети. Един Полет може да се осъществи на точно един Терминал.
* В един Терминал са назначени много Служители. Един Служител е назначен в точно един Терминал.
* За един Полет има много Билети. Един билет е точно за един Полет.
* Едно Работно място работи точно един Служител. Един Служител работи на точно едно Работно място.
* Един Пътник може да има един или повече Билета. Един Билет може да принадлежи на точно един Пътник.
* Един Служител може да продава нула или повече Билети. Един билет може да бъде продаден от точно един пътник.

1. **Ключове**

* Терминал – име: еднозначно определя терминала
* Полети - номер: еднозначно определя полета
* Служители – служебен номер: еднозначно определя служителя
* Работно място - номер: еднозначно определя работното място
* Пътници – имейл: еднозначно определя пътника
* Билети – уникален номер: еднозначно определя билета

1. **E/R модел на данни**



1. **Релационен модел на данни**

Първо преобразуваме isa-йерархиите, после множествата от същности, после връзките много-много и накрая връзките много-един. Връзките много-един оптимизираме.

За преобразуването на isa-йерахия може да използваме три подхода: ER, ООП и NULL. За isa-йерархията WorkPlace-PassportControl-Gate-Desk e по-подходящо да използваме ООП а не ER подхода, защото Работното място може да е само един от трите типа и не би ни трябвало допълнителна таблица, в която да са всички Работни места.

NULL подходът не е подходящ, защото ако всичките Работни места са в една таблица, Паспортният контрол и Изходите няма да се различават понеже нямат никакви атрибути и съответно няма да има NULL поле. Единствената разлика ще е, че Гишетата имат връзка с полети.

За решението на задачата ще използваме ООП подхода.

След преобразуване на isa-йерархията получаваме релационни схеми:

* PassportControl (number, type, emplyeeNum)
* Gate(number, flightNum, type, emplyeeNum)
* Desk(number, flightNum, type, emplyeeNum)
* ~~correspond(number, flightNum)~~
* ~~takesOff(number, flightNum)~~
* Flights(number, company, type, time, startPoint, endPoint, terminalName)

Продължаваме с другите множества от същности и връзки:

* Terminal(name)
* Employee(number, name, position, terminalName)
* Passengers(email, name, type)
* Tickets(uniqueNum, employeeNum, email, flightNum)
* ~~аccomplish(terminalName, flightNum)~~
* ~~appointed(terminalName, employeeNum)~~
* ~~sells(employeeNum, uniqueNum)~~
* ~~has(email, uniqueNum)~~
* ~~works(employeeNum, workPlaceNum)~~

Окончателната схема на базата от данни се състои от следните релационни схеми:

* PassportControl (number: char(3) , type: varchar(15), emplyeeNum: int)
* Gate(number: char(3), flightNum: char(8) , type: varchar(15), emplyeeNum: int)
* Desk(number: char(3), flightNum: char(8) , type: varchar(15), emplyeeNum: int)
* Flights(number: char(8) company: varchar(50), type: char, time: char(5), startPoint: varchar(40), endpoint: varchar(40), terminalName: int)
* Terminal(name: int)
* Employee(number: int, name: varchar(50), position: varchar(20), terminalName: int)
* Passengers(email: char(30), name: varchar(50), type: char)
* Tickets(uniqueNum: int, employeeNum: int, email: varchar(30), flightNum: char(8))

# Релационен модел на данни (релационни схеми)

Схемата на базата от данни се състои от следните релационни схеми:

* PassportControl (number, type, emplyeeNum)
* Gate(number, flightNum, type, emplyeeNum)
* Desk(number, flightNum, type, emplyeeNum)
* Flights(number, company, type, time, startPoint, endPoint, terminalName)
* Terminal(name)
* Employee(number, name, position, terminalName)
* Passengers(email, name, type)
* Tickets(uniqueNum, employeeNum, email, flightNum)

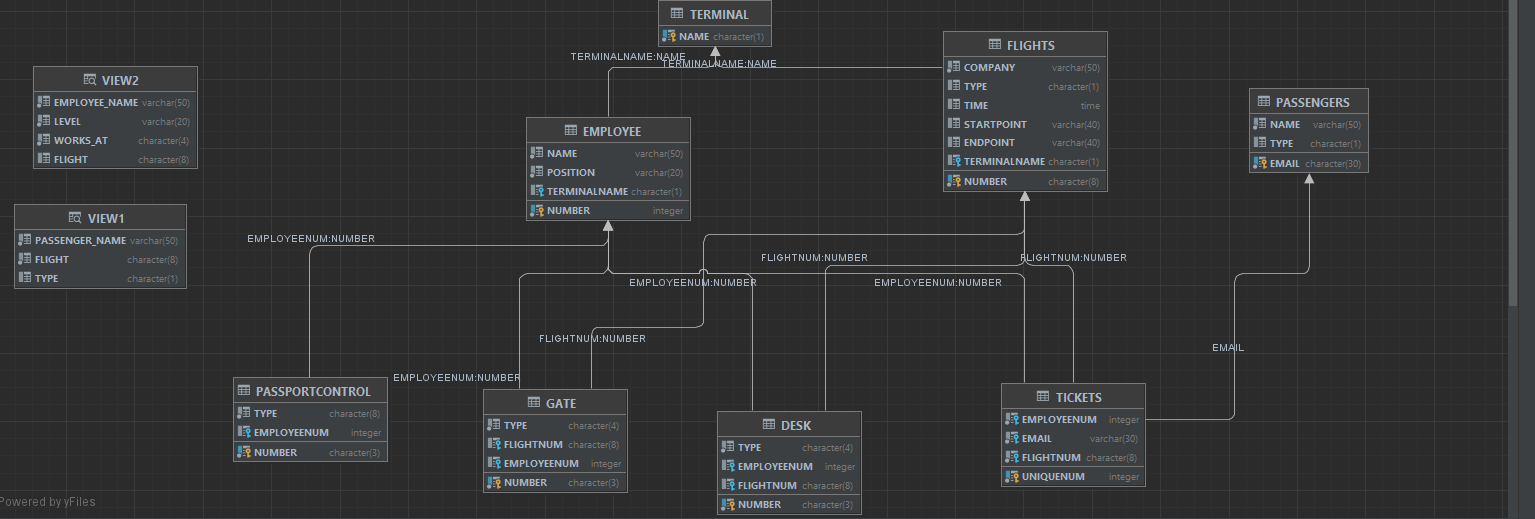
# ФЗ и нормализация

* ФЗ-1: number-> emplyeeNum, type (PassportControl)
* ФЗ-2: number-> fightNum, type, emplyeeNum (Gate)
* ФЗ-3: number-> flightNum, type, emplyeeNum (Desk)
* ФЗ-4: number-> company, type, time, startPoint, endPoint, terminalName (Flights)
* ФЗ-5: number-> name, position, terminalName (Emlpoyee)
* ФЗ-6: email-> name, type (Passengers)
* ФЗ-7: uniqueNum-> employeeNum, email, flightNum (Tickets)

За Terminal няма нетривиалнa ФЗ.

За всички ФЗ на релациите е в сила, че в лявата част се намира суперключ за релацията, следователно всички релации са в НФБК. Не се налага да правим нищо допълнително.

# Схема на базата от данни



# Приложение

# 