Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

Волгоградский государственный технический университет

Факультет «Электроники и вычислительной техники»

Кафедра «Программное обеспечение автоматизированных систем»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

по дисциплине «Проектирование и разработка современных баз данных»

на тему «Проектирование и разработка реляционной базы данных и

приложения пользователя (структура базы данных определяется

индивидуальным заданием). Индивидуальное задание: информационная

система аэропорта»

Выполнил

студент группы ПРИН-1Н

Пахомов Д.А.

Руководитель работы

Аникин А. В.

Волгоград, 2014

Содержание

[1 Описание на естественном языке 3](#_Toc408936719)

[2 Концептуальная ER-модель предметной области 6](#_Toc408936720)

[3 Определите набор задач, которые должна будет решать создаваемая система 6](#_Toc408936721)

[3.1 Транзакционные (задачи учёта) 6](#_Toc408936722)

[3.2 Справочные (оперативные запросы) 7](#_Toc408936723)

[3.3 Справочные расчётные (аналитические запросы) 7](#_Toc408936724)

[3.4 Плановые (расчётные задачи) 7](#_Toc408936725)

[4 Логическую реляционную модель БД 8](#_Toc408936726)

[5 Физическая модель БД 9](#_Toc408936727)

[6 Примеры процедур 9](#_Toc408936728)

[6.1 Показать параметры рейса (сводобные, забронированные места). 9](#_Toc408936729)

[6.2 Ввести данные об осмотре самолета. 10](#_Toc408936730)

[7 Триггеры 10](#_Toc408936731)

[7.1 Предотсвращение установки полета с самолетом, который на ремонте 10](#_Toc408936732)

[7.2 Удаление самолета с рейсов при постановке его на ремонт. 11](#_Toc408936733)

[8 Представления 12](#_Toc408936734)

[8.1 Отделы, их главы, и входящие в них бригады 12](#_Toc408936735)

[8.2 Бригады со списком членов 12](#_Toc408936736)

[8.3 Список вылетов, с пассажирами, местами и количеством багажа 12](#_Toc408936737)

[8.4 Расписание полетов на месяц 13](#_Toc408936738)

[9 Описание разработки и тестирования приложения 13](#_Toc408936739)

[9.1 Описание реализации пользовательского приложения 13](#_Toc408936740)

[9.2 Тестирование 14](#_Toc408936741)

[9.2 Скриншоты 14](#_Toc408936742)

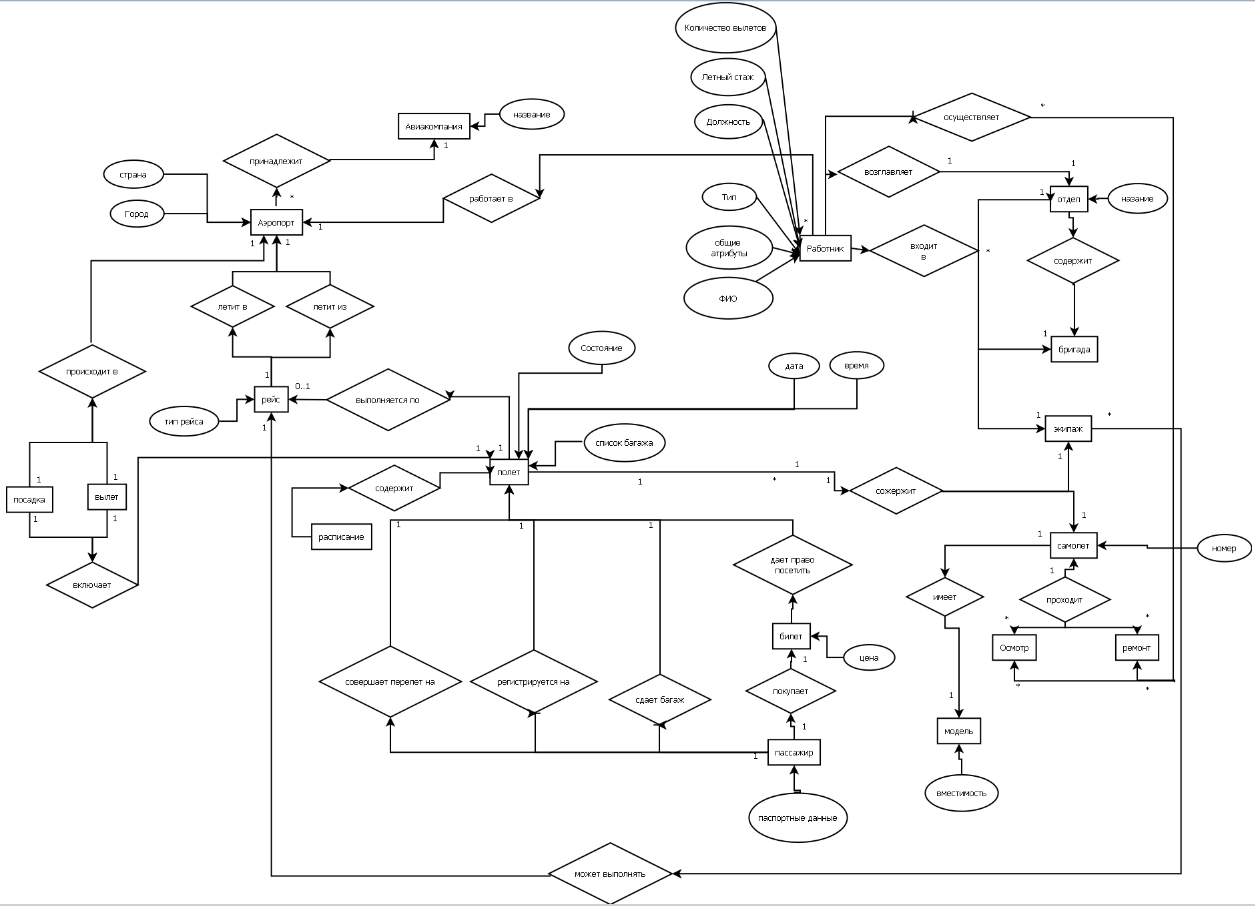
[10 Заключение 15](#_Toc408936743)

# 1 Описание на естественном языке

Информационная система аэропорта.

1. Работники делятся на группы:
   1. администратор;
   2. диспетчер;
   3. техник;
   4. кассир;
2. Каждая из категорий имеет уникальные атрибуты.
3. Каждая категория относится к отделу.
4. В отделах присутствует разбиение на бригады.
5. У отдела есть начальник.
6. Администрация аэропорта состоит из начальников отделовв.
7. Администрации имеет функции:
   1. планирование рейсов;
   2. составление расписаний;
   3. формирование кадрового состава.
8. За самолетом закрепляется бригада из:
   1. пилотов;
   2. техников;
   3. обслуживавшего персонала.
9. Пилоты обязаны проходить медосмотр (раз в год).
10. Пилоты, не прошедшие медосмотр должны быть переведены на другую работу.
11. Перед рейсом самолет осматривается и ремонтируется (если надо).
12. Перед рейсом происходит подготовка, состоящая из:
    1. техосмотр;
    2. заправка;
    3. обслуживание (уборка и пр.).
13. Существует расписание, в котором указано:
    1. тип самолета;
    2. рейс;
    3. дни вылета;
    4. время вылета;
    5. время прилета;
    6. начальный пункт;
    7. конечный пункт;
    8. пункты пересадки;
    9. стоимость.
14. Билеты можно приобрести заранее.
15. Билеты могут распространять другие компании.
16. Билет можно вернуть.
17. Цена билета зависит от маршрута и времени.
18. Рейсы могут быть задержаны или отменены с указанием причины.
19. Категории авиарейсов:
    1. внутренние;
    2. международные;
    3. чартерные;
    4. грузоперевозки;
    5. специальные рейсы.
20. При посадке пассажир характеризуется паспортными данными, билетом и загранпаспортом.
21. Для самолета существует список вещей, отданных в багажное отделение.
22. Каждый аэропорт обслуживает рейсы разных авиакомпаний и имеет международный код и название.
23. Авиакомпания характеризуется названием.
24. У каждой авиакомпании есть несколько рейсов, проходящих через этот аэропорт.
25. Каждому рейсу соответствует один самолёт («воздушное судно»), каждому самолёту несколько рейсов.
26. Самолёт характеризуется номером, маркой, моделью, вместимостью.
27. Каждому рейсу соответствует несколько экипажей, выполняющих их в разное время согласно расписанию.
28. Рейсы могут быть терминальные — завершающиеся или начинающиеся в аэропортуи транзитные, которые используют аэропорт только для временной посадки для заправки и отдыха.
29. Рейс характеризуется номером, типом (терминальные/транизитные), аэропортом
30. отправления, аэропортом назначения, временем отправления, временем прибытия, временем в пути, дальностью, периодичностью (по нечётным дням, по чётным, по выходным, каждый N понедельник/вторник/.../воскресенье).
31. Экипаж состоит из нескольких сотрудников авиакомпании.
32. Каждый член экипажа имеет ФИО, должность (командир, пилот, стюардесса) и лётный стаж, исчисляющийся в количестве вылетов.
33. В аэропорту происходит 2 типа событий — вылет и посадка.
34. Каждое событие — вылет или посадка — характеризуется состоянием — «ожидается» / «состоялось» / «отменён», датой, временем задержки/опережения.

# 2 Концептуальная ER-модель предметной области



# Рис. 1 ЕR модель

# 3 Определите набор задач, которые должна будет решать создаваемая система

## 3.1 Транзакционные (задачи учёта)

1. Отметить посадку самолёта (и актуализировать стаж экипажа).
2. Отметить вылет самолёта.
3. Добавить новый рейс.
4. Перенести полет.
5. Поменять самолёт рейса.
6. Перевести экипаж с рейса на рейс.
7. Снять экипаж с рейса.
8. Назначить экипаж на рейс.
9. Купить билет.
10. Вернуть билет.
11. Поставить самолет в ремонт.
12. Ввести данные об осмотре самолета.

## 3.2 Справочные (оперативные запросы)

1. Показать список сотрудников.
2. Показать список отделов, содержащихся в них бригад и начальников.
3. Показать список экипажей.
4. Показать перечень пилотов, прошедших(непрошедших) медосмотр.
5. Показать список самолетов.
6. Показать запаздывающие рейсы на сегодня.
7. Показать параметры рейса (сводобные, забронированные места).
8. Показать данные об осмотре самолета.

## 3.3 Справочные расчётные (аналитические запросы)

1. Показать статистику по рейсу (колво пассажиров, среднее колво пассажиров).
2. Показать статистику полётов по авиакомпаниям - среднее число полётов в день

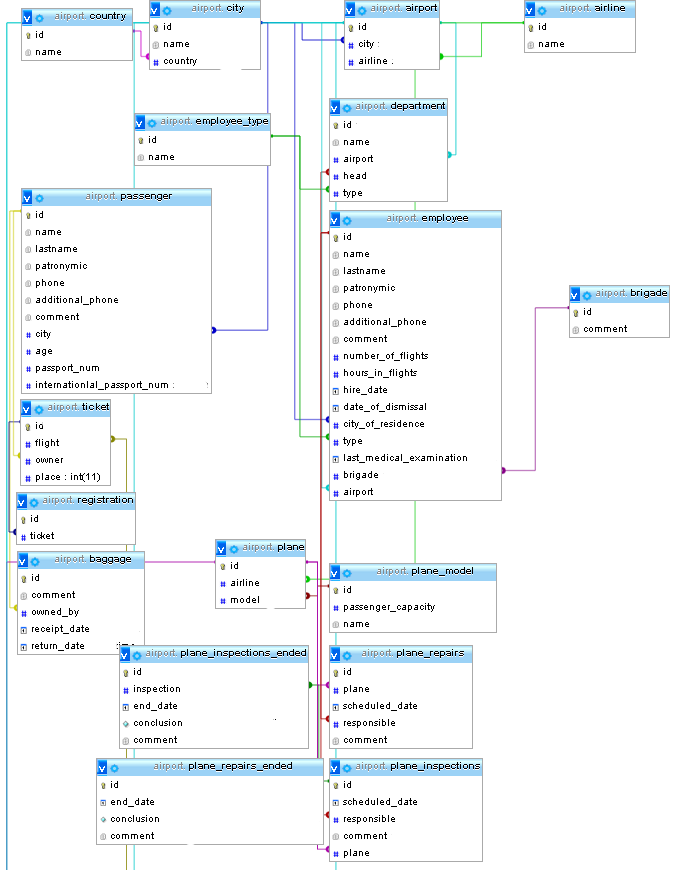
## 3.4 Плановые (расчётные задачи)

1. Составить расписание полётов на месяц.

## 3.5 Доступность задач пользователям

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Администратор | Диспетчер | Техник | Кассир |
| Отметить посадку самолёта |  | + |  |  |
| Отметить вылет самолёта |  | + |  |  |
| Добавить новый рейс |  | + |  |  |
| Перенести полет. |  | + |  |  |
| Поменять самолёт рейса. |  | + |  |  |
| Перевести экипаж с рейса на рейс. |  | + |  |  |
| Снять экипаж с рейса. |  | + |  |  |
| Назначить экипаж на рейс. |  | + |  |  |
| Купить билет. |  |  |  | + |
| Вернуть билет. |  |  |  | + |
| Поставить самолет в ремонт. |  |  | + |  |
| Ввести данные об осмотре самолета. |  |  | + |  |
| Показать список сотрудников. | + |  |  |  |
| Показать список отделов, содержащихся в них бригад и начальников. | + |  |  |  |
| Показать список экипажей. | + | + |  |  |
| Показать перечень пилотов, прошедших(непрошедших) медосмотр. | + | + |  |  |
| Показать список самолетов. | + | + |  |  |
| Показать запаздывающие рейсы на сегодня. | + | + | + | + |
| Показать параметры рейса (сводобные, забронированные места). | + |  |  |  |
| Показать данные об осмотре самолета. | + |  | + |  |
| Показать статистику по рейсу (колво пассажиров, среднее колво пассажиров). | + |  |  | + |
| Показать статистику полётов по авиакомпаниям - среднее число полётов в день | + |  |  | + |
| Составить расписание полётов на месяц. | + |  |  | + |

# 4 Логическую реляционную модель БД



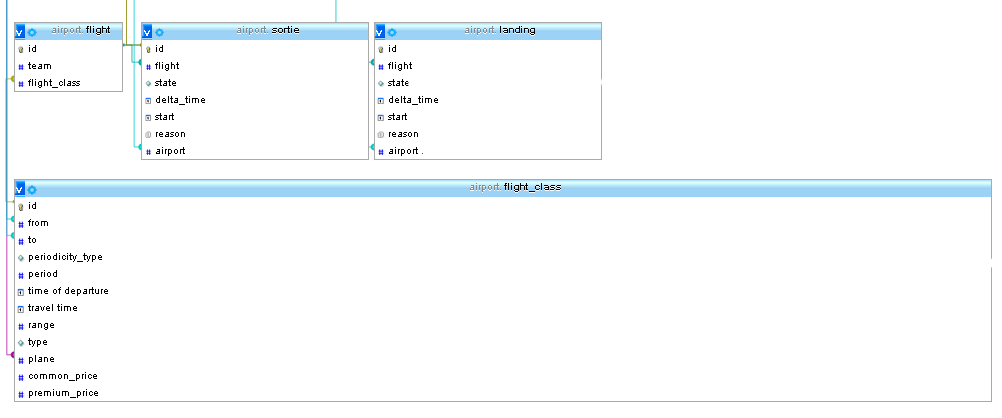
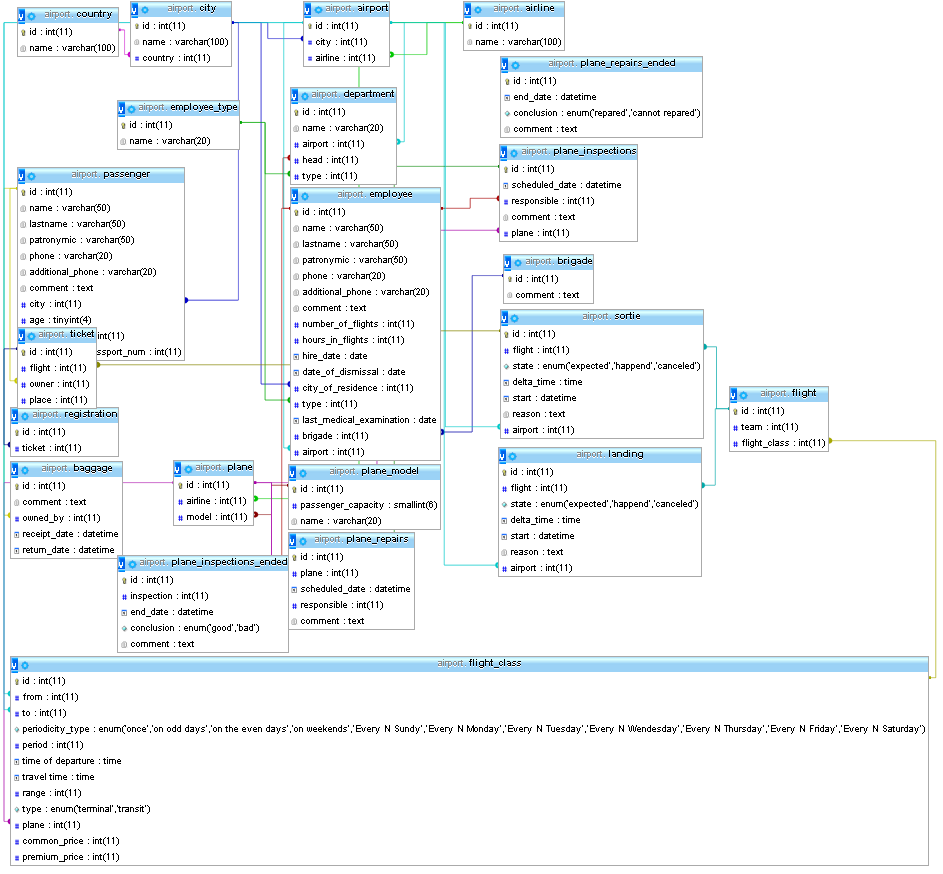


Рис. 2 Логическая модель БД

# 5 Физическая модель БД



# Рис. 3 Физическая модель БД

# 6 Примеры процедур

## 6.1 Показать параметры рейса (сводобные, забронированные места).

CREATE DEFINER=`root`@`localhost`

PROCEDURE `flight\_info`(

IN `flight\_id` INT,

OUT `registered` INT,

OUT `capacity` INT) NOT DETERMINISTIC NO SQL SQL SECURITY DEFINER

Begin

select count(registration.id)

from registration

join ticket

where ticket.flight = flight\_id

into registered;

select plane\_model.passenger\_capacity

from flight

join flight\_class

join plane

join plane\_model

where flight.id = flight\_id

into capacity;

end

## 

## 6.2 Ввести данные об осмотре самолета.

CREATE PROCEDURE `plane\_inspection\_report`(

IN `inspection\_id` INT,

IN `conclusion\_val` ENUM('good', 'bad'),

IN `comment\_val` TEXT) NOT DETERMINISTIC NO SQL SQL SECURITY DEFINER

begin

INSERT INTO `plane\_inspections\_ended`(`end\_date`, `conclusion`, `comment`)

VALUES (

inspection\_id,NOW(),

conclusion\_val,

comment\_val

);

end

# 7 Триггеры

## 7.1 Предотсвращение установки полета с самолетом, который на ремонте

CREATE TRIGGER `flight\_insert\_check\_repair`

BEFORE INSERT

ON `flight`

FOR EACH ROW

begin

declare in\_repair int;

select count(\*)

from plane\_repairs\_ended

join plane\_repairs

inner join flight\_class

on flight\_class.id=NEW.flight\_class

and flight\_class.plane = plane\_repairs.plane

where plane\_repairs\_ended.end\_date > NOW()

or plane\_repairs\_ended.conclusion='cannot repared'

into in\_repair;

if in\_repair > 0 then

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'plane in repair';

end if;

END

## 7.2 Удаление самолета с рейсов при постановке его на ремонт.

CREATE TRIGGER `delete\_flights\_before\_repair`

BEFORE INSERT

ON `plane\_repairs`

FOR EACH ROW

begin

delete flight

from flight join flight\_class

where flight\_class.plane=NEW.plane

end

# 8 Представления

## 8.1 Отделы, их главы, и входящие в них бригады

create view department\_view as

select

department.name as 'departnemt name',

employee.name as 'head name',

employee.lastname as 'head lastname',

GROUP\_CONCAT(distinct ee2.brigade SEPARATOR ',') AS 'brigades'

from department

inner join employee on department.head=employee.id

inner join city on employee.city\_of\_residence = city.id

join employee as ee2 on ee2.type=department.type



## 8.2 Бригады со списком членов

create view brigade\_list as

select employee.brigade, GROUP\_CONCAT(CONCAT(employee.name,' ',employee.lastname,' (',employee\_type.name,')') SEPARATOR ';\n') AS 'brigades'

from employee

inner join employee\_type on employee.type=employee\_type.id

group by employee.brigade



## 8.3 Список вылетов, с пассажирами, местами и количеством багажа

create view sorties\_with\_passengers as

SELECT

sortie.start,

passenger.name,

passenger.lastname,

ticket.place,

count(baggage.id) as 'baggages count'

FROM registration

join ticket

join sortie

join passenger

join baggage on baggage.owned\_by = passenger.id

group by passenger.id

order by sortie.start



## 8.4 Расписание полетов на месяц

create view sortie\_in\_month as

select \* from `sortie` where DATE\_ADD(NOW(), INTERVAL 31 DAY) > `sortie`.`start` and state='expected'



# 9 Описание разработки и тестирования приложения

# 9.1 Описание реализации пользовательского приложения

Разрабатываемое приложение является веб приложением с использованием ASP.MVC и Entity Framework.

Используя инструменты visual studio была сгенерирована модель по базе данных. Которая позволила получать данные и управлять ими из кода C#.

Для валидации данных используется встроенный вариатор ASP.

# 9.2 Описание процесса разработки

Как было отмечено выше, для разработки использовался Entity Framework для связи с базой данных.

Ниже приведена диаграмма классов со связями, для отображения базы данный в ком программы.

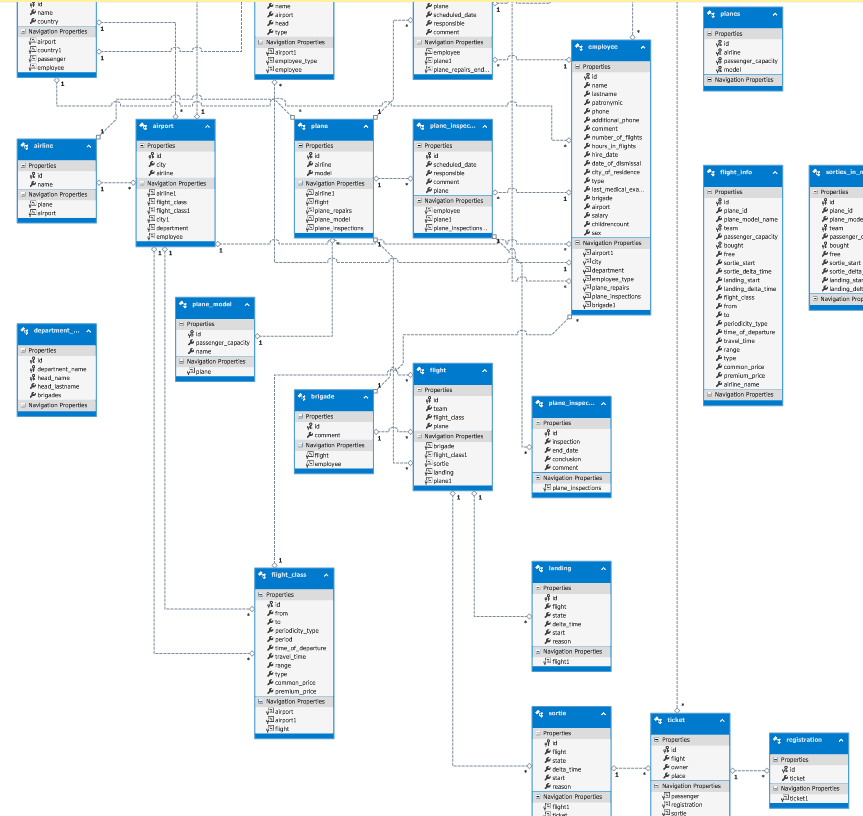


Рис. 4 Диаграмма сущностей Entity Framework (фрагмент)

# 9.3 Тестирование

Тестирование информационной системы происходило в два этапа:

– тестирование созданных запросов к базе данных;

– модульное тестирование отдельных контроллеров приложения;

– функциональное тестирование разработанного приложения в целом

согласно различным сценария для разных групп пользователей.

# 9.4 Скриншоты

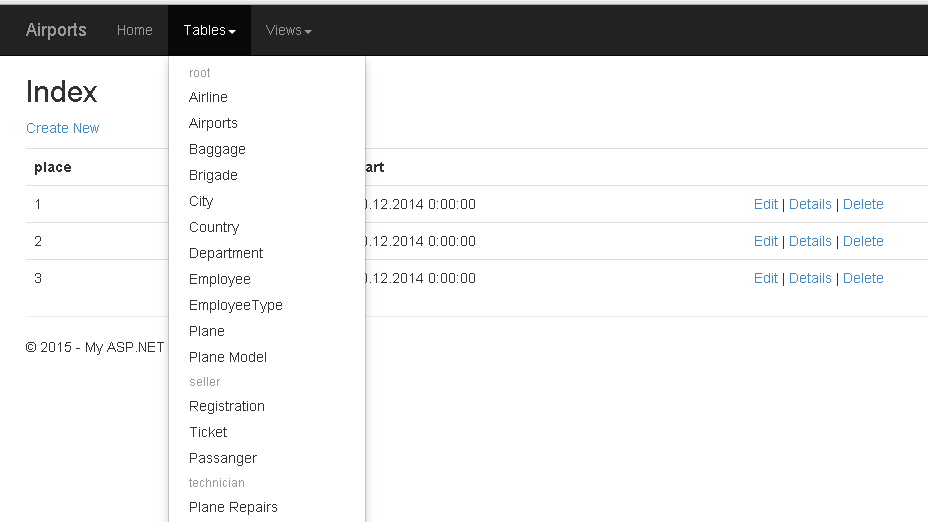


Рис. 5 Вид отображения таблиц

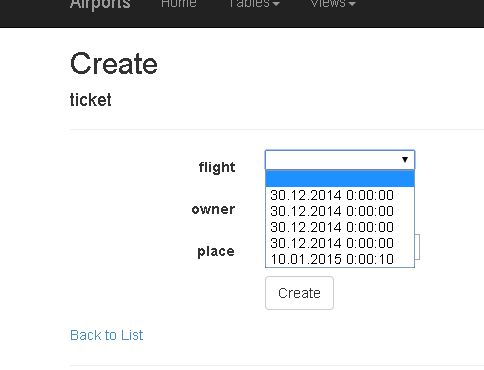


Рис. 6 Вид создания новой записи

# 10 Заключение

В ходе выполнения курсовой работы была спроектирована и создана база данных аэропорта, а также реализована пользовательского приложения для работы с ней.

Приложение позволяет осуществлять базовые операции над базой данных – добавлять, удалять и редактировать записи, а также осуществлять обозначенные запросы и получать соответствующие отчеты