



world skills  
Kazakhstan

# КОНКУРСНОЕ ЗАДАНИЕ

## ПРОГРАММНЫЕ РЕШЕНИЯ ДЛЯ БИЗНЕСА СЕССИЯ 3

Разработано независимой командой по разработке конкурсного задания





## СОДЕРЖАНИЕ

*Этот тестовый проект содержит следующие документы/файлы*

1. DS2018\_TP09\_S3.pdf (Задание к 3 сессии)
2. Session3\_MySQL.sql (SQL Script для создания таблиц и данных в MySQL)
3. Session3\_MSSQL.sql (SQL Script для создания таблиц и данных в MS SQL)
4. Session3\_1C.dt (Выгрузка информационной базы 1С)
5. Testing\_Template.docx (Шаблон тестирования для выполнения)
6. Testing (Рабочая копия системы бронирования)
7. Testing\_Connect\_DB.txt (Инструкция по подключению к БД для просмотра данных, которые используются в рабочей копии системы бронирования)

## Введение

AMONIC Airlines просит разработать систему бронирования билетов, чтобы она могла работать как часть информационной системы, в офисах компании. Чтобы забронировать билет в AMONIC Airlines, пассажир должен либо позвонить в один из офисов компании, либо прийти в один из офисов и предоставить необходимую информацию.

Основные функции сессии, следующие:

- Поиск рейсов по запросу пассажира
- Выпуск билетов с информацией о пассажире
- Тестирование рабочей системы

## Описание проекта и заданий

В ходе разработки проекта убедитесь, что результаты соответствуют базовым требованиям, разработанным различными отделами AMONIC Airlines:

- Руководство по стилю должно быть применено однородно в ходе всей разработки
- Все требуемые модули ПО должны иметь применимые и полезные проверки и сообщения об ошибках, как запрашивает компания
- Где применимо используйте комментарии в коде, для дальнейшего более удобного чтения дальнейшими разработчиками системы
- Используйте соглашение об именовании для всех материалов, которые вам будут предоставляться
- Любая форма или отчет после создания должен отражаться в центре экрана
- Когда форма или диалог активны, операции в других формах должны быть недоступны
- Кнопки Delete и Cancel должны быть красного цвета для избегания случайных нажатий
- При использовании цвета для строк или записей должно быть понятно, что эти цвета означают
- Каркасы форм, приведенные в этом документе, являются только предложением и не должны быть полностью скопированы
- Для любого проекта важно правильно рассчитать время и вовремя подать завершённые рабочие результаты



## **Инструкции для Участника**

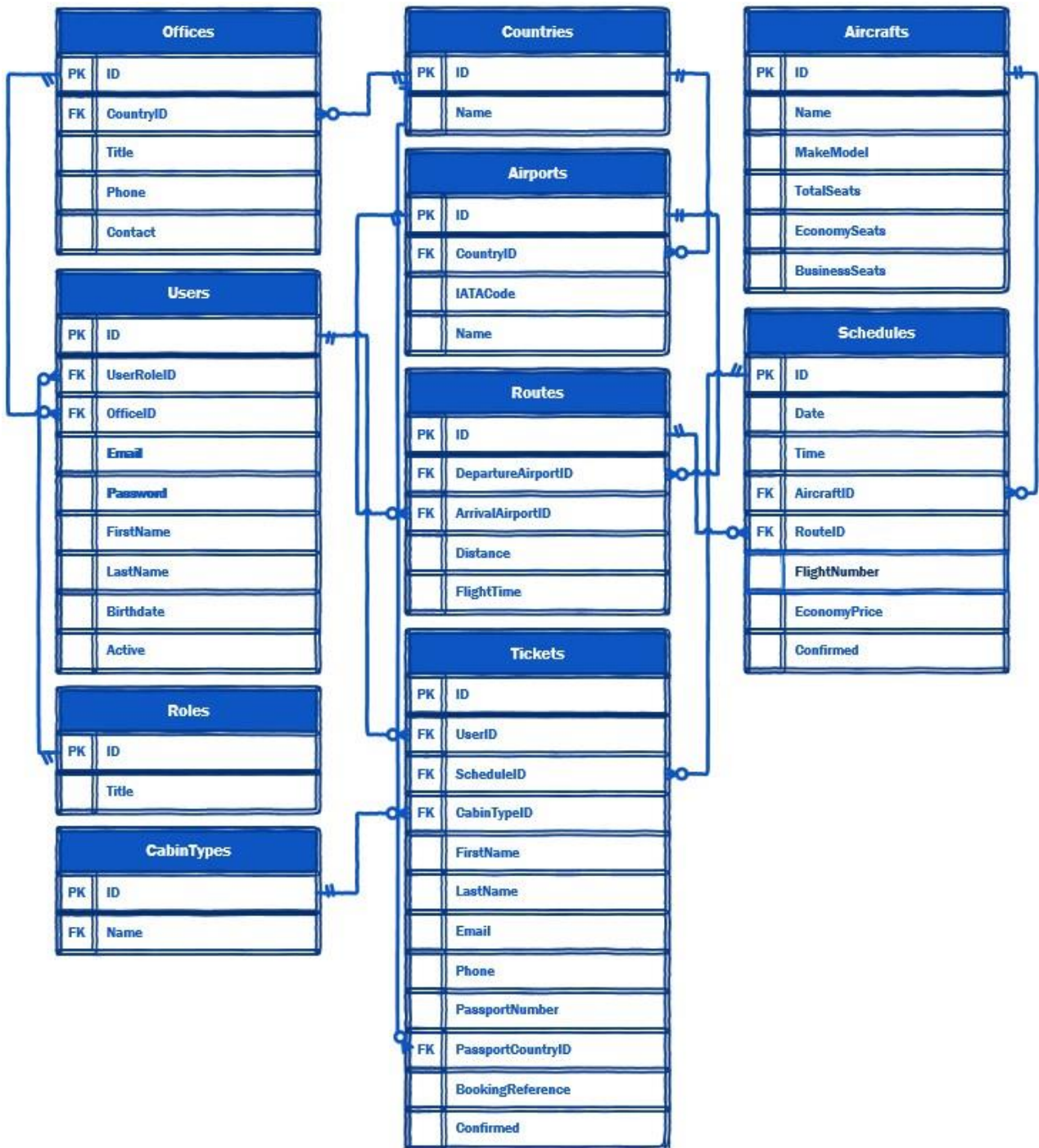
### **3.1 СОЗДАНИЕ БАЗЫ ДАННЫХ**

Создайте базу данных с названием “Session3\_XX” (где XX – номер вашего рабочего места) в предпочитаемой вами платформе (MySQL, Microsoft SQL Server или 1С). Это будет главная и единственная база данных в этой сессии.

### **3.2 ЗАГРУЗКА СТРУКТУРЫ БАЗЫ ДАННЫХ**

В зависимости от предпочитаемой платформы, доступны SQL скрипты и выгрузка информационной базы для 1С. Данные скрипты и выгрузка состоят из структуры базы данных и данных, требуемых для выполнения задания. Данные необходимо загрузить в базу данных, созданную для данной сессии под названием “Session3\_XX”.

Согласно инструкциям дизайнеров, предоставленная структура базы данных для данной сессии не может быть изменена. Речь идет о удалении таблиц, добавлении или удалении любых полей в таблицах или изменении типа данных.



Для лучшего понимания структуры базы данных дизайнеры предоставили ERD. Вышеприведенная диаграмма объясняет концепцию и модель данных, используемых в базе данных.



Search for flights

Search Parameters

From CAI To AUH Cabin Type Economy

Return  One way  Outbound 11/10/2016  Return 15/10/2016

Outbound flight details:  Display three days before and after

From	To	Date	Time	Flight Number(s)	Cabin Price	Number of stops
AUH	CAI	11/10/2016	08:15	[xxxx]	\$405	0

Return flight details:  Display three days before and after

From	To	Date	Time	Flight Number(s)	Cabin Price	Number of stops
AUH	CAI	11/10/2016	08:15	[xxxx]	\$420	0
AUH	CAI	13/10/2016	16:15	[xxxx] - [xxxx]	\$350	1
AUH	CAI	15/10/2016	11:45	[xxxx]	\$420	0
AUH	CAI	16/10/2016	16:15	[xxxx] - [xxxx]	\$350	1

Confirm booking for [xxx] Passengers

### 3.3 ПОИСК РЕЙСОВ

Чтобы забронировать рейс, оператору системы необходимо выяснить есть ли необходимый рейс в расписании.

Чтобы найти нужный рейс, оператор может обозначить несколько или все из следующих параметров:

- Аэропорты отправления и прибытия должны выбираться на форме из выпадающих списков (combo boxes)
- Необходимо выбрать дату отправления. Если пассажир хочет взять обратный билет, можно также указать дату возвращения.
- Пассажир может выбрать либо билет «Туда», либо билет «Туда и обратно». У билета «Туда и обратно» должны быть те же аэропорты вылета и прилета, но в обратном порядке. Обратный рейс может быть забронирован только на даты после даты вылета.
- Тип билета, который влияет на цену, должен быть включен как параметр и выбираться из выпадающего списка. По умолчанию тип билета «эконом»

После того как критерии полета выбраны, список (два списка в случае билета «Туда-Обратно») отображаются со следующими характеристиками:

- Список с вылетами должен быть на экране постоянно, даже если нет результата. Список обратных рейсов будет выведен пользователю, когда пользователь выбирает тип рейса «Туда - Обратно»



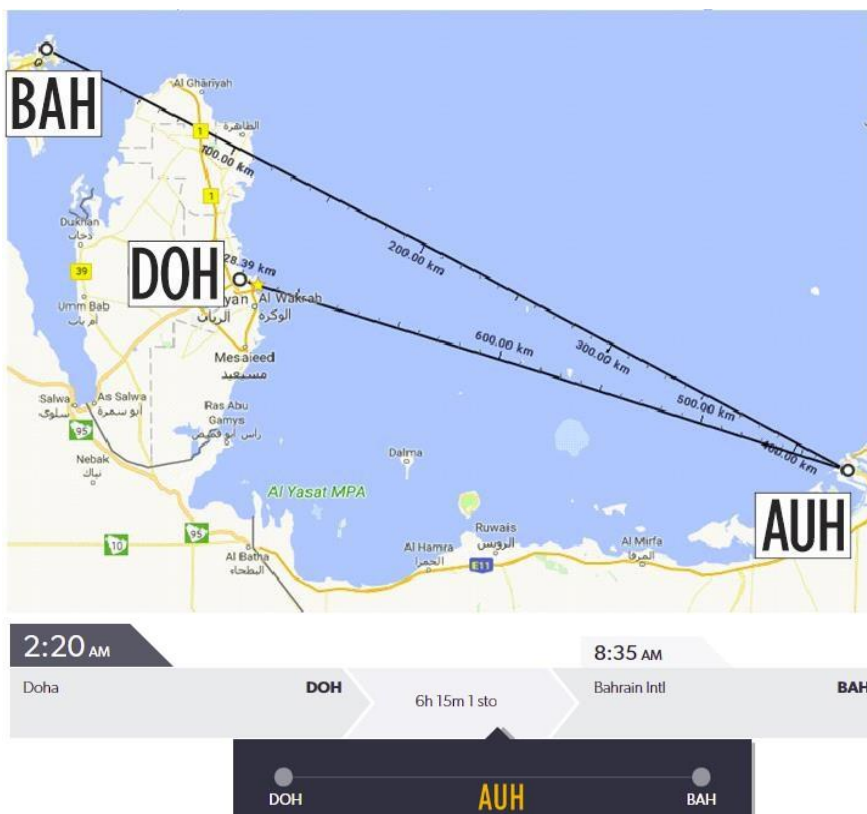


- Используя галочку сверху каждого списка, пользователь может запросить систему показать рейсы за три дня до и три дня после указанной даты. Это нужно для того, чтобы избежать поиска много раз с похожими результатами запроса.
- Элементы списка: аэропорт отправления (From), аэропорт прибытия (To), дата и время отправления, номер рейса или номера рейсов если есть пересадка.
- Цена билета в бизнес классе на 35% дороже билетов эконом класса, а билеты первого класса на 30 % дороже билетов бизнес класса, если не получаются целые значения необходимо округлить до ближайшего целого числа в меньшую сторону.

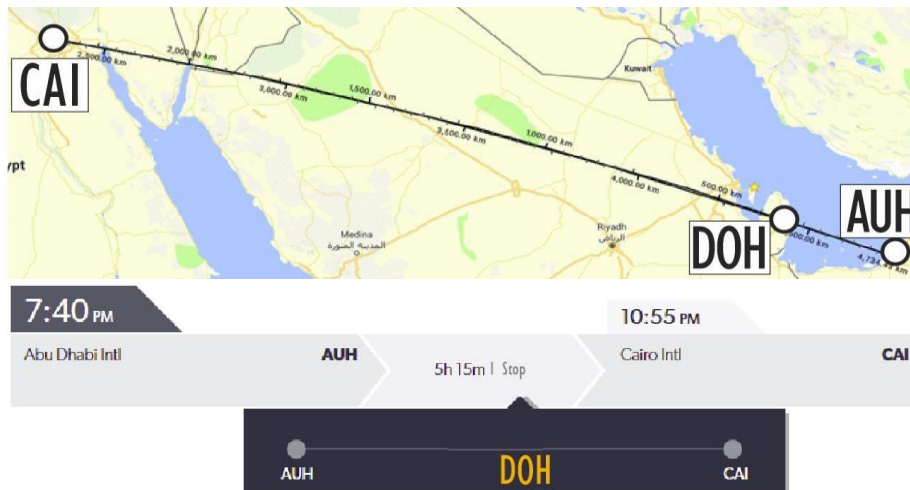
После выбора рейса или рейсов пользователь указывает количество пассажиров, которые бронируют одно и то же путешествие и подтверждает бронирование. Система проверяет, есть ли достаточно мест на выбранных рейсах для пассажиров. Если есть, то можно перейти к форме подтверждения бронирования.

Учитывайте следующие моменты для лучшего понимания работы системы бронирования рейсов:

- У AMONIC Airlines ограниченное число рейсов в регионе. Иногда пассажиры хотят забронировать рейс, для которого не существует прямого рейса (без пересадки)



- Обратите внимание на картинку сверху: у нас нет прямого рейса из Doha в Bahrain, мы предлагаем пассажиру маршрут DOH-AUH-BAH с пересадкой в Abu Dhabi (наш основной аэропорт). Ваша система должна иметь возможность найти и предложить бронирование из всех доступных маршрутов.



- Авиакомпании по всему миру в зависимости от популярности маршрута могут указывать пересадки в рейсах. Это помогает делать рейсы более экономически выгодными. Они называются непрямые рейсы.

Как показано выше маршрут AUH-CAI сам по себе не вызывает много интереса для большого количества рейсов, но если мы указываем пересадку в Doha мы можем предлагать рейсы по как по маршруту AUH-DOH, так и по маршруту DOH-CAI. Наша система бронирования в этих ситуациях предложит маршрут AUH-CAI, но для этого нужно будет выпустить два билета AUH-DOH и DOH-CAI. У нас все равно есть прямые рейсы по этому маршруту в некоторые дни недели.



Booking confirmation X

Outbound flight details

From: CAI To: AUH Cabin Type: Economy Date: 11/10/2017 Flight number: 1908

Return flight details

From: AUH To: CAI Cabin Type: Economy Date: 11/15/2017 Flight number: 1907

Passenger details

Firstname  Lastname  Birthdate

Passport number  Passport country  Phone

Passengers list

Firstname	Lastname	Birthdate	Passport number	Passport Country	Phone
Aleik	Hafezi	11-05-1994	3695005740	Bahrain	+97255369505

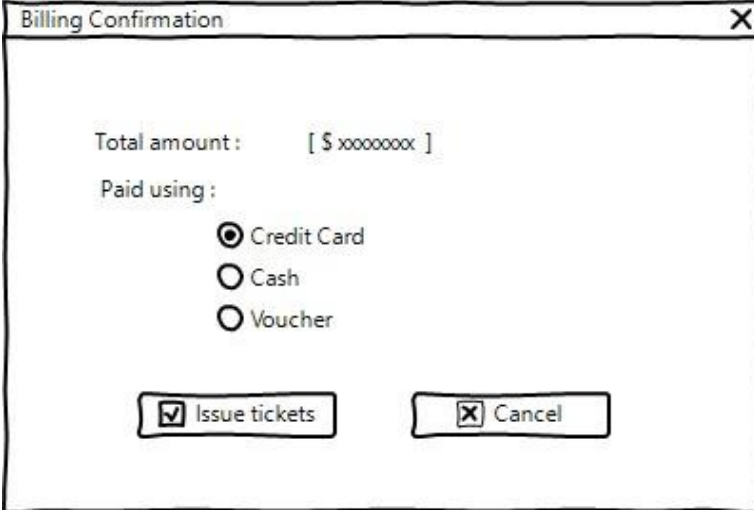
### 3.4 ПОДТВЕРЖДЕНИЕ БРОНИРОВАНИЯ

Как только рейсы для бронирования были выбраны, появляется форма подтверждения бронирования, в которую оператор может ввести данные пассажира.

Эта форма должна включать следующее:

- Информация о рейсе или рейсах для рейса «Туда» и рейса «Обратно».
  - Если никакой обратный рейс не выбран, не нужно отображать информацию по обратному рейсу.
  - Если по маршруту «Туда» или «Обратно» несколько рейсов (с пересадками), эта информация должна быть отражена.
- Информация о пассажире для бронирования:
  - Поля для обязательного заполнения: Имя, Фамилия, дата рождения, Номер паспорта, Гражданство и телефонный номер. ○ Гражданство должно выбираться из выпадающего списка, который берется из базы данных
  - Внесение всей информации о пассажире необходимо для регистрации бронирования.
- Нажимая на кнопку «Добавить пассажира», информация добавляется в список. Пользователь затем не может редактировать записи, но может удалять их с помощью кнопки «Remove».
- Когда вся информация введена, нажимается кнопка «Подтвердить бронирование»





Billing Confirmation

Total amount: [ \$ xxxxxxxx ]

Paid using:

Credit Card

Cash

Voucher

Issue tickets

Cancel

### 3.5 ПОДТВЕРЖДЕНИЕ ОПЛАТЫ

В конце процесса пользователь (менеджер) должен подтвердить получение оплаты. Информация на этой форме должна быть следующей:

- Подсчитать общую сумму к оплате, которая является общей ценой всех билетов, которые будут выпущены по этому бронированию.
- Возможность выбора способа оплаты: кредитная карта, наличные или ваучер. Эта информация нигде не хранится на этой стадии.
- Кнопка подтверждения, которая обозначена на картинке сверху, как «Issue ticket» выпускает билеты и сохраняет их в базе данных:
- Для каждого пассажира каждого рейса, соответствующая запись должна сохраняться в базе данных.
  - Для идентификации бронирования, которое состоит из одного или нескольких рейсов для каждого пассажира, в индустрии активно используется номер бронирования. Номер бронирования — это строка, состоящая из шести знаков из букв и цифр. Эта строка из знаков должна быть уникальной и не может повторяться для бронирований.
  - Выпуская несколько билетов для одного бронирования, генерируется номер бронирования и указывается для всех билетов.
- Кнопка отмены закрывает эту форму и возвращает к подтверждению бронирования.

### 3.6 РЕШЕНИЕ ДЛЯ ТЕСТИРОВАНИЯ ПО

Служба поддержки авиакомпании получила жалобы и отчеты об ошибках в работе системы бронирования из различных офисов. Список был собран в шаблон тестирования (Testing\_Template.docx).

Читая различные сценарии как разработчик ПО вам необходимо будет запустить их и записать результаты проверки ошибок в том формате, который запрашивается.