Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

Высшего профессионального образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной

инженерии

**Первый этап курсовой работы**

по дисциплине «Распределенные программные системы»

«Описание предметной области, функциональная декомпозиция системы. Модель данных»

**Выполнил:**

Ст. гр. ИСТ-113

Жерихов Н.С.

**Принял:**

Тимофеев А. А.

Владимир 2016

Основная информация

Объектом данной курсовой работы является туристическое агентство. Туристическое агентство – это турфирма которая занимаются розничной продажей туров от Тур Операторов. Главная задача Туристического Агентства - это работа с частным лицами и корпоративными клиентами для подбор места проведения туров, разработка маршрутов прибытия и отбытия, организация развлекательных программ, экскурсий и т.д.

Данный курсовой проект предназначен для упрощения контроля расходов и доходов турфирм.

Для полноценного пользования системой человеку необходимо авторизоваться. Затем можно просматривать списки возможных туров. Возможен просмотр рейтинга тура, который зависит от его популярности заказов.

Функционал

1. Работа турами;
2. Работа с доходами от туров;
3. Работа с пользователями.

Данная программная система облегчает процесс поиска тура для отдыха. Обычный пользователь системы – посетитель сайта может просмотреть список туров.

Для работников в программной системе предусматривается удобство создания новых туров, создание новых местоположений и отелей в системе.

Привилегированным пользователям можно вносить данные в журнал отчетов, таким образом, позволяя администратору турагентства вести учет доходов.

В данном проекте будет использоваться второе хранилище данных, которое будет содержать информацию о доходах тура, а так же данные пользователей системы. Данная специфика необходима для обеспечения безопасности данных, связанных с частью системы, отвечающей за финансовую составляющую турагентства и за безопасность персональных данных пользователей.

Диаграмма прецедентов

Диаграммой прецедентов называется диаграмма, на которой показана совокупность прецедентов и актеров, а также отношения между ними. Диаграмма прецедентов представлена на рисунке 1.



Рисунок 1. Диаграмма прецедентов

**Расширенное описание прецедента «Добавить сотрудника»**

* Название: «Добавить сотрудника»
* Предусловие: Пользователь авторизован как администратор и находится на странице с пользователями
* Действующее лицо: Администратор
* Основной поток: Добавление сотрудника. Администратор может добавить нового сотрудника, указать туры , на которых он работает и логин с паролем, под которым пользователь сможет войти в систему
* Альтернативный поток: Администратор не может создавать двух пользователем с одинаковым логином. В этом случае система не создает аккаунт, а предлагает администратору ввести другой логин
* Постусловие: Система добавляет в список указанного пользователя с соответствующими данными. Пользователь может войти в систему под данной комбинацией логина и пароля.

**Расширенное описание прецедента «Составить отчет о доходах»**

* Название: «**Составить отчет**»
* Предусловие: Пользователь авторизован в системе как сотрудник турагентства и находится на странице с турами
* Действующее лицо: Сотрудник
* Основной поток: Составление отчета. Сотрудник нажимает кнопку составить отчет, после чего он попадает на форму создания отчета. Система предлагает сотруднику ввести данные о данном туре. После чего сотрудник нажимает кнопку «Готово» и система заносит данные в базу данных.
* Альтернативный поток: Сотрудник не может составлять отчеты о туре, к которому он не принадлежит. В случае попытки составить отчет не о своем маршруте, сотрудник будет оповещен системой о невозможности внести данные по причине недостатка прав.
* Постусловие: Администратор видит обновленную информацию на странице доходов о доходах маршрутов.

Диаграмма классов

Класс Местоположение (Place) определяет местоположение для тура:

Номер местоположения (idPlace int);

Страна (country string) – название страны;

Город (city string) – название страны;

Название отеля (Hotel string) – название отеля;

Звезды (stars int) – количество звезд у отеля;

Класс Тур (Tour) представляет собой список туров.

Номер тура (idTour int);

Название тура (nameTour string);

Старт тура (startDate date) – дата вылета (начало тура);

Окончание тура (endDate date) – дата возвращения;

Номер местоположения (idPlace int) – ссылка на местоположение;

Количество человек (colPerson int) – количество человек для данного тура.

Цена (price int) – цена на выбранный тур.

Класс Контракт (Contract) определяет состояние контракта клиента с турагентством:

Номер контракта (idPlace int);

Номер пользователя (idUser int);

Номер тура (idTour int);

Состояние (state string) – состояние контракта(одобрен, отклонен, ожидание);

Класс Отчет (Report) – форма отчета о доходе тура.

Номер отчета (idReport int);

Номер тура (idTour int) какому туру принадлежит отчет;

Количество заказов (numberOrder int) – сколько заказов было сделано;

Дата отчета (dateTime date) – время создания отчета;

Прибыль (summa int) – сколько денег заработал тур;

Класс Пользователь (userы) определяет основные атрибуты, которыми обладает пользователь системы.

Номер пользователя (idUser int);

Логин (username string);

Пароль (password string);

Тур пользователя (tour int) определяет, к каким турам относится сотрудник турагентства. Определяет его права на создание отчетов.

Класс роли пользователей (roles) определяет соответствие пользователей с их ролями в системе.

Логин (username string);

Роль (role string);

Диаграмма классов концептуального уровня описывает модель предметной области, в ней присутствуют только классы прикладных объектов. Диаграмма классов первой базы данных представлена на рисунке 2, второй базы данных на рисунке 3.



Рисунок 2. Диаграмма классов первой БД



Рисунок 3. Диаграмма классов второй БД

IDEF диаграммы



Рисунок 4. Функциональный блок



Рисунок 5. Декомпозиция функционального блока