МИНОБРНАУКИ РФ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

«Воронежский государственный университет»

Факультет компьютерных наук

Кафедра информационных систем

Направление подготовки 230400 «Информационные системы и технологии»

**Отчет по производственной практике**

**Тема:** Интернет магазин «Умных часов»

Выполнил студент 4 курса

Зацепин Владимир Вячеславович

Руководитель: доц. Сычев А.В.

Воронеж 2016

Оглавление

[Постановка задачи 2](#_Toc466890308)

[Метод решения задачи 2](#_Toc466890309)

[Средства разработки 3](#_Toc466890310)

[Структура приложения 3](#_Toc466890311)

[Реализация (Зацепин В.В.) 4](#_Toc466890312)

[База данных 4](#_Toc466890313)

[REST API 4](#_Toc466890314)

[Admin page 6](#_Toc466890315)

[Заключение 7](#_Toc466890316)

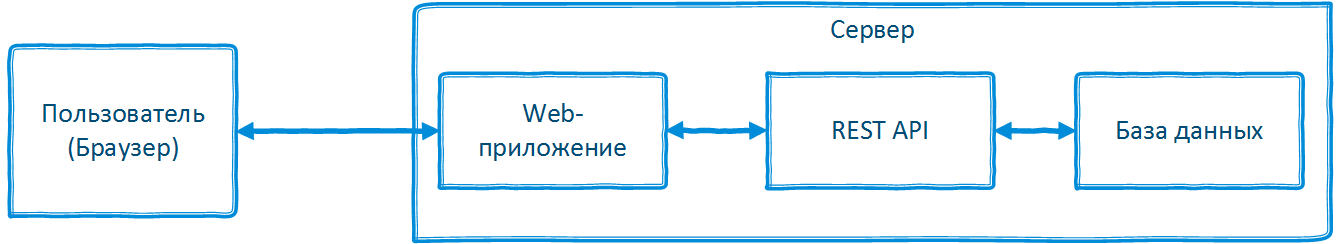
# Постановка задачи

Интернет магазин «Умных» часов должен обеспечить возможность реализации продукции через Интернет. Данное приложение должно выполнять следующие основные требования:

* Предложение ассортимента по производителю
* Предоставление информации по продукту, а именно:
  + Название
  + Цена
  + Описание
  + Характеристики
* Предоставление возможности оставить заявку на приобретение определенного товара
* Запоминание выбранного покупателем товара до оформления заявки
* Отдельная защищенная страничка со всеми последними заказами пользователей

# Метод решения задачи

Для реализации данного приложения была выбрала следующая архитектура:



Клиент, используя браузер, заходит на страничку Web-приложения, которое реализовано как одностраничное Интернет приложение (Single Page Application) при помощи библиотеки AngularJS. Далее, приложение использует REST API (Representational state transfer Application Programming Interface) для связи с базой данных. Такой подход позволяет обеспечить зависимость клиентской и серверной части всего приложения только на уровне API.

# Средства разработки

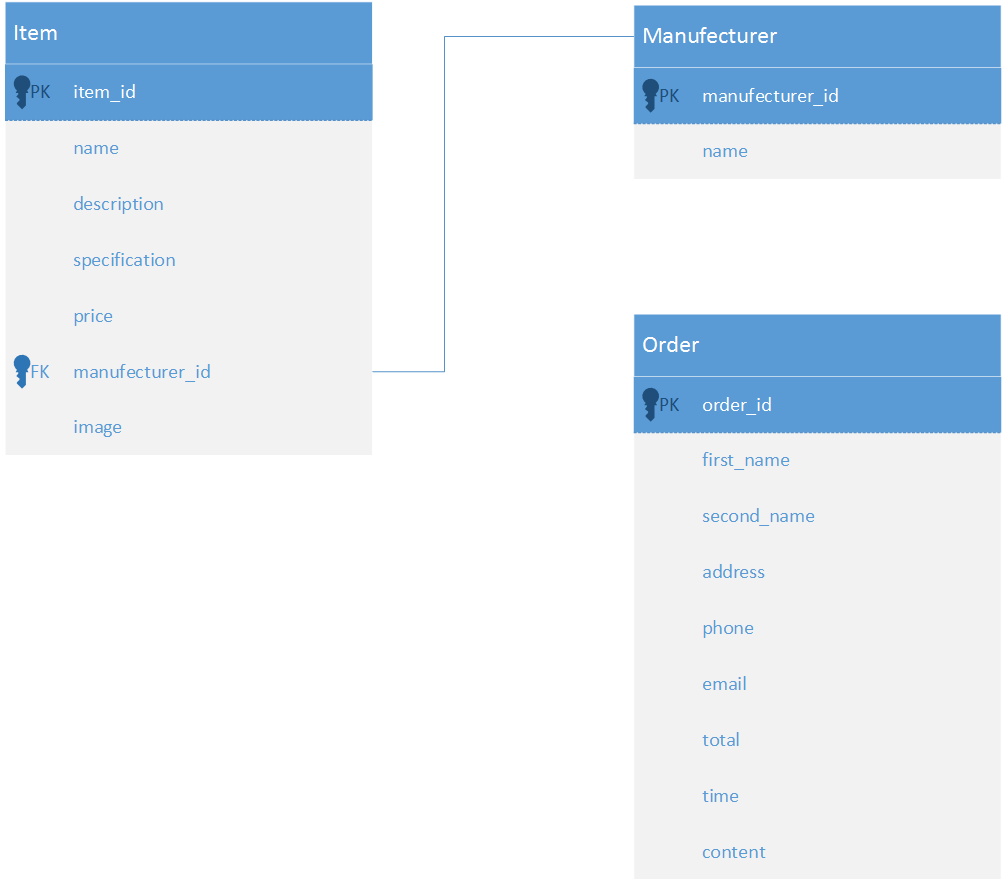
* Intellij IDEA – IDE для написание кода
* MySQL – СУБД
* Endles – локальный Apache сервер
* Microsoft Office Visio – построение диаграмм

# Структура приложения

* api.php - REST API и подключение к базе данных (Зацепин)
* index.html – точка входа для приложения (Гончарова)
  + home.html – главная страничка сайта
    - Приветственное изображение
    - Список лучших предложений
  + catalog.html – страничка-список часов, сгруппированный по производителям
    - Список производителей
    - Список часов относящихся в выбранному производителю
    - Для элементов списка необходимы изображение часов, название, цена
  + item.html – страничка с подробной информацией о конкретных часах
    - Изображение
    - Название
    - Описание
    - Характеристики
    - Цена
    - Возможность добавить в корзину
  + cart.html – страничка оформления заказа
    - Список всех товаров добавленных в корзину
    - Общая стоимость заказа
    - Форма заполнения для оформления заказа (ФИО, телефон, адрес)
* admin.html – страничка со всеми последними заказами

# Реализация (Зацепин В.В.)

## База данных



Поле “content” содержит информацию о заказанных товарах в формате JSON.

## REST API

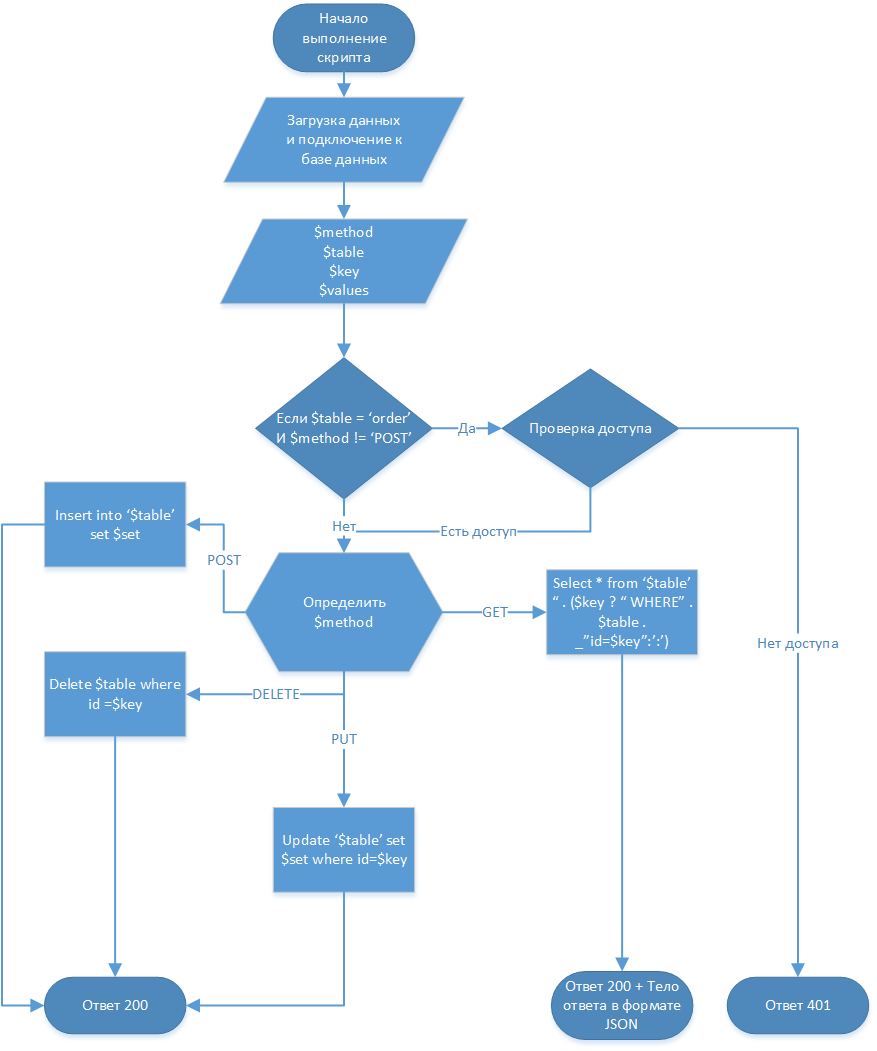
REST API - способ построения клиент-серверной архитектуры. В случае данного приложения это реализовано при помощи одного php скрипта – api.php, который позволяет осуществить запрос к базе данных по следующему принципу:

|  |  |  |
| --- | --- | --- |
| HTTP запрос | HTTP пример запроса | Пример ответа JSON |
| GET <host>/api.php/<table\_name>/<id> | GET localhost/api.php/item/1 | {"item\_id":"1","name":"Apple Watch Nike +"} |
| POST <host>/api.php/<table\_name>  + Тело запроса в JSON | <http://localhost/api/order>  и тело запроса:  {"items":"[{\"item\_id\":\"1}]","total":25000,"first\_name":"Иван","last\_name":"Иванов","phone":"1234567890","address":"г. Воронеж ул. Кольцовская 35","email":"ivan@mail.com","time":"2016-11-14T08:00:45.423Z"} | 200 OK |

Кроме того, для доступа ко всем заказам (таблица order,) реализован алгоритм аутентификации Basic:

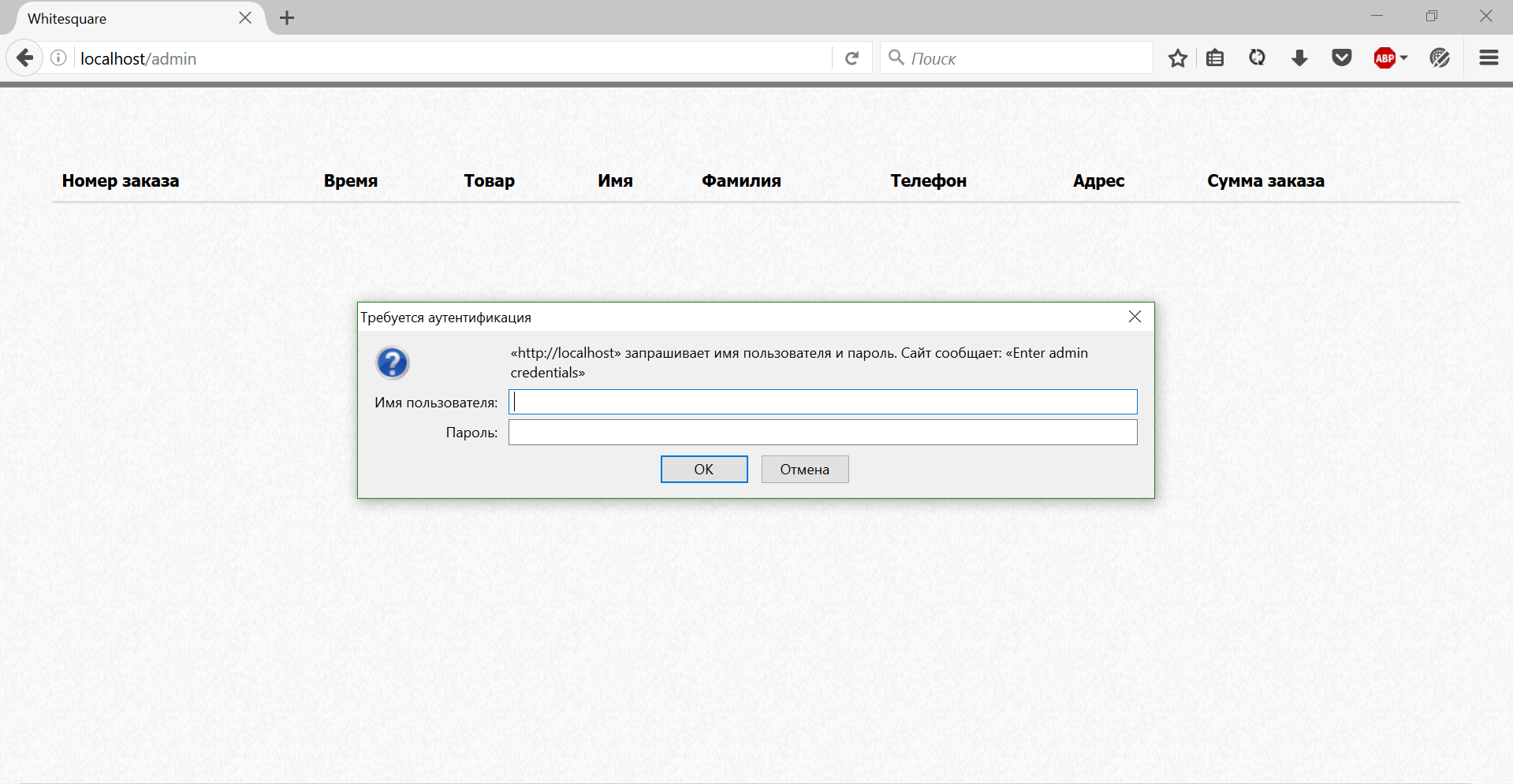
**if** (($table == 'order') && ($method != 'POST')) {  
 **if** (($\_SERVER['PHP\_AUTH\_USER'] != $admin\_name) || ($\_SERVER['PHP\_AUTH\_PW'] != $admin\_password)) {  
 header('WWW-Authenticate: Basic realm="Enter admin credentials"');  
 header('HTTP/1.0 401 Unauthorized');  
 **echo** 'Bad credentials';  
 **exit**;  
 }  
}

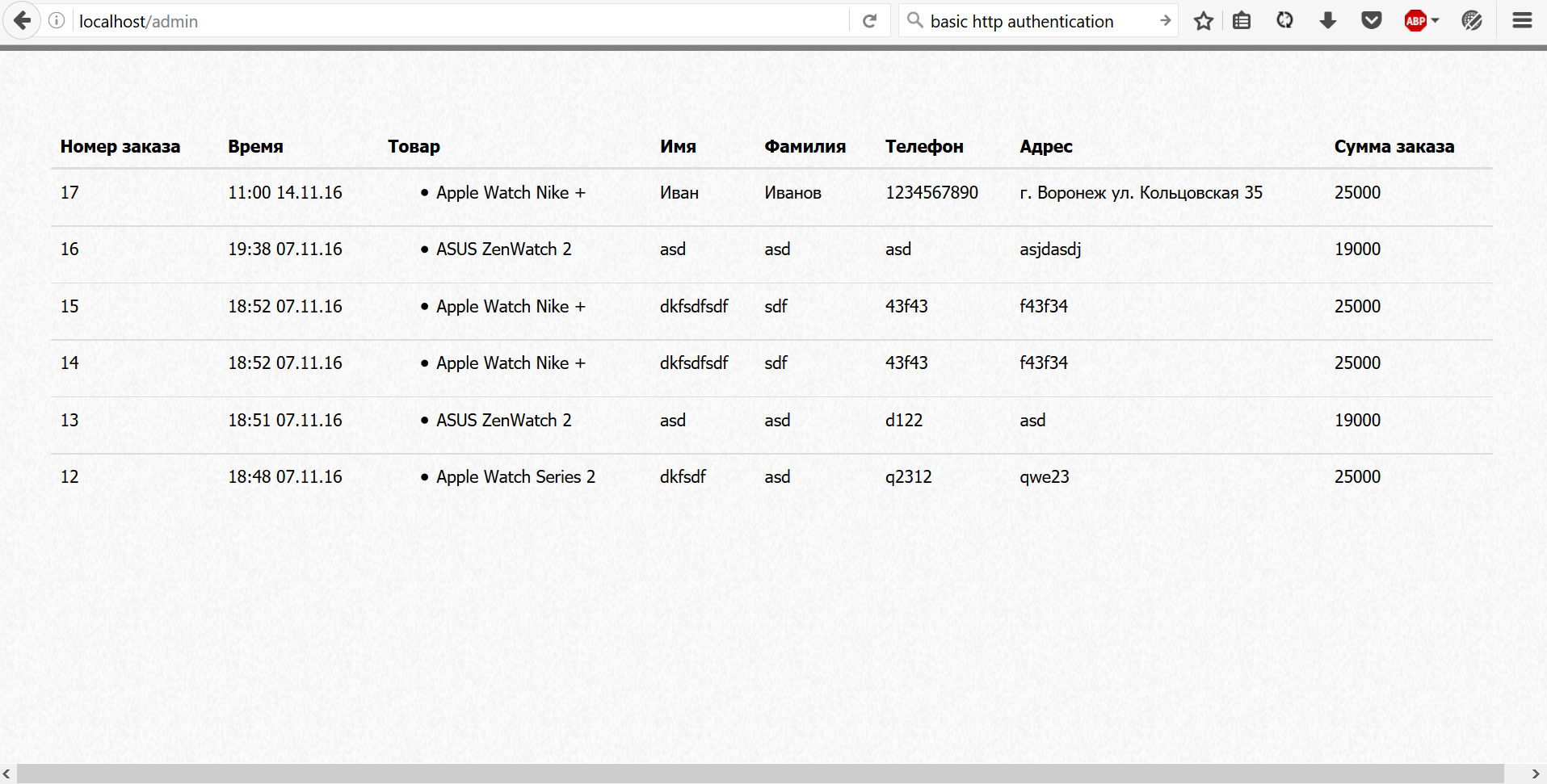
Общая блок-схема алгоритма работы api.php:



## Admin page

Страничка для просмотра всех заказав реализованная при помощи AngularJS:





Заказы отcсортированы по дате (самый последний – самый верхний).

# Заключение

Уровень завершенности проекта следует оценивать, как полностью рабочий, готовый для полноценного использования. Все модули функционируют без ошибок. Обеспечивают полное выполнение всех требуемых от них функций.

Но, следует заметить, что для обеспечения стабильной работы приложения в реальных условиях необходимо обеспечить более качественную безопасность данных.