

Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

Кафедра математической
кибернетики и компьютерных наук

**ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ВЕБ-СЕРВИСА ДЛЯ
АВТОСАЛОНА**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные
технологии
факультета КНиИТ
Байбулова Армана Жулдыгалиевича

Научный руководитель
доцент, к.ф.-м.н.

Ю. Н. Кондратова

Заведующий кафедрой
к.ф.-м.н.

С. В. Миронов

Саратов 2017

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Инструменты разработки	4
2 Функционал веб-сервиса	6
3 Реализация веб-сервиса	7
3.1 БД «Автосалон»	7
3.1.1 Подключение к БД	7
3.1.2 Реализация БД «Автосалон»	7
3.2 Аутентификация и Регистрация новых пользователей	7
3.3 Добавление данных в БД	9
3.4 Редактирование данных в БД	10
ЗАКЛЮЧЕНИЕ	12
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	13

ВВЕДЕНИЕ

90 процентов разработок в Java связана с разработкой именно клиент-серверных приложений, где в качестве сервера используется application server, который выполняет роль сохранения, вычисления, поиск данных и тому подобные сложные и долгие операции, которые затрачивают достаточное количество ресурсов компьютера и времени. А клиент обычно представляется в виде веб-приложения. Это может быть, например, телефон, или настольные приложения, или окно браузера. То есть веб-сервисы позволяют пользователям быстро и удобно воспользоваться множеством услуг почти с любого устройства. Это делает их очень популярными в наше время.

Веб-сервисы широко используются для разных целей: совершения покупок (AliExpress, OZON.ru, Ситилинк и другие), поиск информации (Google, Yandex, Yahoo! и другие), расчет математических формул (WolframAlpha), управление своими картами и счетами (Сбербанк Онлайн, Альфа-Клик и другие) и так далее.

Важным элементом в этой архитектуре является протокол, по которому эти данные передаются. Так как цель работы это разработка веб-сервиса, то используются 2 базовых протокола: http и https. Отличаются они тем, что один открытый, а второй засекреченный.

Несомненным преимуществом веб-сервисов является и то, что их составные части могут быть написаны на разных языках. Например, Windows-C Sharp-клиент может обмениваться данными с Java-сервером, работающим под Linux.

Целью данной работы является проектирование и разработка веб-сервиса для автосалона.

Для достижения цели необходимо решить следующие задачи:

- определить функционал веб-сервиса;
- выбрать инструменты и технологии для разработки веб-сервиса;
- построить логическую и физическую модель базы данных;
- спроектировать и реализовать веб-сервис.

1 Инструменты разработки

Для управления сущностями веб-сервиса будет использоваться JavaScript. JavaScript - это язык управления сценариями просмотра гипертекстовых страниц Web на стороне клиента. Если быть более точным, то JavaScript - это не только язык программирования на стороне клиента. Liveware, прародитель JavaScript, является средством подстановок на стороне сервера Netscape. Однако, наибольшую популярность JavaScript обеспечило программирование на стороне клиента. Одна из главных причин, по которой Web-разработчики приняли JavaScript, - возможность выполнения на стороне клиента многих функций, которые ранее выполнялись исключительно на стороне сервера [1].

Maven был выбран в качестве системы автоматической сборки проекта. Он был выбран по нескольким причинам: во-первых все проекты на Maven имеют похожую структуру, то есть внутри самого инструмента задана жесткая архитектура, по которой должен быть выглядеть проект. Это существенно упрощает процесс разработки и процесс понимания самого приложения. Во-вторых не нужно будет скачивать никакие библиотеки и подключать их вручную. Просто прописываются зависимости, и Maven сам их выкачивает и подключает [2].

Для реализации серверной части используется Apache Tomcat. Apache Tomcat - это контейнер, который позволяет вам использовать интернет приложения такие, как Java сервлеты и JSP (серверные страницы Java). Tomcat используется в качестве самостоятельного веб-сервера, в качестве сервера контента в сочетании с веб-сервером Apache HTTP Server, а также в качестве контейнера сервлетов в серверах приложений JBoss и GlassFish [3].

Для построения логической и физической модели базы данных будет использоваться PostgreSQL. PostgreSQL - это объектно-ориентированная система управления реляционными базами данных. Это означает, что это система для управления данными, которые хранятся в виде отношений. Отношение - это математический термин для таблицы. Понятие хранения данных в таблицах является сегодня таким обычным, что оно может показаться очевидным, однако есть несколько других способов организации баз данных. Файлы и каталоги в Unix-подобных операционных системах являются примером иерархической базы данных. Одно из наиболее современных направлений разработки СУБД - это объектно-ориентированные базы данных [4].

Структурно, наш веб-сервис представляет функциональную бизнес-логику в виде архитектурного паттерна Model-View-Controller (MVC). MVC - это не шаблон проекта, это конструкционный шаблон, который описывает способ построения структуры приложения, сферы ответственности и взаимодействие каждой из частей в данной структуре [5].

В работе, будет под Видом (View) иметься в виду JSP файлы, под Моделью (Model) будет иметься в виду классы Java, а Сервлеты будут Контроллерами (Controller).

2 Функционал веб-сервиса

Основное направление разрабатываемого сервиса - это визуализировать удобно и понятно информацию из БД, например возможность поиска зарегистрированных клиентов по различным параметрам, поиск доступных автомобилей и совершение продажи автомобиля и так далее. Функционал веб-сервиса:

- возможность Аутентификации и Регистрации новых пользователей;
- при регистрации, в зависимости от должности, может быть ограничен функционал пользователя;
- есть возможность выйти из профиля и зайти под другим;
- добавление и редактирование информации о клиентах и автомобилях;
- возможность удаления клиентов и автомобилей из БД;
- при нажатии на кнопку «Совершить продажу с новым клиентом» надо добавить информацию о новом клиенте, а потом добавить автомобиль;
- если клиент уже есть в БД, то можно перейти к его профилю и непосредственно добавить ему автомобиль;
- на главной странице можно осуществлять глобальный поиск по клиентам. А также по определенному полю и применять фильтр на полноту.

3 Реализация веб-сервиса

3.1 БД «Автосалон»

3.1.1 Подключение к БД

Связь с БД обеспечивается в файле *bd.properties*. Здесь прописывается *url* для соединения с БД. Он прописывается так: в зависимости от разных БД значение второго ключа, порт и хост будут меняться. В данном случае сначала прописывается, что используется технология jdbc, порт, хост и название базы. Далее какой класс используется для работы с БД. Username и password это ключи для входа в БД [6].

```
jdbc.url=jdbc:postgresql://127.0.0.1:5432/avtosalon
jdbc.driver_class=org.postgresql.Driver
jdbc.username=postgres
jdbc.password=root
```

3.1.2 Реализация БД «Автосалон»

На рисунке 1 приведена полная ER (сущность – связь) диаграмма разработанной базы данных. SQL код для ее создания приведен в Приложении ??.

3.2 Аутентификация и Регистрация новых пользователей

Аутентификация клиента происходит в сервлете LoginServlet.java:

```
public class LoginServlet extends HttpServlet {

    private DealershipSingleton dealershipSingleton =
        DealershipSingleton.getInstance();

    @Override
    protected void doGet(HttpServletRequest req,
        HttpServletResponse resp) throws ServletException,
        IOException {
        RequestDispatcher dispatcher = req.getRequestDispatcher
            ("view/Login.jsp");
        dispatcher.forward(req, resp);
    }
}
```

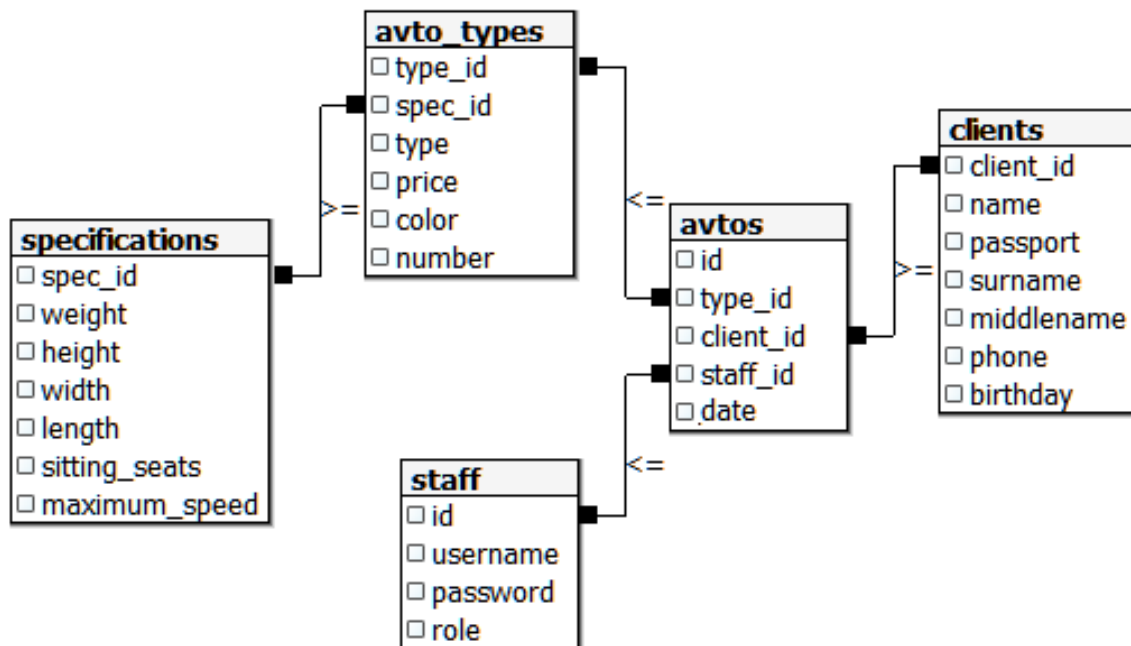


Рисунок 1 – Основная диаграмма.

```

@Override
protected void doPost(HttpServletRequest request,
                      HttpServletResponse response) throws
                      ServletException, IOException {
    boolean isError = false;

    String login = request.getParameter("login");
    String password = request.getParameter("password");

    if (user != null && !isError) {
        HttpSession session = request.getSession();
        session.setAttribute("user", user);
        session.setMaxInactiveInterval(30 * 60);
        Cookie userName = new Cookie("authid",

```



```

        user.getLogin() + ":" + user.getPassword());
        userName.setMaxAge(30 * 60);
        response.addCookie(userName);
        response.sendRedirect("/dealership/view");
    }

```

А процесс Регистрации клиента происходит в сервлете RegistrationServlet.java:

```

public class RegistrationServlet extends HttpServlet {

    private DealershipSingleton dealershipSingleton =
        DealershipSingleton.getInstance();

    @Override
    protected void doGet(HttpServletRequest req,
                          HttpServletResponse resp) throws
        ServletException, IOException {
        RequestDispatcher dispatcher =
            req.getRequestDispatcher("view/Registration.jsp");
        dispatcher.forward(req, resp);
    }

    protected void doPost(HttpServletRequest request,
                          HttpServletResponse response)
        throws ServletException, IOException {
        boolean isError = false;
        String login = request.getParameter("login");
        String role = request.getParameter("role");
        String password = request.getParameter("password");
        String rePassword = request.getParameter("rePassword");
    }

```

3.3 Добавление данных в БД

Добавление клиента в БД происходит в сервлете AddClientServlet.java:

```

public class AddClientServlet extends HttpServlet {
    /** Строковые константы */

```

```

private static final String ADD_CLIENT =
"/view/client/AddClient.jsp";

/**
 * Обработка get-запросов
 * @param req Запрос
 * @param resp Ответ
 */
@Override
protected void doGet(HttpServletRequest req,
HttpServletRequest resp) throws ServletException,
IOException {
    req.setCharacterEncoding("UTF-8");
    forwardTo(req, resp, ADD_CLIENT);
}

/**
 * Обработка post-запросов
 * @param req Запрос
 * @param resp Ответ
 */
@Override
protected void doPost(HttpServletRequest req,
HttpServletRequest resp) throws ServletException,
IOException {
    req.setCharacterEncoding("UTF-8");
    processAddClient(req, resp);
}

```

Аналогично устроен процесс добавления машин в сервлете AddCarServlet.java.

3.4 Редактирование данных в БД

Изменение данных о клиентах происходит в сервлете EditClientServlet.java:

```

private void processUpdateClient(HttpServletRequest req) {

```

```

if (req.getParameter("update") != null) {
    String name = req.getParameter("name");
    String name2 = req.getParameter("name2");
    String name3 = req.getParameter("name3");

    if (!DEALERSHIP.getCurrentClient().getFullName().
equals(name) || !DEALERSHIP.getCurrentClient().
getSurName().equals(name2) || !DEALERSHIP.
getCurrentClient().getMiddleName(). equals(name3))
    {
        DEALERSHIP.renameCurrentClient(name);
        DEALERSHIP.renameCurrentClient(name2);
        DEALERSHIP.renameCurrentClient(name3);
        req.setAttribute("changes", CLIENT_NAME_CHANGED);
    }
}
}

```

Удаление клиентов из БД происходит следующим образом:

```

private void processRemoveClient(HttpServletRequest req,
HttpServletRequestResponse resp) throws ServletException, IOException {
    if (req.getParameter("remove") != null) {
        DEALERSHIP.removeCurrentClient();
        forwardTo(req, resp, DEALERSHIP_VIEW);
    }
}

```

Аналогично устроен процесс редактирования и удаления машин в сервлете EditCarServlet.java.

ЗАКЛЮЧЕНИЕ

В ходе дипломной работы были изучены следующие инструменты для разработки: для выполнения работы сервера - Tomcat, проектирование БД - PostgreSQL, автоматический сборщик проектов Maven .

В результате выполнения работы были отработаны навыки:

- подбор необходимых инструментов для разработки веб-сервиса;
- работа со сторонними библиотеками и фреймворками;
- построение модели базы данных;
- разработка веб-сервиса;
- работа с клиент-серверными приложениями;
- программирование на языке Java.

Был разработан веб-сервис для Автосалона. Данный веб-сервис позволяет сотрудникам автосалона:

1. осуществлять продажу автомобиля с новыми или уже занесенными в БД клиентами;
2. просмотр всех известных в БД клиентов и автомобилей;
3. осуществлять поиск автомобилей и клиентов с использованием разных полей для поиска;
4. редактирование информации о клиентах и имеющихся у них автомобилей;
5. возможность аутентификации и регистрации сотрудников Автосалона, где каждая должность ограничена своим функционалом;
6. возможность сменить пользователя, выйдя из профиля и зайдя под другим.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Использование JavaScript на веб-страницах [Электронный ресурс].— URL: https://professorweb.ru/my/javascript/js_theory/level2/2_1.php (Дата обращения 02.02.2017). Загл. с экр. Яз. рус.
- 2 Системы сборки проектов [Электронный ресурс].— URL: <http://www.apache-maven.ru/index.html> (Дата обращения 01.02.2017). Загл. с экр. Яз. рус.
- 3 Национальная библиотека им. Н. Э. Баумана [Электронный ресурс].— URL: http://ru.bmstu.wiki/Apache_Tomcat (Дата обращения 05.03.2017). Загл. с экр. Яз. рус.
- 4 Что такое PostgreSQL? [Электронный ресурс].— URL: <http://postgresql.ru.net/manual/intro-what-is.html> (Дата обращения 21.04.2017). Загл. с экр. Яз. рус.
- 5 Немного о MVC [Электронный ресурс].— URL: <https://tproger.ru/articles/mvc/> (Дата обращения 20.03.2017). Загл. с экр. Яз. рус.
- 6 Chapter 3. Initializing the Driver [Электронный ресурс].— URL: <https://jdbc.postgresql.org/documentation/head/connect.html> (Дата обращения 20.04.2017). Загл. с экр. Яз. англ.

10.06.2017

