

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**"САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ  
УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»**

**Кафедра математической экономики**

---

**РАЗРАБОТКА WEB-СЕРВИСА ТАКСИ**

наименование темы выпускной квалификационной работы полужирным шрифтом

**АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ**

студента 5 курса 561 группы

по направлению **09.03.03 – Прикладная информатика**

код и наименование направления (специальности)

**механико-математического факультета**

наименование факультета, института, колледжа

---

**БЕГЛОВА ДМИТРИЯ БОРИСОВИЧА**

фамилия. имя, отчество

Научный руководитель

профессор, д.ф.-м.н., доцент

\_\_\_\_\_

подпись, дата

А.Ю.Трынин

инициалы, фамилия

Зав. кафедрой

д.ф.-м.н., профессор

\_\_\_\_\_

подпись, дата

С.И.Дудов

инициалы, фамилия

Саратов 2019

## **Введение**

В последнее время все больше и больше людей пользуются услугами такси, так как это комфортнее общественного транспорта и если поездок не так много, чаще всего обходится дешевле содержания собственной машины. Такси можно вызвать традиционным способом, таким как позвонить по телефону или вызвать через интернет. По телефону удобно вызывать такси, когда нет компьютера или смартфона под рукой, в остальных же случаях, плюсы от вызова через интернет следующие:

- Это зачастую быстрее
- Если не знаете точного адреса, сможете указать точку на карте
- Вы можете посмотреть историю своих поездок (если Вы зарегистрированы), это полезно для аналитики затрат на такси или если вы вдруг забыли какую-то вещь в салоне, всегда можно связаться с таксистом
- Вы сразу знаете точную цену от точки А к точке В
- Вы можете указать промежуточные точки, чтобы забрать, к примеру, друзей, и вам сразу посчитают цену.
- Парку такси не нужно платить зарплату менеджерам и за счет этого могут снизить цены на такси.

Цель данной работы в разработке автоматизированного web-сервиса такси с графическим интерфейсом для клиентов и водителей. Клиенты смогут вызвать себе машину от точки А к точке В и сразу же получить цену и назначенную машину, историю поездок, выбрать класс автомобиля. Водители такси будут иметь возможность зарегистрироваться и выходить на линию, получать заказы, смотреть историю поездок и свой баланс. В первом разделе мы опишем подробнее, возможности web-сервиса такси. Во втором разделе разберем дизайн web-сервиса. Третий раздел включает в себя техническую реализацию. В Четвертом разделе мы проанализируем конкурентоспособность на рынке

## **Анализ предметной области**

Работа такси осуществляется следующим образом: каждый водитель, заступая на смену, связывается с диспетчером, и тот вносит его в карточку

работающих в данный момент. Сведения карточки отображаются на экране. Водители держат обратную связь по радио.

Заказы поступают по телефону к диспетчеру, он записывает необходимые данные в базу заказов. Дата и время поступления заказа диспетчер вводит сам или вводится автоматически.

Клиент может сразу у диспетчера узнать стоимость заказа, и только потом заказывать. Потом из списка свободных водителей выбирает того, кто будет выполнять заказ или водитель сам отвечает по обратной связи, что примет заказ. После выполнения заказа водитель по радиации отчитывается перед диспетчером.

После каждой смены диспетчер формирует отчет, который показывает, сколько заказов поступило, сколько выполнилось, сколько было отменено, и их общую стоимость. Эти отчеты поступают к администратору, на основании этого всего администратор формирует общий отчет (сводный отчет) за определенную дату. В конце месяца администратор, пользуясь своими отчетами, формирует отчет по итогам месяца и отправляет его высшему руководству такси. Еще администратор занимается кадрами. Он формирует дела новых сотрудников, и вносит необходимые поправки в дела уже работающих на этом предприятии. На администраторе лежит ответственность за правильное формирование дел и их сохранность. Проблемы, которые могут возникнуть при осуществлении данной деятельности такие:

1. Потеря данных диспетчером, или неправильная их трактовка;
2. Выбор не оптимального маршрута следования таксистов;
3. Не корректная форма представления отчетов и т.д.

Следующим шагом является разработка базы данных.

## **Методология проектирования реляционных баз данных**

Реляционная база данных — это совокупность взаимосвязанных таблиц, каждая из которых содержит информацию об объектах определенного типа. Методология проектирования реляционных баз данных состоит из 8 этапов:

1. Определение основных типов сущностей, присутствующих в представлении данного пользователя о предметной области приложения. Документирование выделенных типов сущностей.
2. Определение важнейших типов связей, существующих между сущностями, выделенными на предыдущем этапе. Определение кардинальности связей и ограничений участия его членов. Документирование типов связей. При необходимости могут использоваться диаграммы сущность-связь (ER- диаграммы).
3. Связывание атрибутов с соответствующими типами сущностей или связей. Идентификация простых и составных атрибутов, простых и множественных атрибутов, а также производных атрибутов. Документирование сведений об атрибутах.
4. Определение доменов для всех атрибутов в каждой локальной концептуальной модели данных. Документирование сведений о доменах атрибутов.
5. Определение потенциального ключа для каждого типа сущности, если таких ключей окажется несколько, выбор среди них первичного ключа. Специализация или генерация типов сущностей (необязательный этап). Определение суперклассов и подклассов для типов сущностей.
6. Разработка диаграмм сущность - связь (ER- диаграмм), содержащих концептуальное отражение представлений пользователя о предметной области приложения.
7. Обсуждение локальных концептуальных моделей данных с конечными пользователями для получения подтверждения того, что данная модель корректно отражает представления пользователя о приложении и предприятии.

## **Выбор СУБД**

Для реализации построенной инфологической модели требуется СУБД, поддерживающая реляционную модель БД. Для решения поставленной задачи необходимо проанализировать несколько СУБД, поддерживающие ло-

кальные реляционные базы данных., а именно: MS Access, Visual Basic, Visual C++, SQL Server.

Access входит в состав самого популярного пакета Microsoft Office. Основные преимущества: знаком многим конечным пользователям и обладает высокой устойчивостью данных, прост в освоении, может использоваться непрофессиональным программистом, позволяет готовить отчеты из баз данных различных форматов. Предназначен для создания отчетов произвольной формы на основании различных данных и разработки некоммерческих приложений. Минимальные ресурсы ПК: процессор 486DX, Windows 3.1, 95, NT, объем оперативной памяти 12 (16) Мб, занимаемый объем на ЖМД 10-40 Мб.

Visual Basic - это универсальный объектно-ориентированный язык программирования, диалекты которого встроены в Access, Visual FoxPro. Преимущества: универсальность, возможность создания компонентов OLE, невысокие требования к аппаратным ресурсам ЭВМ. Применяется для создания приложений средней мощности, не связанных с большой интенсивностью обработки данных, разработки компонентов OLE, интеграция компонентов Microsoft Office. Минимальные ресурсы ПК: процессор 386DX, Windows 3.1, 95, NT, объем оперативной памяти 6 (16) Мб, занимаемый объем на ЖМД 8-36 Мб.

Visual C++ - наиболее мощный объектно-ориентированный язык программирования, обладает неограниченной функциональностью. Предназначен для создания компонентов приложений для выполнения операций, критичных по скорости.

SQL Server - сервер баз данных, реализует подход "клиент-сервер" и взаимодействует с указанными пакетами. Главные достоинства: высокая степень защиты данных, мощные средства для обработки данных, высокая производительность. Область применения: хранение больших объемов данных, хранение высокоценных данных или данных, требующих соблюдения режима секретности. Минимальные ресурсы ПК: процессор 486DX-33МГц, Windows NT, объем оперативной памяти 16 (32) Мб, занимаемый объем на ЖМД 80 Мб.

Исходя из проведенного анализа для реализации информационной системы таксопарка была выбрана СУБД MySQL.

## **Описание СУБД MySQL**

В данной работе в качестве СУБД использовалась БД MySQL. MySQL - это система управления реляционными базами данных, относящаяся к свободному программному обеспечению: она распространяется на условиях GNU Public License. СУБД MySQL является клиент-серверной системой, включающей многопоточный SQL-сервер, поддерживающий различные платформы, несколько клиентских программ и библиотек, инструменты администрирования и широкий диапазон программных интерфейсов приложений (API-интерфейсов). Сервер MySQL существует также и в форме встраиваемое многопоточной библиотеки, которую можно связывать с разрабатываемыми приложениями, чтобы получить более компактные, быстрые и легкоуправляемые продукты. В качестве интерфейса пользователя для создания базы данных применялся phpMyAdmin. Это приложение, написанное на PHP, которое обеспечивает полноценную работу с базами данных MySQL через браузер.

## **Описание возможностей Web-сервиса такси**

Возможности веб-сервиса:

- Выбор пункта назначения. Клиент заходит на домен web-сервиса ему предлагается ввести «адрес откуда его забрать» после «адрес куда его отвезти», а так же он может отметить на карте, с помощью левой кнопки мыши. Карта масштабируется и прокладывает маршрут, который видит клиент, в соответствии с рисунком 1.

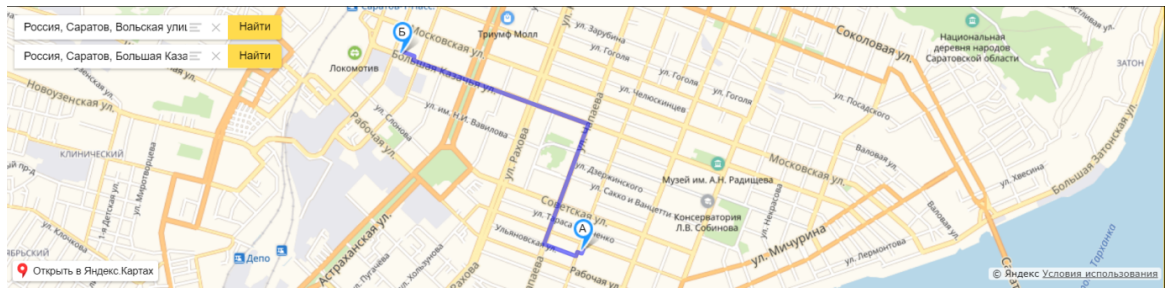


Рисунок 1 — Карта

- Выбор тарифа. Водители самостоятельно указывают цены за км, тем самым создают между собой конкуренцию. Это гарантируют низкую цену клиенту, потому что пассажир может выбрать и поехать за 150 рублей, но без кондиционера или за 180, но в машине, в которой есть кондиционер, а возможно, будет 3 вариант с кондиционером и за 145 рублей. Интерфейс имеет вид в соответствии с рисунком 2.

Сервис вызова такси Заказать такси Войти Зарегистрироваться

**Список свободных таксистов**

Не курит   
  Есть кондиционер   
  Перевоз домашних животных   
  Квитанция об оплате   
  Онлайн оплата   
 Применить

Стоимость за 1 км.	Машина	Гос. номер	Имя	Телефон	Дополнительно	Действия
150 р.	BMW x5 (Чёрный)	B333BB	Иванов Иван	+79778888888	<ul style="list-style-type: none"> <li>• Не курит</li> <li>• Может дать квитанцию об оплате</li> <li>• Принимает оплату онлайн</li> </ul>	<span style="background-color: #007bff; color: white; padding: 2px 10px; border-radius: 4px;">Заказать</span>
50 р.	Хейндэ Солярис (Голубой)	в314хс64	Беглов Дмитрий	+79661651216	<ul style="list-style-type: none"> <li>• Не курит</li> <li>• В машине есть кондиционер</li> <li>• Принимает оплату онлайн</li> </ul>	<span style="background-color: #007bff; color: white; padding: 2px 10px; border-radius: 4px;">Заказать</span>

Рисунок 2 — Выбор тарифа

- Расчет стоимости. В личном кабинете водителя, он указывает цену за 1 км поездки, когда клиент вводит адрес, карта рассчитывает кол-во км, после чего пассажир вводит требования для машины, такие как: «кондиционер в машине», «некурящий водитель», что нужно «перевезти домашнее животное» или после поездки, «потребуется квитанция» и нажимает на кнопку заказать такси. Сайт проверяет всех водителей, которые сейчас со статусом «На месте», а также подходит по параметрам, которые указал клиент и выводит список всех водителей. Пассажир,

может видеть какие сейчас автомобили на линии, сколько будет стоить поездка у каждого из водителей и выбрать из своих предпочтений.

- Выбор способа оплаты. Клиент может выбрать оплату наличными или безналичными, если выбран второй вариант, то будут указаны водители, у которых есть переносной терминал эквайринга для оплаты кредитными и дебетовыми картами. Такие водители будут пользоваться большим спросом, так как это создает дополнительное удобство для клиентов, повышает лояльность молодого поколения к сервису, которые используют в основном безналичный расчет. И несмотря на то, что водителю придется платить % банку за использование эквайрингом, это даст ему преимущество над водителями без терминала.
- Регистрация пользователя. Также клиенту предлагается зарегистрироваться, человек вводит свои данные: «имя», «телефон», «пароль» в форму для регистрации и попадает в свой личный кабинет. Регистрация необязательная, такси можно вызвать и без нее. Форма имеет вид в соответствии с рисунком 3.

Сервис вызова такси

[Заказать такси](#) [Войти](#) [Зарегистрироваться](#)

Регистрация

Ф.И.О.

E-Mail адрес

Пароль

Подтвердите пароль

Тип аккаунта

[Зарегистрироваться](#)

Рисунок 3 — Форма регистрации пользователя

- История поездок. В личном кабинете клиент видит свою историю поездок, в которой отображены «стоимость», «откуда», «куда», «марка авто», «государственный номер автомобиля», «телефон водителя». Это помогает клиенту, анализировать свои расходы на такси и если клиент,



что-нибудь забыл в автомобиле, то пассажир может связаться с водителем и найти пропажу.

- Регистрация водителя. Если регистрация для клиента необязательная, то водитель сделать это обязан. Чтобы пройти регистрацию, для этого ему нужно будет заполнить поля: «имя», «фамилия», «телефон», «e-mail», «марка и модель автомобиля», «цвет автомобиль», «государственный номер автомобиля», также водитель должен указать, есть ли в автомобиле кондиционер, курит ли водитель, приемлемо ли перевозить домашних животных, есть ли квитанции у водителя, принимает ли он безналичный расчет, загрузить отсканированные копии своих документов, таких как: Паспорт, водительское удостоверение, свидетельство транспортного средства.
- Выход на линию. Для того чтобы водителю выйти на линию, ему требуется авторизоваться в личном кабинете, и включить пункт «на месте», после того как его смена закончилась, и чтобы не получать заказы, он ставит статус: «Нет на месте».
- История заказов. Как и для пассажиров, для водителей также есть история заказов, в ней отображены поездки с датой и временем
- Баланс водителя В личном кабинете водителя, скрипт рассчитывает сумму поездок, а также комиссию сервиса, которую водитель может оплатить картой или Яндекс.деньгами прямо в личном кабинете.

## **Заключение**

В ходе работы был проведен анализ предметной области, исследованы требования к информационной системе, проведено сравнение СУБД, рассмотрены и применены на практике средства проектирования реляционных баз данных и спроектирована сама система.

На практике способы разработки web-интерфейса были реализованы с помощью программного кода, написанного на языке PHP. Система является работоспособной, простой в эксплуатации и имеет возможность для дальнейшего расширения.

Проведенные в работе исследования позволили более углубленно изучить процесс проектирования информационных систем, реляционных баз данных и разработки web-интерфейса, а также повысить навыки решения практических задач как теоретическими, так и практическими способами.

## Список использованных источников

1. Шелдон, Р., Мойе, Д. MySQL 5: базовый курс / Р. Шелдон, Д. Мойе. - М.: Диалектика, 2007. - 180 с.
2. Васвани, В. MySQL: использование и администрирование / В. Васвани. - М.: Питер, 2011. - 368 с.
3. Дюбуа, П. MySQL / П. Дюбуа. - М.: Вильямс, 2006. - 168 с.
4. Кузнецов, М., Симдянов, И. MySQL 5. В подлиннике / М. Кузнецов, И. Симдянов, БХВ-Петербург, 2006. - 92 с.
5. Чен, П., Модель сущность-связь шаг к единому представлению о данных / П. Чен. - М.: Вильямс, 2009. - 43 с.
6. Суэринг, С., Конверс Т., Парк, Д. PHP и MySQL. Библия программиста / С. Суэринг, Т. Конверс, Д. Парк. - М.: Диалектика, 2010. - 92 с.
7. Котеров, Д., Костарев, А. PHP / Д. Котеров, А. Костарев. - М.: БХВ-Петербург, 2005. С. 1120
8. Зандстра, М. PHP: объекты, шаблоны и методики программирования / М. Зандстра. -М.: Вильямс, 2010. - 560 с.
9. Дари, К., Баланеску, Э. PHP и MySQL: создание интернет-магазина / К. Дари, Э. Баланеску. - М.: Вильямс, 2010. 356 с.
10. Редько, В.Н.; Бассараб, И.А. Базы данных и информационные системы / В.Н. Редько, И.А. Бассараб. - М.: Знание, 2011. - 602 с.
11. Голицына, О.Л. Базы данных / О.Л. Голицына. - М.: Инфра-М, 2013. - 399 с.