



**Частное учреждение профессионального образования
«Высшая школа предпринимательства»
(ЧУПО «ВШП»)**

КУРСОВОЙ ПРОЕКТ

«РАЗРАБОТКА БАЗЫ ДАННЫХ ДЛЯ АВТОСЕРВИСА»

Выполнил:

студент 3-го курса специальности
09.02.07 «Информационные системы и
программирование»

Марупов Ромиз Алимухаммадович

подпись: _____

Проверил:

преподаватель дисциплины,
преподаватель ЧУПО «ВШП»,
к.ф.н. Ткачев П.С.

оценка: _____

подпись: _____

Тверь, 2024

СОДЕРЖАНИЕ

Введение	3
Глава I.	7
1.1 Изучение функциональной структуры предприятия автосервиса.	7
1.2 Проведение анализа работы предприятия, выявление основных потребностей в учете клиентов и заказов.	11
1.3 Подбор наиболее подходящей технологии системы управления базами данных (СУБД) с учетом требований и особенностей автосервиса.	14
Глава II.	16
2.1. Разработать структуру базы данных, включая таблицы, связи между ними, атрибуты и ключевые поля.	16
2.2. Разработка базы данных, обеспечивающей эффективный учет клиентов и заказов в автосервисе.	16
2.3 Провести тестирование базы данных для обеспечения корректной работы и соответствия требованиям.	23
Заключение.....	32
Список литературы.....	34
Приложение 1.....	35
Приложение 2.....	36

Введение

Предлагаемая вашему вниманию проектная работа посвящена разработке базы данных для автосервиса. В современном мире автосервисная индустрия является ключевым сектором экономики, обслуживающим миллионы автовладельцев по всему миру. С ростом числа автомобилей, находящихся в собственности населения, возрастает и потребность в качественном обслуживании и ремонте. Сложные технологии современных автомобилей требуют особого внимания и профессионализма со стороны автосервисов. Для успешного существования и развития, автосервисы вынуждены предлагать высокое качество услуг и удовлетворять все более сложные потребности клиентов. В связи с этим, эффективное управление информацией о клиентах и их заказах становится стратегически важным аспектом управления автосервисом.

Актуальность:

Разработка базы данных для автосервиса остается весьма актуальной в современной автомобильной индустрии. С увеличением числа автомобилей на дорогах и повышением требований клиентов к качеству обслуживания, необходимость в эффективном управлении информацией о клиентах, их автомобилях, запасных частях и истории ремонтов становится все более важной. Разработка базы данных позволяет автосервисам оптимизировать свои бизнес-процессы, повышать качество обслуживания, обеспечивать безопасность данных клиентов и анализировать информацию для принятия более обоснованных решений. Таким образом, база данных для автосервиса остается ключевым инструментом для успешной работы и конкурентоспособности в современной автомобильной отрасли.

Однако, параллельно с ростом спроса на услуги автосервисов, наблюдается усиление конкуренции на рынке. Автосервисы вынуждены постоянно совершенствоваться и предлагать более высокое качество услуг, чтобы удерживать и привлекать клиентов. В этой связи, эффективное управление информацией о

клиентах и их автомобилях становится важным фактором успеха для автосервисных предприятий.

Соответственно, разработка базы данных для автосервиса представляет собой актуальную задачу, нацеленную на оптимизацию процессов учета клиентов и заказов, улучшение качества обслуживания и укрепление конкурентоспособности автосервиса на рынке.

Разработка базы данных для автосервиса — это крайне важная задача. Она предполагает создание специальной системы, которая помогает автосервисам управлять информацией о своих клиентах и их машинах. Это необходимо потому, что количество заказов и клиентов в автосервисах постоянно растет, и хорошее управление этой информацией делает работу автосервисов более эффективной и успешной, что приятно и удобно как им, так и клиентам. Дальнейшее раскрытие темы включает в себя изучение существующих методов учета и систем управления данными в автосервисах, а также анализ их преимуществ и недостатков. Это позволяет выделить основные требования к разрабатываемой базе данных и определить ключевые параметры ее функциональности.

Продвинутые технологии информационных систем позволяют эффективно автоматизировать процессы учета и обработки данных, что способствует повышению оперативности и качества обслуживания клиентов. Таким образом, разработка базы данных для автосервиса становится необходимым шагом в современном бизнесе, а внедрение такого инструмента позволяет автосервисам быть конкурентоспособными и успешными на рынке услуг.

Обобщая, можно отметить, что разработка базы данных для автосервиса представляет собой сложную и многостороннюю задачу, которая требует глубокого анализа и профессионального подхода. Однако, эта тема уже претерпела значительное исследование и имеет широкий практический опыт применения, что делает ее важной и актуальной для дальнейших исследований и разработок в данной области.

Цели и задачи работы

Целью данной работы является разработка базы данных для учета текущих клиентов автосервиса. Для достижения этой цели были поставлены следующие задачи:

- Изучение функциональной структуры предприятия автосервиса.
- Проведение анализа работы предприятия, выявление основных потребностей в учете клиентов и заказов.
- Подбор наиболее подходящей технологии системы управления базами данных (СУБД) с учетом требований и особенностей автосервиса.
- Разработать структуру базы данных, включая таблицы, связи между ними, атрибуты и ключевые поля.
- Разработка базы данных, обеспечивающей эффективный учет клиентов и заказов в автосервисе.
- Провести тестирование базы данных для обеспечения корректной работы и соответствия требованиям.

Метод исследования:

- Для достижения поставленных целей и задач исследования применяется комплексный методологический подход, включающий в себя несколько основных методов и инструментов.
- Анализ функциональной структуры автосервиса: Этот метод предполагает изучение внутренних процессов и организационной структуры автосервиса. С помощью анализа документации, интервью с сотрудниками и наблюдений за рабочим процессом будет выявлено, как устроено функционирование предприятия, какие процессы происходят внутри и какие требования к базе данных вытекают из этого.
- Анализ работы автосервиса: Этот метод предполагает более глубокий анализ самой деятельности автосервиса, включая процессы приема автомобилей, выявление неисправностей, составление заказов на запчасти, ремонтные работы, учет времени выполнения заказов и т.д. Анализ проводится на основе

статистических данных, а также наблюдений и интервью с персоналом автосервиса.

- Подбор технологии СУБД: для разработки базы данных будет проведен анализ существующих систем управления базами данных (СУБД) с учетом требований автосервиса. Этот метод включает в себя сравнительный анализ функциональных возможностей, производительности, стоимости и прочих характеристик различных СУБД, чтобы выбрать наиболее подходящее решение.
- Разработка базы данных: на основе анализа предприятия и выбора подходящей технологии СУБД будет проведена разработка структуры базы данных. Этот этап включает в себя определение сущностей и их атрибутов, построение связей между таблицами, создание пользовательского интерфейса и установку соответствующих механизмов защиты данных.
- Тестирование и внедрение: после разработки базы данных будет проведено тестирование ее работы на практике с использованием тестовых данных. После успешного завершения тестирования произойдет внедрение базы данных в работу автосервиса с обучением персонала и настройкой процессов работы с системой.

Глава I.

1.1 Изучение функциональной структуры предприятия автосервиса.

Перед разработкой базы данных для автосервиса требуется комплексный анализ предметной области, который охватывает различные аспекты функционирования сервиса и требования его клиентов. Важным этапом данного анализа является изучение бизнес-процессов автосервиса: начиная от момента приема клиента и его автомобиля, через выполнение ремонтных работ и обслуживание, до завершения и выдачи автомобиля владельцу. Кроме того, требуется учитывать особенности работы с запасными частями, учет времени работы сотрудников, а также финансовый учет услуг и материалов. Важно также анализировать потребности клиентов: от возможности записи на сервис онлайн до предоставления информации о статусе ремонта и консультаций по обслуживанию.

Рассмотрим структуру и систему управления станции технического обслуживания, а также разберем особенности учета и анализа данных в автосервисе. Статья будет полезна руководителям и владельцам СТО, а также менеджерам и аналитикам автомастерских на любом этапе развития бизнеса [4].

Структура автосервиса



Корректная работа компании возможна только при правильно разработанной структуре предприятия, что необходимо для контроля эффективности работы сотрудников и отслеживания бизнес-процессов.

При анализе предметной области необходимо учитывать и законодательные требования, особенно в отношении обработки персональных данных. Законодательство о защите данных ставит перед автосервисами серьезные обязательства по обеспечению конфиденциальности и безопасности информации о клиентах и их транспортных средствах. Поэтому в разработке базы данных важно учесть механизмы защиты данных и соответствие требованиям законодательства.

Определение основных сущностей и их атрибутов является ключевым этапом анализа. Основными сущностями, скорее всего, будут клиенты, автомобили, работы по ремонту и обслуживанию, запасные части и услуги. Для каждой из этих сущностей требуется определить соответствующие атрибуты, которые будут хранить информацию, необходимую для полноценного функционирования автосервиса.

Кроме того, необходимо определить взаимосвязи между сущностями. Например, один клиент может обслуживать несколько автомобилей, или один ремонт может включать в себя несколько видов работ. Эти взаимосвязи важны для эффективного функционирования базы данных и обеспечения связности данных.

Анализ процессов внесения и извлечения данных также имеет ключевое значение. Необходимо определить, какие данные и каким образом будут вводиться в систему, а также как пользователи будут получать доступ к этой информации. Это поможет разработать удобный и эффективный интерфейс для работы с базой данных.

Важным аспектом анализа является также учет масштабируемости и безопасности системы. База данных должна быть спроектирована таким образом, чтобы обеспечить возможность расширения и обработки больших объемов данных, а также защитить информацию от несанкционированного доступа и утечек.

Соответственно, анализ предметной области перед разработкой базы данных для автосервиса представляет собой комплексный процесс, охватывающий различные аспекты функционирования сервиса, потребности клиентов, требования законодательства и многие другие факторы. От качественного анализа зависит успешность всего проекта по разработке базы данных и его способность эффективно поддерживать работу автосервиса в современных условиях.

Автосервис представляет собой специализированное предприятие или учреждение, которое специализируется на предоставлении услуг по обслуживанию, ремонту и техническому обслуживанию автомобилей. Это место, куда владельцы автомобилей обращаются для различных видов работ, начиная от регулярного технического обслуживания и замены расходных материалов до комплексного диагностирования и ремонта сложных механизмов и узлов транспортного средства [1].

В автосервисе работают высококвалифицированные специалисты, включая механиков, электриков, диагностов и других профессионалов, обладающих необходимыми знаниями и навыками для проведения различных видов работ по обслуживанию автомобилей. Они используют специализированные инструменты, оборудование и диагностические системы для выявления неисправностей, проведения ремонта и обеспечения надежной работы автомобилей.

Главная цель автосервиса - обеспечить безопасность, надежность и эффективную эксплуатацию автомобилей. Помимо ремонта и обслуживания, автосервис также может предоставлять консультации по вопросам технического обслуживания, рекомендации по уходу за автомобилем и помощь в выборе запасных частей и аксессуаров [5].

Соответственно, автосервис играет важную роль в поддержании автомобилей в исправном состоянии, продлении их срока службы и обеспечении безопасности на дороге для владельцев и пассажиров.

Сущность автосервиса заключается в том, что это специализированное предприятие или учреждение, которое оказывает различные услуги по

обслуживанию, ремонту и техническому обслуживанию автомобилей. Главная цель автосервиса состоит в том, чтобы обеспечить надежное функционирование автомобилей, продлить их срок службы, обеспечить безопасность водителей и пассажиров на дороге. Обслуживание и ремонт автомобилей: Автосервис предоставляет широкий спектр услуг, начиная от обычного технического обслуживания, замены масла и фильтров до сложных ремонтов двигателей, трансмиссий и других узлов автомобиля.

Профессиональные специалисты: В автосервисах работают высококвалифицированные специалисты, такие как механики, электрики, диагносты и другие, которые обладают необходимыми знаниями и навыками для эффективного обслуживания и ремонта автомобилей.

Специализированное оборудование и инструменты: Автосервисы обычно оснащены специализированным оборудованием и инструментами, необходимыми для проведения различных видов работ, начиная от диагностики до ремонта сложных узлов автомобиля.

Клиентоориентированность: Хороший автосервис стремится к удовлетворению потребностей клиентов, предоставляя качественные услуги, консультации и рекомендации, а также обеспечивая прозрачность и честность в отношениях с клиентами.

Соответственно, сущность автосервиса состоит в его способности обеспечивать надежное функционирование автомобилей путем предоставления высококачественных услуг по обслуживанию и ремонту, основанных на профессионализме, опыте и использовании современных технологий и оборудования [2].

Ремонт и обслуживание автомобилей: это основная функция автосервиса. Включает в себя проведение ремонта, замену деталей и узлов, а также выполнение технического обслуживания, такого как замена масла, фильтров, регулировка тормозов и т.д.

Диагностика и выявление неисправностей: Автосервис обеспечивает диагностику автомобиля для выявления любых потенциальных проблем или неисправностей в работе двигателя, системы подвески, электрических компонентов и т.д.

1.2 Проведение анализа работы предприятия, выявление основных потребностей в учете клиентов и заказов.

Для проекта разработки базы данных для автосервиса необходимо провести анализ работы предприятия и выявить основные потребности в учете клиентов и заказов.

Изучение текущих процессов работы автосервиса от момента обращения клиента до завершения заказа позволит определить основные этапы обслуживания, включая прием заказа, диагностику, ремонт, замену запчастей и возврат автомобиля клиенту.

Также необходимо собрать информацию о клиентах, включая контактные данные, историю предыдущих заказов и особенности их автомобилей.

Важным этапом анализа будет изучение процессов учета заказов, включая регистрацию новых заказов, отслеживание статуса выполнения работ и выставление счетов.

Анализ эффективности текущих процессов учета клиентов и заказов поможет выявить узкие места и проблемы, с которыми сталкиваются сотрудники автосервиса.

Для полного понимания потребностей предприятия в учете клиентов и заказов также необходимо провести интервью с сотрудниками, чтобы выявить их потребности и проблемы в работе с учетом клиентов и заказов.

На основе собранной информации можно сравнить текущие процессы с лучшими практиками учета клиентов и заказов в автосервисной отрасли и сформулировать рекомендации по их улучшению.

В конечном итоге, разработка плана действий для внедрения улучшений, включая внедрение специализированного программного обеспечения, оптимизацию

рабочих процессов и обучение персонала, будет ключевым шагом для улучшения учета клиентов и заказов на автосервисе.

Предоставление консультаций и рекомендаций: Специалисты автосервиса могут консультировать клиентов относительно технического состояния и ремонта их автомобилей, а также предоставлять рекомендации по уходу за транспортным средством.

Шиномонтажные услуги: Некоторые автосервисы предоставляют услуги по установке, замене и ремонту шин и дисков.

Обновление и установка дополнительного оборудования: Автосервис также может предложить установку и обновление различного дополнительного оборудования и аксессуаров, таких как аудио- и видеосистемы, сигнализации, средства связи и т.д.

Предоставление гарантийных услуг: Некоторые автосервисы предоставляют гарантийные услуги на проведенные работы и установленные запчасти.

Функции предметной области автосервиса охватывают широкий спектр операций, начиная от приема автомобилей и заканчивая выпуском отремонтированных или обслуженных транспортных средств. Прием автомобиля включает в себя регистрацию клиента, осмотр транспортного средства, определение причины обращения и составление первичного диагностического отчета. После приема автомобиля проводится более подробная диагностика, позволяющая выявить все неисправности и проблемы, а также оценить степень их серьезности и необходимость ремонта.

Основная задача автосервиса - обеспечить качественное обслуживание и ремонт автомобилей. Это включает в себя проведение необходимых работ по замене деталей, ремонту двигателя, коробки передач, системы охлаждения и прочих компонентов автомобиля. Кроме того, в рамках обслуживания могут выполняться работы по замене масла, проверке тормозной системы, а также проведению технического обслуживания [3].

Управление запасными частями является также важной функцией автосервиса. Это включает в себя контроль за наличием необходимых запасных частей на складе, заказ новых деталей у поставщиков, а также учет использованных комплектующих в процессе ремонта или обслуживания автомобилей. Управление запасными частями играет важную роль в автосервисе. В вашей базе данных уже есть таблица `parts`, которая, вероятно, содержит информацию о запасных частях. Она включает в себя поля, такие как `id`, `RepairOrders_id`, `name`, и `quantity`, которые могут быть использованы для отслеживания запасов и использования деталей. Добавление такой функциональности поможет улучшить эффективность управления запасными частями и обеспечить бесперебойную работу автосервиса.

Для успешной работы автосервиса также необходимо эффективно управлять временем работы сотрудников. Это включает в себя планирование графика работы, распределение задач между механиками и другими специалистами, а также учет отработанного времени и оплату труда.

Важным аспектом работы автосервиса является взаимодействие с клиентами. Это включает в себя консультации по вопросам обслуживания автомобилей, предоставление информации о статусе ремонта, а также прием обратной связи от клиентов для улучшения качества предоставляемых услуг.

Наконец, необходимо учитывать финансовый аспект работы автосервиса, включая выставление счетов, учет оплат и расходов, а также составление финансовых отчетов для анализа финансового состояния предприятия.

Соблюдение стандартов и требований производителей: при проведении ремонта и обслуживания автомобилей, автосервис должен соблюдать стандарты и требования, установленные производителями автомобилей, чтобы гарантировать качество и надежность работы транспортного средства.

Эти функции помогают автосервису обеспечивать надежное функционирование автомобилей и удовлетворять потребности клиентов в обслуживании и ремонте своих транспортных средств.

Деятельность автосервиса обладает рядом особенностей, которые определяют его функционирование и специфику работы. Вот некоторые из них:

Диагностика и ремонт современных технологий: Современные автомобили оснащены сложной электроникой и компьютерными системами, что требует высокой квалификации специалистов и использования специализированных программ для диагностики и ремонта [3].

Управление запасными частями и инвентарем: Автосервис должен эффективно управлять запасными частями, инвентарем и расходными материалами для обеспечения бесперебойной работы и удовлетворения потребностей клиентов.

Отношения с клиентами: Качество обслуживания и взаимоотношений с клиентами играет ключевую роль в успешной деятельности автосервиса. Это включает в себя предоставление профессиональных консультаций, прозрачность в ценообразовании и обслуживании, а также решение возможных конфликтных ситуаций.

Эти особенности определяют характер работы автосервиса и влияют на его успешность и конкурентоспособность на рынке.

1.3 Подбор наиболее подходящей технологии системы управления базами данных (СУБД) с учетом требований и особенностей автосервиса.

Для выбора наиболее подходящей технологии СУБД для автосервиса учтены такие требования, как доступность, производительность и надежность. В результате анализа было принято решение использовать MySQL.

MySQL отлично подходит для небольших и средних предприятий благодаря своей бесплатности и открытости. Ее преимущества, такие как высокая производительность, масштабируемость и широкое распространение, сделали ее популярным выбором среди различных организаций. Более того, наличие обширного сообщества пользователей и разработчиков обеспечивает доступ к поддержке, обновлениям и различным решениям интеграции [6].

Кроме того, удобство использования и обучения MySQL делает ее привлекательным выбором для студентов колледжа. Простой и интуитивно понятный интерфейс, а также доступ к обучающим ресурсам и документации делают процесс обучения более эффективным [10].

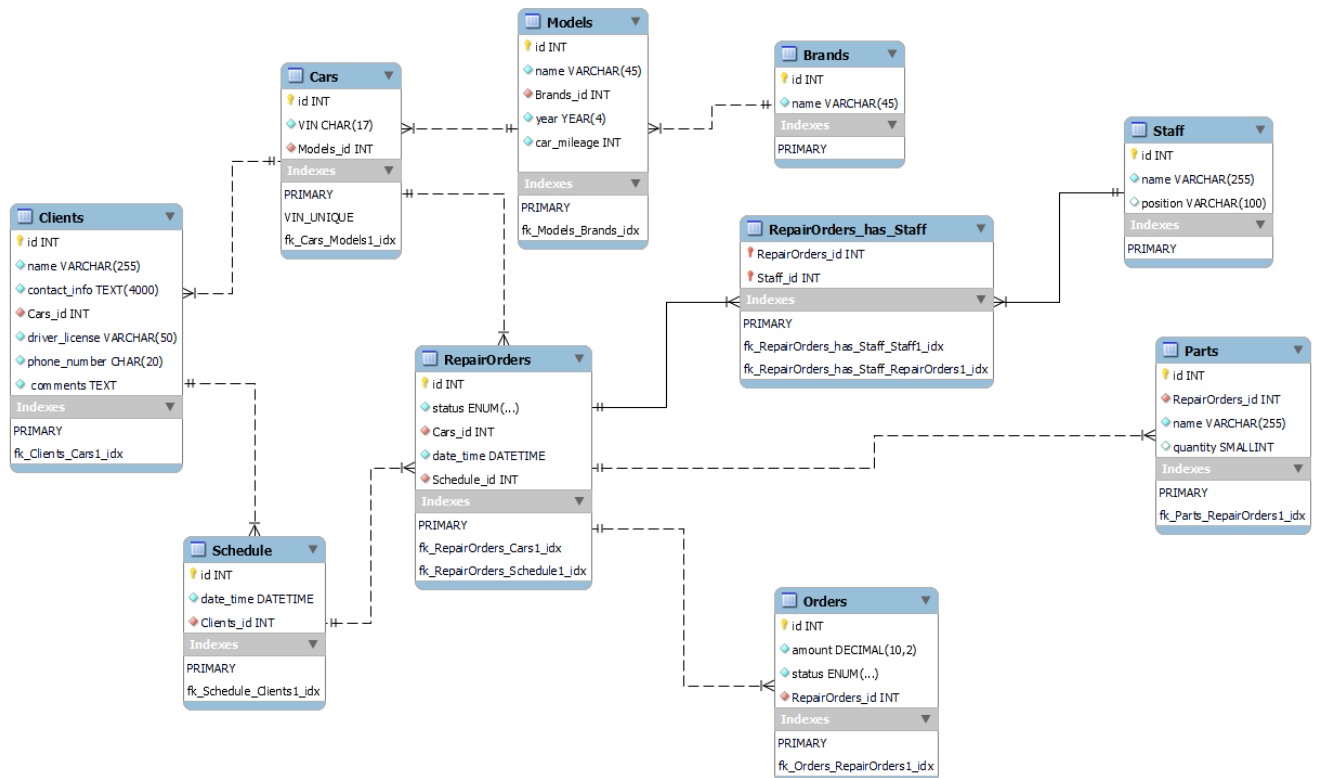
Принятие решения в пользу MySQL также обусловлено вашим текущим обучением в колледже на этой программе. Работа с MySQL в рамках проекта разработки базы данных для автосервиса даст вам дополнительный практический опыт и позволит закрепить ваши знания.

Разумеется, выбор MySQL для проекта обусловлен не только требованиями автосервиса, но и моим текущим курсом в колледже, где мы изучаем эту программу. Это помогает мне лучше понимать материал и применять его на практике. Проект становится не просто заданием, а возможностью применить полученные знания на практике.

Глава II.

2.1. Разработать структуру базы данных, включая таблицы, связи между ними, атрибуты и ключевые поля.

Следующим шагом при создании базы данных была построена схема данных. Каждая таблица имеет связь с другими таблицами. Каждая таблица имеет связь с другими таблицами.



2.2. Разработка базы данных, обеспечивающей эффективный учет клиентов и заказов в автосервисе.

По разработанной концептуальной схеме логического и физического уровней в базе данных «Учёт выполненных работ» были созданы 10 таблиц.

Clients	
id	INT
name	VARCHAR(255)
contact_info	TEXT(4000)
driver_license	VARCHAR(50)
phone_number	CHAR(20)
comments	TEXT
Indexes	
PRIMARY	

Таблица 1

Таблица "Clients" предназначена для хранения информации о клиентах автосервиса. Клиенты - это физические лица или организации, обратившиеся в автосервис для получения услуг по обслуживанию и ремонту своих автомобилей.

Основная цель этой таблицы - предоставить автосервису полную информацию о клиентах, их контактных данных, а также о технических деталях их автомобилей для обеспечения качественного обслуживания [9].

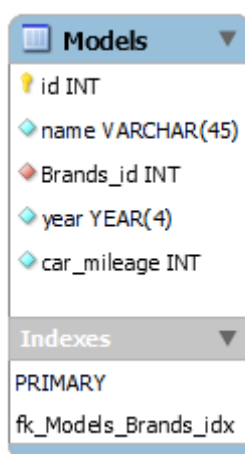
Таблица "Clients" играет важную роль в автосервисе, поскольку предоставляет ценную информацию о клиентах и их автомобилях, помогая автосервису эффективно управлять своими клиентскими отношениями и предоставлять высококачественное обслуживание.

Cars	
id	INT
VIN	CHAR(17)
Models_id	INT
Indexes	
PRIMARY	
VIN_UNIQUE	
fk_Cars_Models1_idx	

Таблица 2

Таблица "Cars" предназначена для хранения информации о конкретных автомобилях, которые обслуживаются в автосервисе. Каждая запись в этой таблице содержит подробную информацию о каждом автомобиле, включая его уникальный идентификатор (ID) и уникальный идентификационный номер (VIN), а также информацию о модели автомобиля и клиенте, которые предоставляют обслуживание данного автомобиля.

Таблица "Cars" играет важную роль в автосервисе, поскольку предоставляет полную информацию о каждом автомобиле, который обслуживается в автосервисе. Эта информация позволяет эффективно управлять автопарком, планировать техническое обслуживание и ремонт, а также обеспечивает удобный доступ к информации для сотрудников автосервиса.



The image shows a screenshot of a database management tool displaying the structure of a table named "Models". The table has five columns: "id" (INT, primary key), "name" (VARCHAR(45)), "Brands_id" (INT, foreign key), "year" (YEAR(4)), and "car_mileage" (INT). Below the columns, there is an "Indexes" section showing a "PRIMARY" index on the "id" column and a foreign key index "fk_Models_Brands_idx" on the "Brands_id" column.

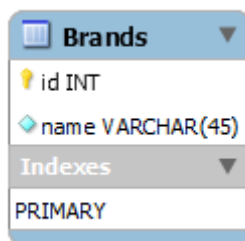
Column	Type	Constraints
id	INT	PRIMARY KEY
name	VARCHAR(45)	
Brands_id	INT	FOREIGN KEY (fk_Models_Brands_idx)
year	YEAR(4)	
car_mileage	INT	

Таблица 3

Таблица "Models" предназначена для хранения информации о моделях автомобилей. Каждая запись в этой таблице содержит подробную информацию о модели, такую как ее уникальный идентификатор, название, год выпуска и текущий пробег.

Таблица "Models" играет важную роль в базе данных для автосервиса, поскольку содержит информацию о различных моделях автомобилей, которые обслуживаются в автосервисе. Эта информация позволяет эффективно управлять информацией о

моделях, планировать техническое обслуживание и ремонт, а также обеспечивает удобный доступ к информации для сотрудников автосервиса.

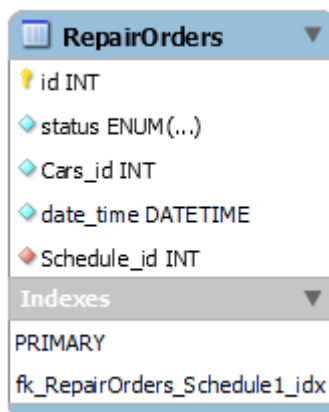


Brands	
id	INT
name	VARCHAR(45)
Indexes	
PRIMARY	

Таблица 4

Таблица "Brands" предназначена для хранения информации о брендах автомобилей. Каждая запись в этой таблице содержит информацию о конкретном бренде, включая его уникальный идентификатор и название.

Таблица "Brands" играет важную роль в базе данных для автосервиса, поскольку содержит информацию о различных брендах автомобилей, которые обслуживаются в автосервисе. Эта информация позволяет эффективно управлять информацией о брендах, планировать техническое обслуживание и ремонт, а также обеспечивает удобный доступ к информации для сотрудников автосервиса.

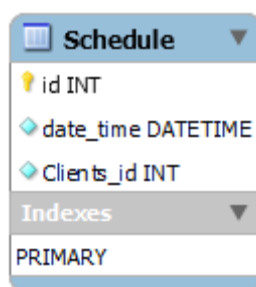


RepairOrders	
id	INT
status	ENUM(...)
Cars_id	INT
date_time	DATETIME
Schedule_id	INT
Indexes	
PRIMARY	
fk_RepairOrders_Schedule1_idx	

Таблица 5

Таблица "Repair Orders" предназначена для учета заказов на ремонт автомобилей в автосервисе. Каждая запись в этой таблице содержит информацию о конкретном заказе на ремонт, включая его уникальный идентификатор, статус выполнения, дату и время создания.

Таблица "Repair Orders" играет важную роль в базе данных для автосервиса, поскольку позволяет эффективно управлять заказами на ремонт, отслеживать их статус и связывать с соответствующими автомобилями и записями на обслуживание. Это обеспечивает более эффективное планирование и выполнение работ по ремонту и обслуживанию автомобилей.



The image shows a screenshot of a database table structure for a table named "Schedule". The table has three columns: "id" of type INT, "date_time" of type DATETIME, and "Clients_id" of type INT. Below the columns, there is a section for "Indexes" which shows a "PRIMARY" index.

Schedule	
id	INT
date_time	DATETIME
Clients_id	INT
Indexes	
PRIMARY	

Таблица 6

Таблица "Service Schedule" предназначена для учета записей на обслуживание автомобилей в автосервисе. Каждая запись в этой таблице содержит информацию о запланированных обслуживаниях, включая уникальный идентификатор и дату и время обслуживания.

Таблица "Service Schedule" играет важную роль в базе данных для автосервиса, поскольку позволяет эффективно управлять запланированными обслуживаниями, отслеживать их даты и временные рамки, а также связывать с соответствующими клиентами. Это обеспечивает более эффективное планирование и выполнение работ по обслуживанию автомобилей, а также улучшает управление клиентскими отношениями.

Orders	
id	INT
amount	DECIMAL(10,2)
status	ENUM(...)
RepairOrders_id	INT
Indexes	
PRIMARY	

Таблица 7

Таблица "Orders" предназначена для учета заказов на ремонт в автосервисе. Каждая запись в этой таблице содержит информацию о конкретном заказе на ремонт, включая его уникальный идентификатор, сумму, статус и связь с соответствующим заказом на ремонт.

Таблица "Orders" играет важную роль в базе данных для автосервиса, поскольку позволяет эффективно управлять заказами на ремонт, отслеживать их статус и связывать с соответствующими заказами на ремонт. Это обеспечивает более эффективное планирование и выполнение работ по ремонту автомобилей.

RepairOrders_has_Staff	
RepairOrders_id	INT
Staff_id	INT
Indexes	
PRIMARY	
fk_RepairOrders_has_Staff_Staff1_idx	

Таблица 8

Таблица "RepairOrders_has_Staff" предназначена для связи сотрудников автосервиса с заказами на ремонт. Каждая запись в этой таблице определяет связь между конкретным заказом на ремонт и сотрудником, который назначен для выполнения работ по этому заказу.

Таблица "RepairOrders_has_Staff" позволяет эффективно управлять назначением сотрудников на заказы на ремонт, что обеспечивает более эффективное планирование и выполнение работ в автосервисе.

Staff	
id	INT
name	VARCHAR(255)
position	VARCHAR(100)
Indexes	
PRIMARY	

Таблица 9

Таблица "Staff" предназначена для хранения информации о сотрудниках автосервиса. Каждая запись в этой таблице содержит данные о конкретном сотруднике, такие как уникальный идентификатор, имя и должность.

Таблица "Staff" предназначена для хранения основной информации о сотрудниках автосервиса. Эта информация может быть использована для управления персоналом, а также для назначения сотрудников на заказы на ремонт и другие работы. Таблица "Staff" предназначена для хранения информации о сотрудниках автосервиса, которые выполняют различные задачи, связанные с обслуживанием и ремонтом автомобилей. Каждая запись в этой таблице содержит уникальный идентификатор сотрудника, его имя и должность в автосервисе, что позволяет эффективно управлять персоналом и назначать сотрудников на выполнение определенных задач. Таблица "Staff" играет важную роль в организации работы автосервиса, обеспечивая наличие квалифицированного персонала для выполнения различных видов работ по обслуживанию и ремонту автомобилей.

Parts	
id	INT
RepairOrders_id	INT
name	VARCHAR(255)
quantity	SMALLINT
Indexes	
PRIMARY	

Таблица 10

Таблица "Parts" предназначена для учета деталей, используемых при выполнении ремонта автомобилей. Каждая запись в этой таблице содержит информацию о конкретной детали, ее уникальном идентификаторе, связанном заказе на ремонт, названии и количестве на складе.

Таблица "Parts" играет важную роль в учете и управлении деталями, используемыми в процессе ремонта автомобилей. Она позволяет эффективно управлять запасами деталей на складе и отслеживать их использование при выполнении заказов на ремонт.

2.3 Провести тестирование базы данных для обеспечения корректной работы и соответствия требованиям.

Вот несколько типовых запросов к базе данных (БД), которые могут отражать соответствующие бизнес-процессы в сфере автосервиса:

1. Добавление нового заказа на ремонт:

```
INSERT INTO RepairOrders (status, Cars_id, date_time, Schedule_id)
SELECT 'new', Cars_id, NOW(), Schedule.id
FROM Clients
JOIN Cars ON Clients.Cars_id = Cars.id
JOIN Schedule ON Clients.id = Schedule.Clients_id
WHERE Cars.id = 12;
```

2. Обновление статуса заказа на ремонт:

```
UPDATE RepairOrders
SET status = 'process'
WHERE id = 6;
```

3. Получение списка активных заказов на ремонт для конкретного клиента:

```
SELECT ro.id, ro.date_time, ro.status
FROM RepairOrders ro
```

```
JOIN Cars c ON ro.Cars_id = c.id
JOIN Clients cl ON c.id = cl.Cars_id
WHERE cl.id = 1 AND ro.status IN ('new', 'process');
```

4. Получение информации о заказе на ремонт и его деталях:

```
SELECT ro.id, ro.date_time, ro.status, c.VIN, m.name AS model_name, b.name AS brand_name
FROM RepairOrders ro
JOIN Cars c ON ro.Cars_id = c.id
JOIN Models m ON c.Models_id = m.id
JOIN Brands b ON m.Brands_id = b.id
WHERE ro.id = 1;
```

5. Получение списка всех заказов на ремонт с количеством заказов для каждого клиента:

```
SELECT cl.name AS client_name, COUNT(ro.id) AS order_count
FROM Clients cl
LEFT JOIN Cars c ON cl.Cars_id = c.id
LEFT JOIN RepairOrders ro ON c.id = ro.Cars_id
GROUP BY cl.id;
```

Присваиваем значения переменным в транзакции.

```
START TRANSACTION;

-- Присваиваем значения переменным
SET @brand_id := 4;
SET @model_id := 4;
SET @vin_code := 'MNO46801357924680';

-- Обновляем бренд
UPDATE Brands SET name = 'Шевроле' WHERE id = @brand_id;
```



```

-- Обновляем модель
UPDATE Models SET name = 'Camaro', Brands_id = @brand_id, year = 2016, car_mileage =
18000 WHERE id = @model_id;

-- Вставляем новый автомобиль или обновляем существующий
INSERT INTO Cars (VIN, Models_id) VALUES (@vin_code, @model_id)
ON DUPLICATE KEY UPDATE Models_id = @model_id;

COMMIT;

```

Данный SQL запрос используется для обновления информации о машине в базе данных. Он начинается с транзакции, что обеспечивает целостность данных. Затем он обновляет информацию о бренде и модели, связывая их с указанным VIN-кодом. Если запись с таким VIN-кодом уже существует, то обновляется информация о модели, иначе создается новая запись. После всех изменений транзакция фиксируется.

Хранимая процедура

```

DELIMITER //

CREATE PROCEDURE UpdateCarInfo(
    IN brandName VARCHAR(45),
    IN modelName VARCHAR(45),
    IN vinCode CHAR(17)
)
BEGIN
    -- Проверяем, существует ли бренд в базе данных
    DECLARE brand_id INT;
    DECLARE model_id INT;

    SELECT id INTO brand_id FROM Brands WHERE name = brandName;
    IF brand_id IS NULL THEN

```

```

-- Если бренда нет, вставляем новый
INSERT INTO Brands (name) VALUES (brandName);
SET brand_id := LAST_INSERT_ID();
END IF;

-- Проверяем, существует ли модель в базе данных
SELECT id INTO model_id FROM Models WHERE name = modelName AND Brands_id =
brand_id;
IF model_id IS NULL THEN
-- Если модели нет, вставляем новую
INSERT INTO Models (name, Brands_id, year, car_mileage) VALUES (modelName,
brand_id, 0, 0);
SET model_id := LAST_INSERT_ID();
END IF;

-- Вставляем новый автомобиль или обновляем существующий
INSERT INTO Cars (VIN, Models_id) VALUES (vinCode, model_id)
ON DUPLICATE KEY UPDATE Models_id = model_id;
END //

DELIMITER ;

```

Данная хранимая процедура используется для обновления информации о машине в базе данных. Она начинается с проверки существования бренда в базе данных. Если бренд не существует, производится вставка нового бренда. Затем процедура проверяет существование модели в базе данных. Если модель не существует, вставляется новая модель. После этого вставляется новая запись об автомобиле или обновляется существующая запись [10].

Представление

```

CREATE VIEW RepairOrdersView AS
SELECT
RO.id AS repair_order_id,

```

```

RO.status AS repair_status,
RO.date_time AS repair_date,
C.name AS client_name,
C.contact_info AS client_contact_info,
C.driver_license AS client_driver_license,
C.phone_number AS client_phone_number,
CA.VIN AS car_VIN,
M.name AS car_model,
B.name AS car_brand,
O.amount AS order_amount,
O.status AS order_status,
P.name AS part_name,
P.quantity AS part_quantity
FROM
RepairOrders RO
JOIN Clients C ON RO.Cars_id = C.Cars_id
JOIN Cars CA ON RO.Cars_id = CA.id
JOIN Models M ON CA.Models_id = M.id
JOIN Brands B ON M.Brands_id = B.id
JOIN Orders O ON RO.id = O.RepairOrders_id
JOIN Parts P ON RO.id = P.RepairOrders_id;

```

Данное представление служит для отображения информации о заказах на ремонт. Оно объединяет данные из различных таблиц, предоставляя информацию о заказах на ремонт, клиентах, их автомобилях, а также деталях заказов и используемых запчастях.

Триггер

```

DELIMITER //

CREATE TRIGGER UpdateRepairOrderStatus AFTER INSERT ON Orders FOR EACH ROW
BEGIN
    DECLARE total_amount DECIMAL(10,2);

```

```

-- Получаем общую сумму заказа
SELECT SUM(amount) INTO total_amount FROM Orders WHERE RepairOrders_id =
NEW.RepairOrders_id;

-- Проверяем, превышает ли общая сумма заказа пороговое значение
IF total_amount > 50000 THEN
    -- Обновляем статус заказа на "completed"
    UPDATE RepairOrders SET status = 'completed' WHERE id = NEW.RepairOrders_id;
END IF;
END;
//

DELIMITER ;

```

Данный триггер обновляет статус заказа на ремонт, если статус заказа изменился на "оплачен".

Пользовательская функция

```

DELIMITER //

CREATE FUNCTION Calculate(order_id INT) RETURNS DECIMAL(10,2)
DETERMINISTIC
BEGIN
    DECLARE total_amount DECIMAL(10,2);

    -- Получаем общую сумму заказа
    SELECT SUM(amount) INTO total_amount FROM Orders WHERE RepairOrders_id =
order_id;

    RETURN total_amount;
END;
//

```

```
DELIMITER ;
```

Данная функция предназначена для вычисления общей суммы заказа на основе переданного идентификатора заказа на ремонт.

Обработчик исключений

```
DELIMITER //
```

```
CREATE PROCEDURE InsertClient(
```

```
    IN client_name VARCHAR(255),
```

```
    IN contact_info TEXT,
```

```
    IN cars_id INT,
```

```
    IN driver_license VARCHAR(50),
```

```
    IN phone_number CHAR(20),
```

```
    IN comments TEXT
```

```
)
```

```
BEGIN
```

```
    DECLARE existing_count INT;
```

```
    -- Проверяем, существует ли запись с таким идентификатором в таблице Cars
```

```
    SELECT COUNT(*) INTO existing_count FROM Cars WHERE id = cars_id;
```

```
    -- Если запись не существует, вызываем исключение
```

```
    IF existing_count = 0 THEN
```

```
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Ошибка: Автомобиль с  
указанным ID не найден.';
```

```
    ELSE
```

```
        -- Если запись существует, вставляем нового клиента
```

```
        INSERT INTO Clients (name, contact_info, Cars_id, driver_license, phone_number,  
`comments`)
```

```
        VALUES (client_name, contact_info, cars_id, driver_license, phone_number, comments);
```

```
    END IF;
```

```
END;
```

```
//
```

```
DELIMITER ;
```

Данная хранимая процедура предназначена для вставки новой записи о клиенте в базу данных. Процедура принимает параметры: имя клиента, контактную информацию, идентификатор автомобиля, номер водительского удостоверения, номер телефона и комментарии.

3NF (третья нормальная форма)

```
---
```

```
CREATE TABLE IF NOT EXISTS Brands (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(45) UNIQUE  
) ENGINE=InnoDB;
```

```
-- Таблица Models
```

```
CREATE TABLE IF NOT EXISTS Models (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(45),  
    Brands_id INT,  
    year INT,  
    car_mileage INT,  
    FOREIGN KEY (Brands_id) REFERENCES Brands(id)  
) ENGINE=InnoDB;
```

```
-- Таблица Cars
```

```
CREATE TABLE IF NOT EXISTS Cars (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    VIN CHAR(17) UNIQUE,  
    Models_id INT,  
    FOREIGN KEY (Models_id) REFERENCES Models(id)
```

```
) ENGINE=InnoDB;  
DELIMITER //  
````
```

Ознакомьтесь с полным кодом, представленным в Приложении 2.

## **Заключение**

В ходе данной курсовой работы была разработана база данных для автосервиса с особым акцентом на учет текущих клиентов и обеспечение комфортного взаимодействия с ними. Основной целью проекта было не только создание инструмента для управления данными, но и обеспечение пользователей современного автосервиса возможностью эффективно и удобно взаимодействовать с услугами предприятия. В процессе разработки базы данных учитывались реальные потребности и особенности автосервиса, что было достигнуто благодаря проведенным бизнес-встречам с владельцем автосервиса. Анализ данных привел к определению необходимых полей, их структуры и типов, а также к построению эффективных связей между объектами базы данных. Результатом этой работы стала функциональная база данных, реализованная в среде управления базами данных MySQL, а также набор командных файлов, предназначенных для управления базой данных и обеспечения ее бесперебойной работы.

В рамках данной работы были успешно выполнены следующие задачи, нацеленные на создание совершенной базы данных для автосервиса:

1. Глубокое изучение функциональной структуры предприятия автосервиса для выявления основных процессов и потребностей.
2. Тщательный анализ рабочих процессов предприятия с акцентом на учет клиентов и заказов, что позволило выделить ключевые аспекты для успешной реализации проекта.
3. Подбор наилучшей технологии СУБД с учетом специфики автосервиса, обеспечивающей оптимальное соответствие требованиям и потребностям бизнеса.
4. Разработка структуры базы данных с определением таблиц, их взаимосвязей, атрибутов и ключевых полей, что сформировало надежный фундамент для хранения информации.



5. Создание функциональной базы данных, ориентированной на эффективный учет клиентов и заказов, что способствует повышению операционной эффективности автосервиса.
6. Проведение тщательного тестирования разработанной базы данных с целью обеспечения ее надежной работы и соответствия высоким стандартам качества.

## Список литературы

1. Автосервис. Менеджмент в автосервисе. Организация работы и управление персоналом / Сидоров В.П. – СПб.: Питер, 2020. – 29 - 416 с.
2. Автосервис. Основы автосервиса. Технологии обслуживания и ремонта / Ковалев А.Н. – М.: Издательский центр "Академия", 2019. – 56 - 320 с.
3. Диагностика и ремонт автомобилей. Современные технологии / Григорьев И.И. – М.: Транспорт, 2018. – 400 - 480 с.
4. Управление автосервисом [Электронный ресурс] - Режим доступа: <https://autodealer.ru/blog/upravlenie-uchyot-analiz-v-avtoservise>
5. Экономика автосервиса. Финансовый учет и анализ / Петров Н.А. – М.: Издательство "Экономика", 2019. – 199 - 384 с.
6. MySQL. Библия пользователя / Форт М. – СПб.: БХВ-Петербург, 2019. – 18-720 с.
7. MySQL. Основы. Программирование. Примеры / Дюрант П. – М.: ДМК Пресс, 2018. – 400 - 416 с.
8. MySQL. Учебное пособие / Джаде К. – М.: Ленанд, 2017. – 256 - 352 с.
9. MySQL. Эффективное использование / Шварц С. – М.: Вильямс, 2018. – 789 - 832 с.
10. MySQL 8 Рецепты программирования / Уэллс П. – М.: Питер, 2020. – 525 - 560 с.

## Приложение 1

QR-кода со ссылкой на репозиторий



Ссылка для ручного ввода:

<https://github.com/Rom1z/Mydb>

## Приложение 2

-- Убедимся, что таблицы находятся в третьей нормальной форме

-- Таблица Brands

```
CREATE TABLE IF NOT EXISTS Brands (
 id INT AUTO_INCREMENT PRIMARY KEY,
 name VARCHAR(45) UNIQUE
) ENGINE=InnoDB;
```

-- Таблица Models

```
CREATE TABLE IF NOT EXISTS Models (
 id INT AUTO_INCREMENT PRIMARY KEY,
 name VARCHAR(45),
 Brands_id INT,
 year INT,
 car_mileage INT,
 FOREIGN KEY (Brands_id) REFERENCES Brands(id)
) ENGINE=InnoDB;
```

-- Таблица Cars

```
CREATE TABLE IF NOT EXISTS Cars (
 id INT AUTO_INCREMENT PRIMARY KEY,
 VIN CHAR(17) UNIQUE,
 Models_id INT,
 FOREIGN KEY (Models_id) REFERENCES Models(id)
) ENGINE=InnoDB;
```

DELIMITER //

```
CREATE PROCEDURE UpdateInfo(
 IN brandName VARCHAR(45),
 IN modelName VARCHAR(45),
 IN vinCode CHAR(17)
)
BEGIN
```

```

-- Проверяем, существует ли бренд в базе данных
DECLARE brand_id INT;
DECLARE model_id INT;

SELECT id INTO brand_id FROM Brands WHERE name = brandName;
IF brand_id IS NULL THEN
 -- Если бренда нет, вставляем новый
 INSERT INTO Brands (name) VALUES (brandName);
 SET brand_id := LAST_INSERT_ID();
END IF;

-- Проверяем, существует ли модель в базе данных
SELECT id INTO model_id FROM Models WHERE name = modelName AND Brands_id =
brand_id;
IF model_id IS NULL THEN
 -- Если модели нет, вставляем новую
 INSERT INTO Models (name, Brands_id, year, car_mileage) VALUES (modelName,
brand_id, 0, 0);
 SET model_id := LAST_INSERT_ID();
END IF;

-- Вставляем новый автомобиль или обновляем существующий
INSERT INTO Cars (VIN, Models_id) VALUES (vinCode, model_id)
ON DUPLICATE KEY UPDATE Models_id = model_id;
END //

DELIMITER ;

```