Департамент образования, науки и молодежной политики Воронежской области

Государственное бюджетное профессиональное образовательное учреждение

Воронежской области «Острогожский многопрофильный техникум»

|  |  |
| --- | --- |
|  |  |

**КУРСОВАЯ РАБОТА**

по МДК 03.01 «Технология разработки программного обеспечения»

по ПМ 03 «Участие в интеграции программных модулей»

по теме: «Проектирование и разработка приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами»

Выполнена студентом: **Путилиным Дмитрием Михайловичем**

Группа **К-41**

Основная профессиональная образовательная программа по специальности

**09.02.03 «Программирование в компьютерных системах»**

Руководитель – преподаватель: **Голдинов Виктор Семенович**

Острогожск 2021

СОДЕРЖАНИЕ

[Введение 3](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883389)

[1. Разработка системного проекта 5](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883390)

[1.1. Анализ предметной области приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами 5](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883391)

[1.2. Требования к программному продукту приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами 5](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883392)

[1.3. Сравнительный анализ существующих программных продуктов 7](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883397)

[2. Практическая реализация проекта 8](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883398)

[2.1. Постановка задачи 8](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883399)

[2.2. Выбор средств и технологии создания приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами 8](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883400)

[2.3. Моделирование предметной области приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами 1](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883401)1

[2.4. Алгоритм функционирования приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами 17](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883402)

[2.5. Описание работы приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами 18](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883403)

[2.6. Тестирование приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами предприятии](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883404) 28

[Заключение](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883405) 27

[Список используемых источников 3](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883406)0

[Приложения 3](file:///C:\Users\User\Desktop\курсац%204%20курс\Kursovaya_Ivanova.docx#_Toc515883407)2

ВВЕДЕНИЕ

В современном обществе количество информации возрастает в геометрической прогрессии. Традиционные инструменты перестали покрывать требования обработки и хранения информации. На помощь приходят информационные технологии, которые значительно повышают надежность хранения и скорость обработки информации.

При современном уровне развития вычислительной техники и средств появилась возможность создания автоматизированных систем управления. Они включают в себя множество автоматизированных рабочих мест (АРМ) сотрудников, средства коммуникации и обмена информацией, другие средства и системы, позволяющие автоматизировать работу управленческого персонала.

Создание автоматизированных рабочих мест позволяет эффективно обрабатывать большие потоки информации, которые имеют определенную структуру, зависящую от особенностей места применения. Это позволяет осуществлять индивидуальный подход к автоматизации именно тех функций, которые выполняются данным подразделением. Введение на предприятии автоматизированных рабочих мест позволяет значительно сократить время выполнения работ и повысить их точность, облегчить труд специалистов.

Под экономическим эффектом подразумевается получение реальной экономической отдачи от использования всей системы или ее отдельных функциональных блоков.

Автоматизированные рабочие места применяются в различных сферах деятельности. В данной курсовой работе произведена разработка программного продукта для учета и контроля работы техников в сети провайдера для менеджера по работе с клиентами.

**Актуальность темы** исследования заключается в том, что программный продукт позволяет вести учет клиентов, заполнение заявок на выполнение обслуживания, контролировать ход выполнения работ.

**Объект исследования**- приложение по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами.

**Предмет исследования** - проектирование и разработка приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами.

**Цель исследования** заключается в разработке качественного, многофункционального программного продукта для учета и контроля работы техников в сети провайдера для менеджера по работе с клиентами.

Для достижения поставленной цели необходимо решить следующие задачи:

* изучить деятельность организации;
* анализ литературных источников;
* провести сравнительный анализ уже известных приложений для учета продукции;
* выбрать инструменты для разработки программного продукта;
* спроектировать модели данных;
* разработать приложение для учета и контролю работы техников в сети провайдера;
* протестировать и разработать руководство пользователя.

**Методы исследования -** изучение литературы, сравнение, анализ существующих подходов к разработке приложения, непосредственная разработка программного продукта для информационной системы.

**Практическая значимость** данной работы состоит в разработке работоспособного приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами, обработки и вывода информации.

**Структура работы** включает в себя введение, разработку системного проекта, его реализацию, заключение, список использованных источников и приложения.

# **РАЗРАБОТКА СИСТЕМНОГО ПРОЕКТА**

## **1.1. Анализ предметной области приложения учету и контролю работы техников в сети провайдера**

Под понятием предметная область понимают фрагмент реальности, который описывается или моделируется с помощью БД и ее приложений. В предметной области выделяется информационные объекты идентифицирующие объекты реального мира, процессы, системы, понятия и т.д., сведения о которых хранятся в БД.

Областью применения приложения является интернет провайдер. Предприятие работает с очень большим объемом информации, как о сотрудниках и клиентах, так и о выполненной работе. Сотрудникам необходимо всегда следить за данными. Для этого нужна общая база данных, включающая всю необходимую информацию.

Данная организация занимается следующими видами деятельности:

* + Подключение пользователей к сети интернет;
  + Устранение неполадок в сети;
  + Установка сетевого оборудование;
  + Настройка сетевого оборудования;
  + Продажа или выдача оборудования в аренду;

Проектируемая система учета заказов предназначена для использования интернет провайдером. В её функции входит

Ежедневно в организацию поступают десятки заявок. Разработанное приложение способно автоматизировать и ускорить процесс обработки заявок, а также позволяет печатать отчеты и накладные.

**1.2. Требования к программному продукту**

**Требования к функциональным характеристикам**

Разработанные компоненты должны выполнять следующие функции:

* функционирование под управлением операционных систем семейства MsWindows;
* предоставление пользователю простого и удобного интерфейса;
* наличие у компонентов интерфейса связи с графическими компонентами и друг с другом.

### Требования к надежности и защите информации

Требования к надежности и безопасности программы:

* срок службы разработанных компонентов не ограничен;
* программа должна работать с абсолютно корректными данными;
* программа должна поддерживать диалоговый режим;

# **Требования к составу и параметрам технических средств. Системные требования**

Разрабатываемые компоненты должны быть совместимы с некоторыми уже существующими компонентами, стабильно вести себя в произвольной связке со смежными компонентами. Функционирование компонентов должно быть возможным под всеми операционными системами Windows (начиная от Windows 7 и выше).

Состав аппаратных средств и их основные характеристики:

* компьютер под управлением Windows 7/8/8.1/10;
* оперативная память 1024 мбайт и выше;
* мышь;
* клавиатура;
* монитор.

# **Требования к персоналу**

Каждый пользователь (администратор) данного программного продукта должен соответствовать ряду требований. Для этого были составлены требования к персоналу:

* ответственность при эксплуатации программного продукта;
* знание соответствующей предметной области;
* опыт работы с персональным компьютером;
* уверенный пользователь печатью;
* знания и навыки работы с принтером.

# **Требования к информационной и программной совместимости**

Безусловным требованием является полная достоверность всей информации. Разрабатываемые компоненты должны быть совместимы с некоторыми уже существующими компонентами, стабильно вести себя в произвольной связке со смежными компонентами.

Как итог, программный продукт должен полностью выполнять все требования, предусмотренные современными установками в области создания программ.

## **Сравнительный анализ существующих программных продуктов**

В настоящее время существует огромное количество приложений и каждый из них имеет свою уникальную особенность. К примеру, к таким приложениям относятся:

# **[Okdesk](https://www.livebusiness.ru/tool/1784/)**

# Облачное решение для автоматизации процессов поддержки и взаимодействия с юр.лицами в малых и средних сервисных компаниях. Обладает функциональностью Help Desk + CRM (учет и обработка заявок, учет договоров, клиентов и контактных лиц с индивидуальными условиями обслуживания, история взаимодействия). Бесплатный встроенный клиентский портал. Регистрация обращений по почте и с сайта. Переписка с клиентом. Функциональность отчетов и дашбордов для руководителей. Отображение заявок на карте.

**Мобифорс**

Простая CRM-система для эффективного управления выездными специалистами из разных сфер: сервис-инженеры, аварийные бригады, монтажники, аварийные комиссары, курьеры, экспедиторы, коллекторы. Наглядный контроль статусов доставки, прочтения, и выполнения задач, группировка завершенных задач по резолюциям позволяет понимать, что происходит как в процессе выполнения задач, так и после их завершения. Возможность после завершения задачи сотрудником переводить ее в статус «на проверке», который оператор может закрыть или отправить задачу на доработку.

# **2. Практическая реализация проекта**

## **2.1. Постановка задачи**

Целью курсовой работы является разработка проектирование и разработка приложения по учету и контролю работы техников в сети провайдера для менеджера по работе с клиентами

Разрабатываемое приложение должно включать в себя следующий основной функционал:

* + Хранение информации о сотрудниках;
  + Хранение информации о клиентах;
  + Учет выполненной работы;
  + Контроль выполненной работы;
  + Поиск нужной информации по базе;
  + Формирование отчетов.

## **2.2. Выбор средств и технологии создания программного продукта**

Современные редакторы программного кода, позаимствовали свой функционал у визуальных текстовых редакторов, а также у специализированных средств текстовой разработки, прилагаемых к различным компиляторам.

Для программирования приложений в интернете применяются специальные средства разработчиков, которые умеют работать с данными Visual Studio

**Visual Studio**

Это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Интегрированная среда разработки (IDE) представляет собой многофункциональную программу, которую можно использовать для различных аспектов разработки программного обеспечения. Помимо стандартного редактора и отладчика, которые существуют в большинстве сред IDE, Visual Studio включает в себя компиляторы, средства авто завершения кода, графические конструкторы и многие другие функции для упрощения процесса разработки.

**MySQL**

Очень быстрая, надежная система управления реляционными базами данных (СУРБД). База данных позволяет эффективно хранить, искать, сортировать и получать данные. Сервер MySQL управляет доступом к данным, позволяя работать с ними одновременно нескольким пользователям, обеспечивает быстрый доступ к данным и гарантирует предоставление доступа только имеющим на это право пользователям. Следовательно, MySQL является многопользовательским, многопотоковым сервером. Он применяет SQL, используемый по всему миру стандартный язык запросов для баз данных

**C#**

Это современный язык программирования, созданный компанией Microsoft. На нём можно писать программы любой сложности для любых платформ и операционных систем.

C# устроен так, чтобы программист мог писать **меньше кода.** Для этого в языке много «магических конструкций» и «синтаксического сахара» — способов выразить идею лаконично.

Это**кроссплатформенный язык,**то есть ваша программа будет работать не только в той операционной системе, в которой она написана, но и в других.

## **2.3. Моделирование предметной области**

Любое предприятие, учреждение или организация представляет собой совокупность различного рода ресурсов связанных между собой процессами. Ресурсами могут быть: рабочие и служащие, оборудование, информация, электроэнергия и т.д. Процесс – это последовательность действий необходимая для преобразования одного вида ресурса в другой. На процесс накладывается технология выполнения этого процесса. Технология в данном случае – это совокупность методов и способов выполнения этого процесса. В зависимости от используемой технологии конечный ресурс может обладать разными свойствами, т.е. быть разного качества.

С целью получения единого представления о взаимодействии ресурсов и процессов, их можно описать в терминах предметной области.

Предметной областью может быть область объектов, универсум рассуждения, универсум рассмотрения, или просто универсум, класс (множество) объектов, рассматриваемых в пределах данного контекста. Под контекстом здесь может пониматься отдельное рассуждение или выражающая его фраза, или совокупность фраз, фрагмент научной теории или теория в целом. Если мы говорим о работе автоматизированной системы управления предприятием, то предметной областью будет все это предприятие. Если мы говорим об автоматизации отдельной задачи, решаемой на этом предприятии, то предметной областью будет вся необходимая для решения этой задачи совокупность объектов, описывающих их свойств, взаимосвязей между этими объектами и процессов, в которых они участвуют.

Программные продукты имеют внутреннюю структуру (внутреннюю организацию), образованную взаимосвязанными программными модулями.

Модуль - это самостоятельная часть программы, имеющая определенное назначение и обеспечивающая заданные функции обработки независимо от других программных модулей.

Структуризация программ выполняется для удобства разработки, программирования, отладки и внесения изменений в программный продукт. Особенно это важно, когда программный продукт разрабатывается коллективом разработчиков.

Структуризация программных продуктов позволяет:

* распределить работы по исполнителям, обеспечив их загрузку и требуемые сроки разработки программных продуктов;
* построить календарные графики проектных работ и осуществлять их координацию в процессе создания программных продуктов;
* контролировать трудозатраты и стоимость проектных работ и др.

Среди множества модулей различают:

* головной модуль, который управляет запуском программного продукта (существует в единственном числе);
* управляющие модули, которые задают последовательность вызова и обеспечивают вызов других модулей на обработку;
* рабочие модули, выполняющие функции обработки;
* сервисные модули и библиотеки, утилиты, осуществляющие обслуживающие функции. Некоторые программные продукты используют модули из готовых библиотек стандартных подпрограмм, процедур, функций, объектов, методов обработки данных.

Информационная связь модулей обеспечивается за счет использования общей базы данных либо межмодульной передачи данных через переменные обмена.

Каждый модуль может оформляться как самостоятельно хранимый файл и для функционирования программного продукта необходимо наличие программных модулей в полном составе.

Спецификации должны однозначно восприниматься как заказ­чиком, так и разработчиком. Обеспечить это требование можно, только разработав некоторую формальную модель этого про­граммного обеспечения. На этапе анализа и определения специ­фикаций можно применить структурный подход. Построим функ­циональную диаграмму, для того чтобы выявить основные функ­ции и составные части проектируемой программной системы и, по возможности, обнаружить и устранить существенные ошибки. Одной из наиболее важных особенностей методологии функцио­нального моделирования является постепенное введение все боль­ших уровней детализации по мере создания диаграмм, отображающих модель.

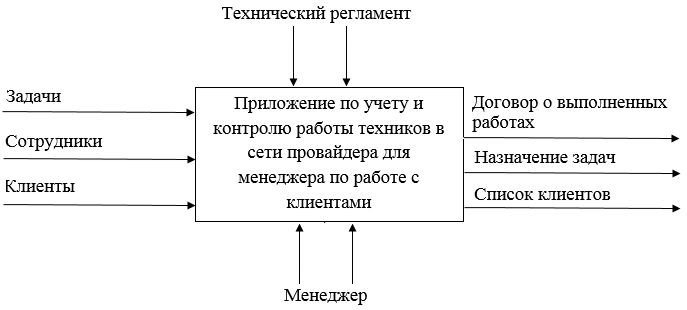


Рис.1. Функциональная диаграмма начального уровня

Диаграмма, показанная на рис. 1, представляет собой диаграмму верхнего уровня. На ней хорошо видно, что служит исходными данными для программы и получения, каких результатов мы ожидаем.

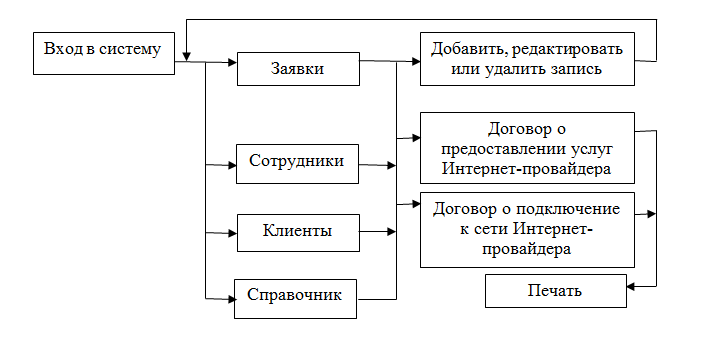


Рис. 2. Функциональная диаграмма нулевого уровня

Диаграмма на (рис. 2) уточняет функции программы: редактирование данных в таблицах; выход на печать договоров.

Теперь необходимо разработать диаграмму «сущность - связь» (ЕR - модель данных), которая обеспечивает способ определения данных и отношений между ними. Модель данных включает сущности и связи между ними. Диаграммы «сущность - связь» в отличие от функциональных диаграмм определяют спецификации структур данных программного обеспечения.

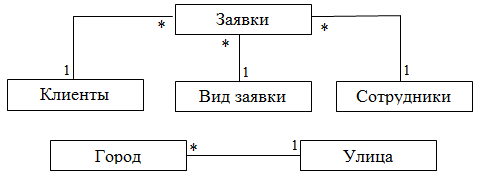


Рис. 3. Диаграмма «Сущность - Связь» для описания базы данных

На рисунке 3 показана связь между субъектом (системным администратором) и программой.

На следующем шаге определяем атрибуты каждой сущности и уточняем их типы (атрибуты, используемые для дополнительной идентификации сущности другой сущностью, не указываются, так как они описываются в соответствующей сущности). Теперь можно внести все это в диаграмму (рис. 4).

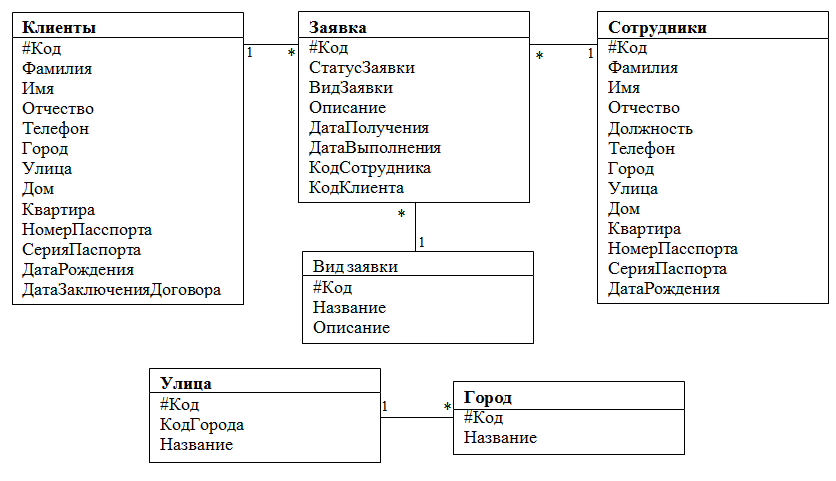


Рис. 4. ER-диаграмма базы данных

Физическое проектирование является вторым и последним этапом создания проекта базы данных, при выполнении которого проектировщик принимает решения о способах реализации разрабатываемой БД. Приступая к физическому проектированию БД, необходимость конкретную целевую СУБД. Основной целью физического проектирования базы данных является описание способа физической реализации логического проекта БД. В случае реляционной модели БД под этим подразумевается следующее:

* создание набора реляционных таблиц и ограничений для них на основе информации, представленной в глобальной логической модели данных;
* определение конкретных структур хранения данных и методов доступа к ним, обеспечивающих оптимальную производительность СУБД.

Разработанная модель находится в 3-й нормальной форме, так как:

* атрибуты сущностей являются атомарными;
* каждый не ключевой атрибут функционально полно зависит от первичного ключа.

В модели отсутствуют транзитивные зависимости не ключевых атрибутов от ключа.

**Детальное проектирование программного обеспечения**

Детальное проектирование - это техническая деятельность, которая следует за выбором архитектуры. Основной целью этой деятельности является как можно более полная подготовка проекта к его реализации. Другими словами, программисты должны иметь возможность реализовать детальный проект, сконцентрировавшись только на проблемах, связанных с созданием программного кода.

Детальное проектирование программного обеспечения включа­ет в себя разработку структурной схемы (рис. 5), которая дает достаточ­но полное представление о проектируемом программном обеспе­чении и начинается по результатам фазы архитектуры и завершается с моментом получения полного плана для фазы программирования.

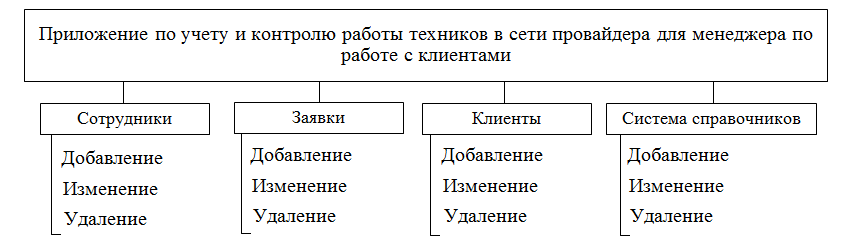


Рис. 5. Структурная схема приложения

На рисунке 5 показана схема основных функций необходимых для работы приложения. В результате проведения детального проектирования план проекта становится более конкретным во многих отношениях. Разработка внутренней структуры программы, пользовательского интерфейса, способа отображения информации.

## **2.4. Алгоритм функционирования программного продукта**

Проектирование алгоритмов и программ - ответственный этап жизненного цикла программных продуктов, определяющий, насколько создаваемая программа соответствует спецификациям и требованиям со стороны конечных пользователей.

Решение задач на компьютере основано на понятии алгоритма. Алгоритм – это точное предписание, определяющее вычислительный процесс, ведущий от варьируемых начальных данных к исходному результату.

Алгоритм означает точное описание некоторого процесса, инструкцию по его выполнению. Разработка алгоритма является сложным и трудоемким процессом. Алгоритмизация – это техника разработки (составления) алгоритма для решения задач на ЭВМ.

Разработка алгоритмов функционирования представляет собой создание модели работы, функционирования программы как единой системы. Именно на данном этапе в проект вкладываются те ограничения, которые могут в будущем мешать при необходимости расширения функциональности, созданной и уже работающей программы. Поэтому это очень важный этап разработки программного обеспечения.

Для начала разрабатывается алгоритм с точным предписанием, определяющим вычислительный процесс, ведущий от исходных данных к искомому результату.

Этап планирования – это то, что можно назвать составление технического задания для приложения. Т.е. эта первая часть технического задания, в котором планируется, что должно быть в проекте.

На первом этапе был разработан общий вид программного продукта.

После чего, происходит оформление скелета проекта по заготовленному ранее шаблону. На данном этапе. Сначала создается каркас, то, что будет расположено на форме (кнопки, надписи, заголовки, компоненты). После, применяются стили (цвета, позиции, шрифты, фон).

После того, как внешний вид был выполнен в соответствии с макетом, идет этап решения задачи. Средствами языка программирования разрабатываются функции, процедуры для работы программного продукта. На рисунке 6 представлена блок – схема алгоритма работы программного продукта «учет заказов и доставки лекарственных средств».

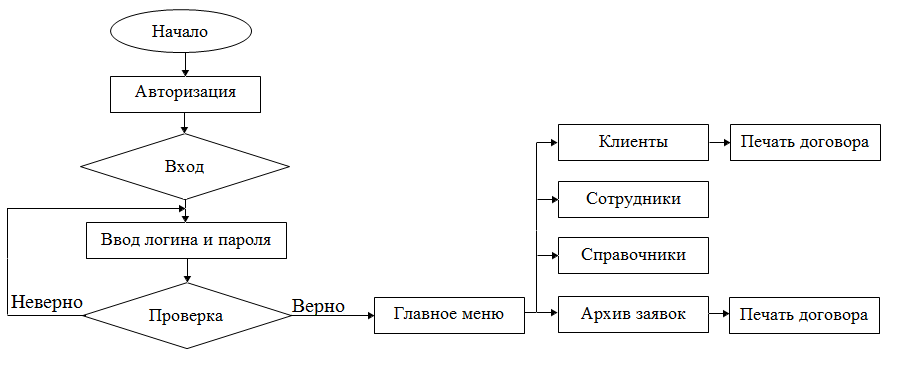


Рис. 6. Алгоритм функционирования функционала «учета и контроля работы техников в сети провайдера для менеджера по работе с клиентами»

## **2.5. Описание работы программного продукта**

Приложение реализуется средствами Visual Studio. При запуске мы видим окно авторизации.

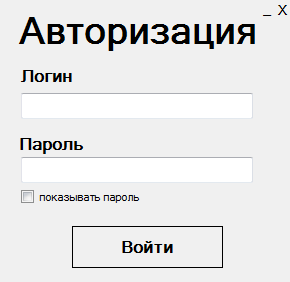


Рис. 7. Форма «Авторизация»

При вводе неправильного пароля появляется сообщение(рис.8).

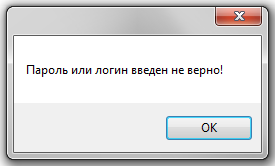


Рис. 8. «Не верный пароль»

При введении правильного пароля открывается главная форма(рис.9)., здесь находятся кнопки, которые позволяют переходить на другие формы.

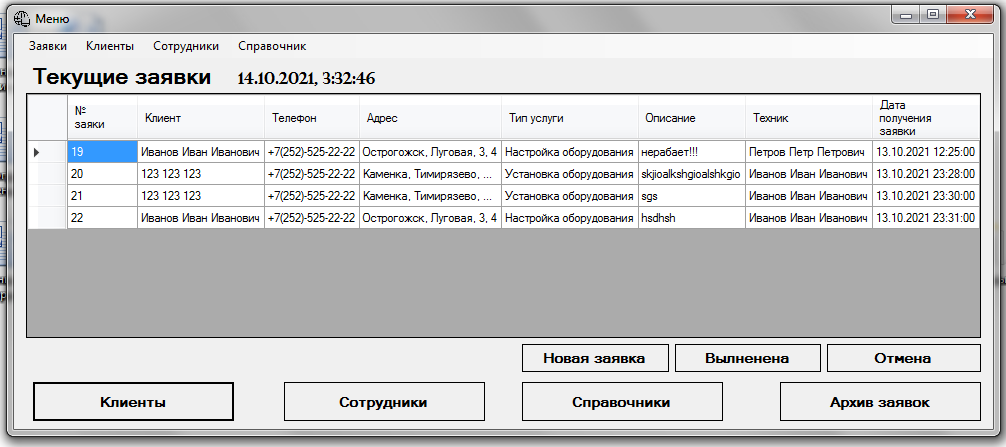


Рис. 9. Форма «Главное меню»

На рисунке 10 показана форма «Клиенты» на которой мы можем добавлять, удалять и редактировать список клиентов, на данной форме также находятся фильтры для быстрого поиска.

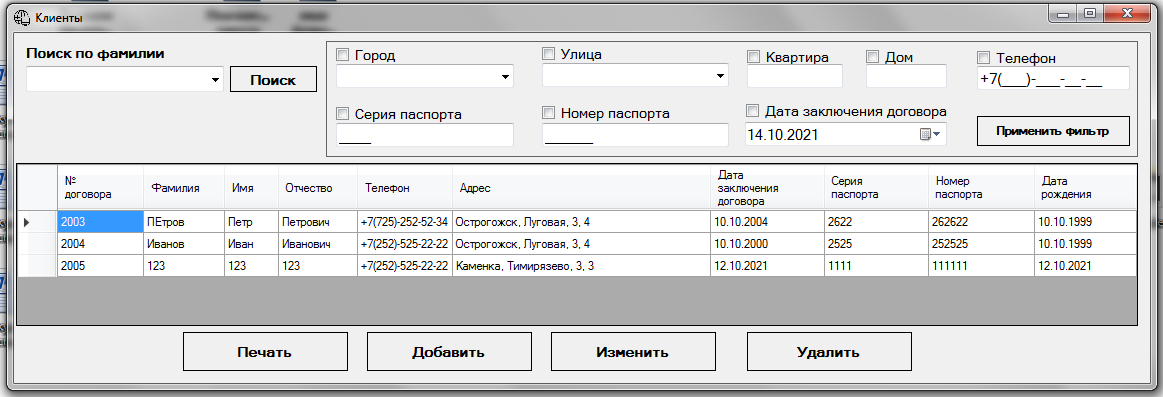


Рис. 10. Форма «Клиенты»

При нажатии кнопки «Добавить» открывается форма, на которой можно добавить нового клиента(рис. 11).

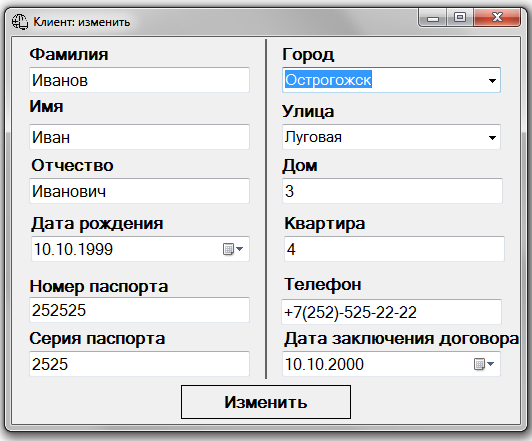


Рис. 11. Форма «Клиент: добавить»

