МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БАШКОРТОСТАН

Государственное бюджетное профессиональное образовательное учреждение

Уфимский колледж статистики, информатики и вычислительной техники

ОТЧЁТ

по преддипломной практике

по специальности 09.02.03 Программирование в компьютерных системах

|  |  |
| --- | --- |
| Заведующий ЦПК  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Б. М. Ибраев/  подпись ФИО | Руководитель практики  от предприятия  \_\_\_\_\_\_\_\_\_\_\_\_/ \_\_\_\_\_\_\_\_\_\_\_\_/  подпись ФИО  М.П. |
| Руководитель практики  от учебного заведения  \_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Р.С. Зиганшина/  подпись ФИО | Студент группы 16П-2  \_\_\_\_\_\_\_\_\_\_\_\_/ Н.С. Батурин/  подпись ФИО  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020 год |

УФА – 2020 год

Министерство образования и науки Республики Башкортостан

Государственное бюджетное профессиональное образовательное учреждение

Уфимский колледж статистики, информатики и вычислительной техники

|  |  |
| --- | --- |
|  | УТВЕРЖДАЮ  Заместитель директора  по учебной работе  З.З. Курмашева  «15» мая 2020 год |

ЗАДАНИЕ

на дипломный проект студенту дневного отделения, группы 16П-2, специальности 09.02.03 Программирование в компьютерных системах.

Фамилия, имя, отчество: Батурин Никита Сергеевич

Тема дипломного проекта: «Автоматизация учета гарантийного оборудования в сервисном центре».

Текст задания:

при выполнении дипломного проекта должны быть решены следующие задачи:

1. разработана структура программы;
2. реализованы функции создания и изменения заказов;
3. печать оформленных актов;
4. построена справочная система.

В результате выполнения дипломного проекта должны быть представлены:

1. пояснительная записка, состоящая из следующих разделов:

Введение

1 Постановка задачи

2 Экспериментальный раздел

3 Экономический раздел

Заключение

Приложения

Список сокращений

Список использованных источников

1. электронный носитель, содержащий разработанный программный продукт;
2. презентация дипломного проекта в электронном виде.

Список рекомендуемых источников:

1. ГОСТ 19.701-90 ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения. [Текст]. – Введ. 1992–01–01 -М.: Изд-во стандартов, 1992.- 24 с.
2. [Агальцов В. П.](http://znanium.com/catalog.php?item=booksearch&code=%D0%B1%D0%B0%D0%B7%D1%8B+%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85&page=1#none) Базы данных. В 2-х кн. Кн. 2. Распределенные и удаленные базы данных: учеб. / В.П. Агальцов. - М.: ИД ФОРУМ: НИЦ Инфра-М, 2013. - 272 с.: ил. - <http://znanium.com/bookread2.php?book=372740>
3. Википедия [Электронный ресурс] // Свободная энциклопедия. – Режим доступа: <http://ru.wikipedia.org/wiki/>, свободный
4. Эванс, Э. Предметно-ориентированное программирование. Структуризация сложных программных систем [Текст]: / Э. Эванс - М.: Вильямс, 2018. – 448 с.

Задание к выполнению получил «30» марта 2020г.

Студент Батурин Никита Сергеевич

Срок окончания «08» июня 2020 г.

Руководитель дипломного проекта Р.С. Зиганшина

Задание рассмотрено на заседании цикловой комиссии информатики   
«25» марта 2020 г. Протокол №5.

Председатель цикловой комиссии информатики О.В. Фатхулова

Изм.

Лист

№ докум.

Подпись

Дата

Лист

2

40.Б-1034-2020 09.02.03 ОП

Разраб.

Батурин Н.С.

Провер.

Зишаншина Р.С.

Реценз.

*Айс К.М.*

Н. Контр.

*Каримова Р.Ф*

Утверд.

*Курмашева З.З*

Автоматизация учета гарантийного оборудования в сервисном центре

Лит.

Листов

37

УКСИВТ 16П-2

АННОТАЦИЯ

Пояснительная записка к дипломному проекту содержит постановку и программу решения задачи «Автоматизация учета гарантийного оборудования в сервисном центре».

Программа service\_center.exe написана на языке C# в среде программирования Microsoft Visual Studio 2019 с использованием сервера баз данных MySQL Server, предназначена для работы в операционной системе MS Windows 10 и отлажена на данных контрольного примера.

СОДЕРЖАНИЕ

|  |  |
| --- | --- |
|  | лист |
| Введение | 4 |
| 1 Постановка задачи | 6 |
| 1.1 Описание предметной области | 6 |
| 1.2 Функциональная модель предметной области | 7 |
| 1.3 Описание входной информации | 8 |
| 1.4 Описание выходной информации | 8 |
| 1.5 Общие требования к программному продукту | 9 |
| 1.6 Описание структуры базы данных | 10 |
| 1.7 Контрольный пример | 13 |
| 2 Экспериментальный раздел | 16 |
| 2.1 Описание программы | 16 |
| 2.2 Протокол тестирование программного продукта | 19 |
| 2.3 Руководство пользователя | 31 |
| Заключение | 35 |
| Список использованных источников | 36 |

# 

# ВВЕДЕНИЕ

В современном мире информация рассматривается, как один из основных компонентов развития общества, а информационные системы и технологии используются в производственной, управленческой, финансовой и многих других профессиональных отраслях.

Отыскание рациональных решений в любой сфере требует больших финансовых и временных ресурсов. А обработка массивного объема информации невозможна без привлечения специальных технологических средств.

Внедрение информационных технологий, современных средств переработки и передачи информации в различные сферы деятельности послужило началом информатизации, которая является реакцией общества на потребность в существенном увеличении производительности труда в различных сферах деятельности человека, а также ее автоматизация и учета, что позволяет облегчить работу ручного труда.

Для успешного развития бизнеса необходимо решить проблемы учета приема и выполнения заявок клиентов.

Целью данной дипломной работы является разработка информационной системы для учета гарантийного оборудования в сервисном центре.

В связи с этим задачами будут являться:

* изучение структуры учреждения, выявление его основных задач и функций;
* анализ информационных процессов, протекающих в учреждении;
* выявление функциональных задач информационной системы;
* установить процессы требующие автоматизации;
* разработка базы данных;
* разработка приложения для работы с базой данных.

Актуальность данной дипломной работы заключается в экономической целесообразности использования денежных ресурсов фирмы заказчика, а также в последующей поддержки программного продукта для исправления непредвиденных неисправностей, модернизации функциональности и реализации новых требований информационной системы.

# 1 Постановка задачи

# Описание предметной области

В сервисный центр по ремонту гарантийного оборудования от клиентов поступают на ремонт устройства различных типов, производителей и моделей в неисправном состоянии их объединяет дистрибьютер, продавший технику клиенту.

Прием оборудования в ремонт осуществляет приемщик – сотрудник сервисного центра. Он регистрирует новые заказы. Для этого с клиента требуется: чек покупки неисправного товара; выдаваемый вместе с чеком гарантийных талон, на котором должен продолжаться период гарантийного обслуживания; ФИО и мобильный телефон клиента.

После регистрации заказа печатается акт приема, который выдается клиенту и является документом, подтверждающим факт приема изделия сервисным центром для ремонта.

Мастер сервисного центра принимает заказ, производит диагностику изделия, выявляет имеющиеся неисправности и целесообразность дальнейшего ремонта. При возможности осуществления ремонта производится ремонт изделия. Мастер проводит необходимые ремонтные работы для устранения неисправности, и вносит в журнал выполненных работ описание выполненных работ, а также отметку о текущем статусе заказа:

* проводится диагностика;
* диагностика проведена, ремонт невозможен;
* диагностика проведена, ремонт возможен;
* диагностика проведена, не гарантийный случай;
* ремонт проведен успешно.

При успешном завершении ремонта клиент ставится в известность об окончании ремонта. Далее клиенту возвращается изделие, и в заказе фиксируется дата выдачи. Заказу присваивается статус «изделие выдано клиенту».

В случае не гарантийной неисправности клиент также ставится в известность и ему требуется забрать изделие из сервисного центра.

# Функциональная модель предметной области

AS-IS – модель «как есть», модель существующего состояния организации. Данная модель позволяет систематизировать протекающие в данный момент процессы, а также используемые информационные объекты. На основе этого выявляются узкие места в организации и взаимодействии бизнес-процессов, определяется необходимость тех или иных изменения в существующей структуре.

Функциональная модель представлена на рисунке 1.2.1.

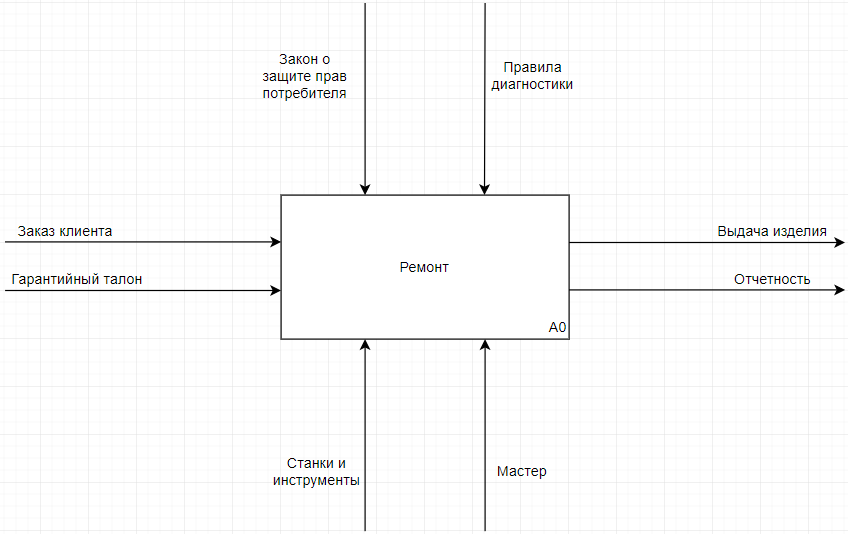


Рисунок 1.2.1 – Функциональная модель

Декомпозиция функциональной модели представлена на рисунке 1.2.2.

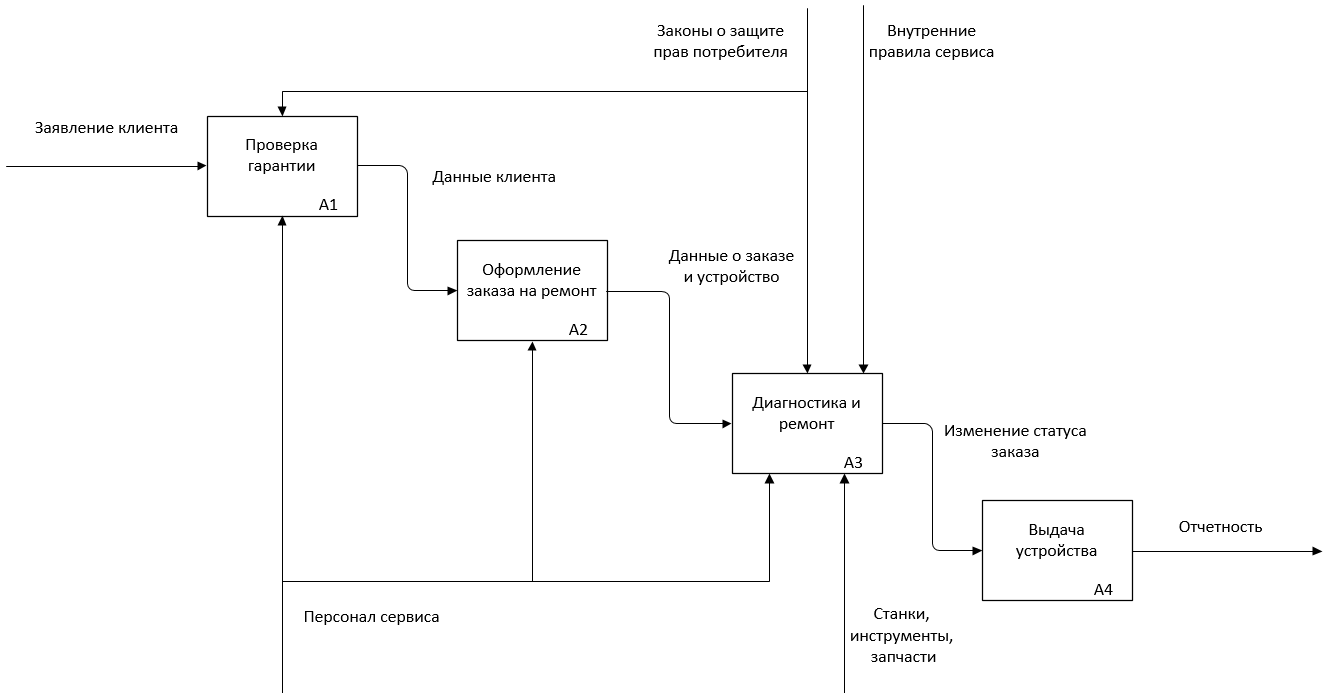


Рисунок 1.2.2 – Декомпозиция функциональной модели

# 1.3 Описание входной информации

Входным документом для задачи будет являться заявление клиента на ремонт, описание которого приводится в таблице 1.3.1.

Таблица 1.3.1 – Описание входных документов

|  |  |  |
| --- | --- | --- |
| Наименование документа (шифр) | Периодичность поступления документа | Откуда поступает документ |
| Заявление клиента | При поступлении | От клиента |

# 1.4 Описание выходной информации

Выходными документами будут являться сформированные акт приема и акт выдачи, в которых указаны данные по заказу.

В таблице 1.4.1 представлено описание выходных документов.

Таблица 1.4.1 – Описание выходных документов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Наименование документа  (шифр) | Периодичность выдачи документа | Кол-во экз. | Куда передаются | Поля сортировки | Поля группировки | Итоги |
| Акт приема | При приеме | 1 | Клиенту | Статус | – | – |
| Акт выдачи | При выдачи | 1 | Клиенту | Статус | – | – |

Формы выходных документов представлены на рисунках 1.4.1 и 1.4.2.

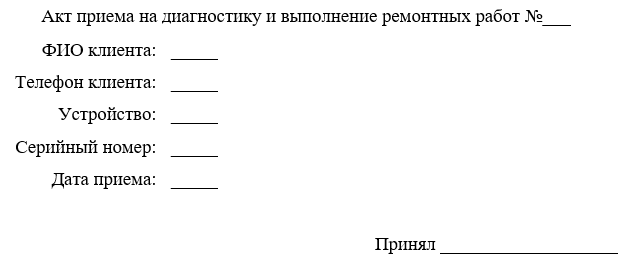


Рисунок 1.4.1– Форма акта приема

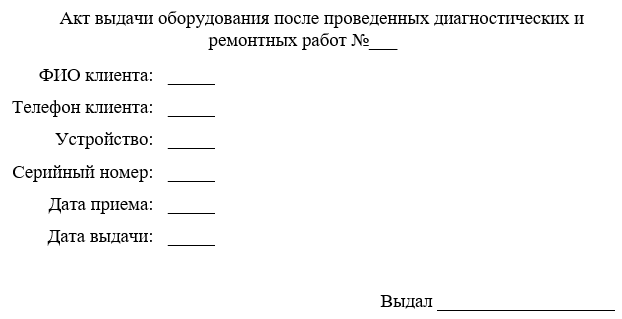


Рисунок 1.4.2 – Форма акта выдачи

# 1.5 Общие требования к программному продукту

Общее наименование информационной системы «Service\_center». При написании кода программного продукта был использован язык программирования C#, язык запросов SQL для СУБД MySQL.

Приложение для персонального компьютера разрабатывается для персональной вычислительной техники со следующими характеристиками:

* + - * объем ОЗУ: 2 ГБ и выше;
* процессор: двухъядерные с тактовой частотой 3 ГГц и лучше;
* место на жестком диске: 4 ГБ и больше;
  + - * ОС: Windows 10 64 bit;
      * монитор;
      * принтер
      * манипулятор типа «мышь»;
      * манипулятор типа «клавиатура»;
      * драйверы: .Net Framework 2.0, 4.5, Visual C++ 2005-2019 (x86, x64).

В результате работы приложения будут генерироваться текстовые документы, необходимо наличие редактора для форматов: doc, docx, rtf.

# 1.6 Описание структуры базы данных

На основании входных документов создаются таблицы базы данных, описание которых приведено ниже в таблице 1.6.1.

Для создания базы данных будет использована СУБД MySQL.

Таблица 1.6.1 – Описание структуры базы данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя поля | Описание поля | Тип данных | Размер поля | Тип ключа[[1]](#footnote-1) |
| 1 | 2 | 3 | 4 | 5 |
| Users(Пользователи) | | | | |
| user\_id | Код пользователя | INT | 4 | PK |
| login | Логин пользователя | VARCHAR | 76 |  |
| name | Имя пользователя | VARCHAR | 301 |  |

Продолжение таблицы 1.6.1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| surname | Фамилия пользователя | VARCHAR | 301 |  |
| patronymic | Отчество пользователя | VARCHAR | 301 |  |
| password | Пароль пользователя | VARCHAR | 301 |  |
| role | Роль пользователя | VARCHAR | 136 |  |
| Catalog(Список товаров) | | | | |
| article | Артикул | INT | 4 | PK |
| title | Название | VARCHAR | 766 |  |
| warranty\_period | Срок гарантии в месяцах | INT | 4 |  |
| Clients(Список клиентов) | | | | |
| client\_id | Код клиента | INT | 4 | PK |
| name | Имя клиента | VARCHAR | 301 |  |
| surname | Фамилия клиента | VARCHAR | 301 |  |
| patronymic | Отчество клиента | VARCHAR | 301 |  |
| phone | Телефон клиента | VARCHAR | 136 |  |
| Statuses(Справочник статусов) | | | | |
| status\_id | Код статуса | INT | 4 | PK |
| name | Название статуса | VARCHAR | 301 |  |

Продолжение таблицы 1.6.1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Orders(Журнал заказов) | | | | |
| order\_id | Код заказа | INT | 4 | PK |
| date\_check | Дата чека | DATE | 3 |  |
| article | Артикул | INT | 4 | FK |
| client\_id | Код клиента | INT | 4 | FK |
| serial\_number | Серийных номер | VARCHAR | 301 |  |
| date\_in | Дата приема | DATE | 3 |  |
| date\_out | Дата выдачи | DATE | 3 |  |
| status\_id | Код статуса | INT | 4 | FK |

На рисунке 1.6.1 представлена схема отношений.

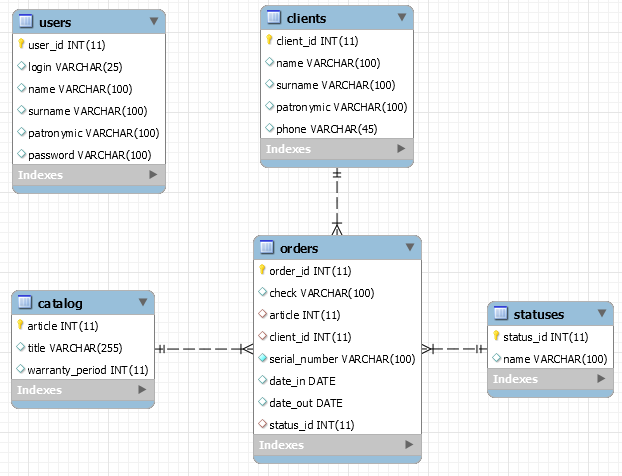


Рисунок 1.6.1 – Схема отношений базы данных

# 1.7 Контрольный пример

Контрольный пример является ручным подсчётом задачи. Он представляет собой вариант задачи с известными результатами и используется для проверки правильности решения на ПЭВМ.

По составленной программе обрабатываются исходные данные контрольного примера. Полученные результаты сравниваются с известными результатами контрольного примера. При несовпадении результатов производится поиск, исправление ошибок, и снова производится выполнение программы. И так до тех пор, пока не будет получен правильный результат.

Входные данные контрольного примера представлены в таблицах 1.7.1 – 1.7.5.

Выходные данные контрольного примера представлены на рисунках 1.7.1 – 1.7.2.

Таблица 1.7.1 – Входные данные для таблицы список пользователей

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Код пользователя | Логин | Имя | Фамилия | Отчество | Пароль |
| 1 | user\_r1 | Антон | Богов | Михайлович | 46h4m1 |
| 2 | user\_m1 | Борис | Гагарин | Семенов | 64n5u9 |

Таблица 1.7.2 – Входные данные для таблицы каталог товаров

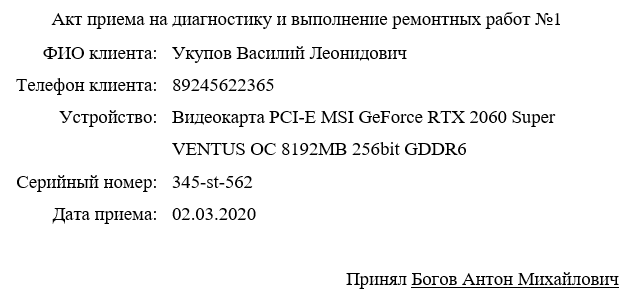
|  |  |  |
| --- | --- | --- |
| Артикул | Название | Срок гарантии |
| 1381298 | Видеокарта PCI-E MSI GeForce RTX 2060 Super VENTUS OC 8192MB 256bit GDDR6 | 36 |

Таблица 1.7.3 – Входные данные для таблицы список клиентов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код клиента | Имя | Фамилия | Отчество | Телефон |
| 1 | Василий | Укупов | Леонидович | 89245622365 |

Таблица 1.7.4 – Входные данные для таблицы справочник статусов

|  |  |
| --- | --- |
| Код статуса | Название |
| 1 | проводится диагностика |
| 2 | диагностика проведена, ремонт невозможен |
| 3 | диагностика проведена, ремонт возможен |
| 4 | диагностика проведена, не гарантийный случай |
| 5 | ремонт проведен успешно |
| 6 | изделие выдано клиенту |

Таблица 1.7.5 – Входные данные для таблицы журнал заказов

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Код заказа | Дата чека | Артикул | Код клиента | Серийный номер | Дата приема | Дата выдачи | Код статуса |
| 1 | 11.01.2020 | 1381298 | 1 | 345-st-562 | 02.03.2020 | 08.03.2020 | 6 |

Рисунок 1.7.1 – Выходной документ акт приема

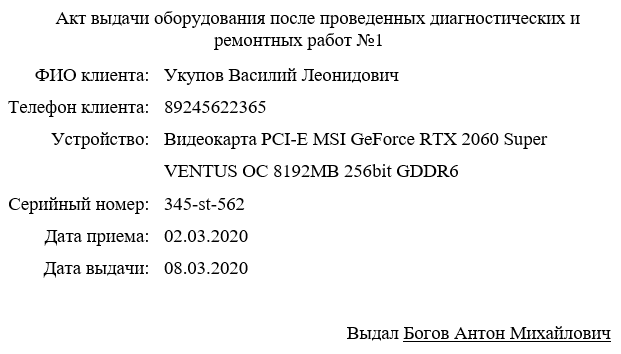


Рисунок 1.7.2 – Выходной документ акт выдачи

2 Экспериментальный раздел

2.1 Описание программы

Программа имеет модульную структуру, она было разработана в среде разработки Visual Studio 2019 с использованием фреймворка Windows Forms и СУБД MySQL Server. Содержит программные модули: Login, MasterForm, ReceiverForm, CreateOrder, OrderList, PrintForm, ConnectionDB. Описание модулей приведено в таблице 2.1.1.

Таблица 2.1.1 – Описание модулей

|  |  |
| --- | --- |
| Модуль | Назначение |
| 1 | 2 |
| Модуль Login – Форма авторизации, главный модуль | |
| LoginForm\_Load | Загрузка формы |
| LoginButton\_Click | Нажатие кнопки входа |
| Модуль MasterForm – Форма мастера для изменения статуса заказа | |
| MasterForm\_FormClosing | Выход из авторизации перед закрытием формы |
| OrderListDGV\_SelectionChanged | Выбор другой строки |
| SearchOrderTextBox\_TextChanged | Поиск одновременно с изменением |
| ConfirmStatusButton\_Click | Подтверждение изменения статуса |
| Модуль ReceiverForm – Форма приемщика для оформления нового заказа и печати актов | |

Продолжение таблицы 2.1.1

|  |  |
| --- | --- |
| ReceiverForm\_FormClosing | Выход из авторизации перед закрытием формы |
| SideBarButton\_Click | Меню для открытия MDI форм |
| Модуль CreateOrder – Форма для оформления заказа | |
| SearchTextBox\_TextChanged | Поиск одновременно с изменением |
| CatalogDGV\_SelectionChanged | Выбор другой строки |
| InfoBoxes\_TextChanged | Событие для первичной валидации введенных данных |
| CheckEmptyBars | Первичная валидация проверяющая пустые поля |
| SpendButton\_Click | Проведение заказа со вторичной валидацией |
| Модуль OrderList – Форма для отображения списка заказов | |
| SearchOrderTextBox\_TextChanged | Поиск одновременно с изменением |
| PrintButton\_Click | Кнопка для печати активирует форму печати |
| Модуль PrintForm – Форма для печати актов приема и выдачи | |
| Replace | Заменяет элемент шаблона в документе |
| ActReceiveButton\_Click | Кнопка для печати акта о приеме |
| ActIssueButton\_Click | Кнопка для печати акта о выдаче |

Продолжение таблицы 2.1.1

|  |  |
| --- | --- |
| Модуль ConnectionDB – Класс для подключения к базе данных и реализации этого подключения | |
| CheckLogin | Метод для верификации логина и пароля пользователя |
| DisplayCatalog | Метод отображает каталог с заданной строкой поиска |
| CompareDublicatesSerialNumbers | Метод для проверки отсутствия дубликатов серийных номеров у одно устройства |
| InsertingClients | Метод для вставки клиента |
| InsertingOrder | Метод для проведения заказа |
| DisplayOrders | Метод отображает журнал заказов с заданной строкой поиска |
| ReportData | Метод для получения данных для актов |
| TitleOrder | Метод для получения названия ремонтируемого устройства |
| StatusList | Метод для получения списка статусов |
| UpdateStatus | Метод обновляет статус у заданного заказа |

Модульная схема программы представлена на рисунке 2.1.1.

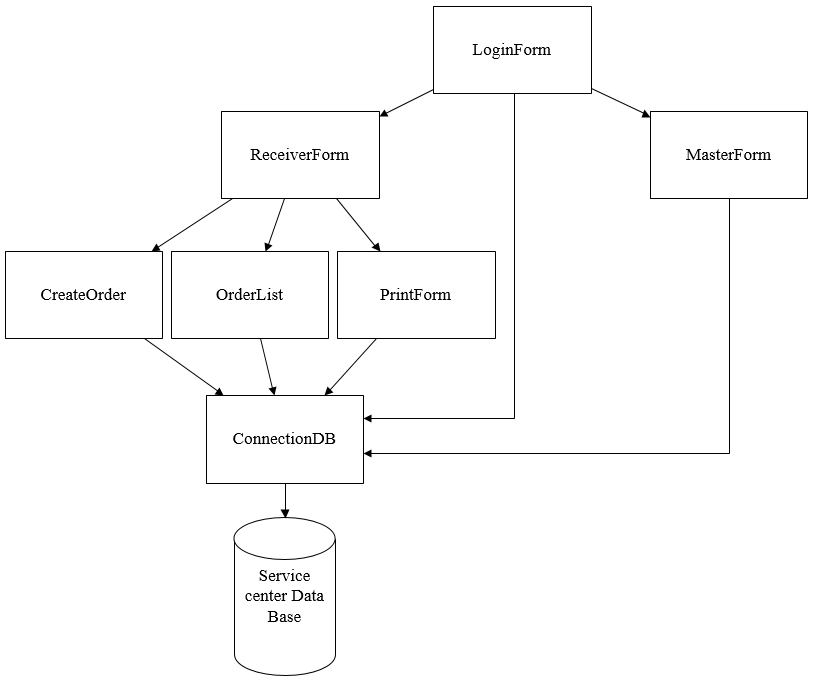


Рисунок 2.1.1 – Модульная схема приложения

2.2 Протокол тестирование программного продукта

В ходе тестирования программного продукта на корректных и некорректных данных не было обнаружено ошибок, которые влияли бы на работу самого программного продукта и всей системы.

Данный программный продукт удовлетворяет всем предъявленным требованиям, имеет комфортный интерфейс и интуитивно понятный функционал, исключает появления системных ошибок.

В таблицах 2.2.1 – 2.2.10 представлена общая информация о тестировании приложения.

Таблица 2.2.1 – Протокол тестирования авторизации на некорректных данных

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Service\_center |
| Номер версии | 1.0 |
| Имя тестера | Батурин Никита Сергеевич |
| Даты тестирования | 10.05.2020 |
| Описание информационных полей для тестирования | |
| Наименование | Авторизация |
| Test Case # | TC\_F1\_1 |
| Приоритет тестирования (Малый/Средний/высокий) | Высокий |
| Название тестирования | Проверка авторизации в модуле LoginForm с неправильными логином и паролем |
| Резюме испытания | Необходимо добиться корректного поведения программы при вводе неправильных данных |
| Шаги тестирования | Введены неверные логин и пароль  Нажата кнопка «войти» |
| Данные тестирования | Логин: qwer  Пароль: qwer |
| Ожидаемый результат | Программа должна оповестить о неверных логине или пароле |
| Фактический результат | В результате тестирования было получено оповещение о том, что были введены неверные логин или пароль (см. рисунок 2.2.1) |
| Предпосылки | Запустить программу |
| Постусловия | Работоспособность системы не нарушена |

Продолжение таблицы 2.2.1

|  |  |
| --- | --- |
| Статус (Pass/Fail) | Pass |

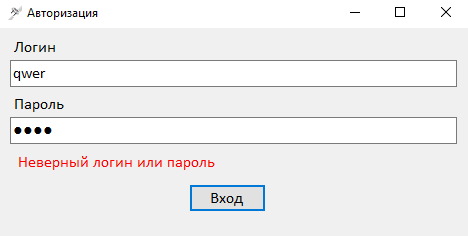


Рисунок 2.2.1 – Сообщение о некорректных данных аккаунта

Таблица 2.2.2 – Протокол тестирования авторизации на корректных данных

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Service\_center |
| Номер версии | 1.0 |
| Имя тестера | Батурин Никита Сергеевич |
| Даты тестирования | 10.05.2020 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Test Case # | TC\_F1\_2 |
| Приоритет тестирования (Малый/Средний/высокий) | Высокий |
| Название тестирования | Проверка авторизации в модуле LoginForm с правильными логином и паролем |
| Резюме испытания | Необходимо добиться корректного поведения программы при вводе правильных данных |
| Шаги тестирования | Введены верные логин и пароль  Нажата кнопка «войти» |
| Данные тестирования | Логин: user\_r  Пароль: 123 |
| Ожидаемый результат | Программа должна подключиться к системе |
| Фактический результат | В результате программа успешно подключилась к системе и отобразилось окно оформления заказа (см. рисунок 2.2.2) |
| Предпосылки | Запустить программу |
| Постусловия | Работоспособность системы не нарушена |
| Статус (Pass/Fail) | Pass |

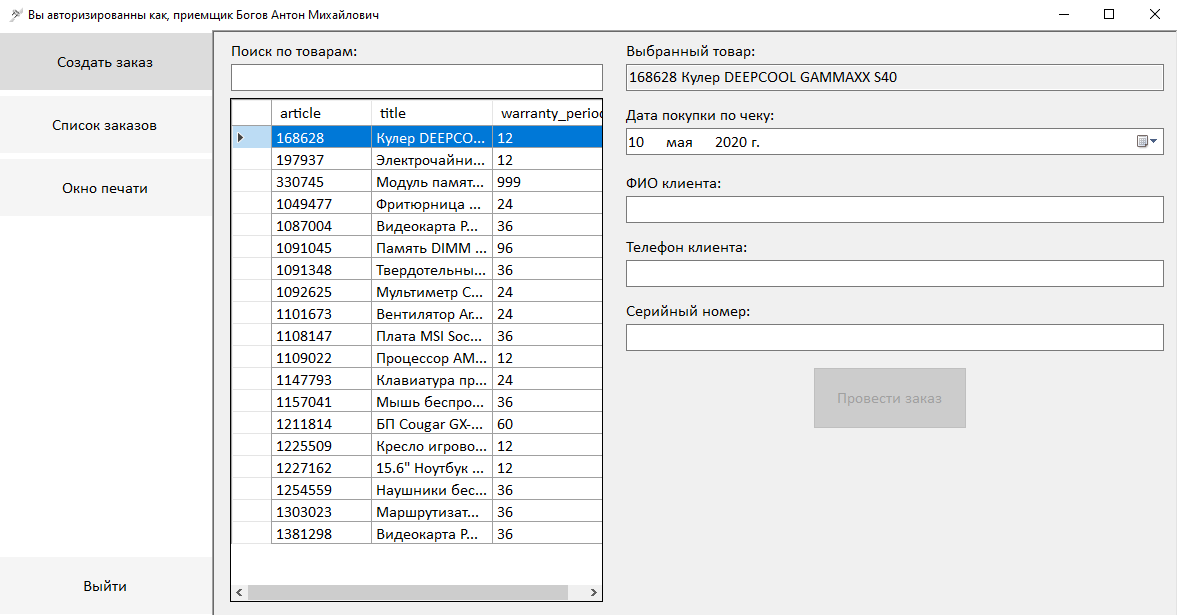


Рисунок 2.2.2 – Форма оформления заказа

Таблица 2.2.3 – Протокол тестирования проведения заказа с отсутствием данных

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Service\_center |
| Номер версии | 1.0 |
| Имя тестера | Батурин Никита Сергеевич |
| Даты тестирования | 10.05.2020 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Test Case # | TC\_F1\_3 |
| Приоритет тестирования (Малый/Средний/высокий) | Высокий |
| Название тестирования | Проверка оформления заказа |
| Резюме испытания | Необходимо добиться корректного поведения программы при проведении заказа с пустыми полями |
| Шаги тестирования | Совершен вход в систему  Нажата кнопка «Провести заказ» |
| Данные тестирования | Логин: user\_r  Пароль: 123 |
| Ожидаемый результат | Кнопка «Провести заказ» должна остаться не активной |
| Фактический результат | После нажатия кнопки «Провести заказ» ничего не произошло, так как она не активна (см. рисунок 2.2.3) |
| Предпосылки | Запустить программу и авторизоваться |
| Постусловия | Работоспособность системы не нарушена |
| Статус (Pass/Fail) | Pass |

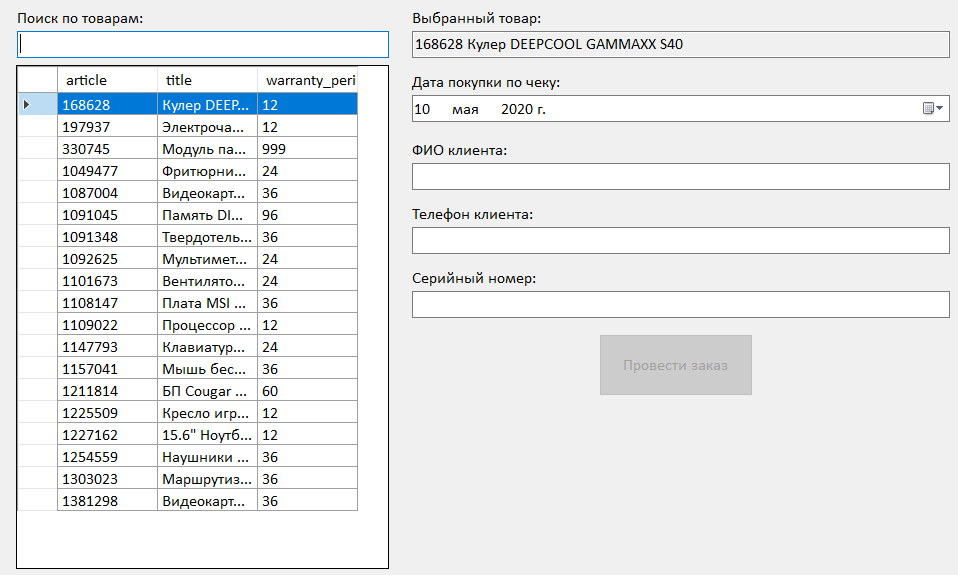


Рисунок 2.2.3 – Невозможность провести заказ с незаполненными полями

Таблица 2.2.4 – Протокол тестирования проведения заказа на некорректных данных

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Service\_center |
| Номер версии | 1.0 |
| Имя тестера | Батурин Никита Сергеевич |
| Даты тестирования | 10.05.2020 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Test Case # | TC\_F1\_4 |
| Приоритет тестирования (Малый/Средний/высокий) | Высокий |
| Название тестирования | Проверка оформления заказа |
| Резюме испытания | Необходимо добиться корректного поведения программы при проведении заказа с некорректными данными |

Продолжение таблицы 2.2.4

|  |  |
| --- | --- |
| Шаги тестирования | Совершен вход в систему  Поля заполнены некорректными данными  Нажата кнопка «Провести заказ» |
| Данные тестирования | Логин: user\_r  Пароль: 123 |
| Ожидаемый результат | Должно высветится уведомление о некорректности введенных данных |
| Фактический результат | После нажатия кнопки «Провести заказ» отобразилось уведомление о некорректности введенных данных (см. рисуноки 2.2.4 – 2.2.6) |
| Предпосылки | Запустить программу и авторизоваться |
| Постусловия | Работоспособность системы не нарушена |
| Статус (Pass/Fail) | Pass |

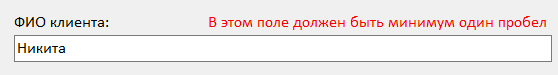


Рисунок 2.2.4 – Уведомление о некорректности ФИО клиента

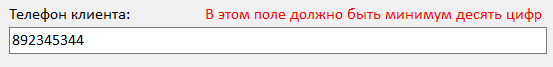


Рисунок 2.2.5 – Уведомление о некорректности телефона клиента



Рисунок 2.2.6 – Уведомление о некорректности серийного номера

Таблица 2.2.5 – Протокол тестирования проведения заказа на корректных данных

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Service\_center |
| Номер версии | 1.0 |
| Имя тестера | Батурин Никита Сергеевич |
| Даты тестирования | 10.05.2020 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Test Case # | TC\_F1\_5 |
| Приоритет тестирования (Малый/Средний/высокий) | Высокий |
| Название тестирования | Проверка оформления заказа |
| Резюме испытания | Необходимо добиться корректного поведения программы при проведении заказа с корректными данными |
| Шаги тестирования | Совершен вход в систему  Поля заполнены корректными данными  Нажата кнопка «Провести заказ» |
| Данные тестирования | Логин: user\_r  Пароль: 123 |
| Ожидаемый результат | Должен оформиться акт о приеме |
| Фактический результат | После нажатия кнопки «Провести заказ» отобразился оформленный акт приема (см. рисуноки 2.2.7) |
| Предпосылки | Запустить программу, авторизоваться и заполнить все поля |

Продолжение таблицы 2.2.5

|  |  |
| --- | --- |
| Постусловия | Работоспособность системы не нарушена |
| Статус (Pass/Fail) | Pass |

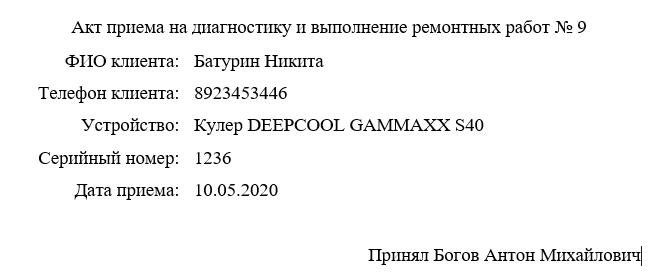


Рисунок 2.2.7 – Оформленный акт приема

Таблица 2.2.6 – Протокол тестирования взятия заказа мастером на диагностику и ремонт и дальнейшего изменения статуса заказа

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Service\_center |
| Номер версии | 1.0 |
| Имя тестера | Батурин Никита Сергеевич |
| Даты тестирования | 10.05.2020 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Test Case # | TC\_F1\_6 |
| Приоритет тестирования (Малый/Средний/высокий) | Высокий |
| Название тестирования | Проверка обновления статуса заказа |

Продолжение таблицы 2.2.6

|  |  |
| --- | --- |
| Резюме испытания | Необходимо добиться корректного поведения программы при изменении статуса заказа |
| Шаги тестирования | Совершен вход в систему  Мастером был выбран заказ и проведена диагностика и ремонт  В выпадающем списке статусов для заданного заказа был выбран «ремонт проведен успешно»  Нажата кнопка «Подтвердить статус» |
| Данные тестирования | Логин: user\_m  Пароль: 123 |
| Ожидаемый результат | Обновления статуса заказа на «ремонт проведен успешно» |
| Фактический результат | После нажатия кнопки «Подтвердить статус» статус изменился на «ремонт проведен успешно» (см. рисунок 2.2.8) |
| Предпосылки | Запустить программу, авторизоваться и указать новый статус |
| Постусловия | Работоспособность системы не нарушена |
| Статус (Pass/Fail) | Pass |

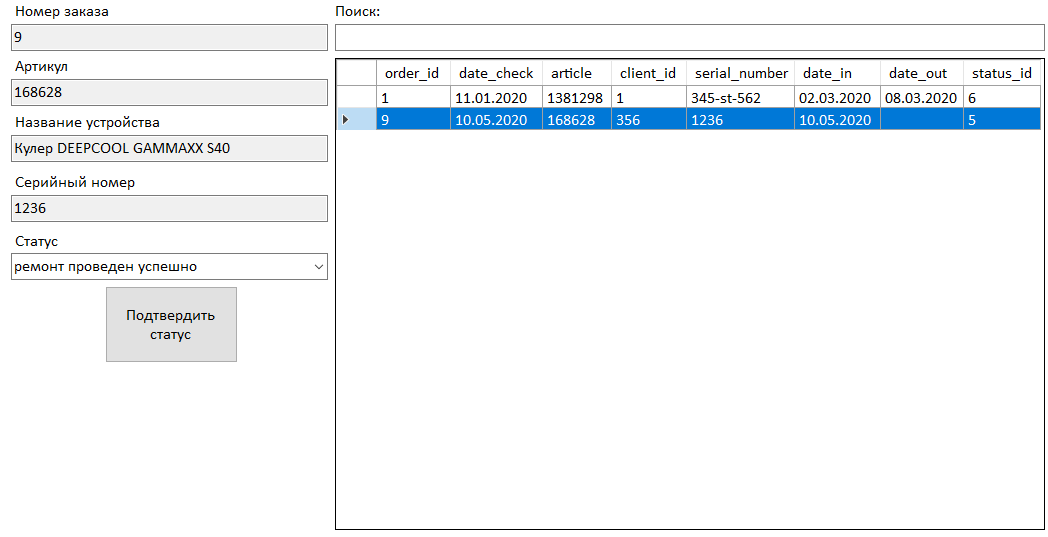


Рисунок 2.2.8 – Успешное обновление статуса заказа

Таблица 2.2.7 – Протокол тестирования печати акта о выдаче

|  |  |
| --- | --- |
| Общая информация о тестировании | |
| Название проекта | Service\_center |
| Номер версии | 1.0 |
| Имя тестера | Батурин Никита Сергеевич |
| Даты тестирования | 10.05.2020 |
| Описание информационных полей для тестирования | |
| Наименование | Описание |
| Test Case # | TC\_F1\_7 |
| Приоритет тестирования (Малый/Средний/высокий) | Высокий |
| Название тестирования | Проверка печати акта о выдаче |
| Резюме испытания | Необходимо добиться корректного поведения программы на печать акта о выдаче и обновления статуса на «изделие выдано клиенту» |
| Шаги тестирования | Совершен вход в систему  Клиент пришел забрать устройство  Печатается акт о выдаче |
| Данные тестирования | Логин: user\_r  Пароль: 123 |
| Ожидаемый результат | Обновления статуса заказа на «изделие выдано клиенту» и печать акта о выдаче |
| Фактический результат | После нажатия кнопки «Подтвердить выдачу» статус изменился на «изделие выдано клиенту» и был оформлен акт о выдаче (см. рисунок 2.2.9) |
| Предпосылки | Запустить программу, авторизоваться и оформить акт о выдаче |
| Постусловия | Работоспособность системы не нарушена |
| Статус (Pass/Fail) | Pass |

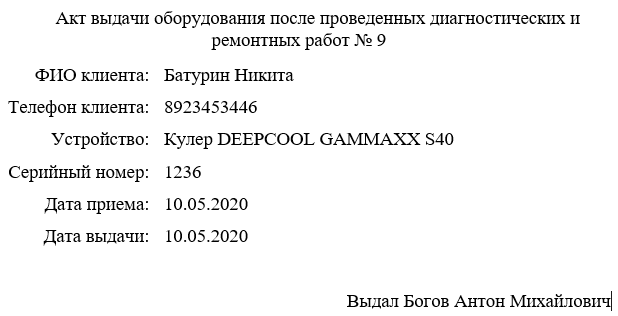


Рисунок 2.2.9 – Сформированный акт о выдаче

2.3 Руководство пользователя

Руководство пользователя относится к эксплуатационной документации. Основная цель руководства пользователя заключается в обеспечении пользователя необходимой информацией для самостоятельной работы с программой или автоматизированной системой.

Для использования программного решения пользователю необходимо иметь персональный компьютер и постоянное подключение к базе данных через интернет.

Пользователями программы являются приемщики и мастера, для каждого пользователя существует роль, которая определяет, какой функционал предоставить пользователю. Приемщики оформляют заказы от клиентов путем заполнения всех необходимых данных в соответствующие поля, далее проводят заказ, которому, в свою очередь задается статус «проводится диагностика». Дальнейшая судьба заказа передается в руки мастеру, он производит диагностику устройства и по ее итогам указывает один из статусов: «диагностика проведена, ремонт невозможен», «диагностика проведена, ремонт возможен» или «диагностика проведена, не гарантийный случай». При заключении невозможности или не гарантийного случая устройство возвращается клиенту, иначе мастером производиться ремонт и записывается статус «ремонт проведен успешно». О чем узнает приемщик и связывается с клиентов для возвращения. При выдаче аппаратуры приемщиком, у заказа указывается статус «изделие выдано клиенту». На этом работа с данным заказом закончена.

Руководство пользователя для приемщика.

При запуске программы открывается окно авторизации, где нужно ввести личные логин и пароль. Окно авторизации представлено на рисунке 2.3.1.

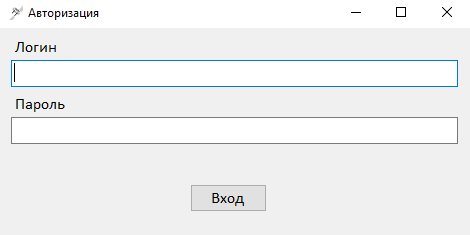


Рисунок 2.3.1 – Окно авторизации

После успешной авторизации для приемщика открывается главное окно, в котором по умолчанию открыта вкладка «Создать заказ», окно представлено на рисунке 2.3.2.

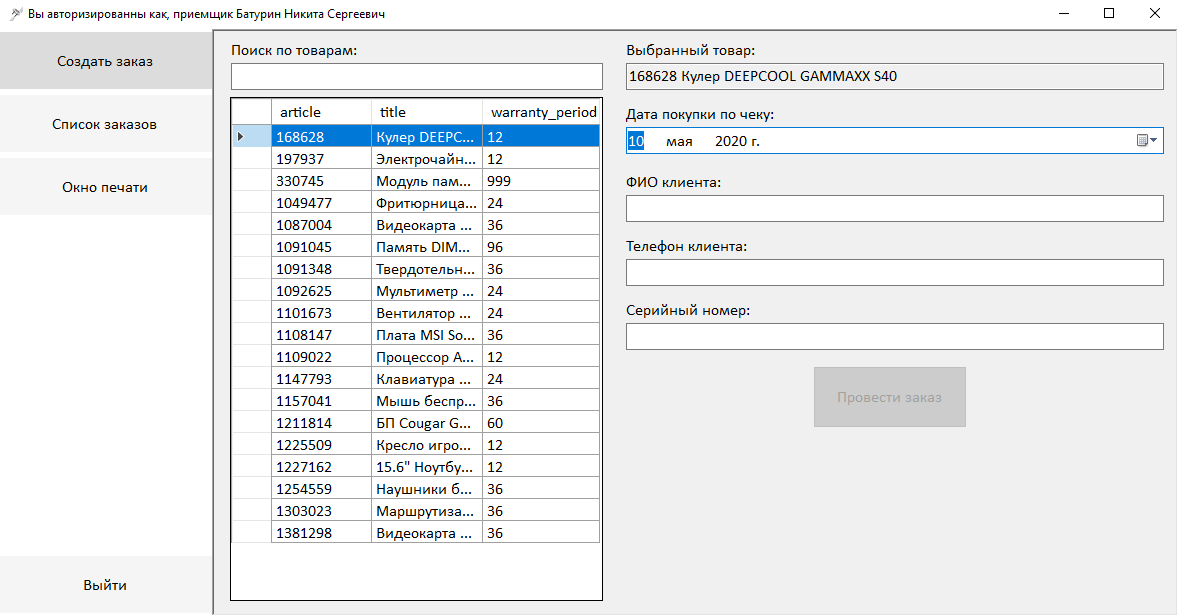


Рисунок 2.3.2 – Главное окно с открытой вкладкой «Создать заказ»

На этом окне заполняются данные по заявлению клиента о гарантийном обслуживании. Сначала, при помощи поисковой строки ищется товар в каталоге. Нужный товар выбирается и доступен для чтения в поле «Выбранный товар», далее с чека покупки заносится дата покупки, для выяснения продолжает ли свое действия гарантия. Если гарантия еще активна, приемщик вносит контактные данные клиента и серийный номер устройства.

При корректном вводе данных и нажатии на кнопку «Провести заказа», будет автоматически сформирован акт о приеме устройства на диагностику и ремонт. Акт представлен на рисунке 2.3.3.

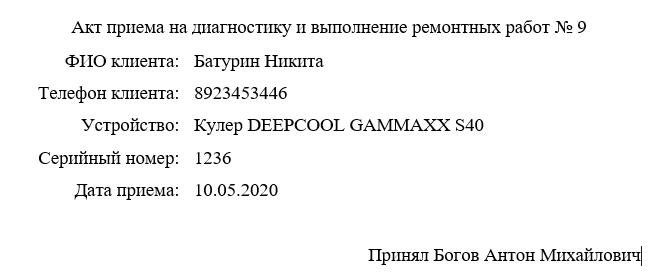


Рисунок 2.3.3 – Акт приема на диагностику и ремонт

После этого устройство и заказ переходят к мастеру, его часть программного решения будет рассмотрена далее, вне зависимости от результата диагностики и ремонта итог один – устройство возвращается к владельцу. Когда клиент возвращается для выдачи ему аппаратуры, также автоматически формируется акт выдачи. Акт выдачи представлен на рисунке 2.3.4.

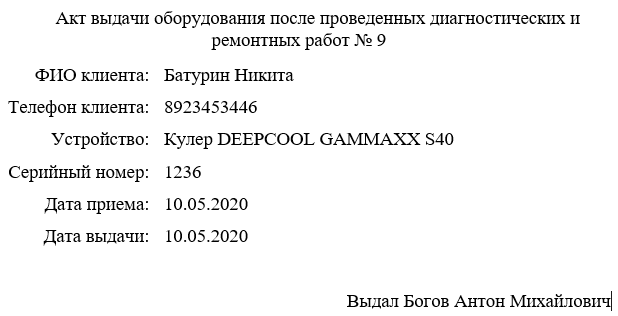


Рисунок 2.3.4 – Акт выдачи оборудования после проделанной работы

Руководство пользователя для мастера.

При запуске программы открывается окно авторизации, где нужно ввести личные логин и пароль. Окно авторизации представлено на рисунке 2.3.1.

После успешной авторизации для мастера открывается окно выбора заказа и возможностью изменить статус заказа. Окно представлено на рисунке 2.3.5.

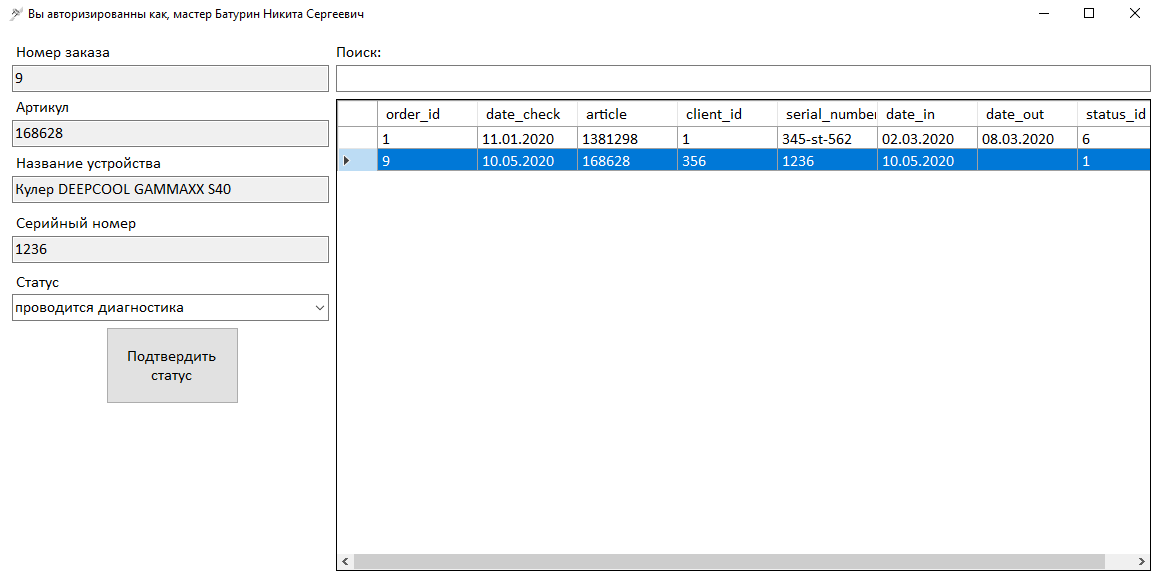


Рисунок 2.3.5 – Окно мастера

Работа мастера заключается в том, чтобы найти заказ в списке для этого есть поисковая строка, далее нужно реальности произвести диагностику, по результату которой, указывается соответствующий статус, далее если итог диагностики означает возможность ремонта, мастер ремонтирует устройство. По завершению ремонта, мастер указывает статус о завершенности ремонта.

Далее заказ переходит снова к приемщику для дальнейшего возвращения оборудования клиенту.

ЗАКЛЮЧЕНИЕ

В результате выполнения дипломного проекта все поставленные цели и задачи были выполнены. Было разработано программное решения для использования его в бизнес среде, выполнено физическое проектирование базы данных, спроектированы экранные формы ввода-вывода и формирования актов.

Разработанное программное обеспечение располагают функциональным решением для выполнения таких задач, как:

– авторизация в системе;

– создание заказов на гарантийное обслуживание;

– формирование актов приема и выдачи;

– просмотр заказов;

– редактирование статуса заказа.

Результатом работы стал программный продукт для автоматизации учета гарантийного оборудования в сервисном центре. Программа разработана на языке программирования C# с использованием технологий Windows Forms и MDI контейнеры, также для работы используется система управления базами данных MySQL.

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Кормен, Т. П. Алгоритмы: построение и анализ [Текст] / Т. Кормен. — М.: МЦНМО, 2013. — 969 с.
2. Скиена, С. Алгоритмы. Руководство по разработке [Текст] / С. Скиена. — 2-е изд.— СПб.: БХВ-Петербург, 2013. — 720 с.
3. Система стандартов безопасности труда [Текст] / Под ред. В. А. Фомичева. – М.: Изд-во стандартов, 2016. – 102 с.
4. Мартин, Р. К. Чистый код: создание, анализ и рефакторинг. Библиотека программиста [Текст] / Р. Мартин. — СПб.: Питер, 2015. — 464 с.
5. Фаулер, М. Рефакторинг: улучшение существующего кода [Текст] / М. Фаулер. – СПб.: Символ-Плюс, 2013. – 432 с.
6. Глушаков, С. В. База данных [Текст] / С. В. Глушаков. – Харьков: Фолио. – М.: ООО «Издательство ACT», 2012. – 232 с.
7. Илюшечкин, В. М. Основы использования и проектирования баз данных: учебник для вузов [Текст] / В. М. Илюшечкин. – М.:Юрайт, 2011. – 2013 с.
8. Иванова, Г. С. Технология программирования [Текст] / Г. С. Иванова. – М.: КноРус, 2011. – 336 с.
9. Давыдова, Н. А. Программирование [Электронный ресурс] / Н. А. Давыдова, Е. В. Боровская. - М.: БИНОМ. Лаборатория знаний, 2012. - 238 с.
10. Гагарина, Л. Г. Технология разработки программного обеспечения [Текст] / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Виснадул. – М.: ИД. «Форум»: ИНФРА-М, 2011. – 400 с.
11. Бужан, В. В. Объектно-ориентированное программирование: учебно-методическое пособие для студентов очной и заочной форм обучения направления подготовки 230100.62 [Текст] / В. В. Бужан. – М.: Краснодар: ИМСИТ, 2013. – 52 с.
12. Бабаш, А. В. Информационная безопасность. Лабораторный практикум[Текст] / А. В. Бабаш. - М.: КНОРУС, 2013.-135 с.
13. Башлы, П. Н. Информационная безопасность и защита информации [Электронный ресурс]: / П. Н. Башлы, А. В. Бабаш, Е. К. Баранова. - М.: РИОР, 2013. - 222 с.
14. Баранова, Е. К. Основы информатики и защиты информации [Электронный ресурс]: / Е. К. Баранова. - М.: РИОР : ИНФРА-М, 2013. - 183 с.
15. Баранова, Е. К. Информационная безопасность и защита информации [Текст]: учеб. пособие.- 2-е изд./ Баранова, Е. К, Бабаш А. В. - М.: РИОР: ИНФРА-М, 2014.-256 с.
16. Карпова, И. П. Базы данных. Учебное пособие. Курс лекций и материалы для практических занятий [Текст] / И. П. Карпова. – М.: СПб.: Питер, 2013. — 240 с.
17. Российская государственная библиотека [Электронный ресурс]/ Центр информ. РГБ; ред. Власенко Т.В.; Web-мастер Козлова Н.В. Электрон. дан. –М.: Рос. гос. б-ка, 1997. – Режим доступа http://www.rsl.ru, свободный. Загл. с экрана – Яз. рус., англ.
18. Википедия [Электронный ресурс] – Режим доступа https://ru.wikipedia.org, свободный. Загл. с экрана – Яз. рус., англ.
19. Электронно-библиотечная система Znanium.com/ разработка Научно-издательского центра ИНФРА-М. – Режим доступа https://znanium.com, свободный. Загл. С экрана – Яз. рус., англ.
20. Рихтер, Дж. CLR via C# программирование на платформе Microsoft .Net Framework 4.5 на языке C# [Текст] / Дж. Рихтер. – 4-е издание – СПб.: Питер, 2016. – 896 с.: ил. – (Серия «Мастер-класс»).

1. PK-первичный ключ

   FK-внешний ключ [↑](#footnote-ref-1)