Министерство образования Тульской области

Государственное профессиональное образовательное учреждение

Тульской области

«Донской политехнический колледж»

Разработка информационной системы учета ремонта мобильных устройств

Курсовая работа МДК 02.01

«Технология разработки программного обеспечения»

|  |  |
| --- | --- |
| Студента группы С-20-1 | П.В. Смыслов |
| Руководитель | С.М. Гвоздев |
| Проверил | \_\_\_\_\_\_\_\_\_\_\_\_\_\_  (дата) |
| Оценка | \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Донской, 2022

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc122925456)

[1. Техническое задание 5](#_Toc122925457)

[1.1 Назначение работы 5](#_Toc122925458)

[1.2 Требования к функциональным характеристикам 5](#_Toc122925459)

[1.3 Требования к надежности и безопасности 6](#_Toc122925460)

[1.4 Требования к составу и параметрам технических средств 6](#_Toc122925461)

[1.5 Требования к информационной и программной совместимости 6](#_Toc122925462)

[1.6 Порядок контроля и приёмки 7](#_Toc122925463)

[2. Разработка технического проекта 8](#_Toc122925464)

[2.1 Анализ требования спецификаций 8](#_Toc122925465)

[2.2 Этап эскизного проектирования 11](#_Toc122925466)

[2.3 Проектирование внутренней структуры 14](#_Toc122925467)

[3. Реализация программного обеспечения 15](#_Toc122925468)

[3.1 Обоснование выбора средств разработки 15](#_Toc122925469)

[3.2 Разработка программного обеспечения 18](#_Toc122925470)

[3.3 Технико-экономические показатели 24](#_Toc122925471)

[ЗАКЛЮЧЕНИЕ 27](#_Toc122925472)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 28](#_Toc122925473)

[ПРИЛОЖЕНИЕ А 29](#_Toc122925474)

[ПРИЛОЖЕНИЕ Б 31](#_Toc122925475)

# ВВЕДЕНИЕ

Сервисные центры по ремонту и обслуживанию мобильных устройств принимают от юридических и физических лиц устройства, нуждающиеся в ремонте, модернизации или каких-либо других действиях, требующих вмешательства специалистов. При этом, в ходе ремонтных работ, в большинстве случаев, специалисты сервисных центров опираются на свой опыт по ремонту и обслуживанию мобильных устройств.

Использование информационной системы «разработка информационной системы учета ремонта мобильных устройств» позволит повысить производительность труда сотрудников сервисного центра, качество и скорость обслуживания клиентов, за счет оперативного анализа неисправностей и сокращения времени на выбор варианта их устранения.

Целью курсовой работы является разработка информационной системы сервисного центра.

Исследование, безусловно, является актуальным, поскольку информационная система учета ремонта мобильных устройств сегодня требуются не только для крупных промышленных предприятий, но и для мелких частных организаций, на которых существуют проблемы с учетом клиентов, деталей, устройств и работой предприятия в условиях компьютеризации регионов России.

Объектом исследования является учет клиентской базы, запчастей и устройств.

Предметом исследования являются смартфоны, планшеты, ноутбуки, запчасти для перечисленного.

В соответствии с поставленной целью необходимо решить следующие задачи:

* проанализировать теоретическую базу, касаемую проектирования и разработок информационных систем;
* рассмотрение классификации информационных систем;
* разработка основных критериев и требований, которым должна соответствовать система;
* разработка информационной системы сервисного центра;
* анализ полученных результатов.

Таким образом, создается информационная система, которая повысит

эффективность, скорость обработки, структурирует и облегчит процесс учета заявок на ремонт.

1. Техническое задание
   1. Назначение работы

Программный продукт «МОБАЙЛВЕКТОР» служит для создания, сохранения, передачи, обработки информации для учета мобильных устройств, их владельцев и запчастей.

Преимуществом программного продукта «МОБАЙЛВЕКТОР» является:

1. Систематизация большого количества информации о клиентах их устройствах и запчастях.
2. Экономия времени – сокращение поиска информации до считаных секунд по одним только номеру или дате.
3. Экономия финансов – расходники для печати на бумажном носителе и сама бумага стоят денег, программа сохранит бюджет.

## 1.2 Требования к функциональным характеристикам

Функциональная характеристика — это набор рабочих, эксплуатационных

параметров любой техники: прибора, устройства, блока, детали, дающая количественную оценку из свойств.

Данный программный продукт служит для создания, сохранения, передачи, обработки информации для учета мобильных устройств, их владельцев и запчастей.

Программный продукт «МОБАЙЛВЕКТОР» должен предоставлять администратору возможность:

* добавлять, редактировать, удалять информацию о клиентах, запчастях, мобильных устройствах;
* управлять учетными записями персонала;
* Программный продукт «МОБАЙЛВЕКТОР» должен предоставлять персоналу возможность:
* Просматривать и редактировать информацию о клиентах, запчастях, мобильных устройствах;

## 1.3 Требования к надежности и безопасности

Основными требованиями надежности и безопасности разрабатываемого программного продукта следует считать:

* Контроль учетных записей.
* Автосохранениеш8 информации.

## 1.4 Требования к составу и параметрам технических средств

Основными требованиями к составу и параметрам технических документов являются:

* Процессор Intel Core i5 10400
* частота: 2.9 ГГц
* оперативная память: 4096 МБ DDR4
* графический процессор: Intel UHD Graphics 630
* разрешение экрана: 1024 x 768.

## 1.5 Требования к информационной и программной совместимости

Для эксплуатации программного продукта необходимо наличие следующих компонентов:

* операционная система семейства Microsoft®Windows®

(не ниже Windows 8.1).

* доступ к сети-Интернет

## 1.6 Порядок контроля и приёмки

Для проверки выполнения заданных функций программного продукта «МОБАЙЛВЕКТОР» устанавливаются следующие виды испытаний:

* тестовые испытания;
* опытная эксплуатация;
* приемочные испытания.

Срок приема – сдачи ПО: 01.10.22 – 26.12.22 (3 месяца).

Условия приема – сдачи:

* работа программного обеспечения «МОБАЙЛВЕКТОР» корректна.
* отладка и тестирования пройдены успешно.
* протокол испытаний.

Протокол испытаний – это документ, содержащий результаты испытаний и другую информацию, относящуюся к испытанию (таблица 1.1).

Таблица 1.1 - Протокол испытаний

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №п/п | Дата | | Испытания | Результат |
| 1 | 20.10.2022 | | Моделирование объекта | Созданы инфологическая, даталогическая, физическая модели, а также ER – диаграмма. |
| 2 | 26.10.2022 | | Составление технического задания | ТЗ составлено и согласованно |
| 3 | 06.11.2022 | | Разработка эскизного проекта | Создана DFD-диаграмма и SADT-модель |
| 4 | 10.11.2022 | | Программирование | Написан код |
| 5 | 23.12.2022 | | Отладка и тестирование | Исправлены ошибки |
| 6 | | 24.12.2022 | Внедрение ПП | Использование ПП |
| 7 | | 25.12.2022 | Сопроводительная документации | Составлена документация |

По завершении проектирования технического задания следует приступить к разработке технического проекта.

1. Разработка технического проекта
   1. Анализ требования спецификаций

Анализ требований к ПО – это процесс работы с требованиями к ПО в процессе его разработки. Фактически, это процесс формирования технического задания на создаваемое ПО и сопровождения ТЗ в процессе разработки ПО.

Спецификация требований программного обеспечения — законченное описание поведения программы, которую требуется разработать. Включает ряд пользовательских сценариев, которые описывают все варианты взаимодействия между пользователями и программным обеспечением.

SADT-МОДЕЛЬ — это точное, полное и адекватное текстовое и графическое описание системы, имеющей конкретное назначение, выполненное в виде иерархически организованной совокупности диаграмм, созданных на основе стандартного представления данных.

IDEF0 — это методология и нотация для функционального моделирования системы, предназначенная для формализации и описания бизнес-процессов, поддерживаемых и автоматизируемых этой системой.

К ее особенностям можно отнести:

* использование контекстной диаграммы;
* поддержка декомпозиции;
* доминирование;
* выделение 4 типов стрелок.

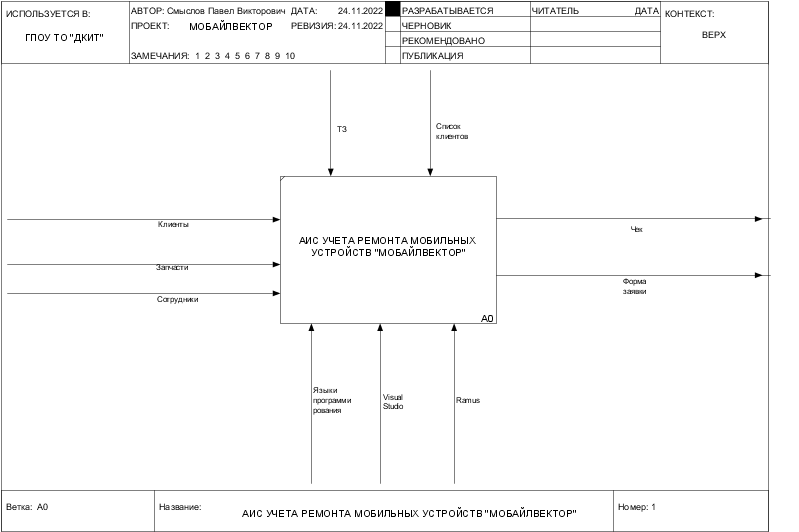


Рисунок 2.1 - IDEF0 АИС УЧЕТА РЕМОНТА МОБИЛЬНЫХ УСТРОЙСТВУЧЕТА РЕМОНТА МОБИЛЬНЫХ УСТРОЙСТВ "МОБАЙЛВЕКТОР"

IDEF1 — это методология моделирования информационных потоков внутри системы, позволяющая отображать и анализировать их структуру и взаимосвязи. Одна из основных ценностей и причин стремительного развития информационных технологий — это высочайший темп наращивания человечеством информации. Хранимая и обрабатываемая, она даёт возможность прогресса, и потребность в том, чтобы наращивать возможности по обработке данных, привела к появлению ЭВМ.

Основные характеристики:

* Помогает выяснить структуру и содержание существующих потоков информации на любом предприятии.
* Помогает определить какие проблемы, выявленные в результате функционального анализа и анализа потребностей, вызваны недостатком управления соответствующей информацией.
* Выявляет, информационные потоки, требующие дополнительного управления для эффективной реализации модели.

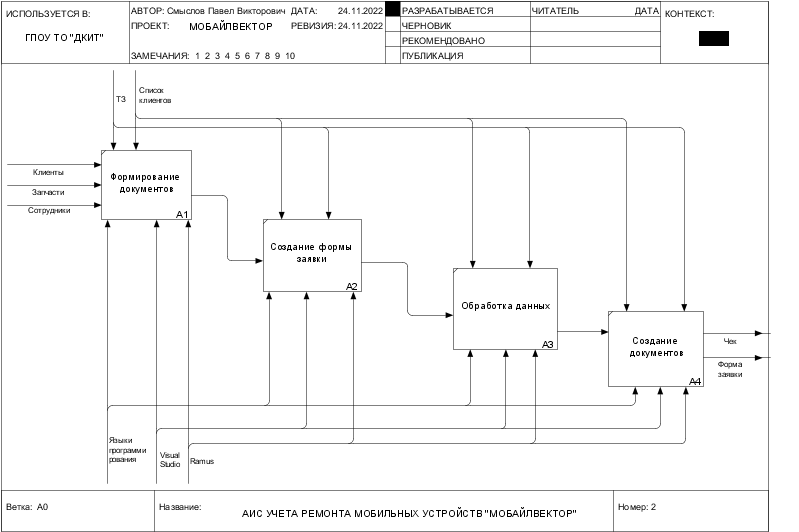


Рисунок 2.2 - IDEF0 АИС УЧЕТА РЕМОНТА МОБИЛЬНЫХ УСТРОЙСТВУЧЕТА РЕМОНТА МОБИЛЬНЫХ УСТРОЙСТВ "МОБАЙЛВЕКТОР"

ER-диаграмма — это разновидность блок-схемы, где показано, как разные «сущности» (люди, объекты, концепции и так далее) связаны между собой внутри системы. ER-диаграммы чаще всего применяются для проектирования и отладки реляционных баз данных в сфере образования, исследования и разработки программного обеспечения и информационных систем для бизнеса.

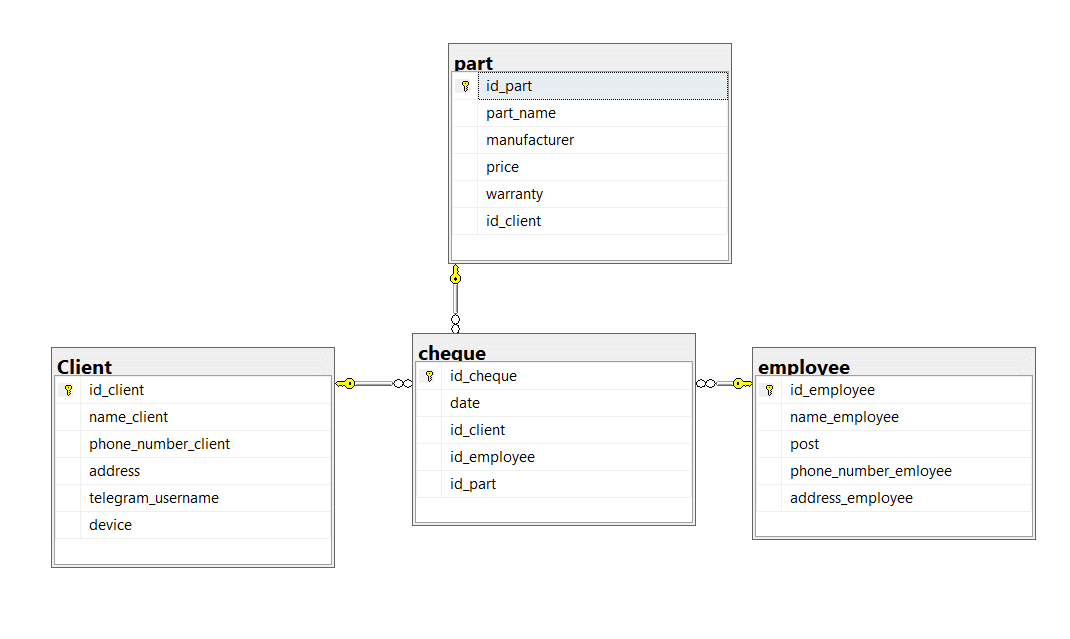


Рисунок2.3 - ER-диаграмма

Рассмотрев требования и спецификации технического проекта, необходимо приступить к этапу эскизного проектирования программного обеспечения.

## 2.2 Этап эскизного проектирования

Эскизный проект представляет собой совокупность конструкторских документов, отражающих принципиальные конструктивные решения, дающие общее представление об устройстве и принципе работы изделия, а также его основные параметры и габаритные размеры.

Прототип — это очень упрощённая версия будущей сложной программы, которая просто позволяет проверить одну конкретную узкую идею. Прототипы нужны, когда стоит задача быстро проверить какую-то конкретную техническую гипотезу.

1. Постановка цели. Происходит на встрече всех участников, среди которых клиент, дизайнер, маркетолог, копирайтер, программист, то есть все те, кто заинтересован в успешности проекта. Чем четче и точнее сформулированы цели, тем легче выдвигать и проверять гипотезы для детализации прототипа.
2. Цели нужно ставить максимально конкретные: например, разработать продающий лэндинг для презентации новой услуги, продумать корпоративный сайт для повышения охвата и узнаваемости бренда.
3. Проведение исследования. Чтобы создать качественный прототип, важно изучить бизнес клиента, особенности его продукта и целевой аудитории.
4. Формулирование гипотез. Важно понять, что именно вы хотите проверить с помощью прототипов. Не «посмотреть, получится ли разместить все блоки на сайте», а «оценить, насколько пользователю удобно будет выбрать товар и сделать заказ таким способом». Это поможет сделать прототипирование максимально эффективным.
5. Создание прототипа. С учетом результатов исследования и сформулированных гипотез создается макет будущего сайта или приложения.

Описав этапы проектирования необходимо перейти к созданию прототипа.

Рисунок 2.4 - прототип по "МОБАЙЛВЕКТОР"

LOGO

Авторизация

Новый

сотрудник

Регистрация клиента

Login

Password

Дизайн-макет — это схематичное изображение финальной идеи с указанием всех деталей. В нем указываются концепция, шрифты, тексты, изображения, расположение всех элементов и общая картина продукта.

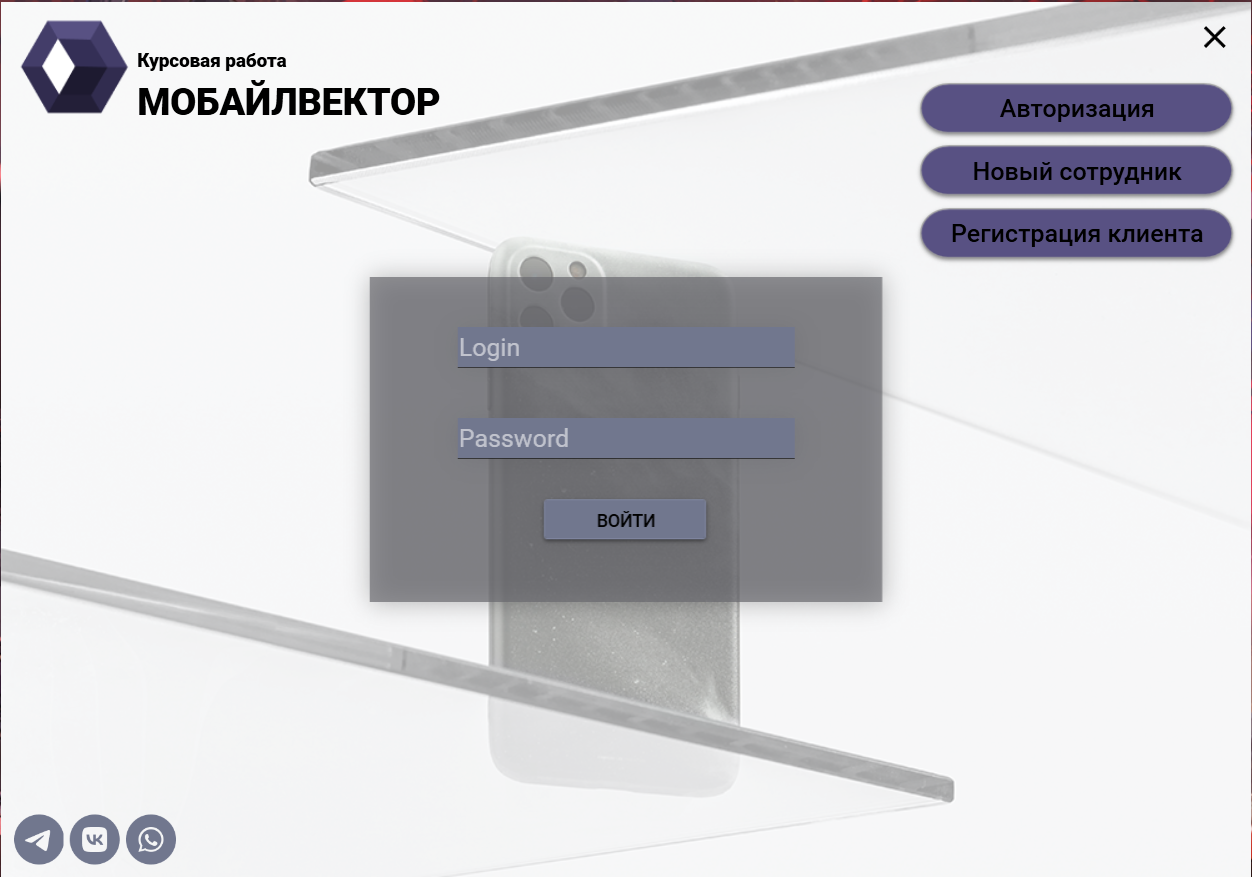


Рисунок 2.5 – дизайн-макет ПО «МОБАЙЛВЕКТОР»

Рассмотрев [этап эскизного проектирования программного](#_2.2_Этап_эскизного) обеспечения, необходимо приступить к этапу проектирования внутренней структуры.

## 2.3 Проектирование внутренней структуры

На этапе создания программного продукта важно грамотно разработать его структуру, так как она может повлиять на получение пользователей из поисковых систем. Для этого необходимо разместить информацию таким образом, чтобы любой посетитель смог быстро ее отыскать.

В программном продукте «МОБАЙЛВЕКТОР» применяется нелинейная структура, которая выглядит в виде деления на две страницы с без возможности перехода из одной в другую (рисунок 2.6).

Страница

авторизации

Услуги

(MainUserPage.xaml)

Портфолио

(MainAdminPage.xaml)

Рисунок 2.6 – Внутренняя структура программного обеспечения

Понятная логическая схема программного продукта упростит его настройку и обслуживание в дальнейшем. Такое ПО легче и быстрее контролировать, проще анализировать их данные, не допуская появления дублирующих и пустых страниц.

Таким образом выполнен этап построения внутренней структуры программного продукта, следовательно, необходимо приступить к этапу разработки программного кода с последующей его проверкой.

1. Реализация программного обеспечения
   1. Обоснование выбора средств разработки

Для реализации программного продукта «МОБАЙЛВЕКТОР» необходимо программное обеспечение:

1. Visual Studio

* IntelliSense. Технология автодополнения Microsoft. Дописывает название функции при вводе начальных букв. Кроме прямого назначения, IntelliSense используется для доступа к документации и для устранения неоднозначности в именах переменных, функций и методов, используя рефлексию.
* Code Anilizer. Функционал, который помогает найти ошибки в коде. Совмещён с IntelliSense, тем, что все ошибки, уведомления, потенциальные ошибки подсвечиваются. Также вся эта информация отображается в окне "Error List".
* Perfomance Analizer. Инструмент, отображающий затраты ресурсов при работе приложения/сервиса в виде статистики и графиков.
* Test Manager. Встроенный менеджер тестов. После создания теста можно с помощью специального окна запускать и настраивать тесты.
* Extension/Updates Manager. Менеджер плагинов, адаптеров, провайдеров. Позволяет легко найти, установить, обновить любое дополнение.
* Nuget. Система управления пакетами для платформ разработки Microsoft, в первую очередь библиотек .NET Framework. Управляется .NET Foundation. Удобная установка библиотек в любой .Net проект.
* Git Manager. Встроенный менеджер контроля версий. Изначально работал только с Team Foundation Server. Сейчас можно подключить Team Explorer (Название менеджера) к любому репозиторию.
* Присутствуют все необходимые функции для работы с git без запросов.
* Archivator. Архиватор проектов. После того, как проект готов, нужно собрать исполняемый файл. Для каждой технологии реализован свой архиватор. Не нужно устанавливать отдельный софт, чтобы сделать установочник.
* File Manager. Для добавления нового файла в проект существует встроенный менеджер файлов. Удобное создание любых файлов на основе шаблонов. Реализовано большое количество стандартных шаблонов (Пример: класс). Также можно добавлять свои. При установке новой технологии - добавляются соответствующие шаблоны.
* Views. Большое количество различных вкладок для отображения различной полезной информации, вроде списка "GOTO", или отображения данных объекта в Debug режиме.
* Customization. Возможность изменить внешний вид Visual Studio под себя. Изменения цветов, темы, шрифтов, отступов и т. д. Расположение окон в удобном вам виде.
* Setting. Настройка всего вышеперечисленного функционала. Настройка быстрых клавиш, уведомлений, быстрый запуск, стартового окна, вкладок, разметки языков и много другого

Благодаря огромному количеству настроек, поддерживаемых технологий, быстродействию и удобству Visual Studio считается одной из лучших сред разработки.

1. Microsoft SQL Server

* Масштабирование системы. Взаимодействовать с ней можно как на простых ноутбуках, так и на ПК с мощным процессором, который способен обрабатывать большой объем запросов.
* Размер страниц – до 8 Кб. Данные извлекаются быстро, а сложную информацию удобнее хранить. Система обрабатывает транзакции в интерактивном режиме, есть динамическая блокировка.
* Автоматизация рутинных административных задач. Например, управление блокировками и памятью, редактура размеров файлов. В программе продуманы настройки, можно создавать профили пользователей.
* Удобный поиск. Его можно осуществлять по фразам, словам, тексту либо создавать ключевые индексы.
* Поддержка работы с другими решениями Майкрософт, в том числе с Excel, Access.

1. Figma

* веб-версия и приложения под разные операционные системы
* бесплатный тариф
* неограниченное файловое хранилище
* командная работа
* интерактивные макеты
* редактор кривых
* библиотека компонентов
* импорт из Sketch
* экспорт SVG

## 3.2 Разработка программного обеспечения

Для создания нового проекта следует запустить приложение MS Visual Studio (Рисунок 3.1).

В рабочей области приложения необходимо выбрать «создание нового проекта» и выбрать пункт «Приложение WPF (.Net Framework)».

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3.1 – Создание проекта

В окне «Настроить новый проект» следует заполнить все необходимые поля

(Рисунок 3.2).

Изображение выглядит как текст, монитор, снимок экрана, экран

Автоматически созданное описание

Рисунок 3.2 – Настройка нового проекта

Для удобства навигации по проекту необходимо создать структуру из директорий (Рисунок 3.3).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3.3 – Директории проекта

Далее следует установить графическую библиотеку «Material Design».

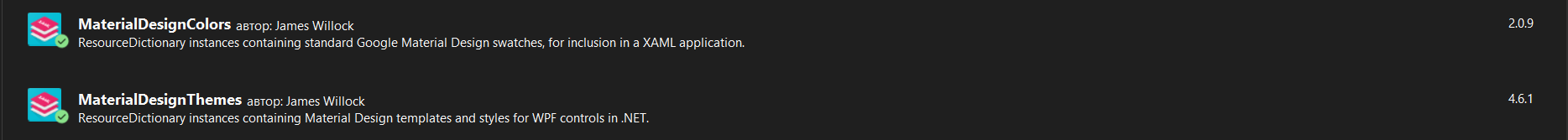


Рисунок 3.4 – Необходимые библиотеки

Интегрировать библиотеку в страницы и окна проекта по инструкции на официальном сайте библиотеки в сети Интернет.

Для реализации переходов между страницами, а также отображения базы данных нужно создать статичный класс «FrameNavigate» в директории «Core».

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3.5 – Класс FrameNavigate

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3.5 – Содержимое класса FrameNavigate

Создание визуальной части программного продукта отображено в рисунках 3.6–3.11

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3.6 – Разметка основного окна (1)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3.7 – Разметка основного окна (2)

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3.8 – Внешний вид главного окна

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3.9 – Разметка страницы администратора

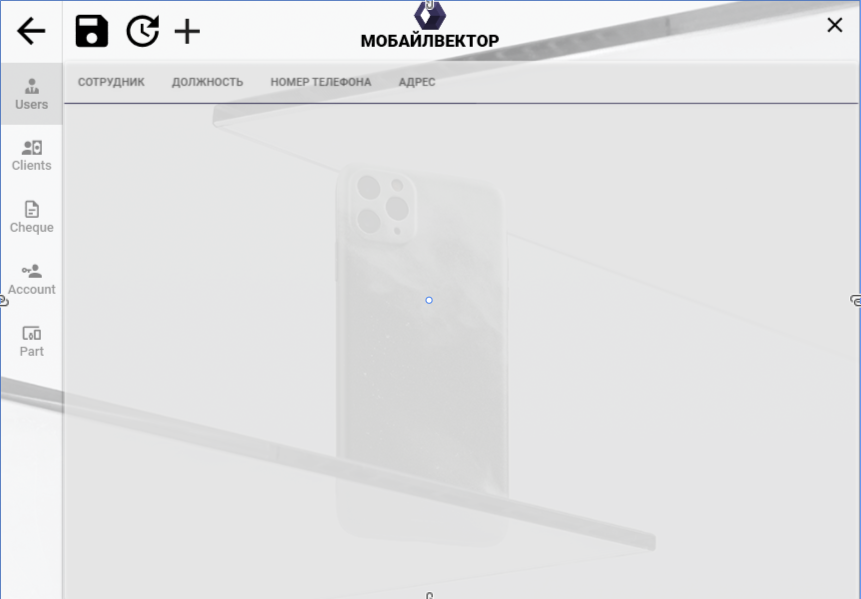


Рисунок 3.10 – Внешний вид страницы администратора

Страница пользователя в данном проекте создается аналогично



Рисунок 3.11 – Внешний вид страницы пользователя

## 3.3 Технико-экономические показатели

В состав основных технико–экономических показателей входят:

1. Затраты на программное обеспечение и аппаратное обеспечение;
2. Затраты на услуги и персонал;
3. Расчёт годовой прибыли.

Расчёт затрат на требуемое в процессе разработки ПП для автоматизации магазина одежды программное обеспечение представлен в таблице 3.1.

Таблица 3.1 – Затраты на ПО

|  |  |  |
| --- | --- | --- |
| Номер | Наименование | Цена |
| 1 | MS Visual Studio Enterprise | 17 000₽ |
| 2 | Figma | 0 ₽ |
| 3 | Web browser MS Edge | 0 ₽ |
| 4 | MS SQL Server | 0 ₽ |

Далее следует рассчитать затраты на аппаратное обеспечение (таблица 3.2)

Таблица 3.2 – Затраты на аппаратное обеспечение

|  |  |  |
| --- | --- | --- |
| Номер | Наименование | Цена |
| 1 | Компьютер Core i5 | 27 000 ₽ |
| 2 | ИБП (650 Вт/ч) | 4 000 ₽ |
| 3 | Компьютерная мышь USB | 1 700 ₽ |
| 4 | Коврик для мыши | 1 000 ₽ |
| 5 | Клавиатура USB | 7 000 ₽ |
| 6 | Принтер со сканером | 7 000₽ |
| 7 | Монитор | 25 500 ₽ |

Проведем расчет затрат на услуги (написание кода, работа программиста), персонал (в лице программиста, ведущего сопровождение ПО на этапе внедрения) (таблица 3.3).

Таблица 3.3 – Затраты на персонал, услуги

|  |  |  |
| --- | --- | --- |
| Номер | Наименование | Цена |
| Услуги | | |
| 1 | Разработка ПО | 65 000 ₽ |
| Персонал | | |
| 1 | З/П | 30 000 ₽ |

Произведя все вычисления, связанные с затратами на разработку получена сумма в размере 194 700 рублей.

Помимо затрат, главным составляющим процесса разработки ПО является расчет годовой прибыли, а соответственно и сроки окупаемости программного продукта.

Расчет годовой прибыли следует выполнить по кварталам, с расчетом Дохода, Расхода и Прибыли (таблица 3.4). Для расчета Дохода требуется рассмотреть количество проданного ПО за каждый квартал и стоимость разработанного ПО за единицу, в данном случае 30 000 рублей по формуле:

Доход = Количество, проданного ПО \* стоимость за единицу (3.1)

К расходам следует отнести заработную плату разработчику в количестве 30 000 рублей в месяц, соответственно 60 000 рублей поквартально.

С экономической точки зрения Прибыль – это разность между доходами и расходами, таким образом в таблице 3.4 в строке «Прибыль» выполнен расчёт.

Таблица 3.4 – Годовая прибыль

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Расчет годовой прибыли | | | | | |
|  | Квартал 1 | Квартал 2 | Квартал 3 | Квартал 4 | Год |
| Кол-во проданного ПО | 7 шт | 12 шт | 54 шт | 34 шт | 107 шт |
| Доход | 210 000 ₽ | 360 000 ₽ | 1 620 000 ₽ | 1 020 000 ₽ | 3 210 000 ₽ |
| Расход | 60 000 ₽ | 60 000 ₽ | 60 000 ₽ | 60 000 ₽ | 240 000 ₽ |
| Прибыль | 150 000 ₽ | 300 000 ₽ | 1 560 000 ₽ | 960 000 ₽ | 2 970 000 ₽ |

Срок окупаемости – ключевой фактор в оценке реализации и внедрения программного продукта. Заказчику важно определить, сколько времени потребуется на то, чтобы «выйти в плюс». Для определения срока окупаемости используется формула (3.2):

РР = K0 / ПЧгс  (3.2)

РР – срок окупаемости, выраженный в годах;

K0 – сумма вложенных средств;

ПЧгс – чистая годовая прибыль.

РР = 194 700 / 2 970 000 = 0, 06 (3.3)

Таким образом срок окупаемости составляет 1 месяц, что говорит о прибыльности и рентабельности созданного программного продукта.

ЗАКЛЮЧЕНИЕ

В ходе выполнения курсовой работы был разработан программный продукт, а именно платформа информационной системы учета ремонта мобильных устройств «МОБАЙЛВЕКТОР», целью которого разработка информационной системы сервисного центра. При разработке данного программного продукта были выполнены следующие задачи:

* составлено техническое задание;
* созданы эскизы и макеты программного продукта для дальнейшей реализации;
* выбраны технологий и средства разработки программного продукта;
* выполнено программирование ПП;
* выполнены отладка и тестирование.

В ходе разработки программного продукта были использованы следующие программные средства: Visual Studio, Microsoft SQL Server, Figma.

Таким образом цель курсовой работы достигнута, поставленные задачи решены.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Ицик Бен-Ган. Microsoft SQL Server 2012. Основы T-SQL
2. Дж. Скит. C# для профессионалов. Тонкости программирования
3. Joe Duffy. Concurrent Programming on Windows
4. Эрик Фримен и др. Head First. Паттерны проектирования
5. Сергей Тепляков. Паттерны проектирования на платформе .NET
6. Дж.Рихтер. CLR via C#
7. Стивен С. Скиена. Алгоритмы. Руководство по разработке
8. Joe Albahari. Threading in C#
9. Проектирование информационной системы, 2017. [Электронный ресурс]. URL: https://prog.bobrodobro.ru/58593 (дата обращения: 15.12.2022)
10. Алексей Васильев. Программирование на С# для начинающих.

Особенности языка

1. Билл Вагнер. Наиболее эффективное программирование на C#
2. Рой Ошероув. Искусство автономного тестирования с примерами на С#
3. Джеффри Рихтер, Мартен ван де Боспурт. WinRT. Программирование на C# для профессионалов
4. Джон Поль Мюллер. C# для чайников
5. Джуст Виссер. Разработка обслуживаемых программ на языке C#

ПРИЛОЖЕНИЕ А

Результат работы программного обеспечения

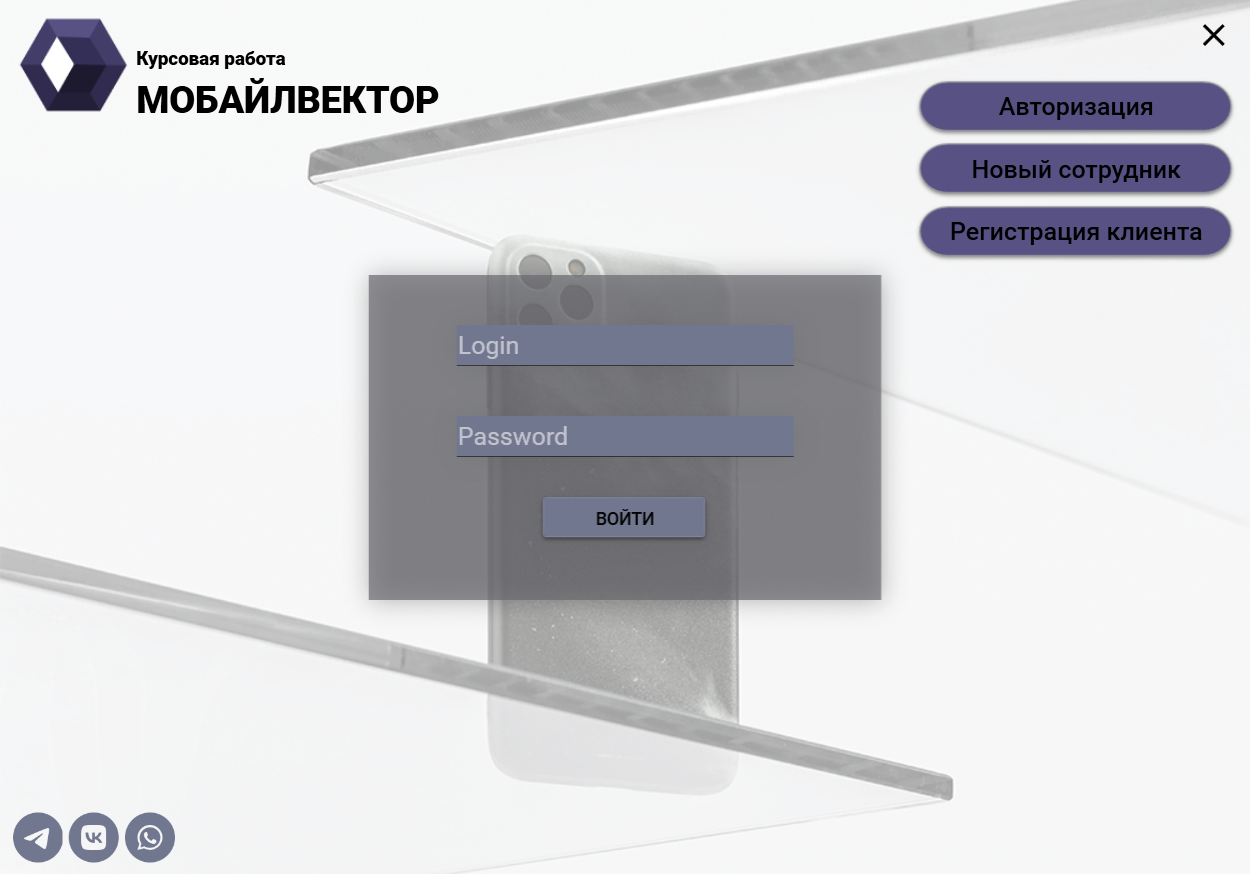


Рисунок A.1 – Внешний вид главного окна

Изображение выглядит как стол

Автоматически созданное описание

Рисунок A.2 – Внешний вид страницы администратора

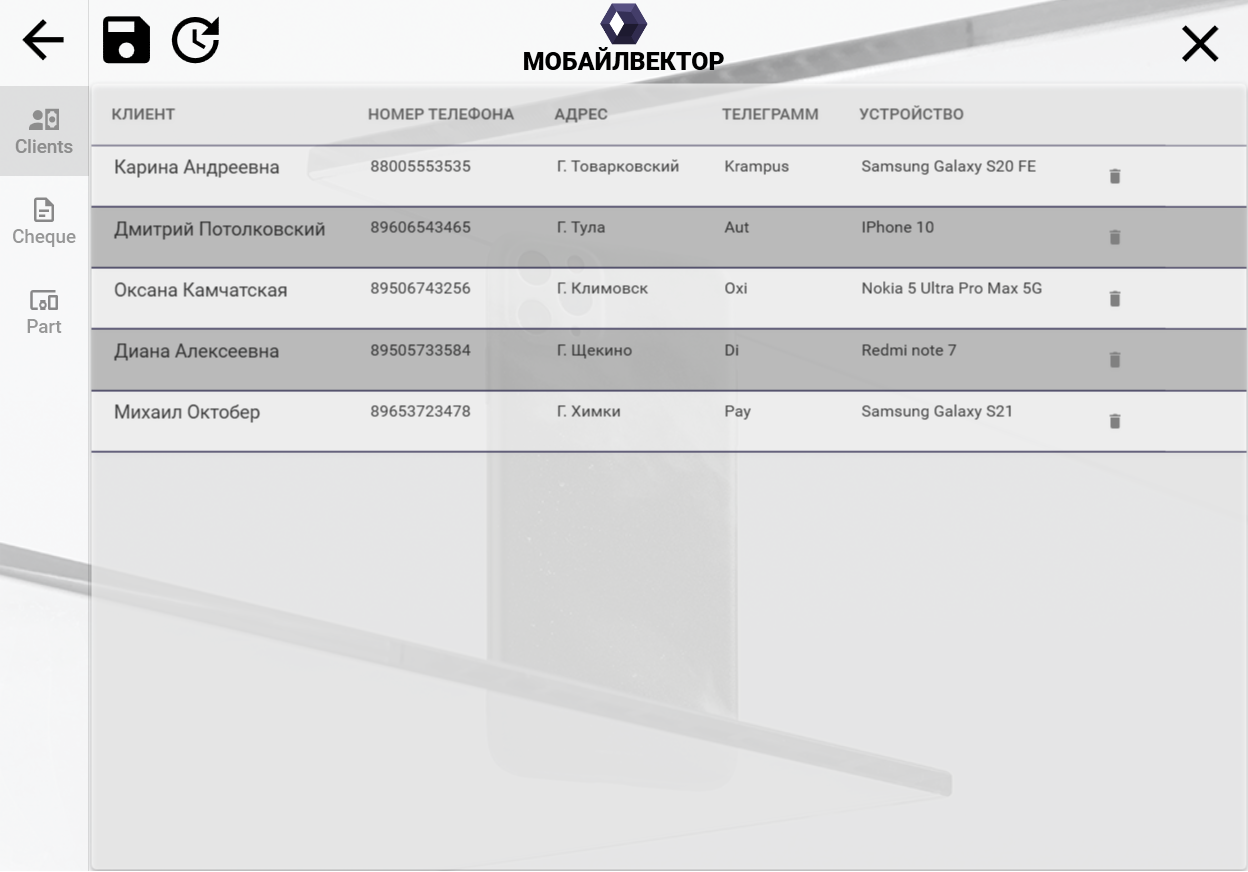


Рисунок A.3 – Внешний вид страницы пользователя

ПРИЛОЖЕНИЕ Б

Листинг программного обеспечения

<Window x:Name="window" x:Class="MOBILEVECTOR.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:materialDesign="http://materialdesigninxaml.net/winfx/xaml/themes"

xmlns:local="clr-namespace:MOBILEVECTOR"

mc:Ignorable="d"

Height="700"

Width="1000"

WindowStyle="None"

WindowStartupLocation="CenterScreen"

AllowsTransparency="True"

Background="Transparent"

FontSize="18"

ResizeMode="NoResize"

FontFamily="{DynamicResource MaterialDesignFont}"

Icon="/View/logo.png" >

<Grid

MouseDown="Grid\_MouseDown" Background="White">

<Grid.Effect>

<DropShadowEffect BlurRadius="30"

ShadowDepth="4"

Opacity="0.4"/>

</Grid.Effect>

<Border

ClipToBounds="True" CornerRadius="10,10,10,10">

<Grid Name="Cap">

<Image Source="/Images/Background.png" OpacityMask="#66FFFFFF" />

<TextBlock FontSize="30"

FontWeight="Black"

Text="МОБАЙЛВЕКТОР"

Foreground="Black" Margin="109,62,591,568"/>

<TextBlock FontSize="15"

FontWeight="Black"

Text="Курсовая работа"

Foreground="Black" Margin="109,38,731,600"/>

<Image x:Name="Logo"

Source="/Images/logo.png"

Width="90"

Height="90"

HorizontalAlignment="Left"

VerticalAlignment="Top" Margin="14,13,0,0"/>

<materialDesign:PackIcon Kind="Close"

Height="30"

Width="30"

HorizontalAlignment="Right"

Margin="0 13 14 0"

MouseDown="PackIcon\_MouseDown"

Cursor="Hand" ToolTip="Закрыть"/>

<StackPanel Orientation="Vertical"

VerticalAlignment="Top"

HorizontalAlignment="Right"

Margin="0 65 14 0">

<Button Name="Btn\_Authorization"

Content="Авторизация"

Background="#585183"

FontSize="20"

Width="250"

Height="40"

materialDesign:ButtonAssist.CornerRadius="22"

Click="Btn\_Authorization\_Click"

Cursor=""/>

<Button x:Name="Btn\_new\_employee"

Content="Новый сотрудник"

Background="#585183"

FontSize="20"

Width="250"

Height="40"

Margin="0 10 0 0"

materialDesign:ButtonAssist.CornerRadius="22"

Click="Btn\_new\_employee\_Click"

Cursor="Hand"/>

<Button x:Name="Btn\_New\_Client"

Content="Регистрация клиента"

Background="#585183"

FontSize="20"

Width="250"

Height="40"

Margin="0 10 0 0"

materialDesign:ButtonAssist.CornerRadius="22"

Click="Btn\_New\_Client\_Click"

Cursor="Hand"/>

</StackPanel>

<StackPanel Orientation="Horizontal"

HorizontalAlignment="Left"

VerticalAlignment="Bottom"

Margin="10 0 0 10">

<Image Source="/Images/telegram.png"

Width="40"

Height="40"

ToolTip="Перейти в телеграм канал МОБАЙЛВЕКТОР"

MouseDown="Image\_MouseDown"/>

<Image Source="/Images/vk.png"

Width="40"

Height="40"

Margin="5 0 0 0"/>

<Image Source="/Images/whatsapp.png"

Width="40"

Height="40"

Margin="5 0 0 0"/>

</StackPanel>

</Grid>

</Border>

<Frame x:Name="MainWindowFrame"

NavigationUIVisibility="Hidden" />

</Grid>

</Window>