

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»
(СПбГУТ)

АРХАНГЕЛЬСКИЙ КОЛЛЕДЖ ТЕЛЕКОММУНИКАЦИЙ
ИМ. Б.Л. РОЗИНГА (ФИЛИАЛ) СПбГУТ
(АКТ (ф) СПбГУТ)

ГРУППОВОЙ ПРОЕКТ

НА ТЕМУ

РАЗРАБОТКА ИНФОРМАЦИОННО-

ОБРАЗОВАТЕЛЬНОГО ЗАЩИЩЕННОГО

ФАЙЛ-СЕРВЕРА «DMN CLOUD»

Л309.23ГП01 ПЗ

(Обозначение документа)

МДК.03.01 Технология разработки программного

обеспечения

Студент	ПКС-01	А.А. Добряков		
	(Группа)	(Подпись)	(Дата)	(И.О. Фамилия)
Студент	ПКС-01	А.А. Дементьев		
	(Группа)	(Подпись)	(Дата)	(И.О. Фамилия)
Студент	ПКС-01	И.Р. Шамсутдинов		
	(Группа)	(Подпись)	(Дата)	(И.О. Фамилия)
Студент	ПКС-01	А.А. Кузнецов		
	(Группа)	(Подпись)	(Дата)	(И.О. Фамилия)
Преподаватель		Д.А.Кривополенов		
		(Подпись)	(Дата)	(И.О. Фамилия)

Архангельск 2023

СОДЕРЖАНИЕ

Перечень сокращений и обозначений.....	3
Введение.....	4
1 Анализ и разработка требований.....	6
1.1 Назначение и область применения.....	6
1.2 Постановка задачи	6
1.3 Описание алгоритма функционирования системы	7
1.4 Выбор состава программных и технических средств	8
2 Проектирование программного продукта	10
2.1 Проектирование причинно-следственной диаграммы.....	10
2.2 Проектирование DFD-диаграммы первого и второго уровня.....	11
2.2 Проектирование UML-диаграммы	13
2.2 Проектирование прототипа программного продукта	14
3 Разработка программного продукта.....	17
3.1 Разработка оконного приложения.....	18
3.2 Создание инсталляционного пакета программного продукта	23
4 Тестирование программного продукта	25
4.1 Системное тестирование методом «чёрного ящика»	25
Заключение	29
Список использованных источников	31

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

В настоящем групповом проекте применяют следующие сокращения и обозначения:

БД – база данных

ОС – операционная система

ПК – персональный компьютер

ПО – программное обеспечение

СУБД – система управления базами данных

IDE – интегрированная среда разработки

LINQ – технология интеграции запросов в язык программирования C#

MS – Microsoft

SQL – язык структурированных запросов

XAML – расширяемый язык разметки

ВВЕДЕНИЕ

Информация в жизни человека является одним из наиболее важных ресурсов и играет большую роль в современном мире. В течение последних 10-15 лет, с расширением рынка ПК и мобильных устройств, значительно увеличился объём электронной информации: у каждого пользователя имеются цифровые фотографии, документы, видео- и аудиозаписи. Впоследствии, возникла острая необходимость в их хранении и защите. Для этих целей актуально и целесообразно создание защищённых файл-серверов.

Информационно-образовательный файл-сервер – это эффективное решение для хранения информации, необходимой для организации образовательного процесса. Незащищённые способы хранения данных могут привести к утечке важных конфиденциальных сведений при несанкционированном доступе злоумышленников или иных третьих лиц.

Целью группового проекта является разработка многопользовательского клиент-серверного информационно-образовательного защищённого файл-сервера «DMN Cloud».

Для достижения поставленной цели требуется решить следующие задачи:

- изучить особенности защищённого хранения данных,
- выполнить сбор требований к файл-серверу,
- проанализировать собранные требования,
- описать спецификацию разрабатываемого ПО,
- спроектировать DFD-диаграммы первого и второго уровней,
- спроектировать причинно-следственную диаграмму,
- спроектировать UML-диаграммы стереотипов и классов,
- выбрать и обосновать выбор средств разработки ПО и БД,
- настроить удалённый FTP-сервер,
- спроектировать структуру локального и облачного хранилища,

- разработать оконное приложение с пользовательским интерфейсом для доступа к локальному и облачному хранилищу,
- реализовать возможность регистрации пользователей в системе,
- реализовать возможность подтверждения адреса электронной почты с использованием почтового SMTP клиента,
- реализовать возможность восстановления пароля пользователя с использованием почтового SMTP клиента,
- реализовать возможность настройки расположения хранилища,
- реализовать возможность просмотра списка хранимых файлов,
- реализовать возможность загрузки, скачивания и удаления файлов,
- выполнить тестирование и отладку разработанного приложения,
- разработать руководство пользователя.

1 Анализ и разработка требований

1.1 Назначение и область применения

Архангельский колледж телекоммуникаций им. Б. Л. Розинга (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича» является учреждением среднего профессионального образования. Во время организации учебного процесса колледж осуществляет различную деятельность, связанную с хранением, передачей и администрированием цифровой информации.

Разрабатываемый информационно-образовательный защищённый файл-сервер предназначен для защищённого хранения образовательных данных в виде файлов пользователей.

1.2 Постановка задачи

Требуется спроектировать и разработать многопользовательскую клиент-серверную систему с оконным интерфейсом пользователя для организации защищённого хранения файлов с образовательными и иными данными.

Система должна обеспечивать работу с файлами, находящимися в хранилище. Должно быть предусмотрено два независимых друг от друга хранилища: одно должно быть расположено на удалённом файл-сервере, доступ к которому предоставляется через FTP-соединение, второе должно являться локальным хранилищем и находиться на ПК пользователя. Необходимо обеспечить автономную работу приложения в общем локальном хранилище без доступа к сети Интернет. Оба хранилища следует защитить паролем, который указывается пользователем.

Для хранения учётных данных пользователей необходимо создать БД.

На форме регистрации должны присутствовать обязательные поля ввода для входа в систему: логин, адрес электронной почты, пароль, подтверждение пароля, а также необязательные: имя, номер телефона. Заполненные в полях ввода данные должны проверяться на корректность. Реализовать отправку кода проверки на электронную почту пользователя для подтверждения регистрации на указанный адрес.

На форме авторизации должна быть реализована возможность перехода к основному окну программы, используя связку «логин-пароль» или «адрес электронной почты - пароль». Требуется реализовать переключатель «Запомнить вход», который сохраняет данные последнего авторизованного в системе пользователя.

При смене пароля учётной записи пользователя, необходимо отправлять соответствующее уведомление на электронную почту.

Также необходимо добавить возможность восстановления пароля посредством отправки на электронную почту проверочного кода.

1.3 Описание алгоритма функционирования системы

Система должна иметь клиент-серверную организацию: на сервере установлены БД и FTP-сервер, соединение к которым выполняется удалённо по сети. На клиентских ПК использование системы происходит через интерфейс оконного приложения, состоящего из нескольких блоков:

- авторизация,
- регистрация,
- восстановление пароля,
- управление файлами.

1.4 Выбор состава программных и технических средств

Согласно цели проекта, требуется создать многопользовательскую

клиент-серверную систему для защищённого хранения данных.

Работа с системой будет осуществляться на ПК с установленной ОС MS Windows версии не ниже 10.

Для проектирования диаграмм выбран онлайн-сервис draw.io, который имеет ряд преимуществ:

- распространение на бесплатной основе,
- удобный, дружелюбный к новым пользователям интерфейс,
- возможность проектировать различные виды диаграмм.

В качестве СУБД выбрана MySQL, так как её главное преимущество – простота использования. Она имеет интуитивно понятный интерфейс и простой язык запросов, который позволяет пользователям легко создавать, изменять и управлять БД. MySQL также поддерживает множество различных языков программирования, что делает его удобным для разработчиков. Также MySQL имеет кроссплатформенный инструмент проектирования – MySQL Workbench.

В качестве FTP-сервера выбран сервер хостинга Sprinthost, по причине его удобства и безлимитного плана на создание FTP и БД.

Приложение будет написано на языке программирования C#, так как это простой, надежный и масштабируемый язык программирования. Динамически типизированный характер C# облегчает разработчикам поиск ошибок в коде. Данный язык содержит простые в применении компоненты для работы базами данных, в частности - с СУБД MySQL.

Для разработки оконного приложения выбрана IDE MS Visual Studio, так как она имеет удобный инструментарий создания оконных приложений на основе фреймворка Windows Presentation Foundation, использующего язык разметки XAML.

Для функционирования системы на стороне сервера достаточны следующие программные и технические средства:

- операционная система Windows 10 / 11 / Windows Server 2016 или более поздней версии,

- сервер БД: MySQL версии не ниже 8.0,
- программное обеспечение для конфигурирования, управления и администрирования сервера БД: MySQL Workbench версии не ниже 8.0,
- процессор x64 с тактовой частотой 2,0 ГГц и выше,
- не менее 6 ГБ свободного места на жёстком диске,
- не менее 4 ГБ оперативной памяти,
- .NET версии не ниже 6.0,
- манипуляторы: клавиатура и мышь.

Для функционирования системы на стороне клиента достаточны следующие программные и технические средства:

- операционная система Windows 10 или более поздней версии,
- процессор x64 с тактовой частотой 1,4 ГГц и выше,
- не менее 1 ГБ оперативной памяти для 32-разрядных систем и не менее 2 ГБ для 64-разрядных систем,
- .NET версии не ниже 6.0,
- манипуляторы: клавиатура и мышь.

2 Проектирование программного продукта

2.1 Проектирование причинно-следственной диаграммы

Причинно-следственные диаграммы – это графический способ исследования и определения наиболее существенных причинно-следственных взаимосвязей между причинами (факторами) и последствиями в исследуемой ситуации или проблеме.

Необходимо спроектировать причинно-следственную диаграмму для понимания функциональности программы (рисунок 1).

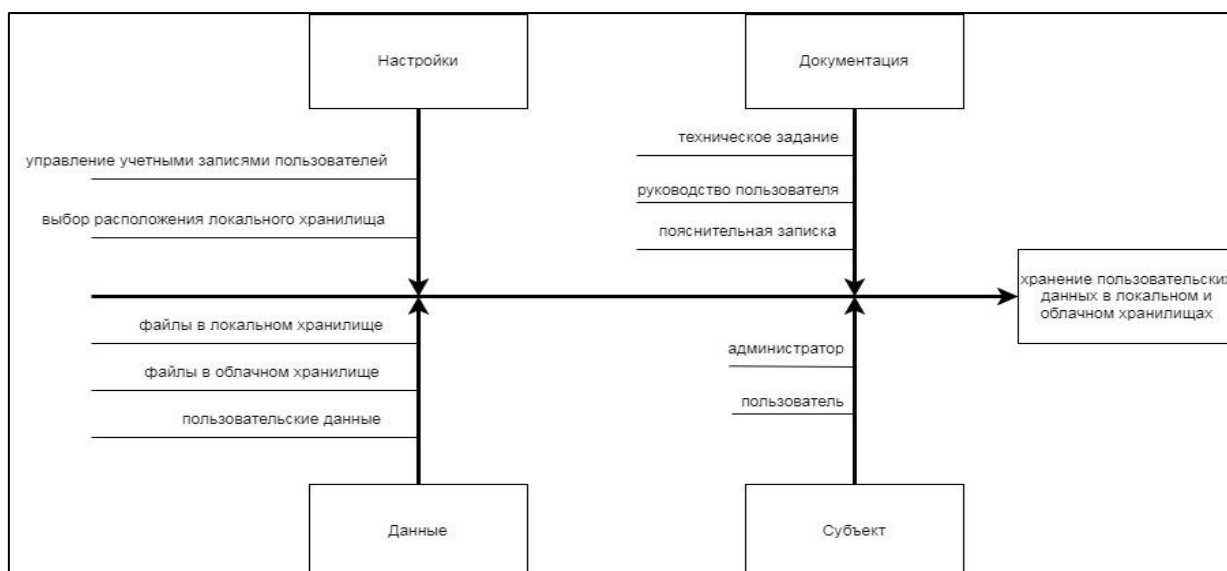


Рисунок 1 — Причинно-следственная диаграмма

Причинно-следственная диаграмма показывает, что программа предусматривает две роли: пользователь и администратор. Для описания программы предусмотрены: пояснительная записка, техническое задание и руководство пользователя. Проблемой разрабатываемого приложения является хранение файлов в локальном и облачном хранилищах. Данная проблема отображена на диаграмме длинной стрелкой, а другие стрелки, примыкающие к основной, усугубляют проблему.

2.2 Проектирование DFD-диаграммы первого и второго уровня

DFD-диаграммы наглядно отображают течение информации в пределах процесса или системы. На рисунке 2 продемонстрирована DFD-диаграмма первого уровня.



Рисунок 2 — DFD-диаграмма первого уровня

На диаграмме показано, как пользователь взаимодействует с программой, отправляя запрос на получение данных из БД и хранилища, где происходит обработка запроса и формируется ответ. Для более подробного представления об элементах контекстной диаграммы сформирована DFD-диаграмма второго уровня, которая продемонстрирована на рисунке 3.

Данная диаграмма демонстрирует взаимодействие пользователя с программой. Пользователь отправляет запрос на открытие программы, получает ответ и попадает на страницу авторизации.

На странице авторизации у пользователя есть возможность зарегистрироваться в приложении или войти в существующую учётную запись.

В случае, если пользователь забудет пароль от аккаунта, у него имеется возможность восстановить данные для входа. Восстановление пароля происходит путём отправки на электронную почту, привязанную к аккаунту, письма с автоматически сгенерированным кодом восстановления. При успешном вводе кода приложение предлагает пользователю сменить пароль на новый.

2.3 Проектирование UML-диаграммы стереотипов

UML – это модель, описывающая часть системы. Она используется для определения функциональности или дизайна системы. Диаграмма должна быть чёткой и лаконичной, чтобы зритель мог без затруднений её понять. Для наглядного описания алгоритма работы приложения необходимо составить UML-диаграмму стереотипов. На рисунке 4 изображена UML-диаграмма стереотипов, созданная на этапе проектирования программного продукта.

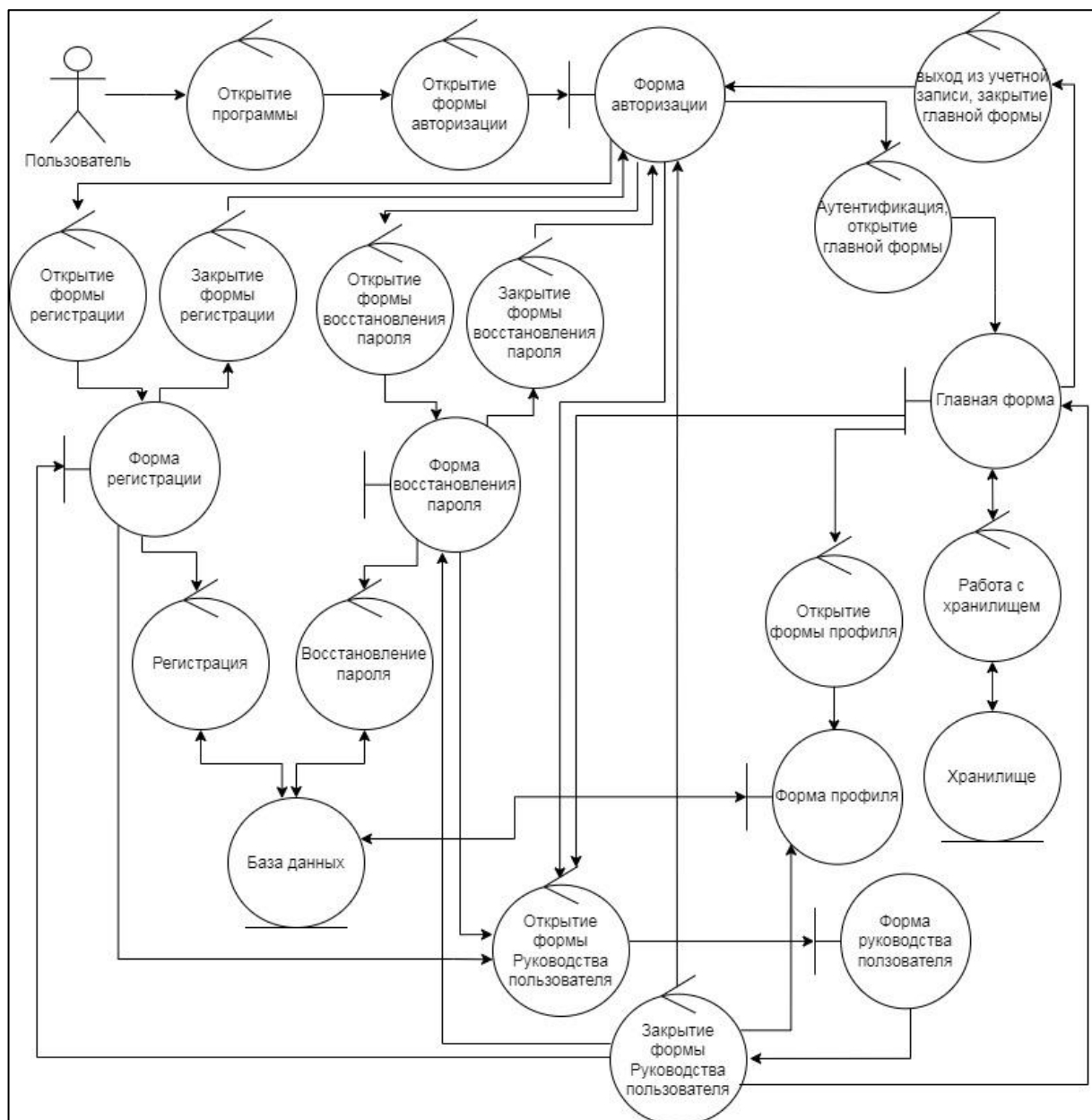


Рисунок 4 — UML-диаграмма стереотипов

Данная диаграмма отображает обращение пользователя к формам «Авторизация», «Регистрация», «Восстановление пароля», «Главная форма», «Руководство пользователя». С помощью формы авторизации пользователь может запустить процессы аутентификации, восстановления пароля или регистрации нового пользователя. При успешной авторизации запускается главная форма приложения, которая запускает процесс вывода информации о файлах в локальном хранилище и файлов, расположенных на файловом сервере приложения.

2.4 Проектирование прототипа программного продукта

Прототип программного продукта – это предварительная версия разрабатываемого программного продукта, созданная с целью демонстрации его функциональности и внешнего вида. Прототип позволяет проверить идеи, концепции и требования, а также получить обратную связь от пользователей и заинтересованных сторон до полного завершения разработки. Он может быть создан в виде простого интерфейса, макета или даже рабочей модели со схожими функциями, но без полной реализации всех возможностей. Прототип помогает определиться с дизайном, функциональностью и улучшить пользовательский опыт до финальной версии программного продукта.

В ходе проектирования разработаны прототипы главной формы программного продукта «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud» и окна авторизации (рисунки 5-7), на которых показана структура форм, предназначенных для работы с файлами локального хранилища и файлами файлового сервера программы.

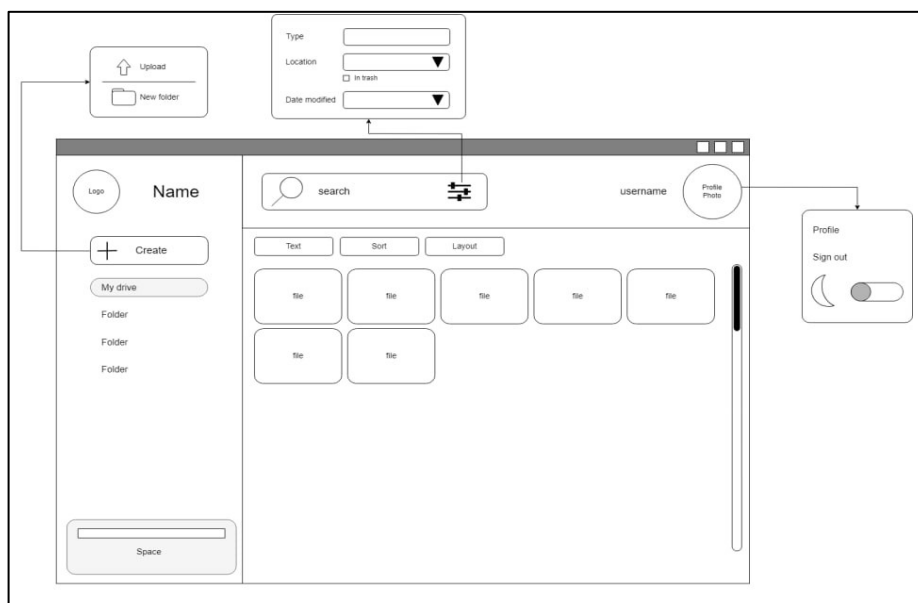


Рисунок 5 — Программа «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud». Вид прототипа главной формы

The image shows a prototype of a 'Sign In' window. At the top, there is a header bar with a circular 'Logo' placeholder and a 'Name' input field. Below the header, the title 'Sign In' is centered. The main content area contains a 'Login' label above a text input field, followed by a 'Password' label above another text input field. Below these fields is a 'Sign In' button. At the bottom of the form area, there are two links: 'Sign up' and 'Forgot Password?'. The entire window is enclosed in a standard desktop window frame with three control buttons in the top right corner.

Рисунок 6 — Программа «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud». Вид прототипа окна авторизации

The image displays two side-by-side prototypes of a 'Sign Up' window. Both windows have a header bar with a 'Logo' placeholder and a 'Name' input field. The left prototype is titled 'Sign up' and features four input fields labeled 'Username', 'Password', 'Confirm password', and 'E-mail', followed by a 'Register' button. The right prototype is titled 'Sign Up' and features the same four input fields, but with an additional 'Code' input field and a 'Register' button. Below the 'Code' field, the text 'Code sent to E-Mail' is displayed. Both windows are framed with standard desktop window controls.

Рисунок 7 — Программа «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud». Вид прототипа окна регистрации

3 Разработка программного продукта

3.1 Разработка оконного приложения

В процессе разработки ПП «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud» создано несколько окон, которые служат для выполнения задач, поставленных на этапе проектирования.

При запуске приложения открывается окно «Авторизация» (рисунок 8). С данного окна возможен переход на окна «Регистрация» (рисунок 9), «Восстановление пароля» (рисунок 10) и «Руководство пользователя» (рисунок 11).

Окно «Авторизация» является стартовым и предназначено для входа в систему для зарегистрированных пользователей.

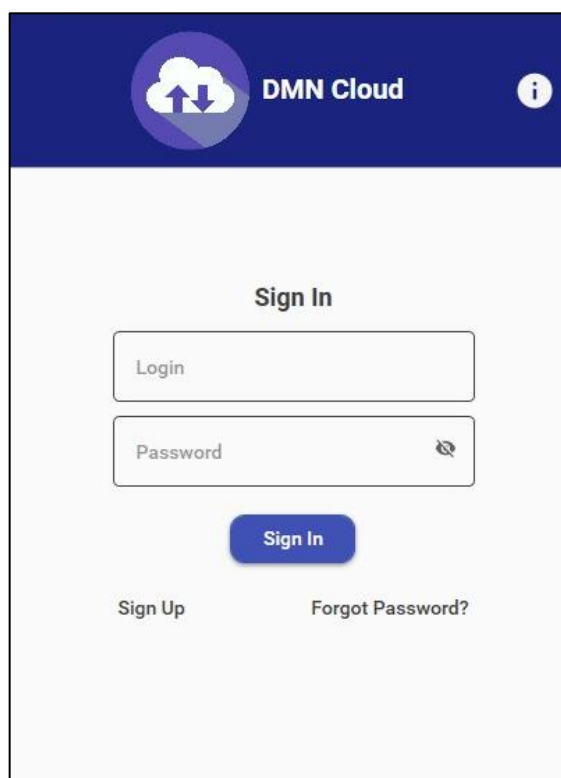
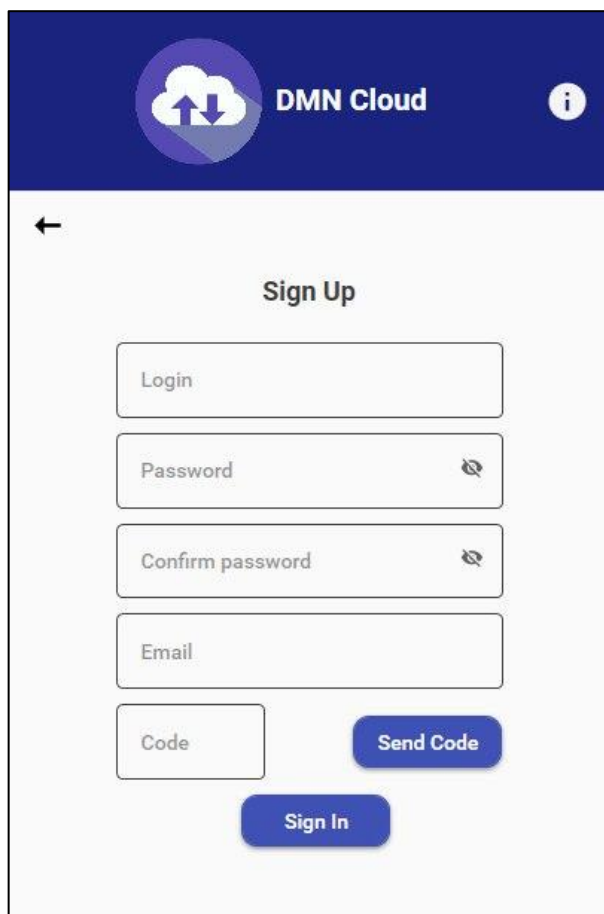


Рисунок 8 — Программа «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud». Вид окна авторизации

С помощью окна «Регистрация» пользователь может создать учётную запись. Для регистрации необходимо заполнить поля «Логин», «Пароль», «Подтверждение пароля», «Адрес электронной почты». Также пользователь должен подтвердить свою электронную почту с помощью письма с проверочным кодом, которое отправляется посредством применения почтового SMTP-клиента.



The image shows a mobile application interface for DMN Cloud. At the top, there is a dark blue header with a cloud icon containing an up and down arrow, the text 'DMN Cloud', and an information icon. Below the header is a light gray area with a back arrow in the top left. The title 'Sign Up' is centered. The form consists of four input fields: 'Login', 'Password' (with an eye icon for toggling visibility), 'Confirm password' (with an eye icon), and 'Email'. Below these fields are two buttons: 'Send Code' (blue) and 'Sign In' (blue). There is also a 'Code' input field next to the 'Send Code' button.

Рисунок 9 — Программа «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud». Вид окна регистрации

В случае, если пользователь не может войти в свою учётную запись, есть возможность восстановить доступ к аккаунту, перейдя на окно «Сброс пароля» путём нажатия на ссылку «Забыли пароль?». Для сброса пароля достаточно заполнить поле «Адрес электронной почты» и нажать кнопку «Отправить код». Далее пользователю на почту придёт письмо с проверочным кодом. При

введении кода из письма пользователю предлагается ввести новый пароль для своей учётной записи.

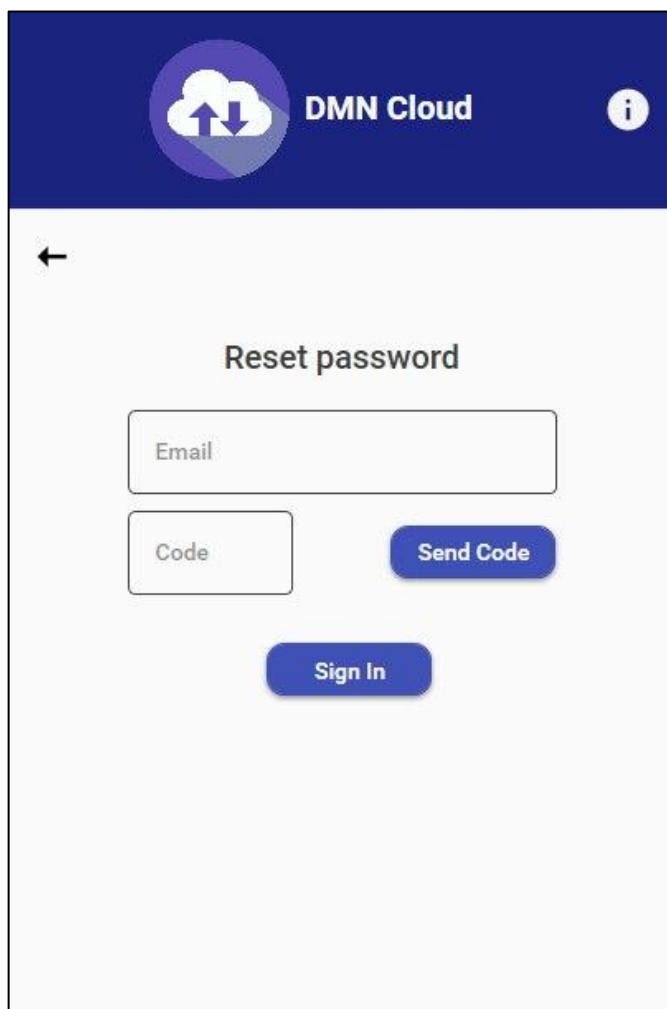
The image shows a mobile application interface for 'DMN Cloud'. At the top is a dark blue header with a logo on the left (a cloud with two arrows pointing up and down) and the text 'DMN Cloud' in the center. To the right of the text is a small white circle containing an 'i' icon. Below the header is a light gray area. In the top left of this area is a black back arrow. Centered in this area is the text 'Reset password'. Below this text are three input fields: a large one labeled 'Email', a smaller one labeled 'Code', and a blue button labeled 'Send Code' to the right of the 'Code' field. At the bottom center of the form area is a blue button labeled 'Sign In'.

Рисунок 10 — Программа «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud». Вид окна восстановления пароля

Окно «Руководство пользователя» содержит в себе текстовую информацию, позволяющую пользователю разобраться в принципе работы с программным продуктом.

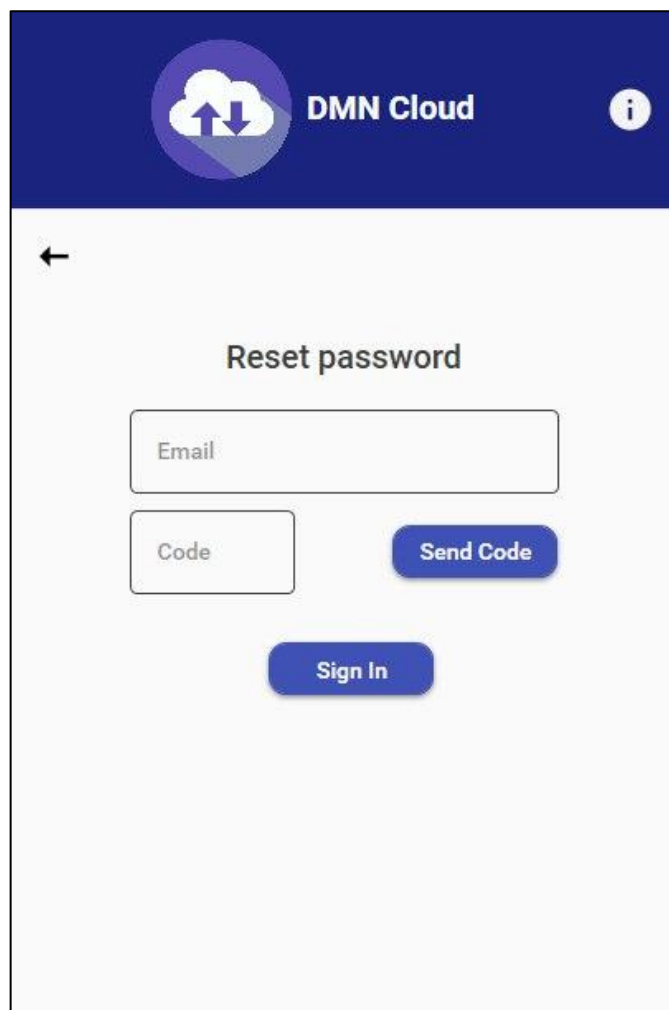


Рисунок 11 — Программа «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud». Вид окна «Руководство пользователя»

При успешной авторизации пользователь попадает на главное окно программы, изображённое на рисунке 12.

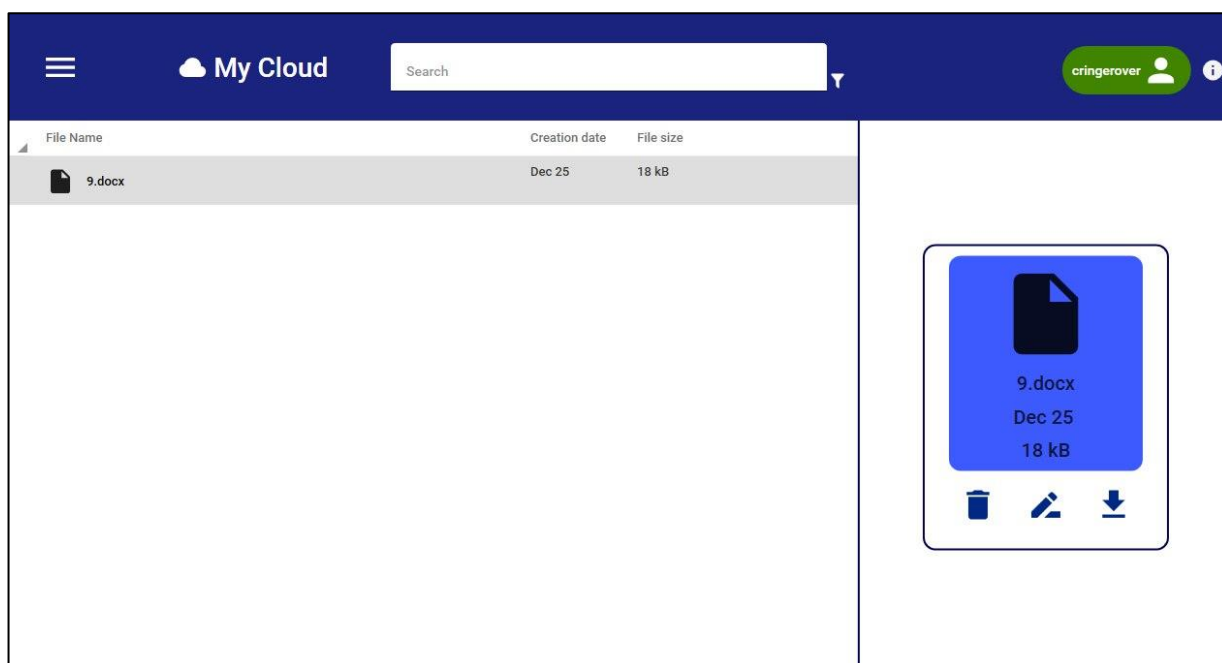


Рисунок 12 — Программа «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud». Вид главного окна

Главная форма приложения отображает меню взаимодействий пользователя с файлами на файловом сервере. Реализована возможность переключения между облачным и локальным хранилищем. Пользователь имеет возможность производить различные действия с файлом с помощью кнопок «Удалить», «Переименовать», «Скачать».

В верхней части интерфейса главного окна расположена поисковая строка, кнопка «Руководство пользователя» и кнопка вызова бокового меню. На боковом меню расположены кнопки переключения между файл-сервером и локальным хранилищем, а также информационная панель, показывающая объём свободного места в хранилище.

В основном меню отображён список файлов, расположенных в хранилище. Метод `GetCloudFiles`, предназначенный для получения списка файлов в облачном хранилище, представлен листингом 1.

Листинг 1 – код метода GetCloudFiles

```
private void GetCloudFiles()
{
    FtpClient ftpClient = new FtpClient("ftp://141.8.192.82",
    "a0885391", "maepsiugur");
    Database database = new Database();

    Regex regex = new Regex(@"^([d-])([rwxt-
]{3}){3}\s+\d{1,}\s+.*?(\d{1,})\s+(\w+\s+\d{1,2}\s+(?:\d{4})?) (\d{1,2}
):\d{2})?\s+(.+?)\s?$",
    RegexOptions.Compiled | RegexOptions.Multiline |
    RegexOptions.IgnoreCase | RegexOptions.IgnorePatternWhitespace);

    ftpClient.ChangeWorkingDirectory(UserName);
    List<FileDirectoryInfo> list =
ftpClient.ListDirectoryDetails()
    .Select(s =>
    {
        Match match = regex.Match(s);
        if (match.Length > 5)
        {

            string size = "";
            size = (Int32.Parse(match.Groups[3].Value.Trim()) /
1024).ToString() + " kB";

            return new FileDirectoryInfo(size,
match.Groups[6].Value, match.Groups[4].Value, ftpClient.uri);
        }
        else return new FileDirectoryInfo();
    }).ToList();

    list.RemoveRange(0, 2);

    CloudFileDataGrid.ItemsSource = list;
    CloudFileDataGrid.SelectedIndex = 0;
    fileList = list;
}
```

Каждый загружаемый в хранилище файл доступен для скачивания. Также интерфейс выводит следующую информацию о файлах:

- название файла,
- дата загрузки файла,
- размер файла.

Код метода UploadFileButton_Click представлен в листинге 2 и предназначен для загрузки файлов на файл-сервер.

Листинг 2 – код метода UploadFileButton_Click

```
private void UploadFileButton_Click(object sender, RoutedEventArgs e)
{
    try
    {
        FtpClient ftpClient = new
FtpClient("ftp://141.8.192.82", "a0885391", "maepsiugur");
        OpenFileDialog openFileDialog = new
OpenFileDialog();

        if (openFileDialog.ShowDialog() == true)
        {
            ftpClient.UploadFile(openFileDialog.FileName,
${UserName}/{openFileDialog.SafeFileName}");
            MessageWindow messageWindow = new
MessageWindow("Success!");
            messageWindow.Show();
            GetCloudFiles();
        }

    }
    catch (Exception ex)
    {
        MessageWindow messageWindow = new(ex.Message);
        messageWindow.Show();
    }
}
```

3.2 Создание инсталляционного пакета программного продукта

Инсталляция ПО – это процесс установки программного обеспечения на компьютер или другое устройство. В ходе инсталляции происходит копирование файлов программы на жесткий диск, создание необходимых системных записей и настройка окружения для работы программы.

Инсталляция ПО может включать в себя выбор компонентов для установки, настройку параметров программы, проверку системных требований, а также предоставление пользователю информации о лицензионном соглашении и других условиях использования программы.

Для создания инсталляционного пакета установлено расширение MS Visual Studio 2022 «Installer Project».

Внешний вид проекта установочного пакета изображён на рисунке 13.

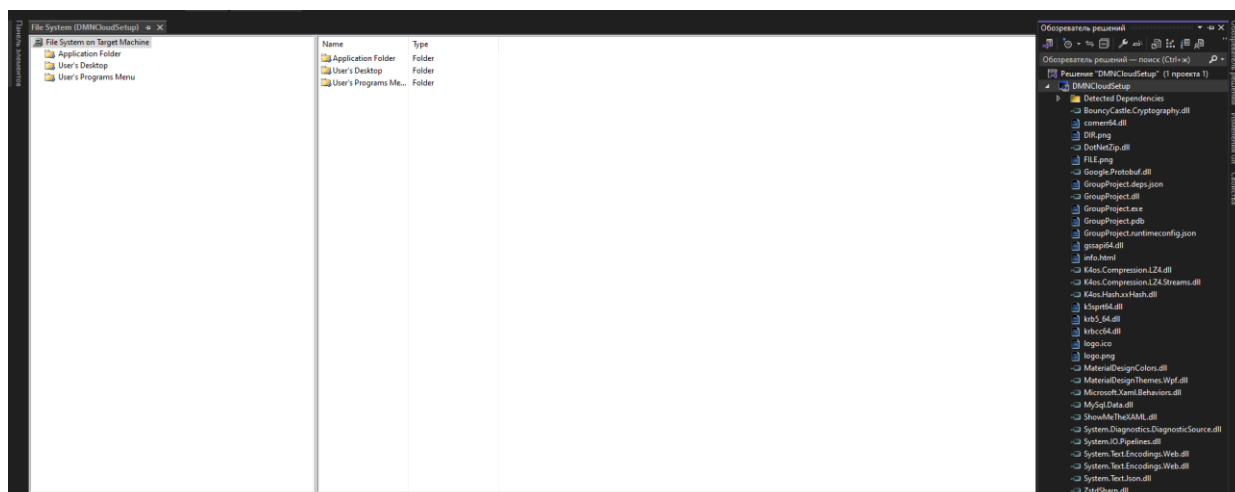


Рисунок 13 — MS Visual Studio 2022 «Installer Project». Вид интерфейса меню создания инсталляционного пакета

Далее в узел «Application's Folder» добавлены файлы приложения. Эти файлы являются основными и должны находиться в папке программы.

В узлах «User's Desktop» и «User's Programs Menu» созданы ярлыки исполняемого файла программы. Данные ярлыки появляются на рабочем столе и панели программ пользователя при установке.

Для установки и последующего использования программы используется файл-установщик с названием «DMNCloudSetup.msi». После распаковки файлов на ПК установится исполняемый файл программы «GroupProject.exe».

4 Тестирование программного продукта

4.1 Системное тестирование методом «чёрного ящика»

Тестирование методом «черного ящика» – это метод тестирования программного обеспечения, при котором тестирующий не имеет доступа к внутренней структуре или реализации тестируемой системы. Вместо этого, тестирование проводится на основе внешних спецификаций и требований к системе. В таблице 1 показан набор тестов при разработке оконного приложения.

Таблица 1 – Набор тестов разработанного приложения

Действие	Ожидаемый результат	Полученный результат
Запустить программу, перейти на окно регистрации, в полях Логин, Пароль ввести «123», в поле E-mail ввести «agr4ce@gmail.com»	Получение на электронную почту письма с проверочным кодом	Совпадает с ожидаемым
Открыть окно авторизации, в поле Логин написать «123», в поле Пароль написать «123», нажать кнопку «Войти»	Открытие главного окна приложения	Совпадает с ожидаемым
Открыть главное окно, нажать кнопку открытия бокового слайдера, нажать кнопку «Загрузить», в диалоговом окне выбрать файл «1.txt»	Загрузка файла «1.txt» в облачное хранилище и появление в списке файлов	Совпадает с ожидаемым

Продолжение таблицы 1

Действие	Ожидаемый результат	Полученный результат
Открыть главное окно, нажать кнопку бокового слайдера, нажать кнопку «Локальное хранилище», нажать кнопку «Загрузить», в диалоговом окне выбрать файл «2.txt»	Загрузка файла «2.txt» в облачное хранилище и появление в списке файлов	Совпадает с ожидаемым
Открыть главное окно, нажать на пункт «1.txt», в правом боковом меню нажать кнопку «Удалить»	Удаление файла «1.txt» с файл-сервера	Совпадает с ожидаемым
Открыть главное окно, нажать кнопку открытия бокового слайдера, нажать кнопку «Локальное хранилище», нажать на пункт «2.txt», в правом боковом меню нажать кнопку «Скачать», в диалоговом окне выбрать путь C://Temp	Появление файла «2.txt» по пути C://Temp	Совпадает с ожидаемым
Запустить программу, нажать кнопку «Забыли пароль?», ввести в поле адрес электронной почты, ввести в поле полученный на почту код, ввести новый пароль	Смена пароля для учетной записи, привязанной к указанному адресу электронной почты	Совпадает с ожидаемым

Продолжение таблицы 1

Действие	Ожидаемый результат	Полученный результат
Запустить программу с отключенным интернет-соединением, ввести данные в поля Логин и Пароль, нажать кнопку «Войти»	Обработанное исключение с указанием на отсутствующее интернет-соединение	Необработанное исключение
Открыть главное меню, нажать кнопку открытия бокового слайдера нажать кнопку загрузить, выбрать файл «1.txt», повторить загрузку файла «1.txt» во второй раз	Необработанное исключение	Файл перезаписан
Запустить программу и войти с одного аккаунта на двух разных ПК	Необработанное исключение	Нормальное функционирование программы
Запустить программу, нажать кнопку «Руководство пользователя»	Открытие руководства пользователя во встроенном браузере	Совпадает с ожидаемым
Открыть окно регистрации, ввести логин «test1», пароль «test123», подтверждение пароля «test321», E-mail «testemail@gmail.com» нажать кнопку «Зарегистрироваться»	Обработанное исключение, вывод сообщения «Введённые пароли не совпадают»	Совпадает с ожидаемым

Продолжение таблицы 1

Действие	Ожидаемый результат	Полученный результат
Открыть окно регистрации, ввести логин «test1», пароль «test123», подтверждение пароля «test123», E-mail «testemail»	Обработанное исключение, вывод сообщения «Введён некорректный E-mail»	Совпадает с ожидаемым
Запустить приложение, на стартовом окне ввести логин «test2», пароль «test321»	Вывод сообщения «Пользователь с введённым логином/паролем не найден»	Совпадает с ожидаемым
Открыть главное окно, нажать на пункт «1.txt», в правом боковом меню нажать на название файла, в текстовом поле написать «123.txt», нажать кнопку «Переименовать»	Переименование файла «1.txt» на «123.txt»	Совпадает с ожидаемым

В ходе тестирования выявлено одно необработанное исключение. В целях исправления ошибки, в обработчики событий основных функциональных кнопок была добавлена проверка на наличие интернет-соединения.

ЗАКЛЮЧЕНИЕ

В результате работы была достигнута основная цель группового проекта, т.е. разработан программный продукт «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud».

В процессе разработки были задействованы различные информационные ресурсы для расширения теоретических знаний о среде разработки и языке программирования C#.

В ходе работы по созданию программного продукта «Облачный защищённый информационно-образовательный файл-сервер «DMN Cloud» выполнены следующие задачи:

- изучены особенности защищённого хранения данных,
- выполнен сбор требований к файл-серверу,
- проанализированы собранные требования,
- описана спецификация разрабатываемого ПО,
- спроектированы DFD-диаграммы первого и второго уровней,
- спроектирована причинно-следственную диаграмму,
- спроектированы UML-диаграммы стереотипов и классов,
- выбран и обоснован выбор средств разработки ПО и БД,
- настроен удалённый FTP-сервер,
- спроектирована структура локального и облачного хранилища,
- разработано оконное приложение с пользовательским интерфейсом для доступа к локальному и облачному хранилищу,
- реализована возможность регистрации пользователей в системе,
- реализована возможность подтверждения адреса электронной почты с использованием почтового SMTP клиента,
- реализована возможность восстановления пароля пользователя с использованием почтового SMTP клиента,
- реализована возможность настройки расположения хранилища,
- реализована возможность просмотра списка хранимых файлов,

- реализована возможность загрузки, скачивания и удаления файлов,
- выполнено тестирование и отладка разработанного приложения,
- разработано руководство пользователя.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Введение в архитектуру клиент-серверных систем. – Текст : электронный // masters.donntu.org : [сайт]. 2021. – URL: <https://masters.donntu.org/2010/fknt/chernobay/library/translate.htm> (дата обращения 18.10.2023).
2. Разное сетевое оборудование. Что такое файл-сервер, сервер приложений, коммуникационный и принт-сервер. – Текст : электронный // ixbt.com : [сайт]. – 2021. – URL: https://www.ixbt.com/comm/ev/nq00_q.html (дата обращения 25.10.2023)
3. Дадян, Э. Г. Данные: хранение и обработка : учебник / Э. Г. Дадян. – Москва : ИНФРА-М, 2020. – 205 с. – URL: <https://znanium.com/catalog/document?id=346013>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.
4. Мартишин, С. А. Базы данных. Практическое применение СУБД SQL и NoSQL-типа для применения проектирования информационных систем : Учебное пособие / С. А. Мартишин, В. Л. Симонов, М. В. Храпченко. – Москва : ИД «ФОРУМ» : ИНФРА-М, 2018. – 368 с. – URL: <http://znanium.com/bookread2.php?book=926871>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.
5. Павловская, Т. А. С#. Программирование на языке высокого уровня: Учебник для вузов / Т. А. Павловская. – Санкт-Петербург : Питер, 2021. – 432 с. – URL: <https://ibooks.ru/bookshelf/377952/reading>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.
6. Что такое DFD (диаграммы потоков данных). – Текст : электронный // habr.com/ru/company/trinion/blog/340064/ (дата обращения 17.11.2023).