|  |
| --- |
| Казанский (Приволжский) федеральный университет Институт вычислительной математики и информационных технологий |
| Образовательные семинары: приложение для организации семинаров и мастер-классов по различным предметам |
| Техническое задание |
|  |

Работу выполнила:

Меркулова Софья Денисовна

|  |
| --- |
| 2025 |

ОГЛАВЛЕНИЕ

[1. Введение 3](#_Toc195818247)

[1.1. Цель 3](#_Toc195818248)

[1.2. Краткая сводка возможностей 3](#_Toc195818249)

[1.3. Определения, акронимы и сокращения 3](#_Toc195818250)

[1.4. Краткое содержание 4](#_Toc195818251)

[2. Обзор системы 4](#_Toc195818253)

[2.1. Определение типов пользователей системы 4](#_Toc195818254)

[2.2. Окружение системы 5](#_Toc195818258)

[2.3. Обзор вариантов использований 5](#_Toc195818259)

[2.4. Нефункциональные требования 7](#_Toc195818260)

[3. Детальные требования 7](#_Toc195818261)

[3.1. Функциональные требования 7](#_Toc195818262)

[3.2. Требования к пользовательским интерфейсам 11](#_Toc195818263)

[3.3. Требования к логической структуре БД 17](#_Toc195818264)

## 1. Введение

### 1.1. Цель

Цель документа - представить детальное описание создаваемого приложения. Он описывает цели и возможности системы, пользователей системы, сценарии использования (пользовательские истории) и строение системы.

# **1.2. Краткая сводка возможностей**

1)  Интерфейс приложения поддерживает следующие характеристики события: его название, дата, время, список участников и категория;

2) У пользователя есть возможность добавить новое событие, редактировать его или удалять, а также просматривать введенные события.

3) Приложение предусматривает фильтрацию событий по категориям и дате;

4) Приложение оповещает пользователя, если он вводит некорректные данные;

5) Пользователю предоставляется возможность генерации отчётов по выбранным категориям.

# **1.3.** **Определения, акронимы и сокращения**

Windows Forms — это интерфейс программирования приложений (API), отвечающий за графический интерфейс пользователя и являющийся частью Microsoft .NET Framework;

Microsoft SQL Server (MS SQL) — это система управления реляционными базами данных (СУБД), разработанная корпорацией Microsoft. Её используют для хранения и извлечения данных из других программных приложений.

Поток в программировании — это процесс вычислений, который выполняет различные шаги для выполнения задачи. Программные шаги изначально пишутся в виде исходного кода и обычно компилируются компилятором;

GitHub — это веб-сервис для хостинга IT-проектов и их совместной разработки. По сути, это визуальный интерфейс для работы с системой Git;

База данных (БД) — это структурированное хранилище данных, которое позволяет эффективно организовывать, хранить и извлекать информацию.

# **1.4. Краткое содержание**

### Приложение написано на языке C# и взаимодействует с с реляционной базой данных. Приложение доступно для администраторов (авторов события), которые наделены специальными возможностями и гостей. У администраторов есть доступ к странице входа, добавлению событий и их характеристик, сохранению, редактированию, удалению и фильтрации, гости же наделяются возможностью просмотра данных и осуществления их фильтрации. Добавление событий пользователем осуществляется следующим образом: пользователь нажимает на кнопку «Добавить событие», в готовую форму с колонками осуществляет ввод данных, нажимает на кнопку «сохранить», данные сохраняются и формируется отчёт. Просмотр событий гостем осуществляется иначе: система отображает раннее введенные события администратором и гость просматривает события.

### 2. Обзор системы

### 2.1. Определение типов пользователей системы

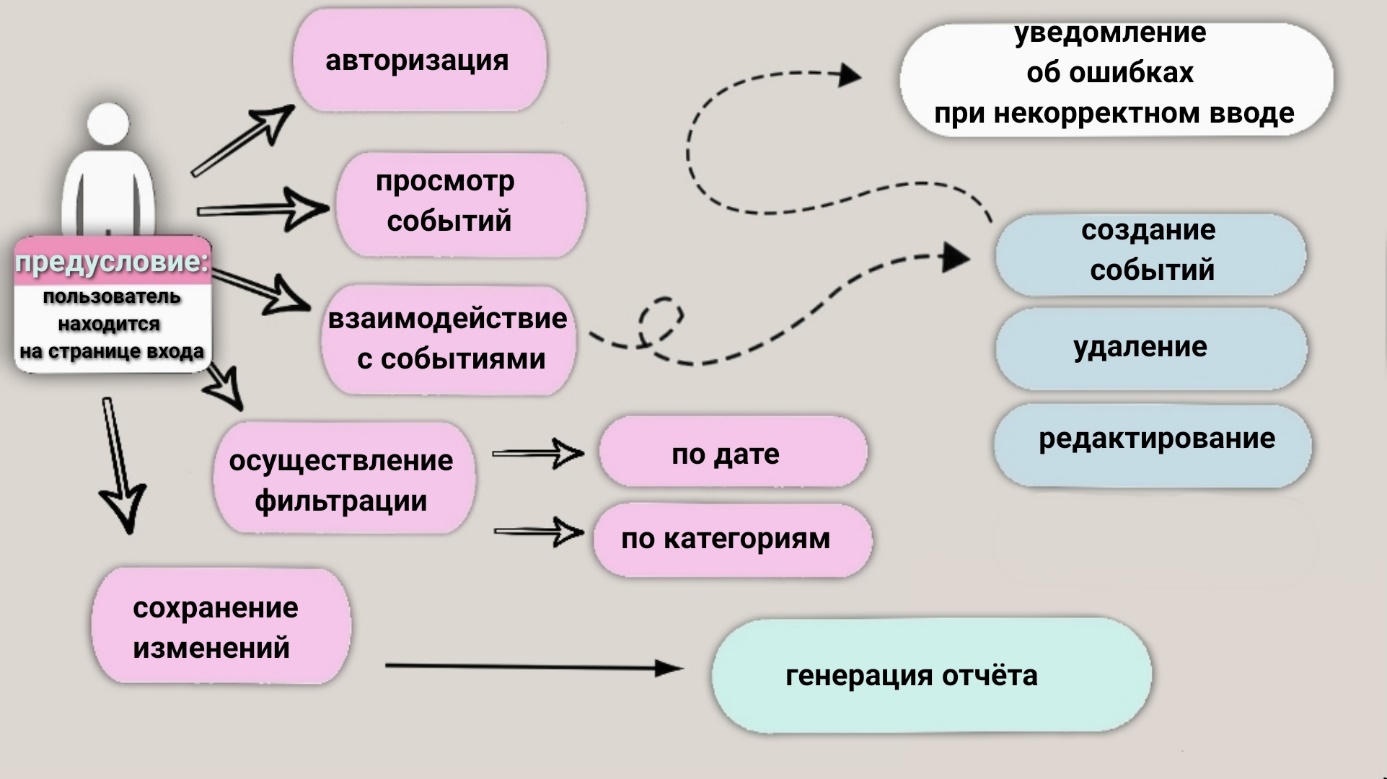
### Приложение обеспечивает доступ и предоставляет специальные возможности следующим категориям пользователей:

### Администратор (автор события) – предоставление доступа к просмотру события, его редактированию и удалению при необходимости.

### Гость (пользователь, использующий приложения для ознакомления с событиями) – предоставление доступа к просмотру события.

### 2.2. Окружение системы

Варианты использования приложений для пользователей, осуществивших различный вход в приложение.





## 2.3. Обзор вариантов использований

***Для администраторов (создатели событий):***

Как администратор я хочу осуществлять вход в приложение с помощью авторизации, чтобы иметь особые возможности.

*Для осуществления критерия необходимо:*

*Добавить «страницу входа», где осуществляется ввод логина и пароля, после чего пользователь переходит на страницу с событиями.*

Как администратор я хочу вводить характеристики к своим событиям: название, дату, время, список участников и категорию, чтобы увеличить понимание образовательного семинара;

*Для осуществления критерия необходимо:*

*Добавление полей для указанных характеристик.*

Как администратор я хочу управлять созданными мною событиями, редактировать их, фильтровать по категориям и дате, удалять их, чтобы улучшить читаемость вводимых данных и их поиск.

*Для осуществления данного критерия необходимо:*

*Добавление кнопок к событию: удаление и редактирование, а также добавление сортировки по дате и по категориям.*

Как администратор я хочу видеть указание, если вводимые данные не соответствуют требуемому формату для того, чтобы избежать ошибок;

*Для осуществления данного критерия необходимо:*

*Вывод на экран диалогового окна с уведомлением о неверном формате.*

Как администратор я хочу генерировать отчет по введенной информации в формате отдельного файла.

*Для осуществления данного критерия необходимо:*

*Кнопка «Сохранить отчет» на форме с таблицей, которая загружает файл в формате таблицы Excel.*

***Для гостей:***

Как гость я хочу входить в приложение без дополнительной регистрации, чтобы просмотреть список события.

*Для осуществления критерия необходимо:*

*Добавить «страницу входа», где так же присутствует кнопка «войти как гость», после которой следует страница с событиями.*

Как гость я хочу иметь доступ к фильтрации событий для облегчения поиска.

*Для осуществления критерия необходимо:*

*Добавление фильтрации по дате и категориям, доступное для всех пользователей приложения.*

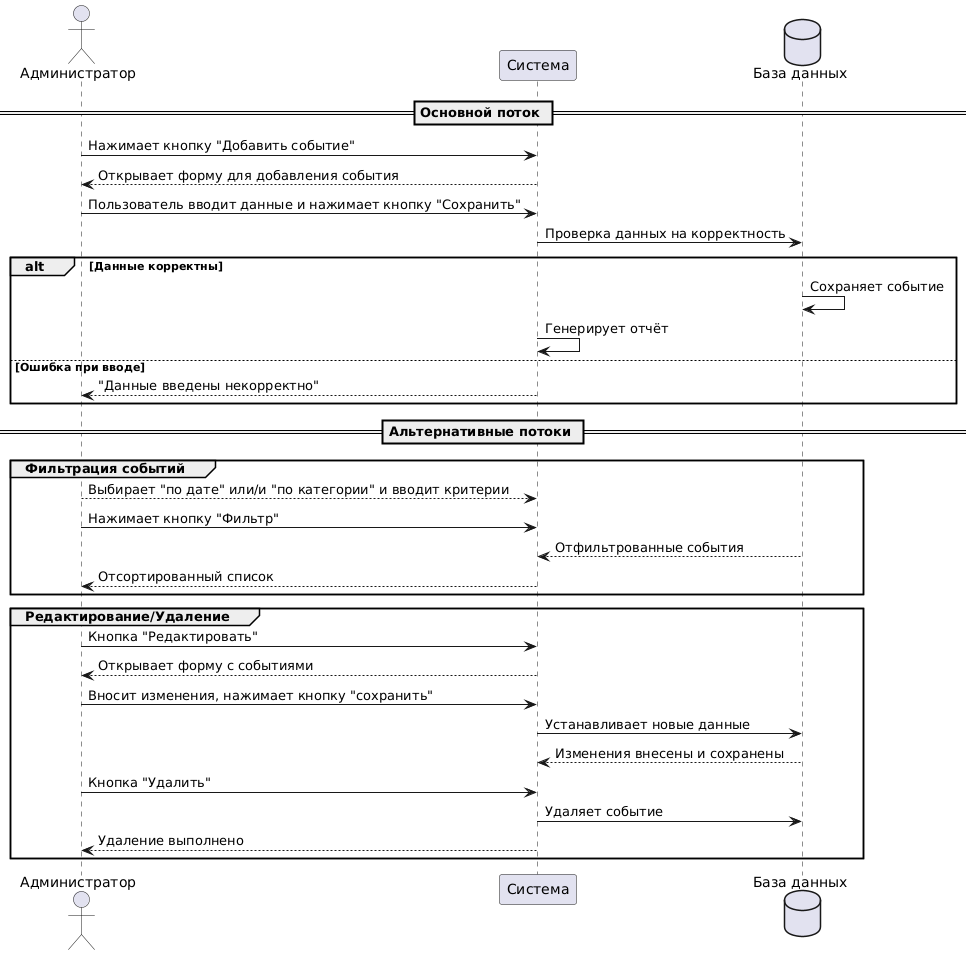
## 2.4. Нефункциональные требования

* Приложение написано на языке C#;
* Используемый интерфейс: Windows Forms;
* Приложение взаимодействует с реляционной базой данных MS SQL;
* Используется система контроля версий: GitHub;

## 3. Детальные требования

## 3.1. Функциональные требования

Диаграмма последовательности №1 (для администратора):



*Название:* Добавление события пользователем;

*Акторы:* Администратор;

*Предусловия:* Пользователь выполнил вход в приложение;

*Основной поток:*

1. Пользователь нажимает на кнопку «Добавить событие»;
2. В готовую форму с колонками (составленным по раннее указанным характеристикам) пользователь осуществляет ввод данных;
3. Нажимает на кнопку «сохранить событие»;
4. Данные сохраняются в таблице главной формы;
5. После пользователь нажимает кнопку «сохранить отчет» для генерации отчета;
6. Осуществляется загрузка файла в формате таблицы Excel;

*Альтернативные потоки:*

* Если события нужно отсортировать, пользователь нажимает на «фильтрацию» и выбирает: «по дате»/ «по категории», приложение осуществляет фильтрацию.
* Если событие необходимо исправить, пользователь нажимает кнопку «редактировать»;
* Если событие необходимо убрать, пользователь нажимает кнопку «удалить»;

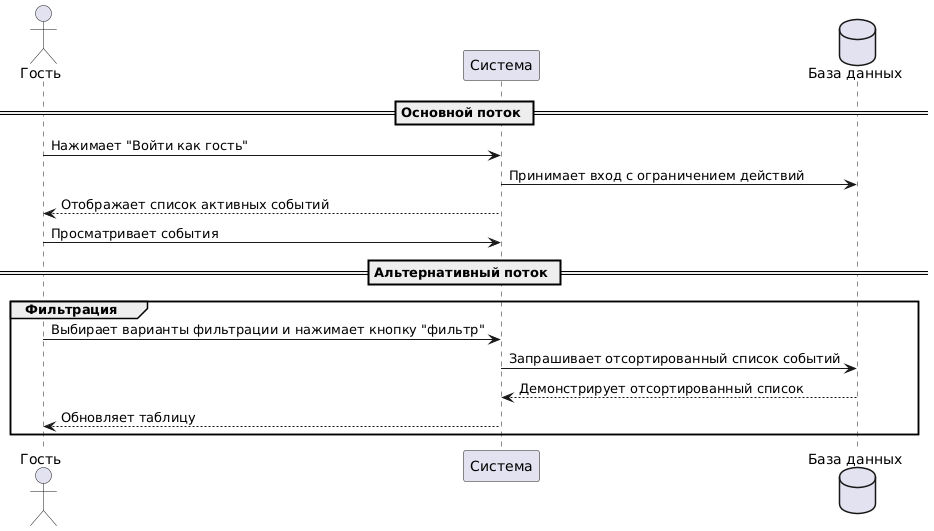
*Постусловия*: Событие создано и сохранено.

*Исключения:* Если ввод данных был некорректным, то уведомить об этом пользователя для исправления ошибки.

*Название:* Просмотр событий пользователем;

*Акторы:* Гость;

Диаграмма последовательности №2 (для гостя):



*Предусловия:* Пользователь выбрал «войти как гость» раннее;

*Основной поток:*

1. Система отображает раннее введенные события администратором;

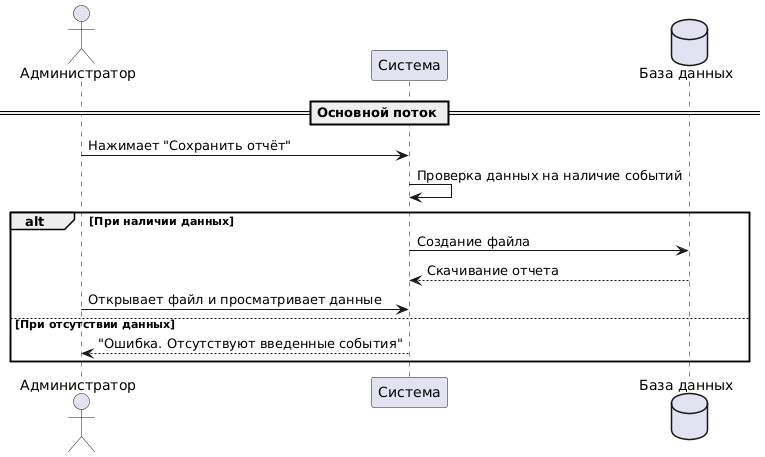
2. Пользователь просматривает события и их характеристики;

*Альтернативные потоки:*

* Если события нужно отсортировать, пользователь нажимает на «фильтрацию» и выбирает: «по дате»/ «по категории», приложение осуществляет фильтрацию.

*Постусловия*: События открыты и просмотрены.

Диаграмма последовательности №3 (для отчетов):



*Название:* Сохранение отчета;

*Акторы:* Администратор;

*Предусловия*: Пользователь ввел событие и сохранил его;

*Основной поток:*

1. Пользователь нажимает на кнопку «Сохранить отчет»;

2. Происходит проверка введенных данных на их наличие;

3. В файл переносятся данные;

4. Осуществляется его загрузка;

5. После пользователь открывает файл и просматривает данные отчета;

*Альтернативные потоки:*

• Если событие не было введено, то пользователь будет уведомлен об ошибке.

### 3.2. Требования к пользовательским интерфейсам

При запуске приложения пользователь находится на странице входа, где ему предоставляется доступ для использования приложения в зависимости от его категории: гость/администратор.

Форма №1 позволяет осуществить вход в приложение с помощью любого логина, ограниченного лишь англоязычным вводом, а также пароля, который зафиксирован изначально: administrator555.

Для такого интерфейса используются нижеперечисленные элементы:

* Кнопки (button): «Войти как Гость» и «Вход».

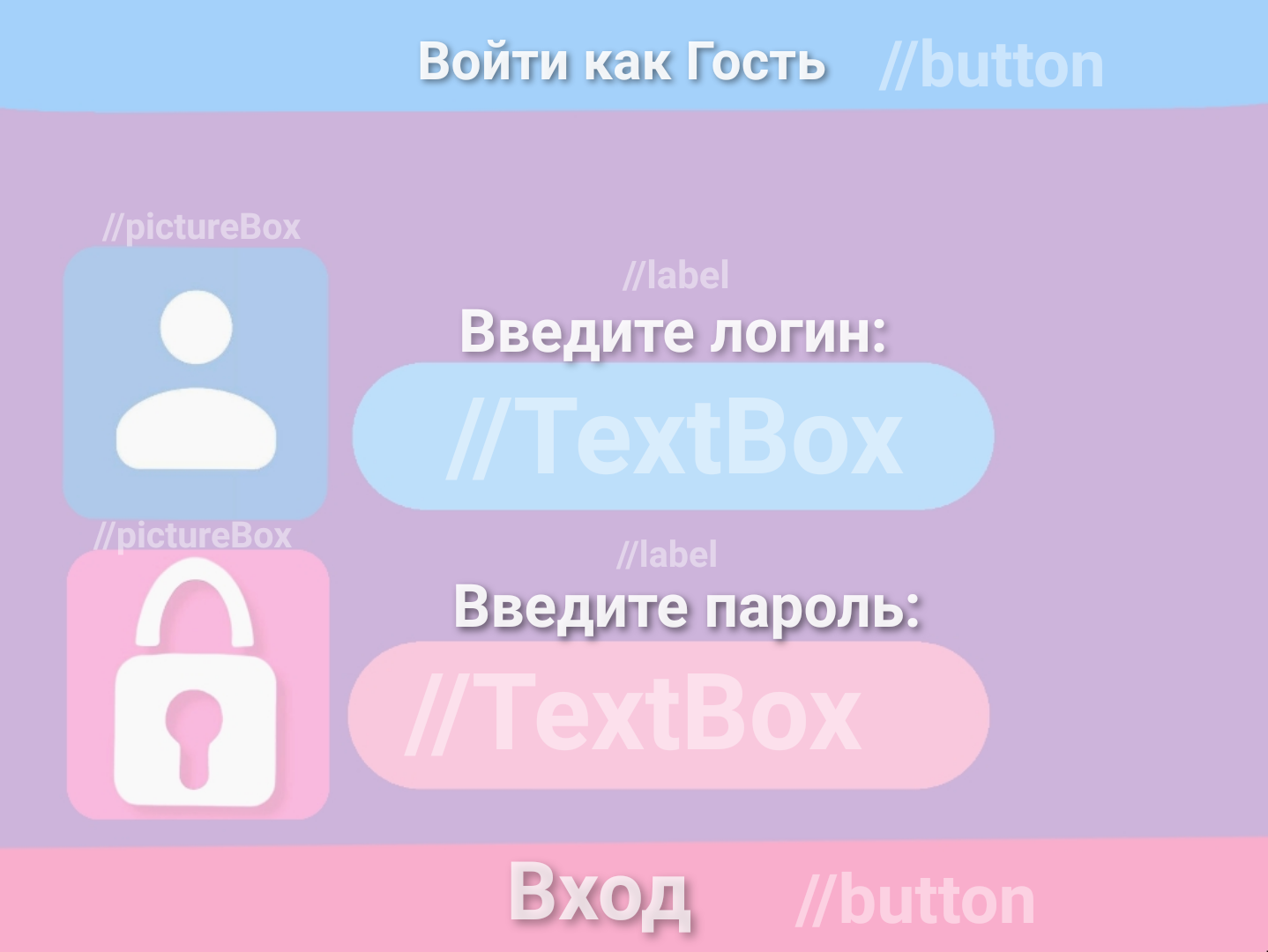
- «Войти как Гость» предоставляет доступ к следующей форме без ввода логина и пароля, но при этом возможность изменения событий будет ограничена.

- Кнопка «Вход», в свою очередь, переводит пользователя к главной странице при условии ввода логина английскими буквами и верным вводом установленного пароля. При таком входе в приложение пользователю открыты все возможности программы.

* Поля для ввода данных (TextBox), в которые осуществляется ввод логина и пароля для входа в качестве администратора.

- На ввод данных наложены ограничения: для первого TextBox – это англоязычный формат, а для второго – определенный пароль, если одно из условий не будет выполнено, пользователь не сможет осуществить переход к следующей странице.

* Элементы для улучшения интерфейса: буквенные указатели (label) и картинки (pictureBox), которые помогают легко сориентироваться в необходимых для выполнения действиях, в данном случае — это вход в приложение.



Форма №1-Вход в приложение.

После осуществления входа в приложение пользователю предоставляется просмотр следующей формы, в которую он может вносить изменения (для администратора) или исключительно осуществлять фильтрацию (для гостя).

Форма №2 осуществляет разграничение пользователей, поэтому, если пользователь входит как гость, то функции добавления событий, их удаление, редактирование и сохранение не будут доступны. В ином случае, пользователю доступны все функции главной страницы.

Для создания формы применимы такие элементы как:

* Комбинированный список (ComboBox), который представляет собою выпадающую строку с списком категорий для события.

- Работает для любого типа пользователей, необходим для осуществления сортировки по категориям.

* Виджет для выбора даты и времени (DateTimePicker).

-Как и категории он работает для любого типа пользователей и необходим для осуществления сортировки по дате и времени.

* Кнопки (button) «фильтр», «добавить событие», «удалить», «редактировать», «сохранить», которые позволяют взаимодействовать со списком событий.

- «Фильтр» принимает условия для сортировки по дате и/или по категории. С его помощью можно ограничить отображение событий, которые не совпадают с определенной датой/категорией. Кнопка доступна и для гостей, и для администраторов.

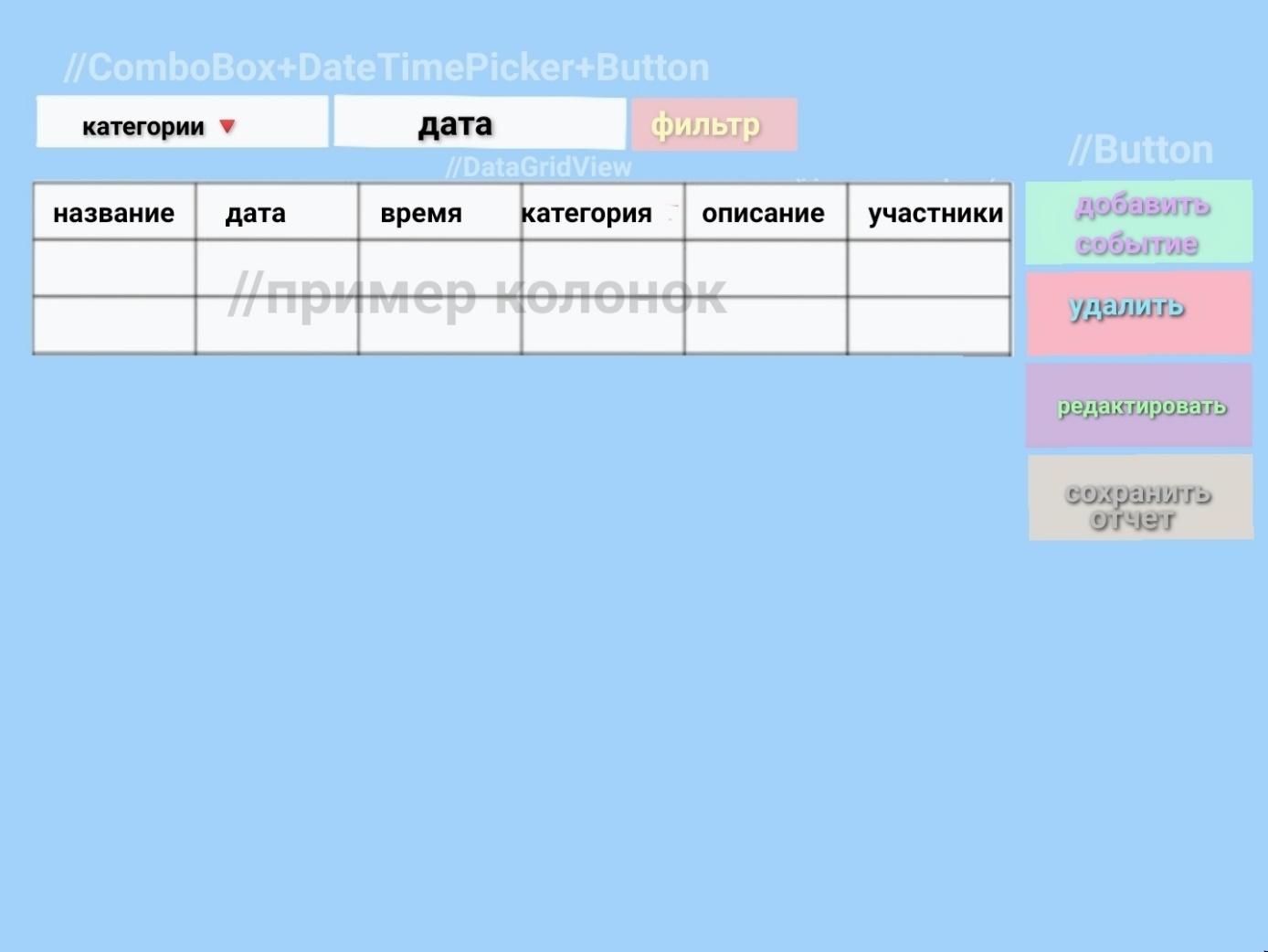
-«Добавить событие» и «редактировать» переводят пользователя в следующую форму, где он вносит изменения. Кнопки доступны только администраторам.

-«Удалить» устраняет определенное событие из списка. Доступ к этой кнопке предоставлен только для администраторов.

- Кнопка «Сохранить отчет » сохраняет и автоматически загружает файл с отчетом . Доступна для администраторов.

* Таблица со всеми введенными событиями (DataGridView).

-В таблице должны быть колонки с названиями: «название», «дата», «время», «категория», «описание», «участники». После сохранения отдельного события в Форме №3 событие высвечивается в таблице все с теми же критериями. Для просмотра такая таблица доступна и гостям, но переходить к 3-ей форме пользователи такого типа не смогут.



Форма №2 - Главная страница.

При добавлении и редактировании события пользователь переходит на следующую форму – Форма №3. Она позволяет осуществить ввод характеристик идентичных тем, что указаны в Форме №2.

Доступ к форме осуществляется с помощью кнопок «добавить событие» и «редактировать», соответственно, он не должен быть доступен для гостей, поскольку эти кнопки у них не осуществляют никаких действий.

Для этой формы характерны следующие элементы:

* Комбинированный список (ComboBox), который представляет собою выпадающую строку с списком категорий для события.
* Текстовые указатели (label) для ясности интерфейса.

-Их текст совпадает с названием характеристик события, находится напротив поля для ввода и выпадающей строки.

* Поля для ввода (TextBox) для записи конкретных качеств определенного события.
* Виджет для выбора даты (DateTimePicker).
* Кнопка (button) «Сохранить событие», которая возвращает пользователя на 2-ую форму, где событие добавляется автоматически.



Форма №3 - Для работы с событиями.

Для просмотра отдельного события (функция доступна для всех пользователей) введен двойной клик, который открывает Форму №4 с все теми же характеристиками события формата DataGridView, но в единичном формате. Выглядит это следующим образом:



Форма №4 – Функция двойного клика.

### 3.3. Требования к логической структуре БД

Таблица для “Events”:

|  |  |  |  |
| --- | --- | --- | --- |
| Название поля | Тип данных | Описание | Пример использования |
| EventId | Int, Primary Key | Id для события | 1 |
| Title | String | Название события | «Урок рисования» |
| Description | String | Описание события | «На семинаре Вас научат основным техникам работы с акварелью» |
| Date | DateTime | Дата проведения события | 2025-04-20 |
| Time | TimeSpan | Время проведения события | 17:30:00 |
| CategoryId | Int, Primary Key | Id категории (здесь нужен для связи с “Category») | 1 |
| Participants | String | Список участников события | «Алла Николаевна, Дмитрий Викторович, Елена Ильинична» |

Таблица для “Category”:

|  |  |  |  |
| --- | --- | --- | --- |
| Название поля | Тип данных | Описание | Пример использования |
| CategoryId | Int, Primary Key | Id категории | 1 |
| NameOfCategory | String | Название категории | «Творчество» |

Схема для БД:

