|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **РОСЖЕЛДОР**  **ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**  **ВЫСШЕГО ОБРАЗОВАНИЯ**  **СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**  **ПУТЕЙ СООБЩЕНИЯ (СГУПС)**   |  |  |  |  | | --- | --- | --- | --- | | **К защите:** |  |  | | | **Заведующий кафедрой** | **Информационные** | | | **технологии транспорта** | | | |  | д-р техн. наук, профессор | | |  |  | В.И. Хабаров | | | *подпись* |  | *инициалы, фамилия* | | |  |  |  | | | *дата* |  |  | |   **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**  **(БАКАЛАВРСКАЯ РАБОТА)**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Тема:** | Создание цифрового помощника для ведущего документоведа кафедры | | | | |  | | | | | |  |  | БР.БПИ.01.2022 |  |  | |  |  | *шифр документа* |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  |  | **Выполнил** |  |  |  | **Руководитель** | |  |  | Д. В. Аксенова |  |  |  | канд. техн. наук, доцент  А. А. Уланов | | *подпись* |  | *инициалы, фамилия* |  | *подпись* |  | *инициалы, фамилия* | |  |  |  |  |  |  |  | | *дата* |  |  |  | *дата* |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | | Нормоконтролер работы |  | *дата* |  | ст. преп.  Т. А. Распопина | |  |  | *подпись* |  | *инициалы, фамилия* | |  |  | *дата* |  |   **2022 г.** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**  **ПУТЕЙ СООБЩЕНИЯ (СГУПС)**  Факультет: Бизнес-информатика  Кафедра: Информационные технологии транспорта  Направление: 09.03.03 «Прикладная информатика»  Профиль: «Корпоративные информационные системы на транспорте»   |  |  |  | | --- | --- | --- | |  | ***УТВЕРЖДАЮ****:* | *зав. кафедрой «Информационные технологии транспорта»* | |  |  | д-р техн. наук, профессор  В. И. Хабаров | |  |  | *«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_ 2022 г.* |   **З А Д А Н И Е**  **на выполнение выпускной квалификационной работы**  студенту Аксеновой Дарье Васильевне   1. Тема «Создание цифрового помощника для ведущего документоведа кафедры» утверждена приказом № 203/с от «30» мая 2022 г. 2. Задание выдано «12» мая 2022 г. 3. Срок сдачи законченной работы на кафедру «20» июня 2022 г. 4. Исходные данные: положение о выпускающей кафедре, структура организации, бизнес-процессы работы ведущего документоведа кафедры   5.Содержание расчетно-пояснительной записки   |  |  |  | | --- | --- | --- | | Наименования разделов и вопросов | Примерное  количество  страниц | График  (сроки)  выполнения | | Введение | 1 | 14.05.2022 | | Изучение предметной области и ключевых процессов работы ведущего документоведа | 15 | 14.05.2022 | | Моделирование бизнес-процессов на языке UML | 6 | 15.05-18.05.22 | | Реализация программного продукта | 10 | 18.05-27.05.22 | | Руководство пользователя | 10 | 27.05-28.05.22 | | Заключение | 1 | 28.05.2022 |   6.Содержание и объемы графической части   |  |  |  | | --- | --- | --- | | Наименование графического документа  (чертежа, схемы, графика) | Количество  листов  формата А1 | График  (сроки)  выполнения | | Презентация в формате PowerPoint |  |  |      |  |  |  | | --- | --- | --- | | Руководитель |  | А. А. Уланов | |  | *(подпись руководителя)* |  | | Задание к исполнению принял |  | Д. В. Аксенова | |  | *(подпись студента)* | |

УДК 004.94

**АННОТАЦИЯ**

В работе 49 страниц, 25 рисунков, 5 таблиц, 15 источников.

Ключевые слова: *кафедра, приложение, программный продукт, событие, база данных, система, проектирование.*

Объектом исследования являются бизнес-процессы выпускающей кафедры «Информационные технологии транспорта» Сибирского государственного университета путей сообщения (СГУПС).

Предметом исследования – приложение для планирования событий в соответствии с установленным календарным планом. Хранение базы данных осуществляется на жестком диске ПК, что позволяет избежать использование резервных серверов. Программный продукт позволяет редактировать базу данных, производить настройку работы уведомлений и отслеживать статус выполнения события с помощью сохранения ответов пользователя на каждое уведомление в журнале событий.

**ABSTRACT**

The work contains 49 pages, 25 figures, 5 tables, 15 sources.

Keywords: *department, application, software product, event, database, system, design.*

The object of the research is the business processes of the graduating Department of «Information Technologies of Transport» of the Siberian State University of Railways (STU).

The subject area is an application for planning events in accordance with the established calendar plan. The database is stored on the PC's hard disk, which avoids the use of backup servers. The software product allows you to edit the database, configure notifications and monitor the status of the event by saving the user's responses to each notification in the event log.

СОДЕРЖАНИЕ

[Введение 5](#_Toc105455175)

[1 Аналитическое исследование 6](#_Toc105455176)

[1.1 Общая характеристика предметной области 6](#_Toc105455177)

[1.2 Положение о кафедре Сибирского государственного университета путей сообщения (СГУПС) 6](#_Toc105455178)

[1.3 Основные функции кафедры 7](#_Toc105455179)

[1.4 Дополнительные функции выпускающей кафедры 9](#_Toc105455180)

[1.5 Взаимодействия кафедры с учебным отделом 10](#_Toc105455181)

[2 Выделение основных событий на выпускающей кафедре 12](#_Toc105455182)

[3 Аналоги программного продукта 14](#_Toc105455183)

[4 Постановка задачи и формирование технического задания 17](#_Toc105455184)

[5 Моделирование бизнес-процессов на языке UML 21](#_Toc105455185)

[5.1 Описание языка UML 21](#_Toc105455186)

[5.2 Проектирование бизнес-процессов программного продукта 23](#_Toc105455187)

[6 Реализация программного продукта 27](#_Toc105455188)

[6.1 Обоснование выбора СУБД 27](#_Toc105455189)

[6.2 Обоснование выбора среды разработки 29](#_Toc105455190)

[6.3 Описание языка программирования 30](#_Toc105455191)

[7 Разработка базы данных приложения 32](#_Toc105455192)

[8 Описание алгоритма работы программы 35](#_Toc105455193)

[9 Руководство пользователя 38](#_Toc105455194)

[9.1 Системные требования 38](#_Toc105455195)

[9.2 Интерфейс программного продукта 38](#_Toc105455196)

[Заключение 47](#_Toc105455197)

[Список использованных источников 48](#_Toc105455198)

4

# ВВЕДЕНИЕ

На сегодняшний день в организационной структуре вуза наблюдается значительное увеличение количества управлений, отделов и служб, то есть увеличение числа сотрудников управленческого аппарата. Однако это имеет не только положительные, но и отрицательные моменты, такие как - снижение качества управления высшим учебным учреждением вследствие несогласованности работы данных подразделений с деканатами и кафедрами по вопросам планирования, учета, контроля и отчетности по результатам деятельности. Вместе с внедряемыми обновлениями вводятся и новые формы документов, невостребованные при организации учебно-воспитательного процесса на кафедрах, но необходимы управлениям, отделам и службам для отчётов по своей работе.

Анализ и решение проблемы регулирования огромного массива данных, циркулирующих в образовательном процессе (документы, отчеты, новости, поручения), легли в основу выпускной квалификационной работы.

Целью дипломного проекта стало создание программного продукта, позволяющего упорядочить список функций ведущего документоведа, обеспечивающие отлаженную работу процессов кафедры.

Использование разработки в рабочем процессе должно позволить:

* осуществлять своевременную подготовку всей нужной документации;
* сократить время, затрачиваемое на поиск событий и сопутствующих документов;
* производить оперативное оповещение новой информацией;
* осуществлять полный контроль над планированием времени.

Разработка персонального цифрового помощника в сфере образования позволит с помощью корпоративного банка данных упростить работу структурным подразделениям организации.

# Аналитическое исследование

## 1.1 Общая характеристика предметной области

Предметной областью выпускной квалификационной работы является проектирование и разработка планировщика событий с расширенными функциональными возможностями для выпускающей кафедры.

Разрабатываемое приложение должно включать следующую информацию:

* справочную базу, хранящую информации о предстоящих событиях (наименование события, дата, сопутствующий перечень документов);
* журнал событий, отображающий историю ответов пользователя на оповещения.

К основным задачам программного продукта следует отнести:

* хранение информации и обмен информацией о важных событиях и задачах, которые необходимо выполнить в строго определенное время;
* уведомление о важных событиях или запланированных задачах за несколько дней до дедлайна;
* обеспечение возможности редактирования, добавления и удаления событий из базы данных.

Данная разработка в своем роде уникальна и имеет практическое значение для работы выпускающей кафедры.

## 1.2 Положение о кафедре Сибирского государственного университета путей сообщения (СГУПС)

Согласно положению о кафедре Сибирского государственного университета путей сообщения, утвержденным приказом СГУПС от 04.06.2008 года к общим положениям относится:

* 1. «Кафедра является основным учебно-научным структурным подразделением университета, осуществляющим учебную, учебно-методическую, научно-методическую работу по одной или нескольким родственным дисциплинам, научные исследования, а также подготовку научно-педагогических кадров и повышение их квалификации.
  2. Официальное наименование кафедры устанавливается при ее создании или реорганизации Ученым советом университета и должно соответствовать для выпускающих кафедр наименованию направления (специальности) подготовки специалистов, для не выпускающих кафедр отражать наименование области знаний, по которой эта кафедра ведет подготовку студентов.
  3. Кафедра юридическим лицом не является.
  4. Кафедра в своей деятельности руководствуется решениями Ученого совета факультета, университета, организационно-распорядительными документами администрации университета, Уставом университета, Федеральным законом "О высшем и послевузовском профессиональном образовании" и Типовым положением об образовательном учреждении высшего профессионального образования (высшем учебном заведении) Российской Федерации.
  5. Содержание и регламентацию работы кафедры определяют годовые и перспективные планы работы кафедры»[1].

## 1.3 Основные функции кафедры

На сегодняшний момент в подавляющем большинстве высших учебных заведений кафедра является основным первичным подразделением, основными функциями которой являются:

* проведение по всем формам получения образования: очной, очно-заочной (вечерней), заочной - все виды учебных занятий и учебной работы;
* разработка рабочих программ дисциплин на основе Государственных образовательных стандартов; утвержденных квалификационных характеристик специалистов; учебных планов специальностей и типовых программ дисциплин с отражением новейших достижений науки, техники и культуры, перспектив их развития, отраслевых и региональных условий и особенностей подготовки специалистов; внутри междисциплинарных логических связей, роли фундаментальных наук в теоретической и практической подготовке специалистов;
* осуществление комплексного методического обеспечения всех учебных дисциплин, закрепленных за кафедрой;
* участие в научной деятельности университета по теоретическим, научно-техническим, социально-экономическим, научно-методическим направлениям и педагогическим проблемам высшего профессионального образования;
* обсуждение, дача заключений и рекомендаций по законченным научным разработкам об опубликовании их результатов, внедрении и использовании на практике и в учебном процессе;
* изучение, обобщение и распространение опыта лучших преподавателей; оказание помощи начинающим преподавателям в овладении педагогическим мастерством; разработка и осуществление мероприятий по использованию современных технических средств и новых технологий обучения;
* обеспечение делопроизводства и документоведения по функционированию кафедры.

В функционировании кафедры нет ни одного вопроса, ни одной проблемы, не связанной с учебной и методической работой, с проведением и внедрением научных разработок, участием сотрудников в научных совещаниях, симпозиумах или конференциях. Именно поэтому, триединство, созданное образовательной деятельностью, методической работой, научно-исследовательской деятельностью, следует рассматривать как атрибутное свойство кафедры.

## 1.4 Дополнительные функции выпускающей кафедры

Ведущую роль в организационной структуре вуза играют выпускающие (профилирующие) кафедры, так как они несут прямую ответственность за выпуск специалистов (бакалавров, магистров) высшей квалификации по той или иной специализации или, в некоторых случаях, по направлению.

Выпускающая кафедра в процессе осуществления всех видов обучения по очной, очно-заочной, заочной формам, в соответствии со своей спецификой выполняет дополнительные функции:

1. Совместно с деканом факультета разработка учебного плана по направлению (специальности), согласование, при необходимости, с УМО и утверждение на Ученом совете СГУПС.
2. Введение изменений и корректировок в учебный план специальности с последующим обсуждением их на Совете факультета.
3. Контроль разработки и согласование УМКД по направлению (специальности) в соответствии с требованиями ГОС ВПО.
4. Разработка программ по всем видам практик, предусмотренных учебным планом специальности, и организация проведения этих практик. Содействие Отделу производственного обучения в разработке и согласовании договоров с предприятиями и организациями на проведение всех видов практик [2]. Руководство практиками.
5. Организация и проведение итоговой государственной аттестации по специальности.
6. Формирование у обучающихся профессиональных навыков по избранному направлению подготовки и гражданской позиции.
7. Подготовка необходимых отчетных документов по организации и ведению учебного процесса по направлению (специальности).
8. Подготовка документов к лицензированию и аккредитации для действующих и вновь вводимых образовательных программ.
9. Обеспечение возможности получения лицензионных прав СГУПСу вести образовательную деятельность и аккредитации по соответствующему направлению (специальности).
10. Организация и руководство: практиками, курсовым и дипломным проектированием студентов, способствуя приближению условий их проведения к реальным условиям производственной и исследовательской деятельности.
11. Содействие в трудоустройстве выпускников кафедры, анализ данных об их трудоустройстве и их информационная поддержка, а так же поддержание контакта с выпускниками с целью осуществления их обратной связи с СГУПС.
12. Изучение рынка трудовых ресурсов по направлению (специальности), по которым осуществляет подготовку специалистов, оказание содействия заключению индивидуальных трехсторонних договоров студентов с Университетом и предприятиями, учреждениями и организациями.

## 1.5 Взаимодействия кафедры с учебным отделом

Являясь основной структурной единицей  [факультета](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%BA%D1%83%D0%BB%D1%8C%D1%82%D0%B5%D1%82) в [высшем учебном](https://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%81%D1%88%D0%B5%D0%B5_%D1%83%D1%87%D0%B5%D0%B1%D0%BD%D0%BE%D0%B5_%D0%B7%D0%B0%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5) заведении, кафедра не может существовать и функционировать отдельно от деятельности всего университета. Её деятельность основывается на постоянном взаимодействии с учебно-методическим и научным отделами.

Учебный отдел передает на кафедру следующие сведения: плановый контингент студентов, график учебного процесса, количество дипломников, график практик и другие данные. На основе этих данных в апреле составляется плановая учебная нагрузка (основная, по совместительству, почасовая, по заочному факультету), уточнения по которой производятся к началу учебного года. В конце учебного года кафедрой сдается отчет о выполненной нагрузке.

Также в учебный отдел сдаются такие сведения как:

* ежемесячные отчеты о выполнении почасовой нагрузке по каждому преподавателю и по кафедре в целом;
* индивидуальные планы работы каждого преподавателя и отчеты о второй половине дня.

Помимо перечисленных функций учебный отдел передает на кафедру и Заведующему лабораторным комплексом расписание (по кафедре, по факультету), который отвечает за составление расписания занятий в компьютерных классах. Кафедра, в свою очередь, передает в учебный отдел учебный план специальности, итоги самоаттестации (по кафедре, по специальности), а также рейтинг специальности. При взаимодействии с учебным отделом на кафедре составляется план-отчет работы кафедры.

План работы кафедры на следующий учебный год составляется заведующим кафедрой и утверждается на заседании кафедры перед началом учебного года, подписывается заведующим кафедрой, согласовывается с деканом факультета и передается в учебный отдел вместе с плановым расчетом учебной нагрузки [3]. Далее представляется на утверждение проректорами по учебной и научной работе.

Взаимодействие кафедры с учебным отделом позволяет координировать все виды методической работы профессорско-преподавательского состава университета, а также повысить эффективности учебного процесса.

# Выделение основных событий на выпускающей кафедре

Контроль оформления документов и составление организационной, учебно-распорядительной документации по деятельности кафедры являются одними из основных функций специалиста в области [делопроизводства](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%BB%D0%BE%D0%BF%D1%80%D0%BE%D0%B8%D0%B7%D0%B2%D0%BE%D0%B4%D1%81%D1%82%D0%B2%D0%BE) - документоведа.

Документовед должен выполнять возложенные на него обязанности, в сроки, обеспечивающие возможность для эффективной организации работы кафедры.

Сроки выполнения основных видов работ представлены в таблице 2.1.

Таблица 2.1 – Основные события и их сроки выполнения

|  |  |  |  |
| --- | --- | --- | --- |
| Событие | Сроки сдачи документов | Период оповещений | Документы |
| |  | | --- | | Подать нагрузку на новый учебный год | | 1 апреля | с 1 марта по 1 апреля | Табель нагрузки на учебный год |
| |  | | --- | | Подать план повышения квалификации на новый учебный год | | 30 сентября | с 1 сентября по 30 сентября | План повышения квалификации на учебный год |
| |  | | --- | | Оформить ППС | | 1 сентября | с 15 августа по 1 сентября | Справка об отсутствии судимости  Справка с место основной работы  Заверенная копия трудовой книжки  Паспорт  СНИЛС  ИНН  Диплом об образовании  Диплом ученой степени /при наличии  Аттестат доцента (профессора) /при наличии  Флюорография  Согласие на обработку персональных данных /сайт СГУПС, раздел «Управление кадрами и делами»  Личный листок /отдел кадров / сайт СГУПС, раздел «Управление кадрами и делами» |

Продолжение таблицы 2.1

|  |  |  |  |
| --- | --- | --- | --- |
| Событие | Сроки сдачи документов | Период оповещений | Документы |
| |  | | --- | | Сдать отчёт о прохождении летней практики | | 1 октября | с 1 сентября по 1 октября | Отчёт о прохождении летней практики |
| Подать табель | не позднее 18-го числа каждого месяца | за 5 дней до окончания сроков | Табель |
| |  | | --- | | Подать почасовые | | не позднее 18-го числа каждого месяца | за 5 дней до окончания сроков | Почасовые |
| |  | | --- | | Заявка на базы практики | | 10 октября | с 10 сентября по 10 октября | Заявка на базу практики |
| |  | | --- | | Оформить приказ на дипломирование | |  | за 10 дней до начала практики | Приказ на дипломирование |
| Назначить председателей ГЭК | 30 сентября | за 5 дней до окончания сроков | Служебная записка по назначению председателей ГЭК |
| |  | | --- | | Оформить служебную записку на ГЭК (состав) | | 18 марта | за 5 дней до окончания сроков | Служебная записка на ГЭК |
| |  | | --- | | План издания методических указаний | | 25 марта | с 15 марта по 25 марта | План издания методических указаний |
| |  | | --- | | План закрепления руководителей практики | | 30 октября | с 30 сентября по 30 октября | План закрепления руководителей практики |
| |  | | --- | | Скорректировать систему рейтинга (дисциплины) | | 30 сентября,  30 января | за 10 дней до окончания сроков |  |
| |  | | --- | | Составить график взаимопосещений на семестр (ОКО) | | 30 сентября,  30 января | за 20 дней до окончания сроков | График взаимопосещений на семестр (ОКО) |
| Отчет по нагрузке | 30 июня | с 30 мая по 30 июня | Отчет по нагрузке |
| Отчет по НИР | 10 января | с 20 декабря по 10 января | Отчет по НИР |
| |  | | --- | | Оформить план-отчет | | 18 марта | за 5 дней до окончания сроков | Служебная записка на ГЭК |
| |  | | --- | | Заседание кафедры | | 20 декабря  20 июня | 20 декабря  20 июня | Протокол заседания кафедры |

Закрепленные за каждым событием определенные временные рамки позволяют не только контролировать исполнения документов, но и определять личную эффективность сотрудников.

1333

# Аналоги программного продукта

С целью разработки корректного и полезного программного продукта был произведен поиск аналогов приложения по типу «Планировщик событий», по итогу которого удалось выделить три наиболее популярных программы, а именно:

* LeaderTask – надежный планировщик личных задач и таск-менеджер для управления поручениями. Российская разработка, доступная на всех платформах (Windows, macOS, iOS, Android, Web), которые синхронизируются между собой [4, 5];
* TickTick – многофункциональный планировщик дел. Синхронизируется через iOS, Android, Mac, Windows и существует как расширение для браузеров Chrome и FireFox;
* Microsoft To Do – бесплатное приложение для планирования списка дел информация в котором синхронизируется на iPhone, устройствах с Android и Windows 10, а также с веб-приложением.

Интерфейсы приложений представлены на рисунках 3.1-3.3.

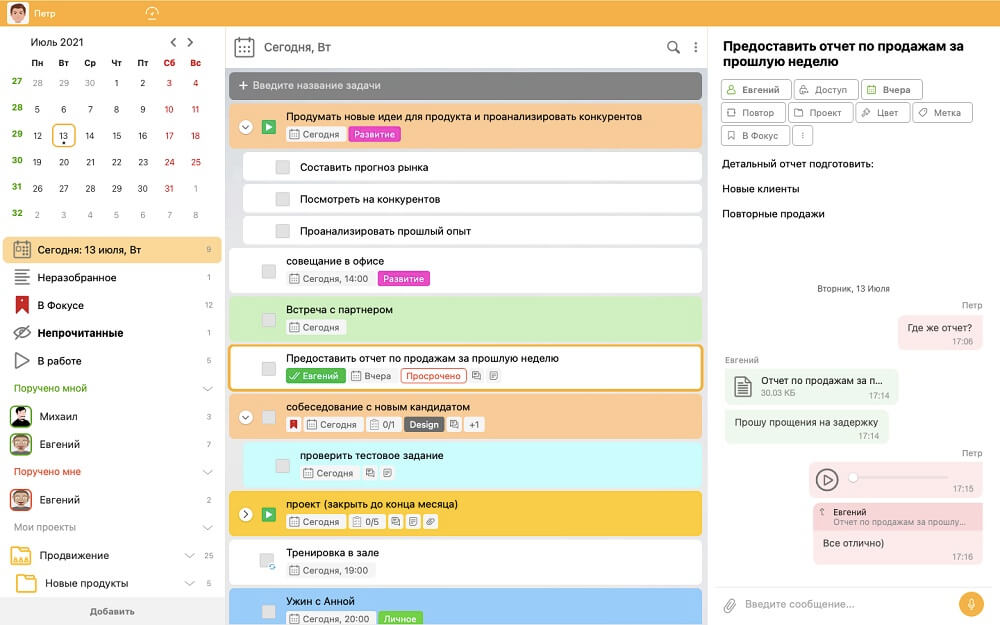


Рисунок 3.1 – Интерфейс приложения LeaderTask

1433

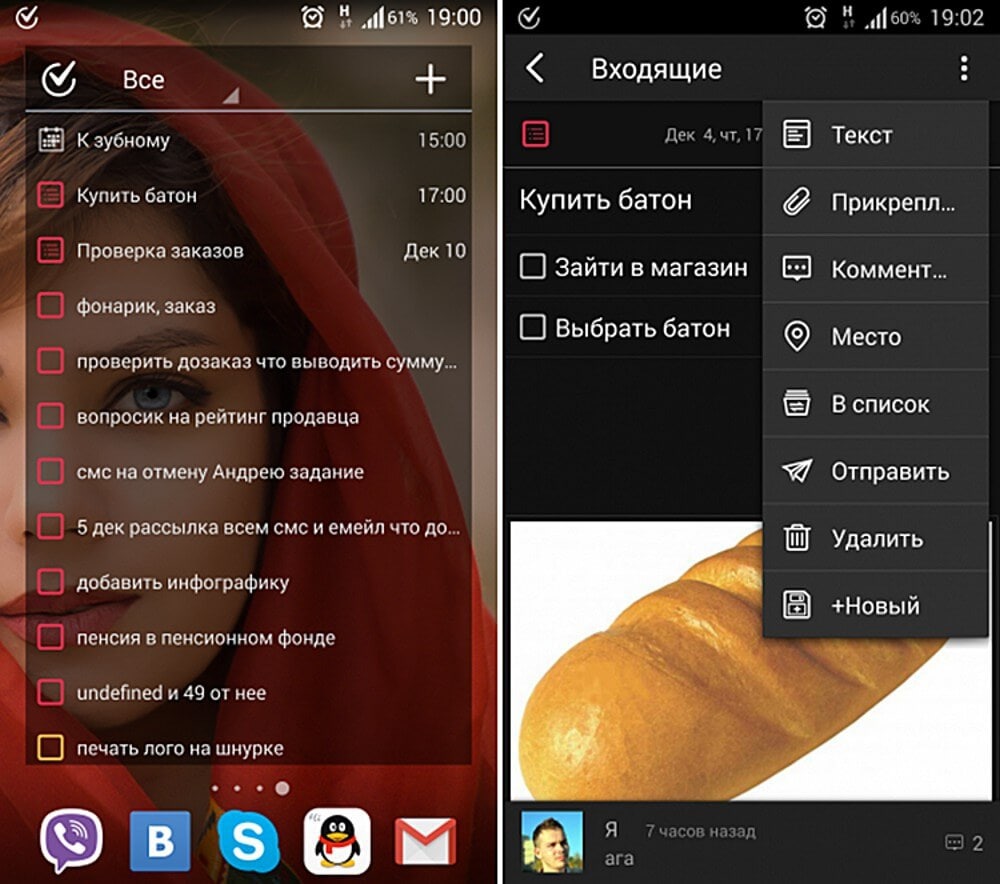


Рисунок 3.2 – Интерфейс приложения TickTick

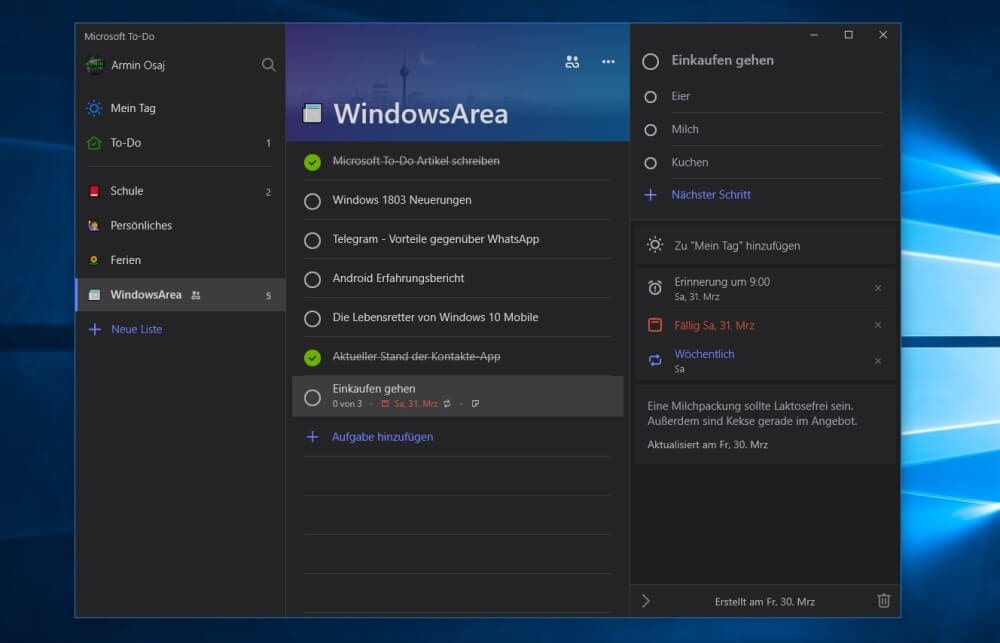


Рисунок 3.3 – Интерфейс приложения Microsoft To Do

По итогам анализа характеристик каждого продукта была составлена сравнительная таблица функциональности, таблица 3.1.

Таблица 3.1 – Сравнительные характеристики аналогов приложения

|  |  |  |  |
| --- | --- | --- | --- |
| Функциональность | LeaderTask | TickTick | Microsoft To Do |
| Удаление информации | + | + | + |
| Добавление новых событий | + | + | + |
| Изменение характеристик/вложений события | + | + | - |
| Возможность вложения файлов (документации) | + | +- | +- |
| Интеграция с календарными приложениями | + | + | + |
| Автоматическое выстраивание иерархии событий | + | + | - |
| Поиск по ключевым тегам | + | - | + |
| Автоматические оповещения | + | + | +- |
| Автоматическое обновление списка задач | + | + | + |
| Работа на ПК | + | + | + |
| Особенности: | Различия в функционале на разных платформах;  Многие функции доступны только в платной версии | Различия в функционале на разных платформах;  Ограничение на размерность прикрепляемых файлов | Нестабильная работа уведомлений;  Ограничение на размерность файлов – не более 25 МВ |

Несмотря на широкий функционал рассмотренных программных продуктов, каждый из них имеет некоторый перечень недочетов, которые могут привести к неустойчивой работе приложения и существенно сказаться на качестве хранимой информации.

# Постановка задачи и формирование технического задания

После проведения анализа возможностей и интерфейсов аналогов программного продукта было сформировано техническое задание на проектируемое ПО.

Название: «Цифровой помощник для ведущего документоведа кафедры».

Целью разработки и внедрения системы является оптимизация функций документоведа и повышение эффективности работы структурного подразделения кафедры информационных технологий транспорта за счет обеспечения контроля выполнения событий; уменьшения времени и трудозатрат на подготовку и поиск необходимых документов; повышения эффективности исполнения заданий и документов, улучшения контроля исполнительской дисциплины.

Система должна решать следующие задачи:

* обеспечить добавление и хранение описания запланированных событий с возможностью прикрепления электронных шаблонов оформления документации;
* обеспечить работу над документами – создание, редактирование, поддержка версионности, поиск документов;
* хранить добавленные и отредактированные документы в единой базе данных;
* предоставить механизм уведомлений, работающий заблаговременно, согласно установленной дате, с предоставлением полной информации о предстоящем событии;
* обеспечить безопасность и доступность сервиса.

Классы и характеристики пользователей. Пользователь (документовед): пользователь системы, имеющий доступ к базе данных с возможностью ее редактирования. В его перечень возможностей входят: добавление и редактирование событий, удаление событий, запись наименований прилагаемых документов к соответствующим событиям, задание сроков начала режима оповещения.

1733

На рисунке 4.1 представлена диаграмма потоков данных, описывающая внешние для системы потоки, источники и хранилища данных, к которым имеется доступ.

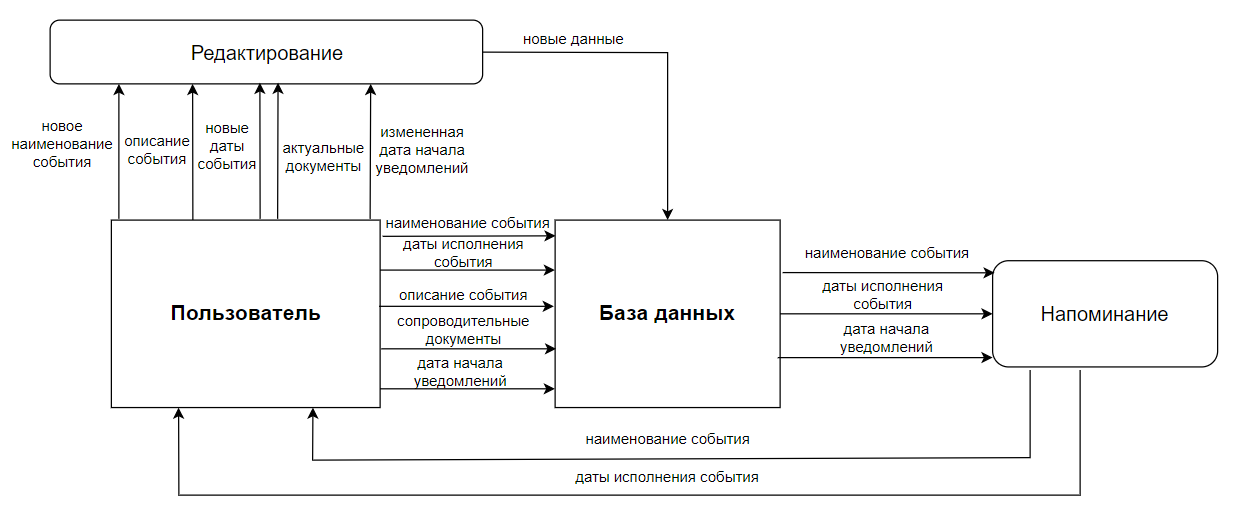


Рисунок 4.1 – DFD-модель движения данных в системе

Для удобной навигации пользователя в проектируемом приложении необходимо предусмотреть удобный интерфейс, включающий три основные вкладки.

Первая вкладка содержит полный перечень уже имеющихся в базе данных событий, отсортированных в порядке ближайших дат.

Вторая вкладка предназначено для добавления, редактирования или удаления события. Данное окно содержит следующее:

* поле для поиска необходимого события или записи нового события;
* поля для задания начала и окончания события;
* поле для прикрепления некоторого перечня документов для соответствующего события;
* поле для настройки работы режима оповещений;
* кнопки: «удалить», «сохранить».

Третья вкладка предназначена для сохранения историй оповещения и совершенными над ними действиями: «отложить» или «принять».

С целью оперативного оповещения пользователя о предстоящем событии система будет открываться автоматически совместно со всплывающими уведомлениями при запуске компьютера.

Рамки и ограничения проекта. Работающая программа будет ограничена данными выпускающей кафедрой «Информационные технологии транспорта» (ИТТ).

Ограничения по разработчикам. Разработка программного продукта осуществляется одним студентом факультета «Бизнес-информатика».

Ограничения по времени. Система должна быть реализована в течение шести месяцев: конечный срок – 30.05.2022.

Нефункциональные требования. Атрибуты качества: надежность, удобство сопровождения, эффективность использования, эксплуатационная пригодность.

Функциональные требования:

* учет данных о важных событиях и задачах;
* хранение данных о важных событиях и задачах;
* автоматическое выстраивание иерархии событий;
* обеспечение информирования о наступлении важного события или задачи;
* фиксирование ответов пользователя на уведомления.

Требования к надежности:

* предусмотреть контроль вводимой информации и блокировку некорректных действий пользователя при работе с системой;
* обеспечить корректное завершение вычислений с соответствующей диагностикой при превышении имеющихся вычислительных ресурсов;
* обеспечить целостность информации, хранящейся в базе данных.

Стадии и этапы разработки:

* сбор данных;
* анализ данных;
* описание бизнес-процессов;
* разработка интерфейса;
* разработка базы данных;
* выбор среды разработки;
* написание кода (проектирование);
* тестирование и отладка.

Порядок сдачи проекта:

* установить программный продукт на рабочем месте документоведа;
* обучить документоведа работе с программным продуктом;
* протестировать программный продукт на рабочем месте документоведа в течение месяца;
* выявить замечания/ошибки;
* внести изменения в код программы;
* составить руководство пользователя.

Внедрение разработанной программы должно упростить работу документоведа кафедры, поскольку в ее функционал будет входить: заблаговременное напоминание о приближающемся событии сотруднику кафедры, автоматическое обновление списка задач, а также возможность прикрепления списка документов, которые необходимо собрать и подписать к конкретно назначенному сроку. Благодаря синхронизации с календарными приложениями программа позволяет настраивать даты и предполагает указывание степени подготовки к событию (выполнено, ожидает выполнения).

# Моделирование бизнес-процессов на языке UML

## 5.1 Описание языка UML

Унифицированный язык моделирования (Unified Modeling Language – UML) – стандартный инструмент для разработки «чертежей программного обеспечения», предназначенный для определения, представления, проектирования и документирования программных систем, организационно-экономических систем, технических систем и других систем различной природы [6, 7].

Язык UML предоставляет стандартный способ написания проектной документации на системы, включая концептуальные аспекты, такие как бизнес-процессы и функции системы, а также конкретные аспекты, такие как выражения языков программирования, схемы баз данных и повторно используемые компоненты ПО.

Достоинствами UML можно считать:

* универсальность – технология, которую знают большинство программистов и аналитиков;
* широкое применение – не зависит от языка программирования проекта;
* большое множество типов диаграмм;
* удобные инструменты;
* UML объектно-ориентирован;
* диаграммы UML сравнительно просты для чтения после достаточно быстрого ознакомления с его синтаксисом;
* распространенность применения;
* в большинстве случаев возможен перенос диаграмм из одного инструмента в другой.

К недостаткам в свою очередь можно отнести:

* избыточность языка. Стандарт UML включает много избыточных и практически неиспользуемых диаграмм и конструкций;

2133

* неточная семантика. UML лишен определенных техник формального описания;
* необходимо изучение правил UML;
* сложности в изучении и внедрении UML для начинающих пользователей.

UML прежде всего предназначен для моделирования и разработки программных систем. Наиболее эффективно его применение в следующих областях, как: корпоративные информационные системы, банковские и финансовые услуги, телекоммуникации, транспорт, оборона, авиация, космонавтика, розничная торговля, медицинская электроника, наука, распределенные Web-сервисы [8].

Стоит учесть, что основное назначение UML — предоставить, с одной стороны, достаточно формальное, с другой стороны, достаточно удобное, и, с третьей стороны, достаточно универсальное средство, позволяющее до некоторой степени снизить риск расхождений в толковании спецификаций.

Стандарт UML предлагает следующий набор диаграмм.

Структурные (structural) модели:

* диаграммы классов (class diagrams) – для моделирования статической структуры классов системы и связей между ними;
* диаграммы компонентов (component diagrams) — для моделирования иерархии компонентов (подсистем) системы; диаграммы размещения (deployment diagrams) — для моделирования физической архитектуры системы.

Модели поведения (behavioral):

* диаграммы вариантов использования (use case diagrams) — для моделирования бизнес-процессов и функциональных требований к создаваемой системе;
* диаграммы взаимодействия (interaction diagrams): диаграммы последовательности (sequence diagrams) и кооперативные диаграммы (collaboration diagrams) — для моделирования процесса обмена сообщениями между объектами;
* диаграммы состояний (statechart diagrams) — для моделирования поведения объектов системы при переходе из одного состояния в другое;
* диаграммы деятельности (activity diagrams) — для моделирования поведения системы в рамках различных вариантов использования, или потоков управления.

Рассмотрение UML как модели предполагает точный подход, требующий использования инструментального средства моделирования для подробного описания программной системы. UML модель активно поддерживается и становится важным поставляемым артефактом проекта.

## 5.2 Проектирование бизнес-процессов программного продукта

Предметная область – программа-планировщик, заблаговременно уведомляющий пользователя о предстоящих событиях и хранящий базу данных мероприятий с их основными характеристиками: описание, дата наступления, наименования сопроводительных документов. Программный продукт запускается автоматически при включении ПК и не требует подключения к сети Интернет. Интерфейс системы позволяет настраивать работу уведомлений и отслеживать историю о готовности пользователя к предстоящему событию.

Для представления основных процессов была выбрана нотация UML, позволяющая продемонстрировать работу системы наиболее многогранно.

Взаимодействие между пользователем и программным продуктом отражено с помощью диаграммы вариантов использования, рисунок 5.1.



Рисунок 5.1 – Диаграмма вариантов использования для актера «Пользователь»

На диаграмме вариантов использования показано взаимодействие между вариантами использования и действующими лицами. Она отражает семь основных функциональных требования к системе с точки зрения пользователя, а именно:

* просмотр предстоящих событий и занимаемых ими статусов – «Событие выполнено» или «Событие ожидает выполнения»;
* добавление нового события и соответствующих ему характеристик (описание, дата наступления, перечень документов);
* редактирование событий;
* добавление к событиям сопроводительных документов;
* удаление записи о событии из базы данных;
* отслеживание ответов пользователя на уведомления о готовности к предстоящим событиям, хранящиеся в журнале событий;
* очистка истории журнала событий.

Таким образом, варианты использования — это функции, выполняемые системой, а действующие лица, в данном случае – пользователь, по отношению к создаваемой системе.

Последовательность событий, происходящих в рамках варианта использования, отражена в виде диаграммы последовательности на рисунке 5.2, где действующее лицо – пользователь и объекты, требуемые системе для выполнения варианта использования расположены в верхней части диаграммы.

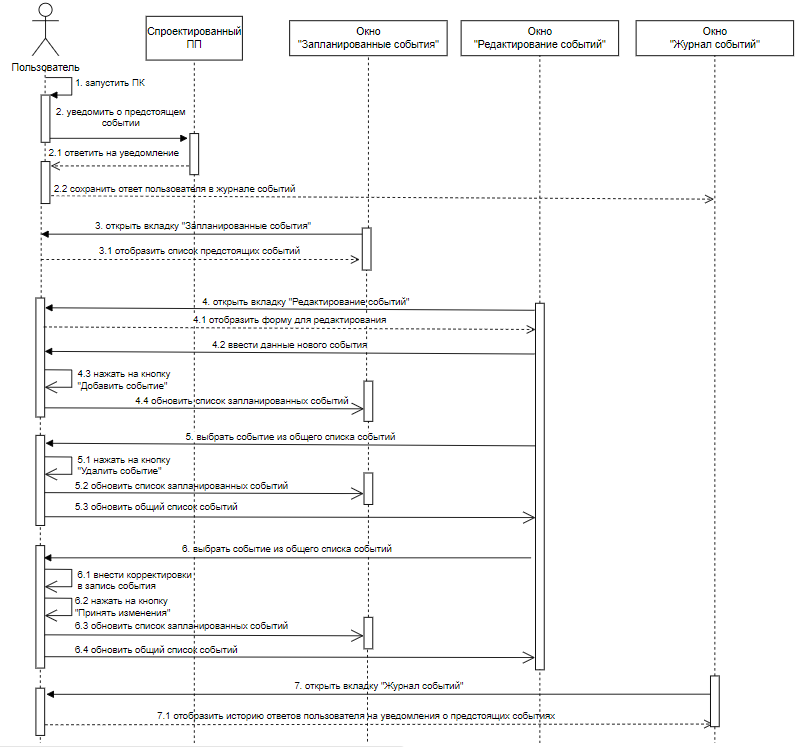


Рисунок 5.2 – Диаграмма последовательности «Взаимодействие пользователя и программного продукта»

Основным предназначением проектируемой системы является уведомление пользователя о предстоящем событии. Процесс оповещения и фиксирования статуса о готовности к событию представлен с помощью диаграммы состояний, рисунок 5.3.

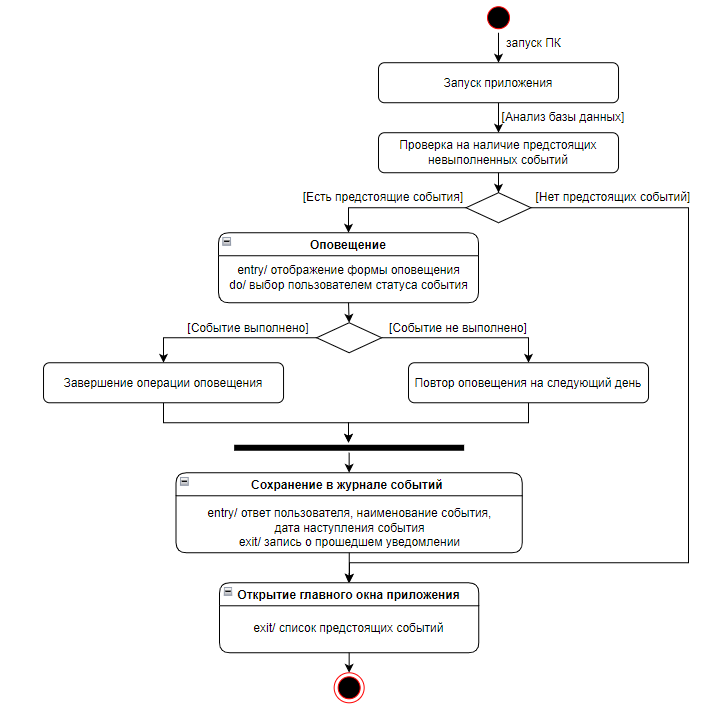


Рисунок 5.3 – Диаграмма состояний объекта «Уведомление о предстоящем событии»

Диаграммы состояний определяют все возможные состояния, в которых может находиться система, а также процесс смены состояний системы в результате наступления некоторых событий, например – ответа пользователем на уведомление о выполнении события.

# Реализация программного продукта

## 6.1 Обоснование выбора СУБД

Проектируемая система «Цифровой помощник для ведущего документоведа кафедры» является локальной однопользовательской системой, главная задача которой заключается в предоставлении пользователю базы событий и уведомлении о подготовке к предстоящему мероприятию. На основании технического задания в реализации программного продукта принято решение использовать реляционную систему управления базами данных для определения всей хранимой информации как набора связанных записей и атрибутов в таблице.

СУБД (система управления базами данных) – это информационная среда для создания баз данных и управления хранимыми в них сведениями об объектах реального мира [9].

Одной из главных задач перед проектированием любой системы, является правильный выбор базы данных. Для выбора подходящей необходимо определить ряд критерий. И главная проблема, которая вытекает, в том, что не имеется четко сформулированных требований к базе данных и поэтому, она должна удовлетворять наиболее широкому кругу потребителей. Которыми являются различные компании. Так как перед любой компанией всегда стоит цель на дальнейший рост, значит СУБД должна обеспечивать по максимуму все возможности на среднем уровне с возможностью дальнейшего роста.

Выбор системы управления баз данных (СУБД) представляет собой сложную многопараметрическую задачу и является одним из важных этапов при разработке приложений баз данных. Выбранный программный продукт должен удовлетворять как текущим, так и будущим потребностям предприятия, при этом следует учитывать финансовые затраты на приобретение необходимого оборудования, самой системы, разработку необходимого программного обеспечения на ее основе, а также обучение персонала. Кроме того, необходимо убедиться, что новая СУБД способна принести предприятию реальные выгоды [10].

2733

Реляционные базы данных пользуются популярностью уже долгое время. Благодаря удачным реализациям реляционных моделей в системах управления и предоставлении удобной работы с данными спрос на них только возрастает.

В ходе сравнительной оценки трех наиболее распространенных реляционных систем управления базами данных – [SQLite](https://www.sqlite.org/), [MySQL](https://www.mysql.com/) и [PostgreSQL](https://www.postgresql.org/), конкурирующих на рынке программного обеспечения по основным показателям, составлена таблица 6.1, отражающая итоги проведенного анализа.

Таблица 6.1 – Сравнение РСУБД

|  |  |  |  |
| --- | --- | --- | --- |
| Критерии оценки | [SQLite](https://www.sqlite.org/) | [MySQL](https://www.mysql.com/) | [PostgreSQL](https://www.postgresql.org/) |
| Тип | Реляционная | Реляционная | Объектно-реляционная |
| Операционная система | Microsoft Windows, Blackberry, Symbian, Maemo, Android, MeeGo, WebOS, NetBSD, FreeBSD, illumos, Solaris 10, Tizen | Microsoft Windows, Oracle Solaris,  macOS, FreeBSD | Linux,  Microsoft Windows, Oracle Solaris,  IBM AIX,  macOS,  HP-UX,  QNX |
| Язык программирования | C, C#, C++, Java, PHP, Python, Ruby | C, Java, SQL, C++, Perl, Python, Tcl, PHP | PL/Tcl, PL/pgSQL, PL/Python, PL/Perl |
| Облачная версия | Да | Да | Нет |
| Исходный код | Открытый | Открытый | Открытый |
| Требование к установке | Нет | Да | Да |
| Хранение БД | Жесткий диск ПК | Веб-хранилище | Резервный сервер |

Проанализировав предложенные РСУБД, выбор сделан в пользу библиотеки языков программирования – [SQLite](https://www.sqlite.org/), Данная система управления базами данных имеет библиотеку в памяти, вызвать и использовать которую возможно без установки или настройки. Стоит учесть, что [SQLite](https://www.sqlite.org/) предполагает хранение базы данных на жестком диске и требует для этого выделенного сервера.

## 6.2 Обоснование выбора среды разработки

Для реализации поставленной задачи была выбрана в качестве среды разработки Microsoft Visual Studio 2019. Так как она наиболее подходит для создания данной автоматизированной информационной системы. Один из главных достоинств является наличие удобного интерфейса большого набора функций и инструментов.

Microsoft Visual Studio – среда разработки программного обеспечения на базе платформы .NET Framework, которая поддерживает новые и улучшенные объекты. Из инструмента программиста, пишущего и отлаживающего код, Microsoft Visual Studio, превратилась в полноценное инструментальное средство, позволяющее автоматизировать деятельность всех членов команды, работающих над проектом [11].

Средства, входящие в состав Visual Studio для Windows, можно использовать для создания привлекательных инновационных приложений для Магазина Windows в среде Windows 8.1. Эти средства включают полнофункциональный редактор кода, мощный отладчик, специальный профилировщик и широкие возможности языковой поддержки, которые позволяют выполнять сборку приложений, написанных на языках HTML5/JavaScript, C++, C# и Visual Basic. В состав Visual Studio для Windows также входит имитатор устройств, который можно использовать для тестирования приложений Магазина Windows на устройствах различных видов [12]. Данная среда программирования позволяет работать с данными различного типа. Большая интеграция со многими языками программирования и программами. Быстро осуществляется компиляция программного кода, выявление и устранение различных ошибок.

## 6.3 Описание языка программирования

C# (произносится как «си шарп») — простой, современный объектно-ориентированный и типобезопасный язык программирования. C# относится к широко известному семейству языков C, и покажется хорошо знакомым любому, кто работал с C, C++, Java или JavaScript.

C# – это объектно-ориентированный язык, который включает в себя программирование.

В C# приложение – это коллекция одного или нескольких классов, структур и других типов. Класс определяется как набор данных и методов, работающих с ними.

Разработка современных приложений все больше склоняется к созданию программных компонентов в виде автономных и самоописательных пакетов, реализующих индивидуальный функционал. Важной особенностью таких компонентов является модель программирования, основанная на свойствах, методах и событиях. Каждый компонент имеет атрибуты, предоставляющие декларативные сведения о компоненте, а также встроенные элементы документации. C# предоставляет языковые конструкции, благодаря чему отлично подходит для создания и применения программных компонентов [13, 14].

Когда говорят C#, нередко имеют в виду технологии платформы .NET (WPF, ASP.NET). И, наоборот, когда говорят .NET, нередко имеют в виду C#. Однако, хотя эти понятия связаны, отождествлять их неверно. Язык C# был создан специально для работы с фреймворком .NET, однако само понятие .NET несколько шире [15].

Инструменты платформы .NET Framework представляют собой основу для создания элементов управления, компонентов и приложений .NET. .NET Framework содержит инструменты, предназначенные для следующих задач:

* представление базовых типов данных и исключений;
* инкапсуляция структур данных;
* операции ввода-вывода;
* доступ к сведениям о загруженных типах;
* вызов проверок безопасности .NET Framework;
* доступ к данным, предоставление графического пользовательского интерфейса на стороне клиента и управляемого сервером графического пользовательского интерфейса на стороне клиента.

.NET Framework предлагает расширенный набор интерфейсов, а также абстрактных и конкретных (неабстрактных) классов. Можно использовать существующие конкретные классы, кроме того, во многих случаях на их основе можно создавать собственные производные классы.

Как часть MS.NET Framework С# содержит много различных классов, которые упорядочены в пространствах имен. Пространство имен (namespace) – это набор связанных классов. Пространство имен также может содержать вложенные пространства имен. Основное пространство имен – это System. К объектам пространств имен можно обращаться при помощи полного имени, используя префиксы.

# Разработка базы данных приложения

Стержневые идеи современных информационных технологий базируются на концепции баз данных. Согласно этой концепции, в основе информационных технологий лежат данные, необходимые для организации баз данных с целью адекватного отображения изменяющегося реального мира и удовлетворения информационных потребностей пользователей.

На начальном этапе проектирования базы данных разрабатывается концептуальная модель данных для предметной области, описывающая связи между основными сущностями с учетом их характеристик, рисунок 7.1.

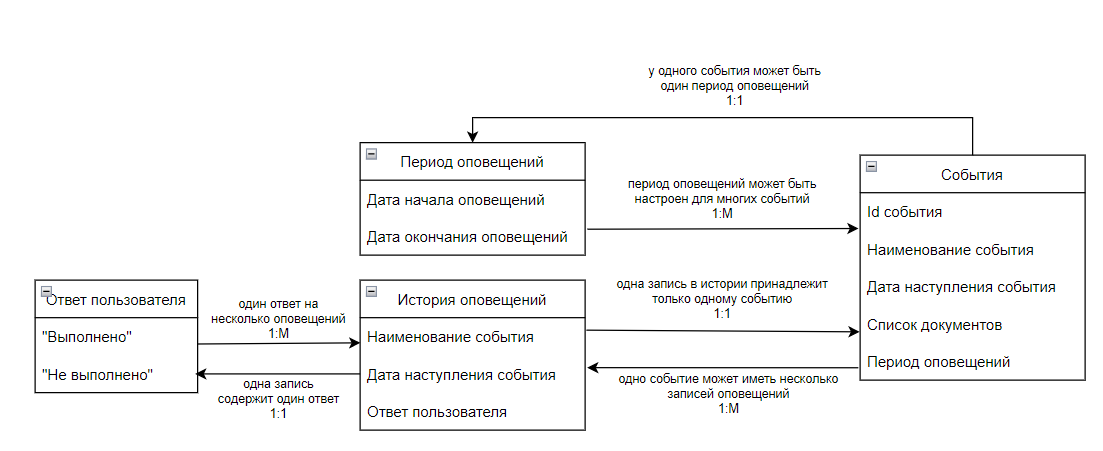


Рисунок 7.1 – Концептуальная модель для цифрового помощника

При создании модели данных необходимо учитывать:

* исключение или сведение к минимуму повторяющихся данных путем задания нормализованных структур;
* обеспечение всем пользователям быстрого доступа к данным;
* возможность расширения структуры модели данных;
* обеспечение целостности данных;
* предотвращение несанкционированного доступа к данным;
* предоставление делегированного доступа к данным отдельному пользователю или группе пользователей;
* снижение сложности создания приложений, предназначенных для ввода, редактирования, обработки данных, а также анализа скрытых закономерностей в структурах данных.

3233

Сведения в базе данных проектируемого приложения должны быть организованы таким образом, чтобы обеспечить минимальную долю избыточности информации, именно поэтому была выделена основная сущность – событие, включающая в себя все необходимые для полноценной информации характеристики.

В таблице 7.1 представлены составляющие каждой из сущностей.

Таблица 7.1 – Описание атрибутов выделенных сущностей

|  |  |
| --- | --- |
| Сущность | Атрибут |
| Событие | Номер события: уникальный идентификатор |
| Наименование события |
| Статус:  «Выполнено» / «Не выполнено» |
| Дата начала уведомлений |
| Дата наступления события |
| Описание события |
| Наименования прилагаемых документов |

Физические модели баз данных определяют способы размещения данных в среде хранения и способы доступа к этим данным, которые поддерживаются на физическом уровне. В таблице 7.2 представлено описание созданной БД программы для программы планировщик событий.

Таблица 7.2 – Описание таблицы базы данных

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица | Поля | Тип данных | Ключевое поле |
| events | Id | integer | PK, AI |
| eventName | text |  |
| statusOfEvent | integer |  |
| dateOfNotification | text |  |
| dateOfEvent | text |  |
| eventDescription | text |  |
| files | text |  |

Ключевыми полями (PK) в базе данных являются номер события. Стоит учесть, что поле «Id», определенное в таблице «События», имеет свойство autoincrement (AI), позволяющее добавлять данные в таблицу без увеличения индекса ключа.

На рисунке 7.2 отображена логическая модель базы данных, построенная на основе концептуальной модели данных.

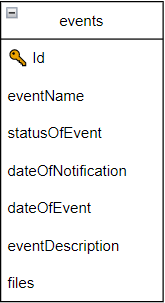


Рисунок 7.2 – Логическая модель данных

Логическая модель данных является прототипом будущей базы данных базы данных и отражает представление пользователя о предметной области программного продукта.

# Описание алгоритма работы программы

С целью точного представления процесса перехода от первичных данных к желаемому результату составлены диаграммы активностей для алгоритмов работы системы, описывающие такие процессы как:

* работа уведомлений, рисунок 8.1;
* добавление нового события в базу данных, рисунок 8.2;
* редактирование событий, рисунок 8.3.

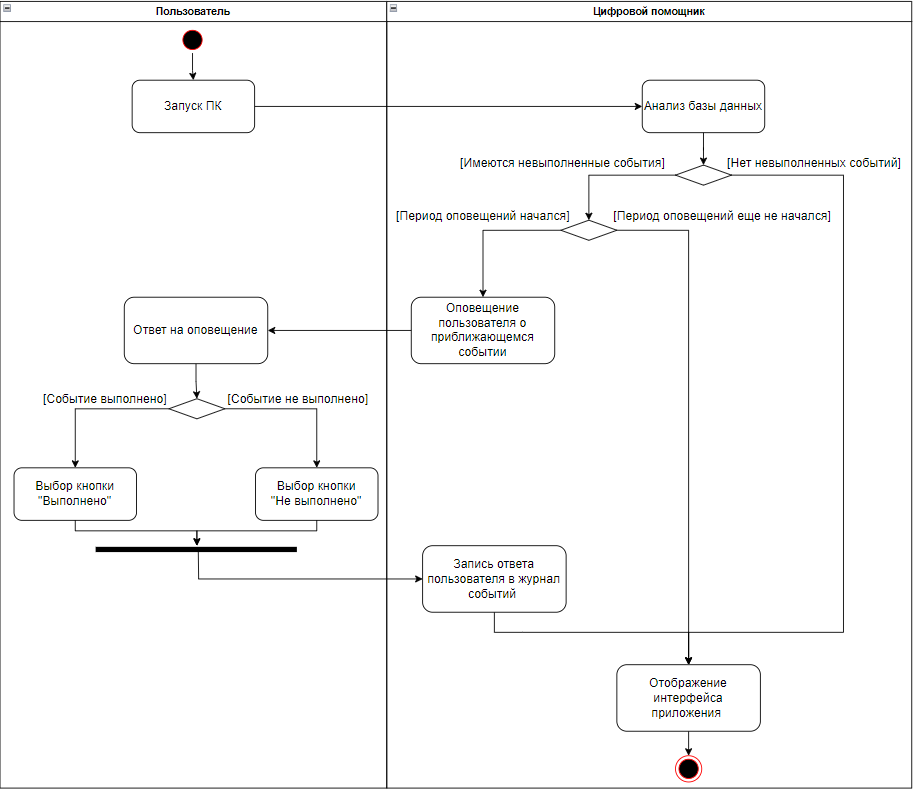


Рисунок 8.1 – Диаграмма активности «Работа системы уведомления о предстоящем событии»

Разработанный программный продукт предусматривает обработку ошибок при наличии некорректно введенных данных в процессе редактировании события или добавления новой записи в базу данных. В случае таких инцидентов как:

3533

* не задано наименование события;
* в любом из полей «Год:», «Месяц:», «День:» внесена отличная от целого числа запись (буква, специальный символ, десятичное число);

программа оповещает пользователя об ошибке и не вносит изменения в базу данных.

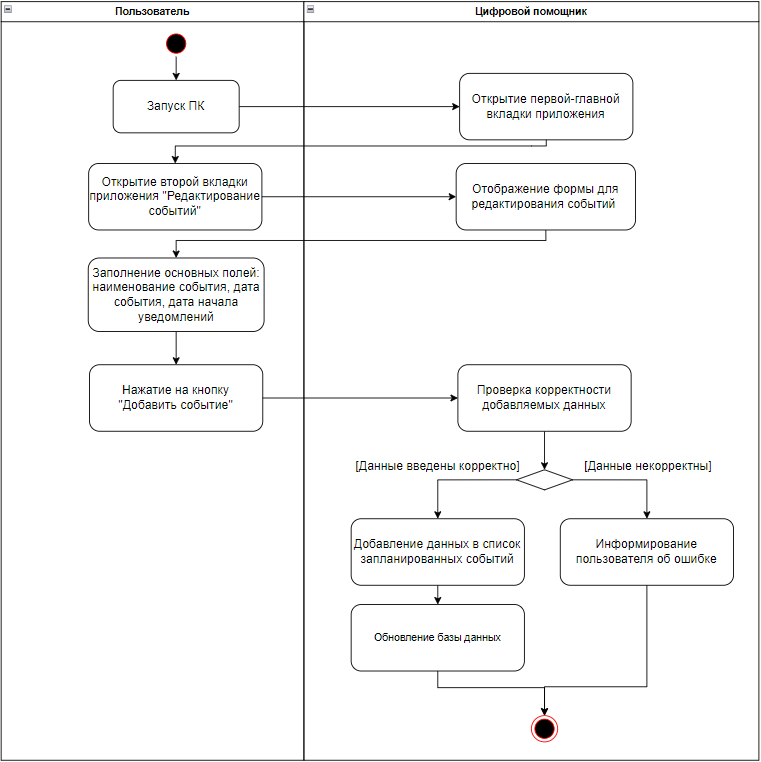


Рисунок 8.2 – Диаграмма активности «Добавление нового события в базу данных системы»

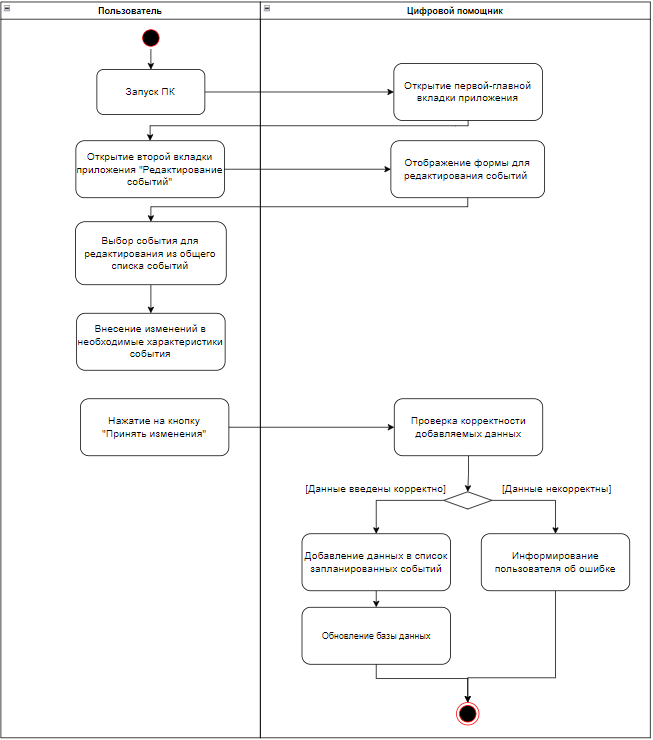


Рисунок 8.3 – Диаграмма активности «Редактирование события из базы данных системы»

Представленные диаграммы активности UML позволяют более детально визуализировать конкретный случай использования и отобразить поток событий в бизнес-процессе.

# Руководство пользователя

## 9.1 Системные требования

Минимальные системные требования:

* операционная система Windows 7, 8/8.1, 10;
* оперативная память (RAM): 256 МБ;
* 220 МБ Свободного места на жестком диске;
* .NET Framework версии 4.7.2.

## 9.2 Интерфейс программного продукта

Для структурирования необходимых данных при разработке интерфейса приложения было выделено три основных вкладки.

Первая вкладка содержит полный перечень уже имеющихся в базе данных запланированных событий, отсортированных в порядке ближайших дат. В данной вкладке отображаются такие сведения как: наименование события, дата наступления (дата/месяц/год), описания события, список необходимых к заполнению документов, рисунок 9.1.

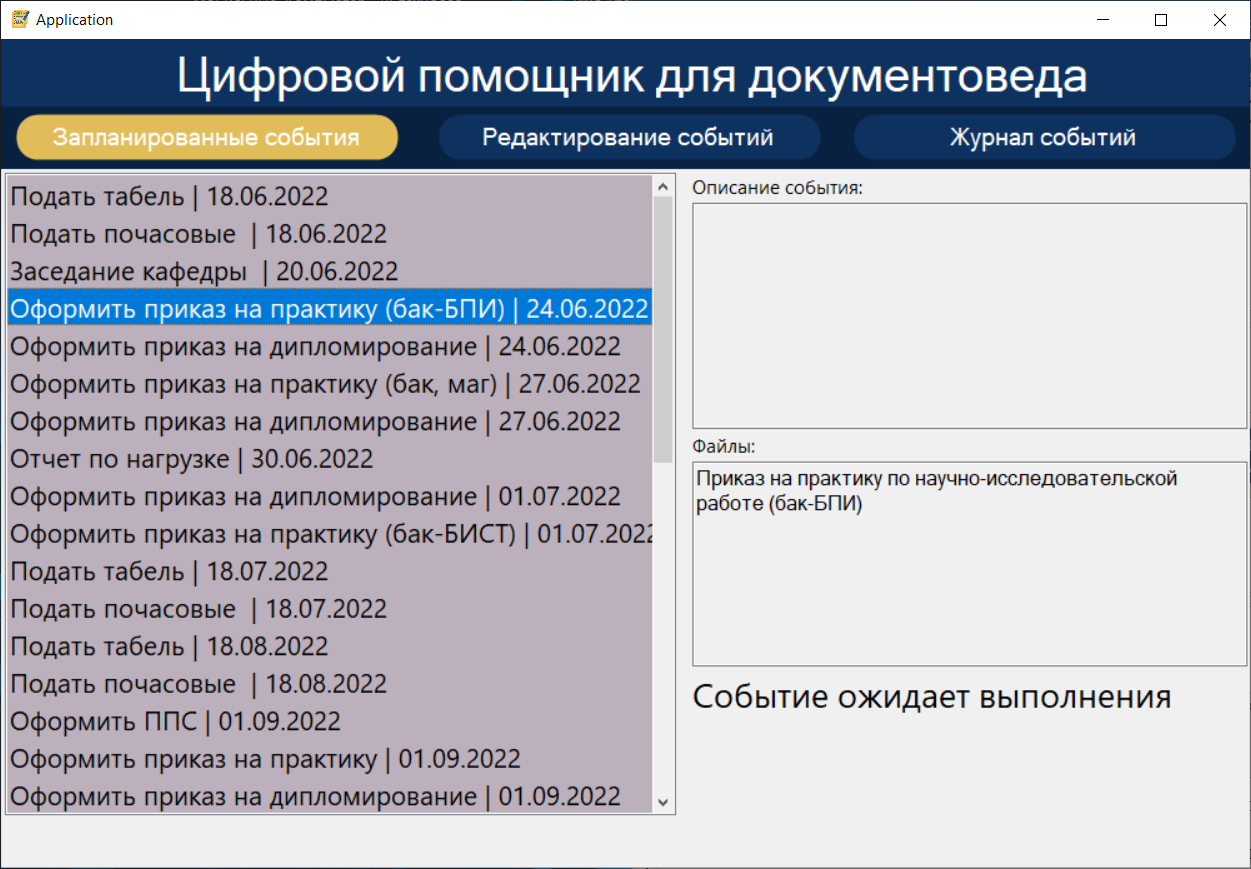


Рисунок 9.1 – Вкладка «Запланированные события»

3833

При выборе какого-либо события из имеющегося списка в правой нижней части окна высвечивается текущий статус выбранного события.

Вторая вкладка предназначено для добавления, редактирования или удаления события, рисунок 9.2. Данное окно содержит следующее:

* перечень всех запланированных и прошедших события с указанием их дат свершения;
* поле для записи нового или редактирования имеющегося события;
* поля для задания наступления события (год, месяц, день);
* поля для настройки начала работы режима оповещений (год, месяц, день);
* поле для описания события;
* поле для ввода перечня документов к соответствующему события;
* кнопки: «Добавить событие», «Удалить событие», «Принять изменения».

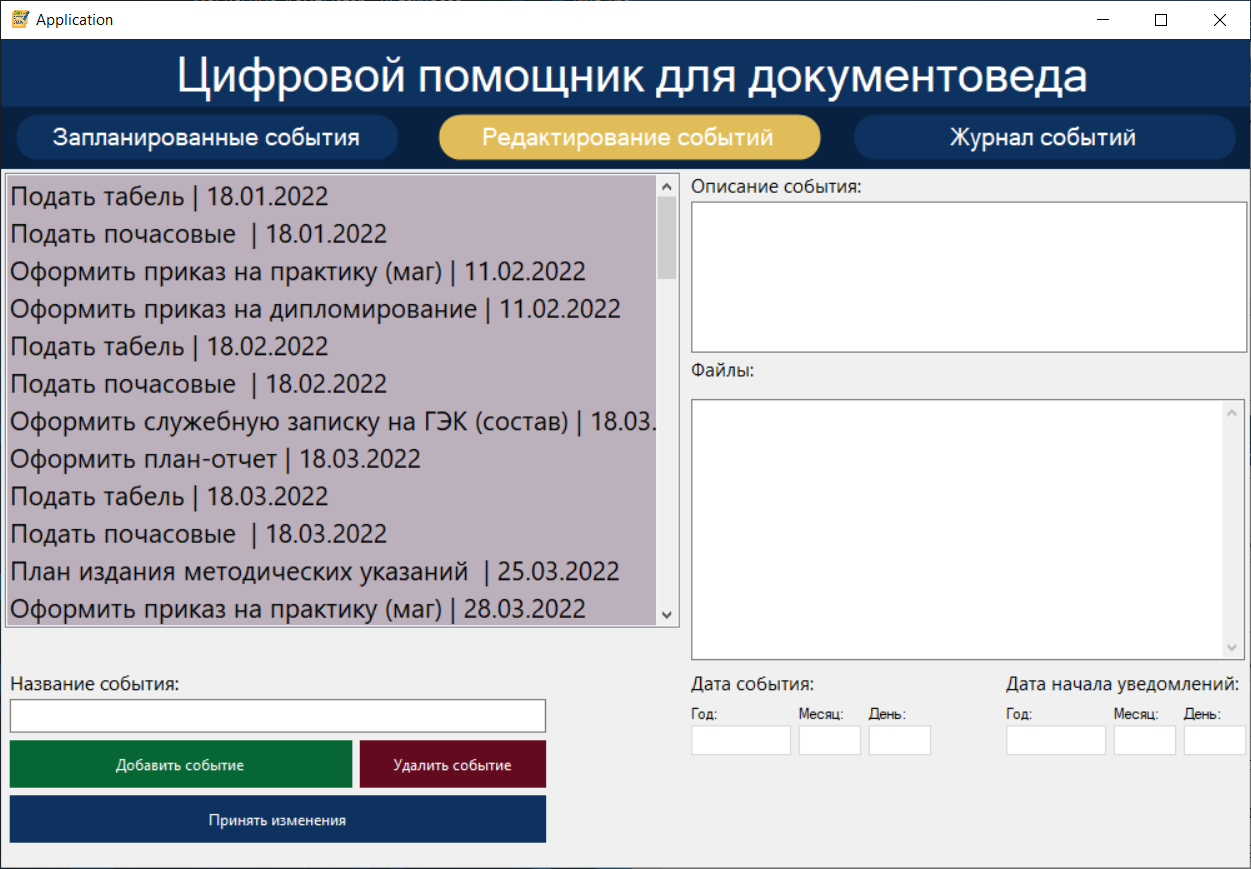


Рисунок 9.2 – Вкладка «Редактирование событий»

Для редактирования необходимого события необходимо выбрать его из представленного в левой части списка, внести корректировки и нажать на кнопку «Принять изменения».

Выберем событие «Подать табель» для добавления описания и изменения дат наступления события и начала работы режима оповещений. На рисунках 9.3-9.4 представлено отображение информации до и после редактирования соответственно.

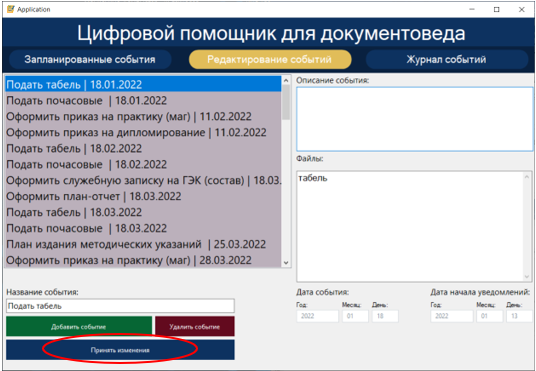


Рисунок 9.3 – Отображение информации до редактирования события

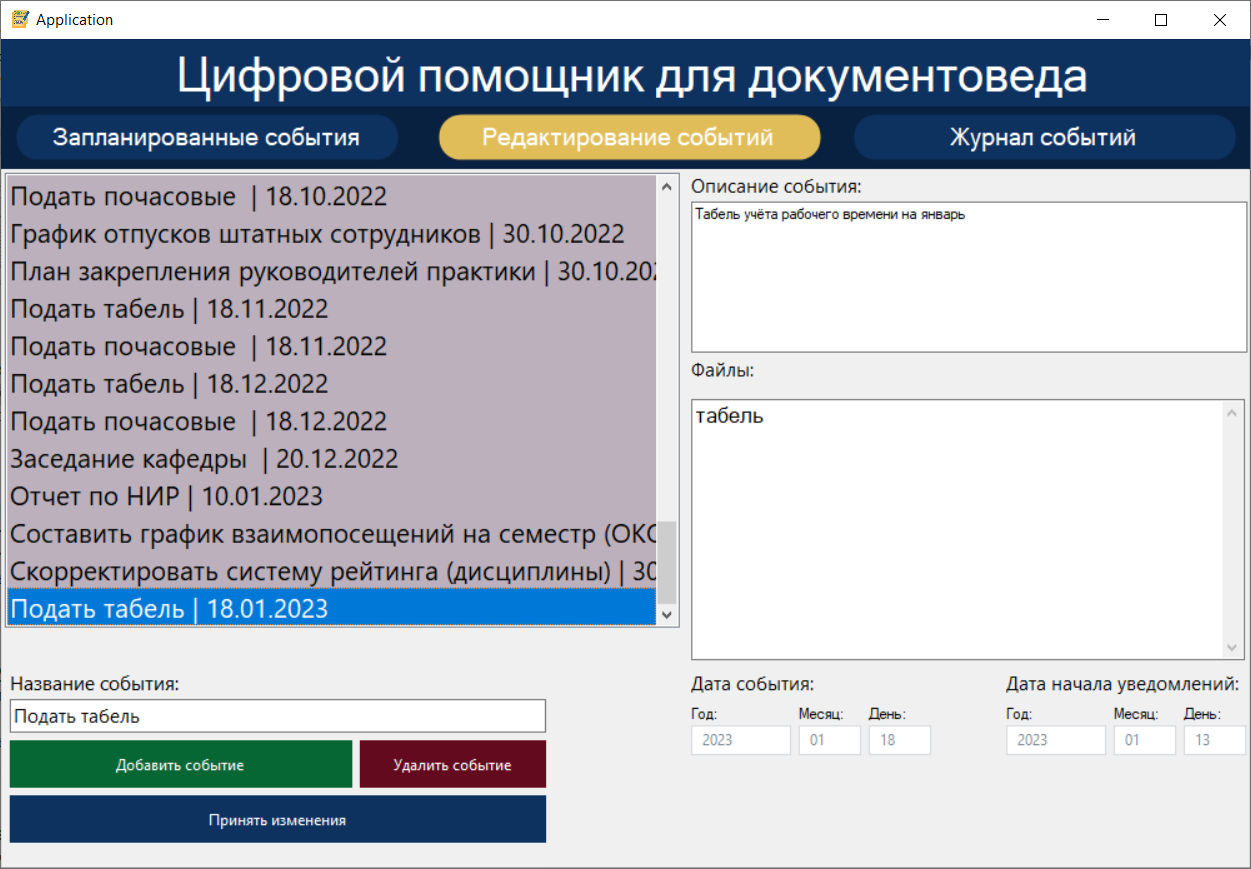


Рисунок 9.4 – Отображение отредактированного события

Удаление события производится путем выбора необходимой позиции из списка и нажатия на кнопку «Удалить событие».

Произведем удаление события «Подать почасовые», которое состоялось 18.01.2022. Отображение перечня событий до и после удаления представлено на рисунках 9.5-9.6, соответственно.

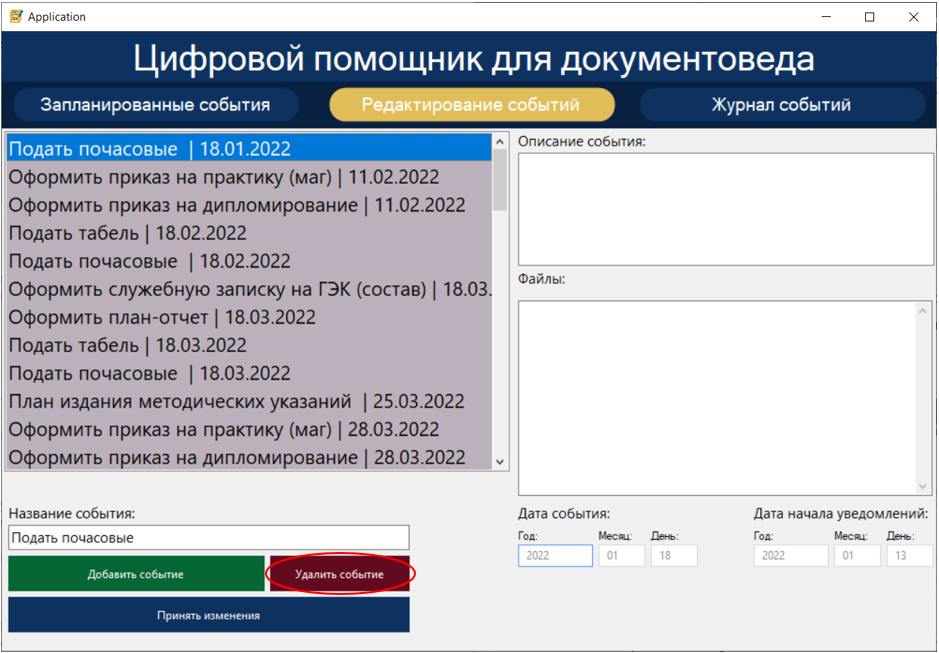


Рисунок 9.5 – Перечень событий до выполнения удаления

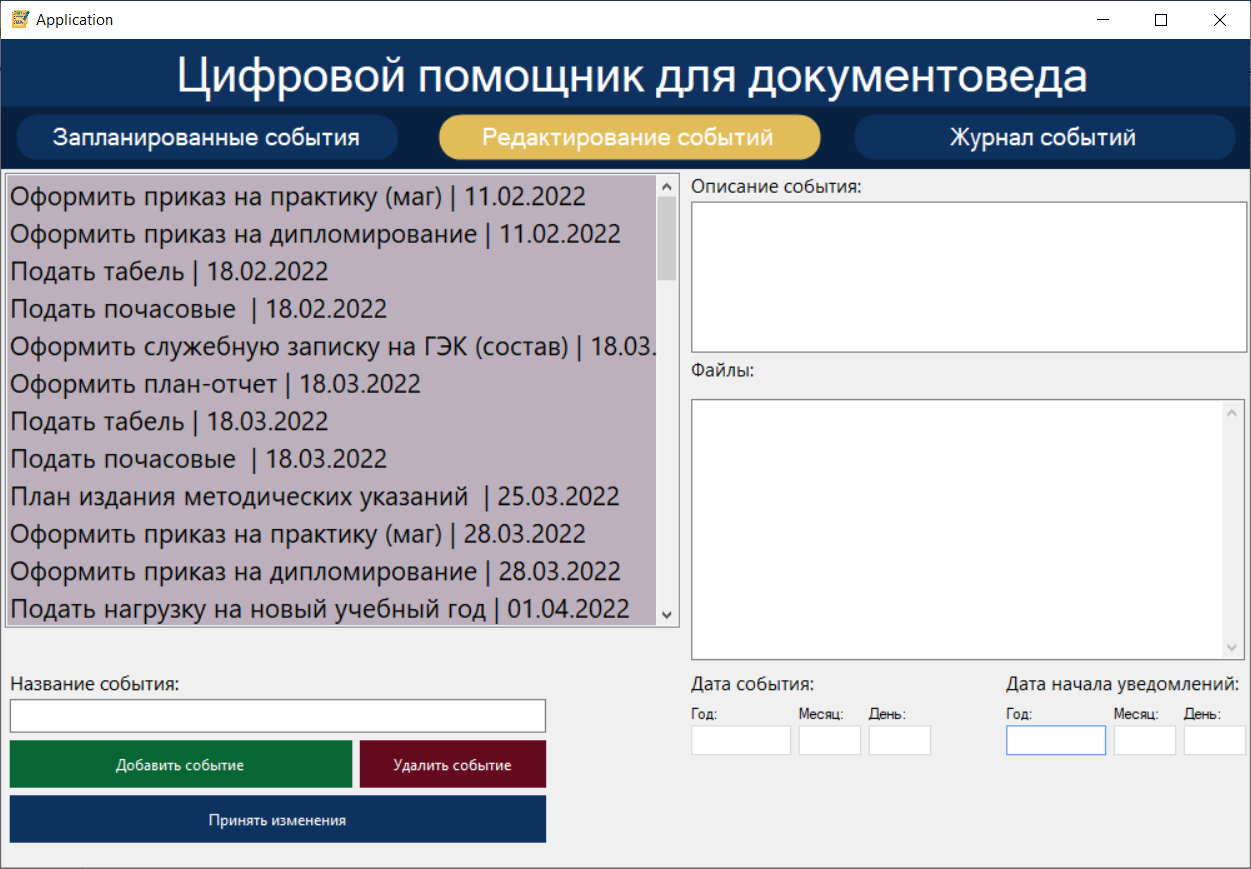


Рисунок 9.6 – Перечень событий после выполнения удаления

Для полноты описания события интерфейс приложения предусматривает добавление файлов/наименования документов, необходимых для сдачи к установленному сроку.

Запись наименований документов для оформления производится в узком поле (TextBox), расположенном в левой половине окна после абзаца «Файлы:». Сохранение внесенных файлов производится автоматически при добавлении нового события путем нажатия на кнопку «Добавить событие», или же, путем редактирования уже имеющегося в списке события и нажатия на кнопку «Принять изменения».

Для примера добавим наименования файлов для события «Оформить приказ на практику (бак, маг)». На рисунках 9.7-9.8 представлено отображение информации по выбранному событию до и после внесения новых файлов соответственно.

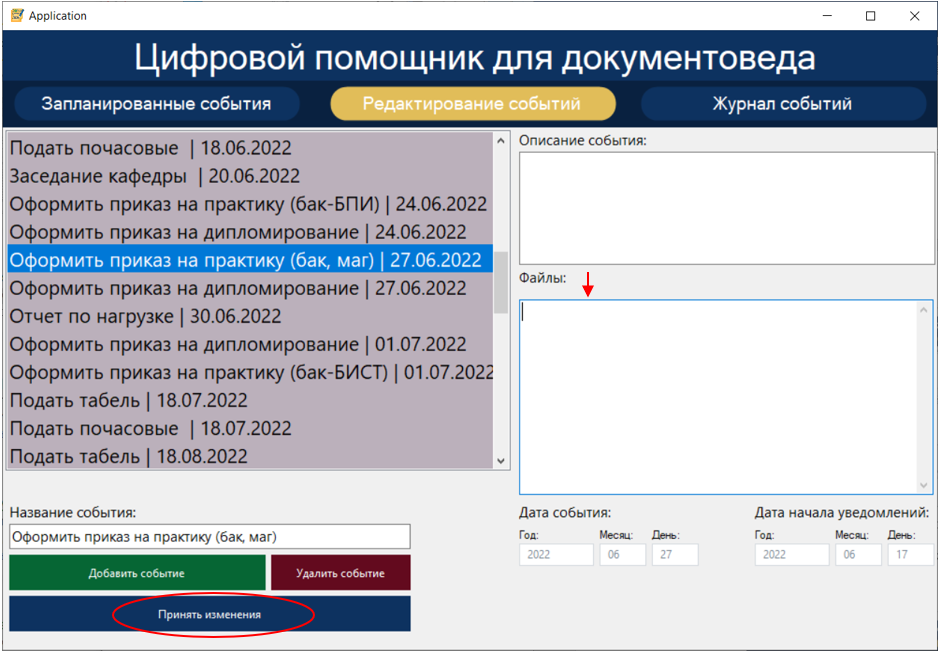


Рисунок 9.7 – Добавление файлов к событию

Проверим добавление прилагаемых к событию документов, перейдя во вкладку «Запланированные события».

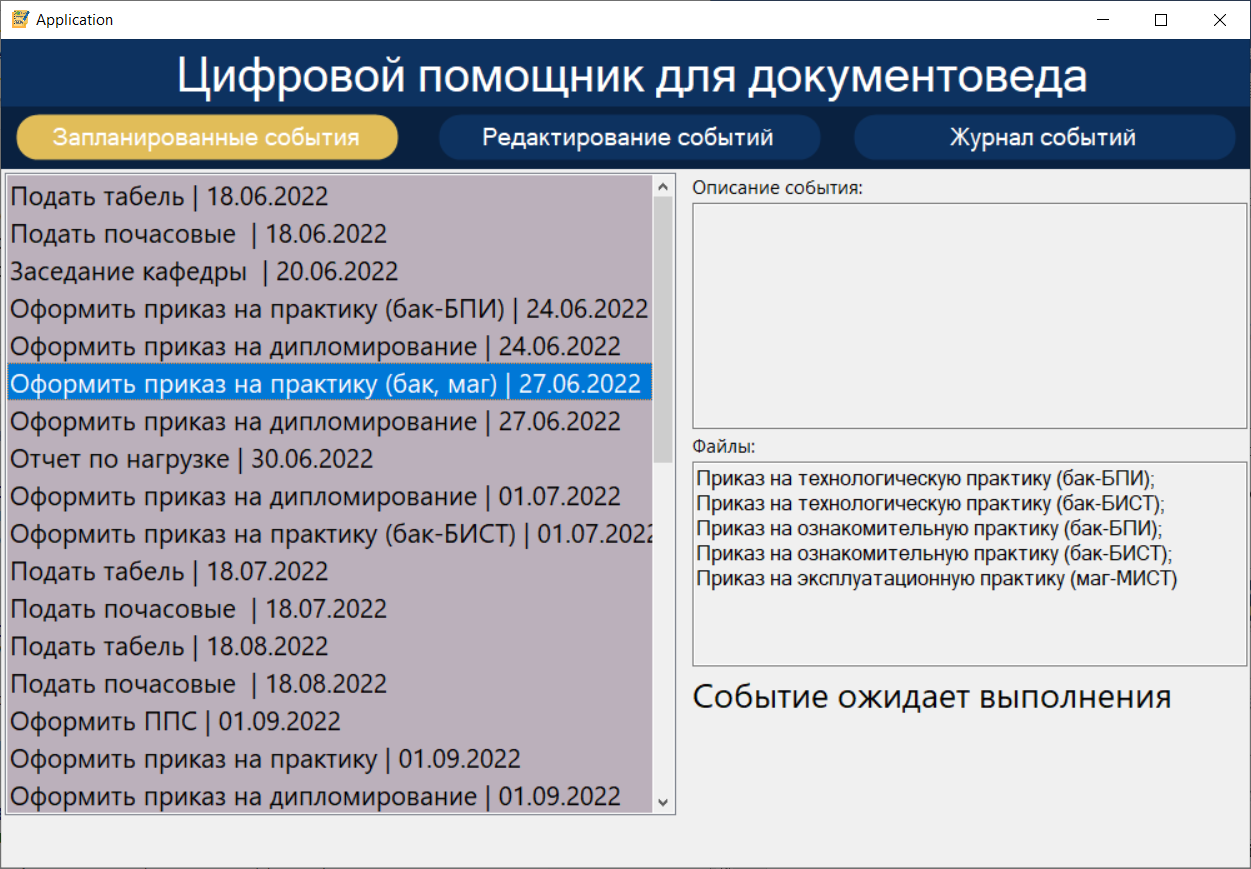


Рисунок 9.8 – Добавление файлов к событию

Для добавления события обязательным условием является заполнение таких данных как: наименование события, дата события и дата начала уведомлений, рисунок 9.9.

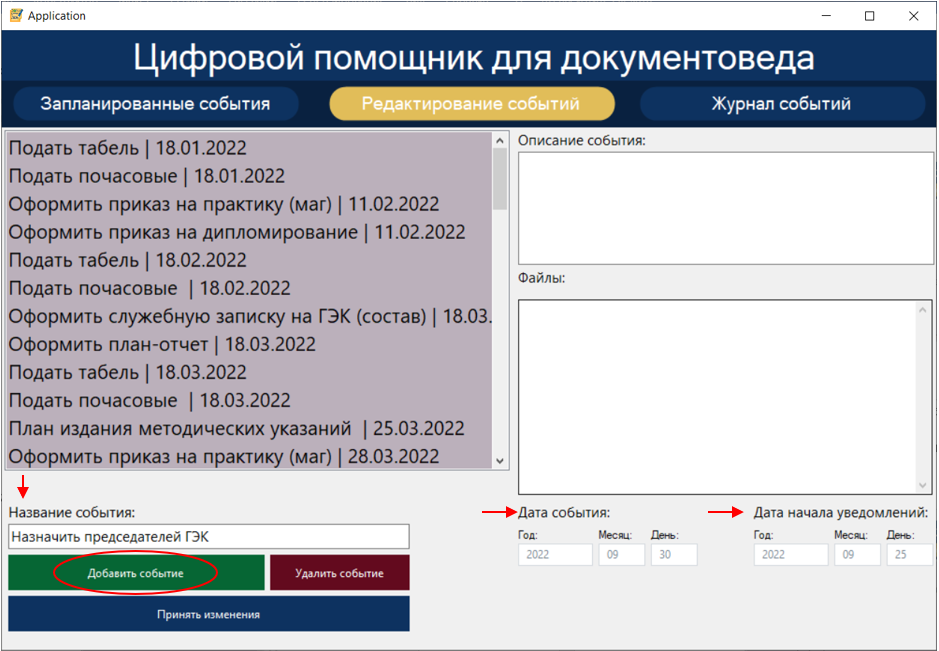


Рисунок 9.9 – Добавление нового события

Отображение добавленного события «Назначить председателей ГЭК», дата исполнения которого 30.09.2022, проверим, перейдя на главную вкладку «Запланированные события», рисунок 9.10.

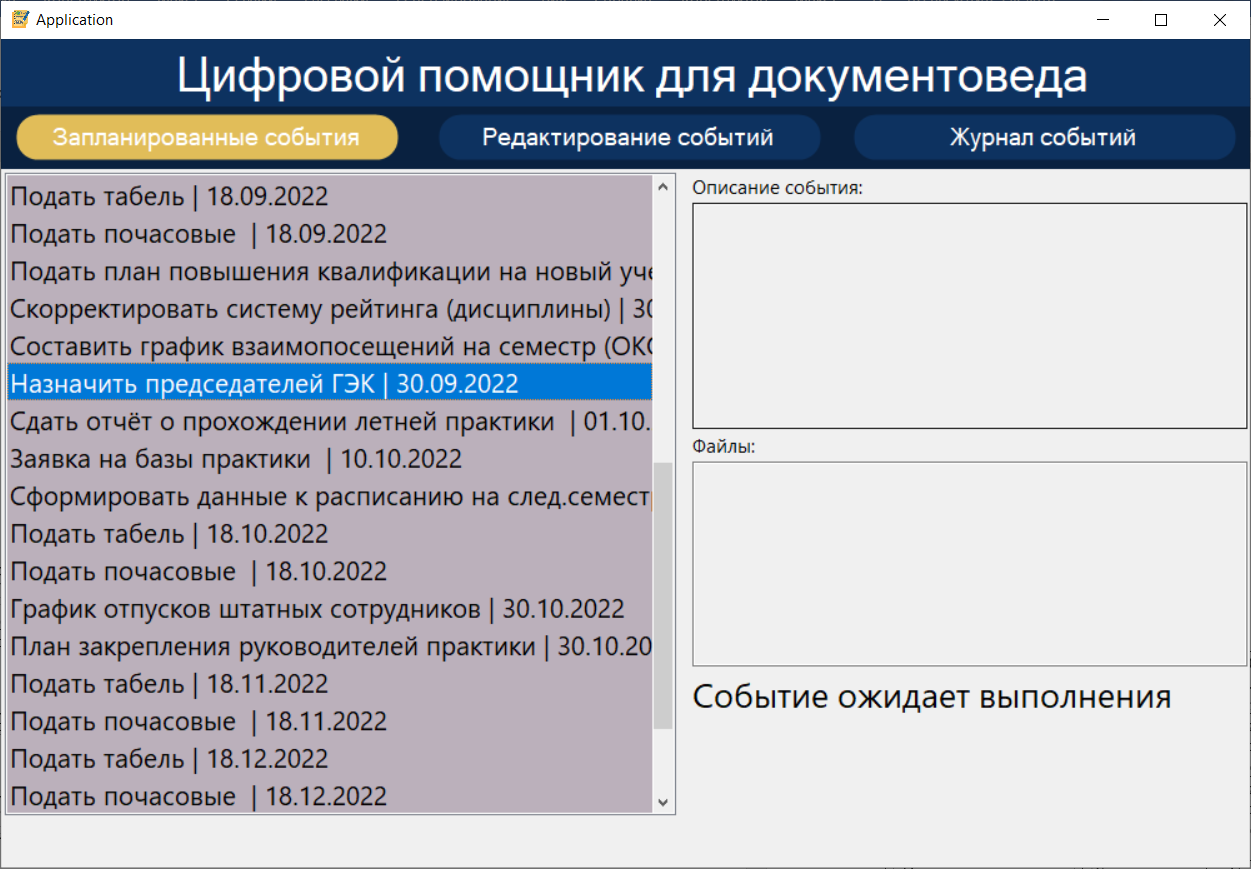


Рисунок 9.10 – Результат добавления нового события

Стоит учесть, что при записи в полях для дат букв, специальных символов или десятичных чисел приложение сообщит пользователю об ошибке, рисунки 9.11-9.13.

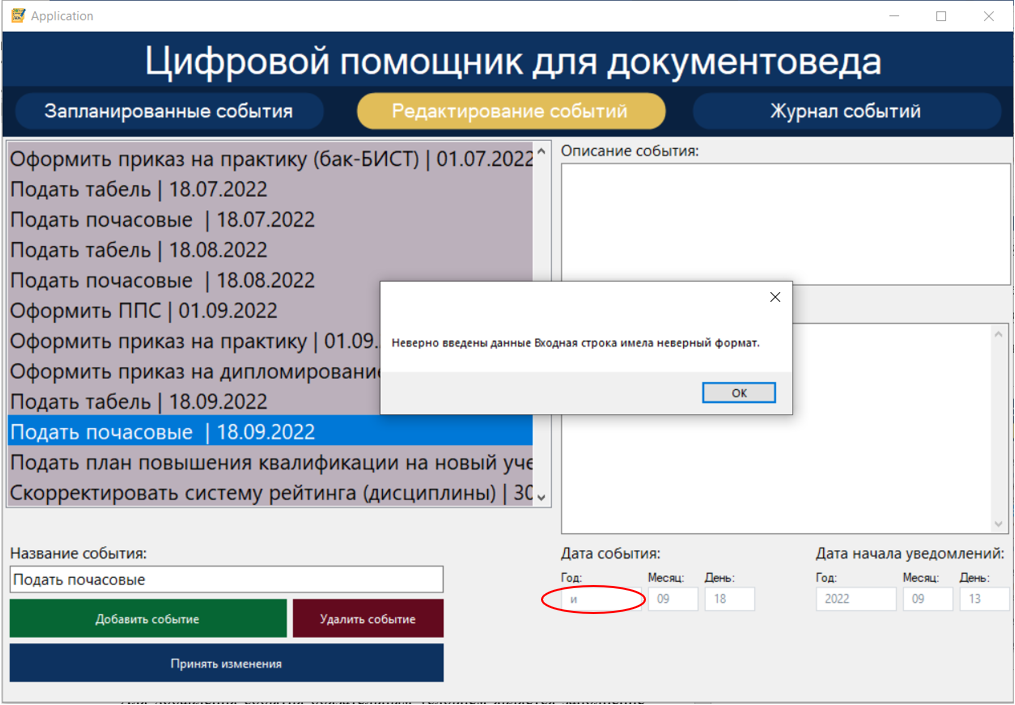


Рисунок 9.11 – Предупреждение об ошибке при вводе буквы в поле «Год:»

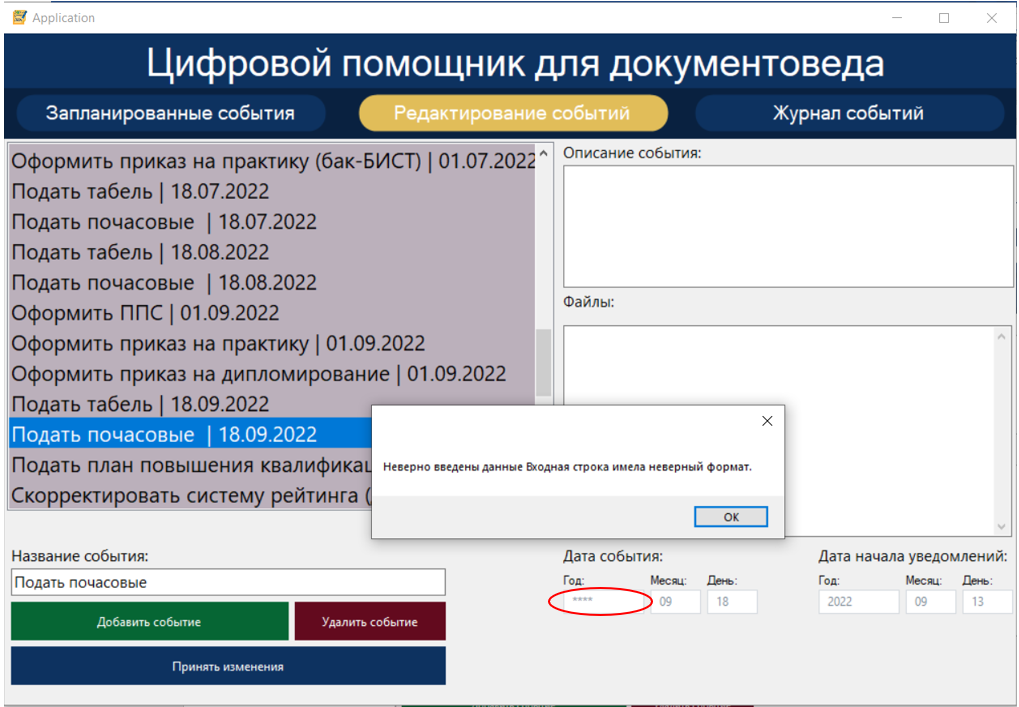


Рисунок 9.12 – Предупреждение об ошибке при вводе специальных символов в поле «Год:»

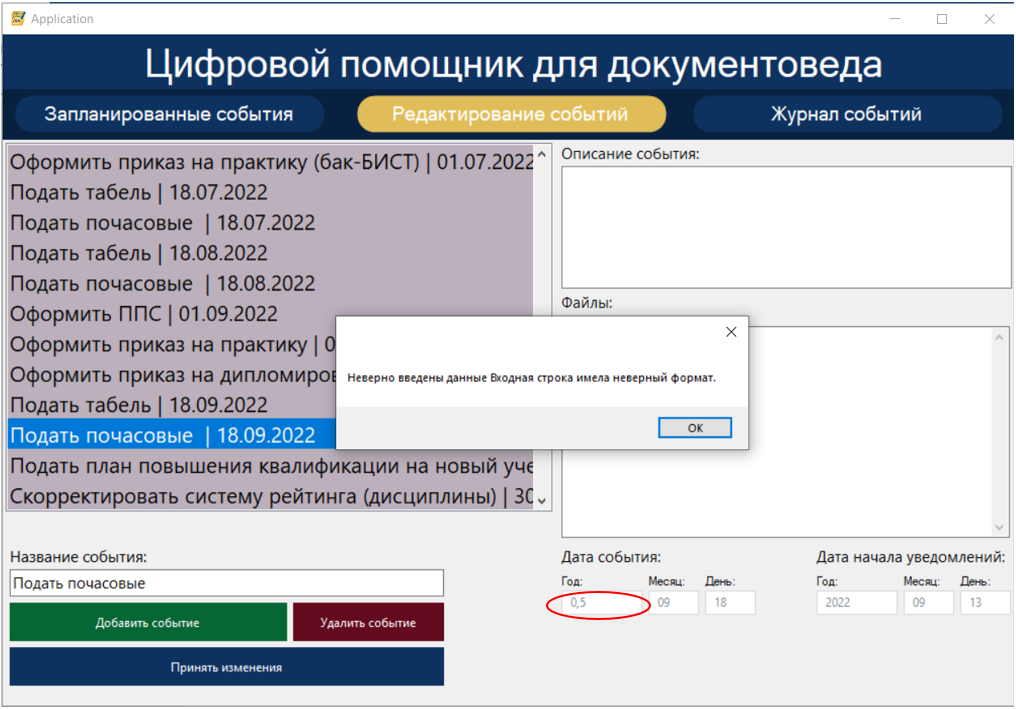


Рисунок 9.13 – Предупреждение об ошибке при вводе десятичного числа в поле «Год:»

Все изменения (добавление, удаление, принятие изменений), внесенные во вкладке «Редактирование событий», синхронизируется с первой вкладкой – «Запланированные события» и отображаются в общем списке событий.

Третья вкладка предназначена для ведения журнала событий. В данном окне отображаются такие данные как: наименование события, дата наступления, исход события (выполнено / не выполнено в срок) и принятое от пользователя сообщение о статусе работы (выполнено / не выполнено), рисунок 9.14.

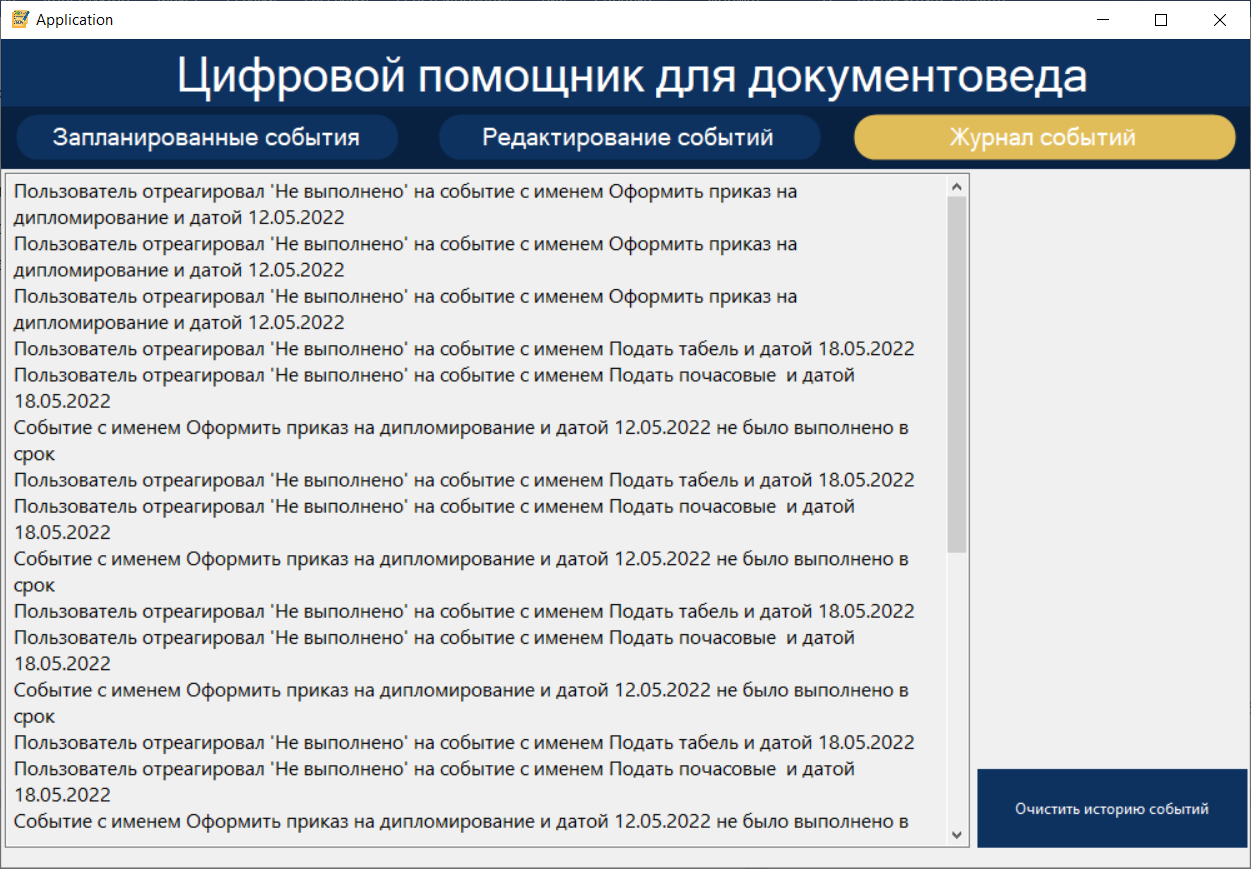


Рисунок 9.14– Вкладка «Журнал событий»

Пример окна оповещения представлен на рисунке 9.15.

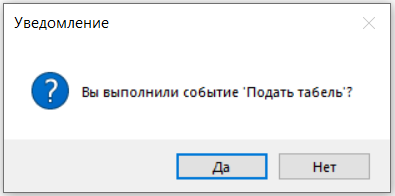


Рисунок 9.15 – Уведомление

Сведения в базе данных проектируемого приложения организованы таким образом, чтобы обеспечить минимальную долю избыточности информации, именно поэтому предусмотрен удобный и понятный в использовании интерфейс.

# ЗАКЛЮЧЕНИЕ

В ходе написания выпускной квалификационной работы было осуществлено подробное описание выпускающей кафедры «Информационные технологии транспорта» Сибирского государственного университета путей сообщения (СГУПС), выделены основные события для своевременного выполнения в соответствии с календарным планом.

На основе изученной информации было составлено техническое задание, включающее в себя:

* наименование разработки;
* цели задания;
* задачи, решаемые спроектированной системой;
* функциональные требования;
* требования к надежности;
* общие видение продукта;
* стадии и этапы разработки;
* порядок сдачи проекта.

Этапом перед проектированием приложения стало изучение и проведение сравнительного анализа характеристик аналоговых программ с целью выявления преимуществ и недостатков: визуальных, функциональных и конструкционных.

Выбрана среда разработки, смоделированы бизнес-процессы.

Итогом проведенной работы стала разработка программного продукта, предназначенного для хранения базы событий и уведомления пользователя о подготовке к предстоящим мероприятиям.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Положение о кафедре Сибирского государственного университета путей сообщения : принято Ученым советом СГУПС протокол №9 от 30.05.2008.
2. О повышении эффективности работы кафедры [Электронный ресурс] // URL https://cyberleninka.ru/article/n (дата обращения: 16.04.2022).
3. Положения о структурных подразделениях [Электронный ресурс] // URL: https://www.stu.ru/about/index.php?page=2382 (дата обращения: 16.04.2022).
4. Планировщик дел: 10 лучших программ [Электронный ресурс] // URL: https://amssoft.ru/amsblog/planirovshchik-zadach.php (дата обращения: 26.04.2022).
5. Планировщик событий // ЛидерТаск [Электронный ресурс] – URL: https://www.leadertask.ru/content/view/596 (дата обращения: 26.04.2022).
6. Язык UML. Руководство пользователя / Г. Буч, Д. Рамбо, И. Якобсон [Электронный ресурс] – URL: http://needworld.ru/attachments/buch-g-rambo-d-jakobson-i-jazyk-uml-rukovodstvo-polzovatelja-2-e-izdanie-2006-pdf.22103/ (дата обращения: 27.04.2022).
7. UML // Википедия. [Электронный ресурс] – URL: http://ru.wikipedia.org/ wiki/UML (дата обращения 27.04.2022).
8. Учебно-методическое пособие по дисциплине «Анализ и проектирование на UML» / Ф. А. Новиков [Электронный ресурс] – URL: <https://books.ifmo.ru/file/pdf/424.pdf> (дата обращения: 27.04.2022).
9. Тарасов С. СУБД для программиста. Базы данных изнутри. М.: Соломон, 2017. – 320 с.
10. Обоснование выбора СУБД // Studwood.ru [Электронный ресурс] – URL: https://studwood.ru/1883953/informatika/issledovatelskaya\_chast(дата обращения 03.05.2022).
11. Microsoft Visual Studio [Электронный ресурс] – URL: https://www.visualstudio.com/ (дата обращения 03.05.2022).

4833

1. Visual Studio // StudFiles [Электронный ресурс] – URL: https://studfiles.net/preview/5733938/page:3/(дата обращения 03.05.2022).
2. Краткий обзор языка C# // Micrоsoft [Электронный ресурс] – URL: https://docs.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/(дата обращения 03.05.2022).
3. Введение в C# // METANIT.COM [Электронный ресурс]– URL: https://metanit.com/sharp/tutorial/1.1.php(дата обращения 03.05.2022).
4. Базовые элементы C# [Электронный ресурс] – URL: <http://smolapo.ru/sites/default/files/Prepod/Efremova/pril/13_2.pdf> (дата обращения 03.05.2022).