**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(ВлГУ)**

**Колледж инновационных технологий и предпринимательства**

КАФЕДРА ФИЗИКИ И ПРИКЛАДНОЙ МАТЕМАТИКИ

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К КУРСОВОЙ РАБОТЕ**

**ПО ДИСЦИПЛИНЕ «Технология разработки программного обеспечения»**

**на тему «Разработка веб-приложения: «Агрегатор календарей»»**

Студента Рушева Алексея Михайловича

Направление подготовки СПО 09.02.03 "Программирование в компьютерных системах"

Группы ПКсп-117

*Руководитель*: преподаватель КИТП \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Лексин А.Ю.

Владимир 2021

ВЛАДИМИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Зав. кафедрой

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

ЗАДАНИЕ

НА КУРСОВУЮ РАБОТУ

по дисциплине «Технология разработки программного обеспечения»

Выдано студенту Рушеву Алексею Михайловичу группы ПКсп-117

Направление подготовки СПО 09.02.03 «Программирование в компьютерных системах»

1. Тема работы «Разработка веб-приложения "Агрегатор календарей"»

2. Срок сдачи законченной работы 08.04.2021

3. Исходные данные к курсовой работе постановка задачи.

4. Содержание расчетно-пояснительной записки (перечень подлежащих разработке вопросов)

Введение.

1. Анализ предметной области.
2. Постановка задачи
3. Проектирование приложения.
4. Разработка приложения.
5. Тестирование.
6. Руководство пользователя.

Заключение.

5. Постановка задачи

ЧАСТЬ 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

1. Выполнить описание предметной области.
2. На основании описания предметной области провести обзор существующих аналогов.

ЧАСТЬ 2. ПОСТАНОВКА ЗАДАЧИ

1. Определить требования и задачи для достижения поставленной цели.
2. Определить минимальные требования для работы программы.

ЧАСТЬ 3. ПРОЕКТИРОВАНИЕ ПРИЛОЖЕНИЯ

1. На основании описания предметной области выполнить разработку ER-модели, реляционной модели данных.
2. Составить словарь данных, в котором представлены все поля таблиц базы данных с указанием ключа, наименования, типа, обязательности заполнения и заметок.
3. Представить диаграмму прецедентов, диаграмму активностей, диаграмму развертывания.
4. Расписать основные методы, используемые в проектировании интерфейса.

ЧАСТЬ 4. РАЗРАБОТКА ПРИЛОЖЕНИЯ

1. Описать выбор инструментальных средств разработки приложения.
2. На основании ER-диаграммы базы данных, описать физическую реализацию базы данных. Если в работе с базой данных используются представления, то описать подробно каждое из них.
3. Описать все классы приложения с пояснением для каких целей они предназначены и методы с описанием: назначения, входных значений и выходных значений каждого метода.
4. Кратко описать структуру и компоненты, используемые при разработке интерфейса.

ЧАСТЬ 5. ТЕСТИРОВАНИЕ

1. Составить тест-требования, определяющие, что должно быть протестировано, но не определяющие, как это должно быть сделано.
2. По тест-требованиям разработать тест-план, в каждом тестовом примере которого обязательно перечислены все входные значения и ожидаемые выходные значения, а также сценарий, описывающий последовательность действий, которые необходимо выполнить для выполнения тестового примера.
3. Указать результаты тестирования некоторого функционала и общую оценку от проведенного тестирования.

ЧАСТЬ 6. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

1. Описать действия пользователя в приложении, с момента установки его и до получения конечного результата, рассмотрев весь возможный функционал приложения.

Дата выдачи задания 03.02.2021

Руководитель Лексин А.Ю.

Задание принял к исполнению Рушев А.М.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 5](#_Toc68182232)

[1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 6](#_Toc68182233)

[1.1 Описание предметной области 6](#_Toc68182234)

[1.2 Обзор существующих решений задачи 7](#_Toc68182235)

[1.2.1 Google Календарь. 7](#_Toc68182236)

[1.2.2 Яндекс.Календарь. 8](#_Toc68182237)

[2 ПОСТАНОВКА ЗАДАЧИ 10](#_Toc68182238)

[2.1 Определение требований и задач 10](#_Toc68182239)

[2.2 Минимальные требования для работы программы 10](#_Toc68182240)

[3 ПРОЕКТИРОВАНИЕ ПРИЛОЖЕНИЯ 12](#_Toc68182241)

[3.1 ER-диаграмма базы данных 12](#_Toc68182242)

[3.2 Словарь данных 13](#_Toc68182243)

[3.3 Диаграмма прецедентов 15](#_Toc68182244)

# ВВЕДЕНИЕ

АФВЫВФА

# АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

## 1.1 Описание предметной области

Целью разрабатываемого проекта является создание веб-приложения: «Агрегатор календарей», для отображения событий из календарей с разных сервисов (google, outlook и т.д.).

Онлайн календарь – это сервис для лёгкой организации дел, событий, их проверки, а также добавления новых мероприятий.

Сервис обладает данным функционалом:

* Добавление события в календарь;
* Редактирование события в календаре;
* Удаление события в календаре;
* Переключение между месяцами;
* Переключение на текущий день.

Пример календаря представлен ниже (Рисунок 1).

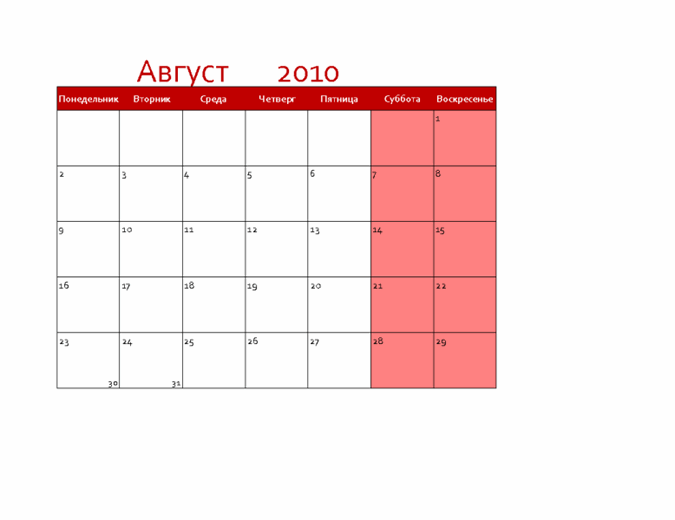


Рисунок 1 – Календарь

## 1.2 Обзор существующих решений задачи

На сегодняшний день существует большое количество календарей, с помощью которых можно просматривать напоминания, корректировать их и добавлять новые, а также удалять.

Примеры таких календарей представлены ниже:

### 1.2.1 Google Календарь.

Данный календарь разработан компанией «Google» и он охватывает весь необходимый функционал для календарей.

Данный проект имеет следующий функционал:

* Создание нового напоминания для даты, а также в разное время;
* Просмотр календаря (целый месяц и по неделям);
* Выбор нужного месяца и года;
* Варианты просмотров (дни рождения, задачи, напоминания, праздники);
* Настройка напоминания;
* Поиск по ключевым словам в календаре;
* Настройки календаря.

Интерфейс Google календаря представлен ниже (Рисунок 1).

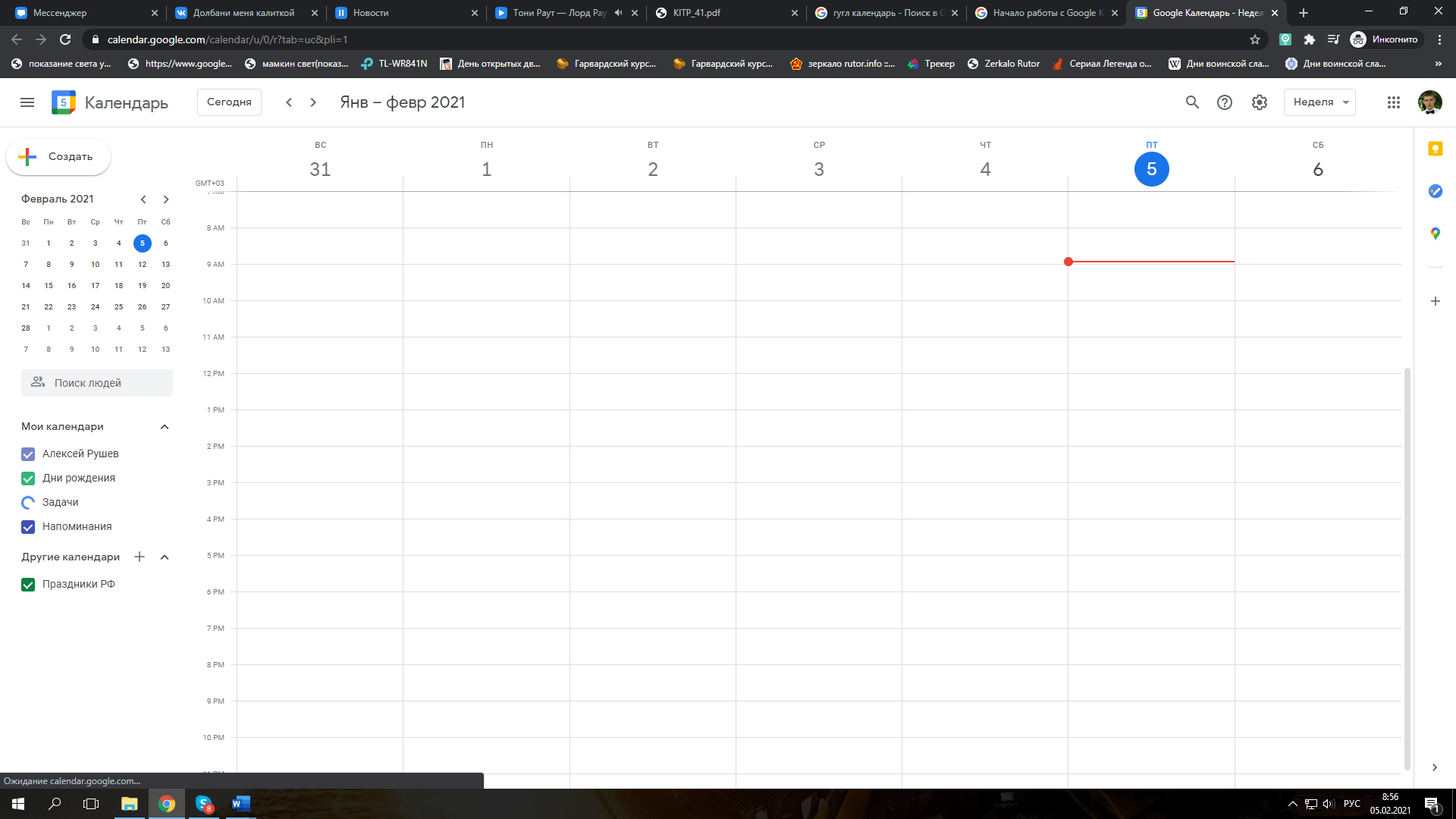


Рисунок 1 – Google календарь

### 1.2.2 Яндекс.Календарь.

Данный проект имеет следующий функционал:

* Создание нового напоминания для даты, а также в разное время;
* Просмотр календаря (целый месяц, по неделям, дням);
* Выбор нужного месяца и года;
* Варианты просмотров (дни рождения, задачи, напоминания, праздники);
* Поиск людей (для показа календаря);
* Настройка напоминания;
* Настройки календаря;
* При создании напоминания есть выбор куда отправится уведомление (почта, смс, CalDAV).

Интерфейс Яндекс календаря представлен ниже (Рисунок 2).

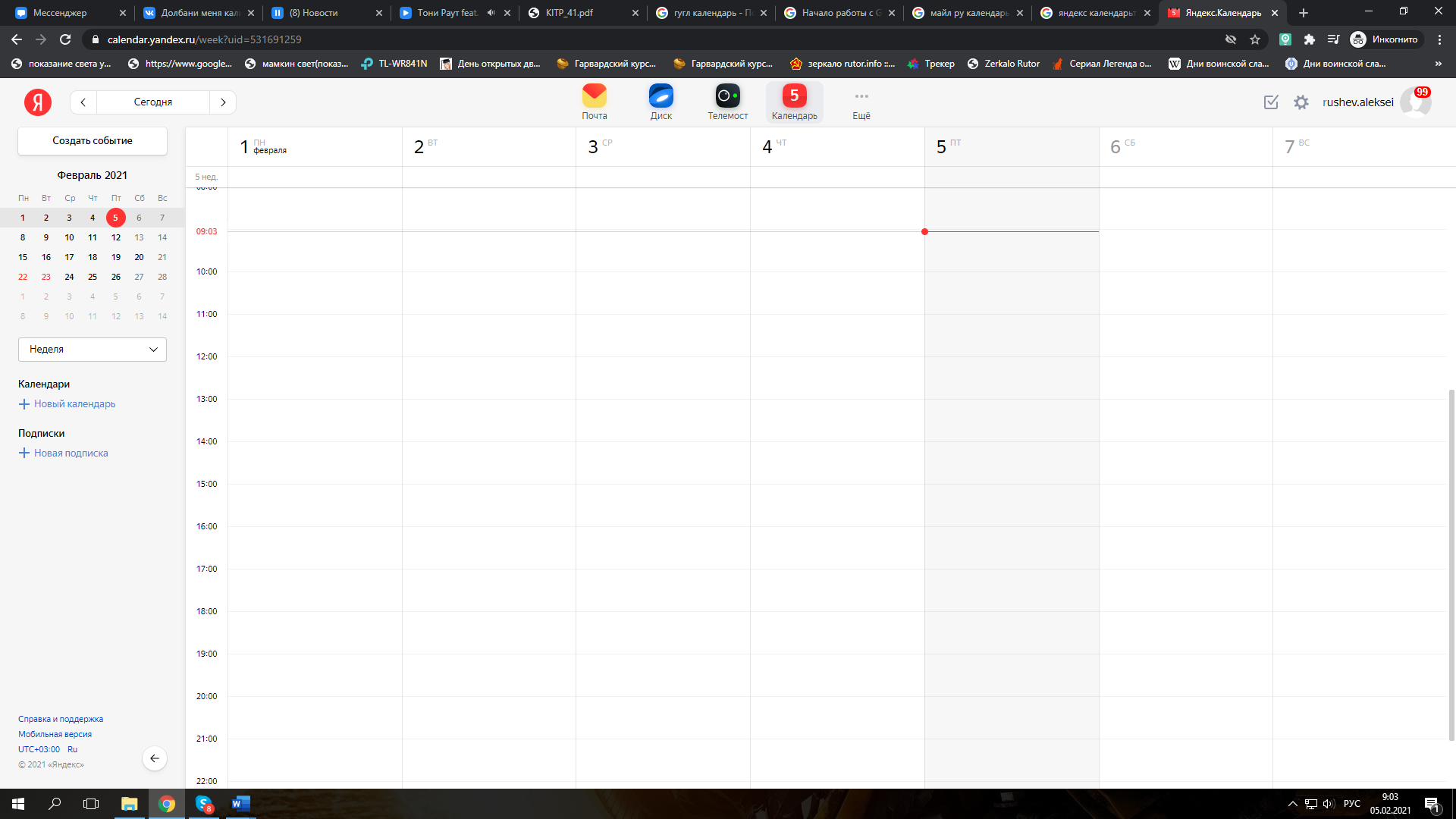


Рисунок 2 - Яндекс.Календарь

Первый календарь удовлетворяет многим функциональным возможностям. Второй же календарь аналогичен первому, но у него есть возможность уведомлять по смс о напоминании в календаре.

Рассмотренные аналоги обладают многим нужным требованиям для работы с календарем обычному пользователю. Оба приведенные в качестве примера. Аналоги, являются бесплатными, для них лишь нужно иметь почту того сервиса, чей календарь используешь. Исходя из вышеперечисленного, можно сделать вывод, что требуется разработка проекта, удовлетворяющему большинству требованиям обоих аналогов.

# 2 ПОСТАНОВКА ЗАДАЧИ

## 2.1 Определение требований и задач

Из описания предметной области, можно прийти к тому, что разрабатываемое приложение должно формировать сетку календаря, навигацию по нему и добавление события, удаления, а также изменения события.

Веб-приложение должно выполнять следующие задачи:

* Формирование сетки календаря;
* Переключение между месяцами;
* Переключение на текущий день в календаре;
* Добавление события в календарь;
* Удаление события из календаря;
* Изменение события в календаре.

## 2.2 Минимальные требования для работы программы

Для работы программы необходимо предоставить место на устройстве:

* 54МБ потребуется для готовой БД;
* 1ГБ потребуется для веб-приложения;
* ОС: Windows 7 и выше;
* Любой веб-браузер, поддерживающий HTML5.

ПО разрабатывается при помощи основных технологий:

* Microsoft SQL Server Management Studio 18;
* Microsoft Visual Studio Code;
* Node.js.

# 3 ПРОЕКТИРОВАНИЕ ПРИЛОЖЕНИЯ

## 3.1 ER-диаграмма базы данных

Для разрабатываемого приложения была разработана ER-модель

схемы данных, которая включает в себя 4 сущности предметной области.

Данная ER-диаграмма представлена ниже (Рисунок 3).

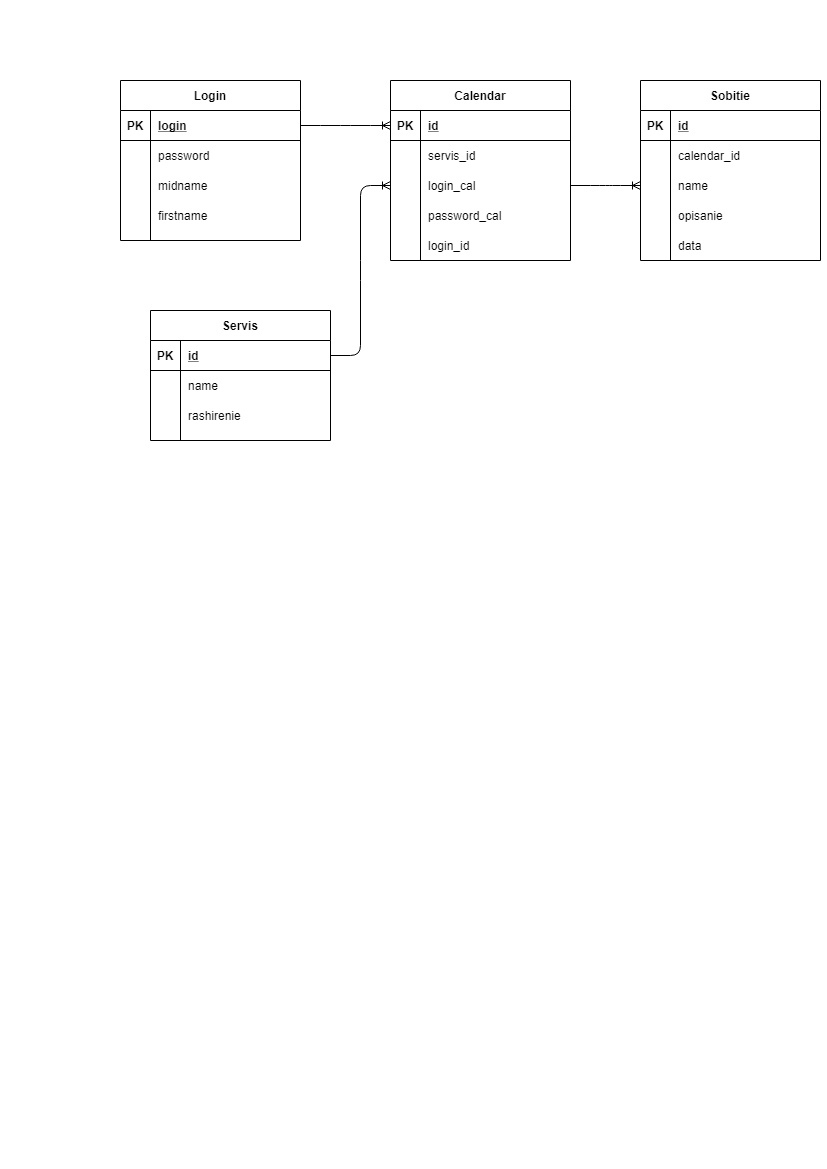


Рисунок 3 – ER-диаграмма приложения

## 3.2 Словарь данных

Словарь данных представлен в таблице 3.1. Данный словарь отражает сущности и поля из ER-диаграммы, в нём представлены все поля таблиц с указанием типов, обязательное заполнение и заметок.

Таблица 3.1 – Словарь данных

|  |  |  |  |
| --- | --- | --- | --- |
| **Словарь данных** | | | |
| **Login** | | | |
| **Ключ** | **Поле** | **Обязательное** | **Примечание** |
| Первичный | login | Да | Логин пользователя в приложении |
|  | password | Да | Пароль пользователя в приложении |
|  | midname | Нет | Фамилия пользователя |
|  | firstname | Нет | Имя пользователя |
| **Servis** |  |  |  |
| **Ключ** | **Поле** | **Обязательное** | **Примечание** |
| Первичный | id | Да | Уникальный идентификатор таблицы |
|  | name | Да | Наименование сервиса |
|  | rashirenie | Да | Расширение сервиса (@google.com) |

Продолжение таблицы 3.1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sobitie** |  |  |  |
| **Ключ** | **Поле** | **Обязательное** | **Примечание** |
| Первичный | id | Да | Уникальный идентификатор таблицы |
| Внешний | calendar\_id | Да | Внешний ключ к таблице Calendar |
|  | name | Да | Наименование события |
|  | opisanie | Нет | Описание события |
|  | data | Да | Дата события |
| **Calendar** |  |  |  |
| **Ключ** | **Поле** | **Обязательное** | **Примечание** |
| Первичный | id | Да | Уникальный идентификатор таблицы |
| Внешний | servis\_id | Да | Внешний ключ к таблице Servis |
|  | login\_cal | Да | Логин пользователя в календаре |
|  | password\_cal | Да | Пароль пользователя в календаре |
| Внешний | login\_id | Да | Внешний ключ к таблице Login |

## 3.3 Диаграмма прецедентов

Диаграмма прецедентов – диаграмма, на которой отражены отношения, существующие между актерами и прецедентами и являющиеся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

На рисунке 4. представлена диаграмма, на которой показаны все возможные функции, который реализован в приложении для пользователя. На диаграмме представлено описание, того, что может делать пользователь.

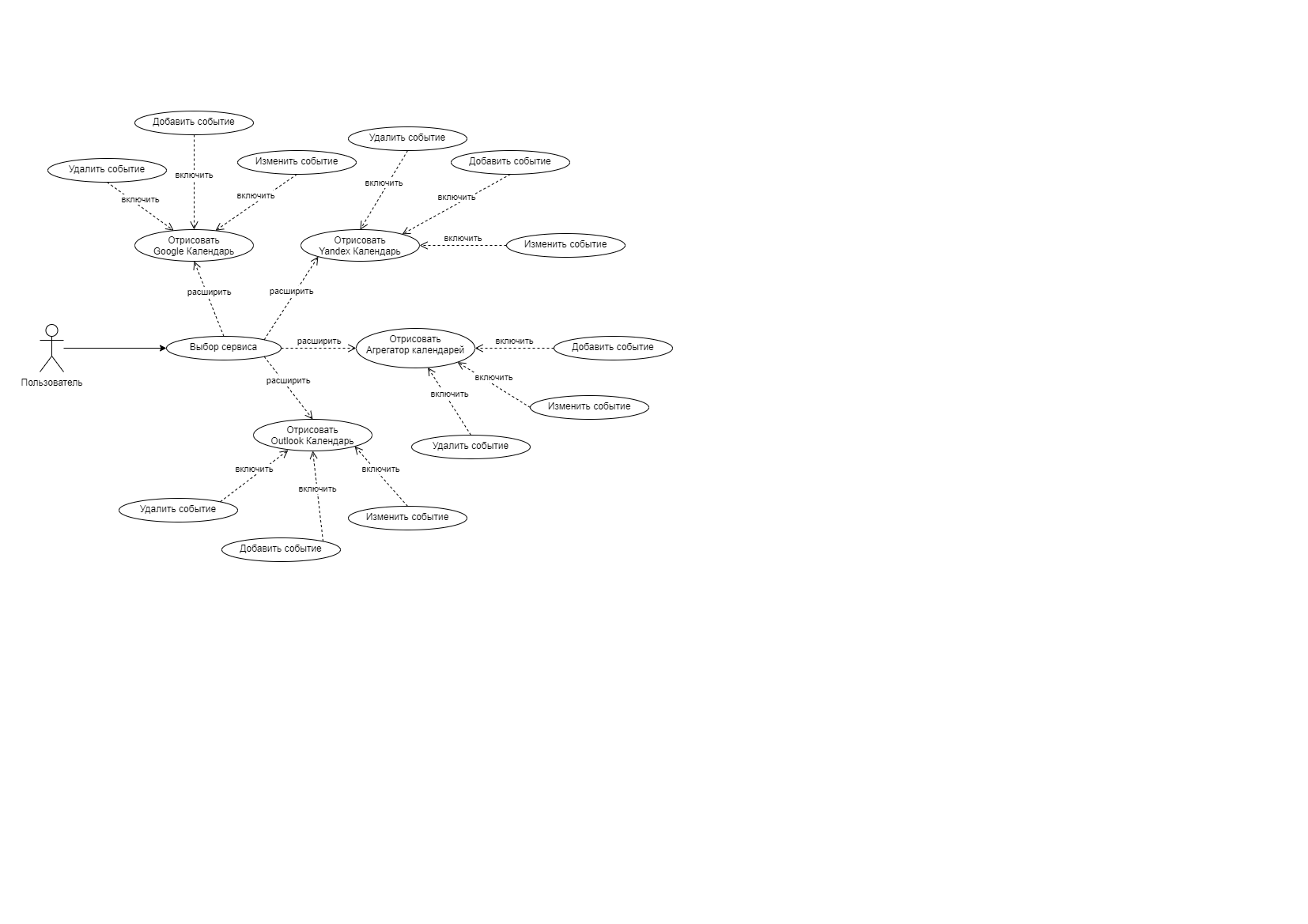


Рисунок 4 – Диаграмма прецедентов