Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизации обработки информации (АОИ)

ВЕБ-ПРИЛОЖЕНИЕ «ЗАМЕТКИ»

Отчёт по дисциплине  
Методология и технология программирования

|  |  |  |
| --- | --- | --- |
|  |  | Выпонили  Студент гр. 429-3    А. А. Бабец  (подпись)    А. А. Барстаков  (подпись)  « » 2021 г.  (дата) |
|  | (оценка) | Руководитель:  старший преподаватель АОИ  С. С. Пекарская  (подпись)  «07» июня 2021 г.  (дата) |

Томск 2021

# Оглавление

[1 Постановка задачи 3](#_Toc73529476)

[2 Анализ идеи веб-сайта 4](#_Toc73529477)

[3 Описание требуемых функциональных возможностей 5](#_Toc73529478)

[4 Модель предметный области 6](#_Toc73529479)

[5 Эскиз интерфейса 7](#_Toc73529480)

[6 Варианты использования 10](#_Toc73529481)

[7 Описание требуемых функциональных возможностей веб-сайта 15](#_Toc73529482)

[8 Используемые frontend-решения 16](#_Toc73529483)

[9 Используемые backend-решения 17](#_Toc73529484)

[10 Описание работы созданного веб-сайта 18](#_Toc73529485)

[11 Заключение 21](#_Toc73529486)

# Постановка задачи

Разработать веб-приложение заметок с возможностью создания, удаления заметок, присвоения им категорий, отбора заметок по времени или категории.

# Анализ идеи веб-сайта

Для начала проведём анализ идеи веб-сайта. Для этого воспользуемся Lean Model Canvas. Результат, изображённой на рисунке 1.

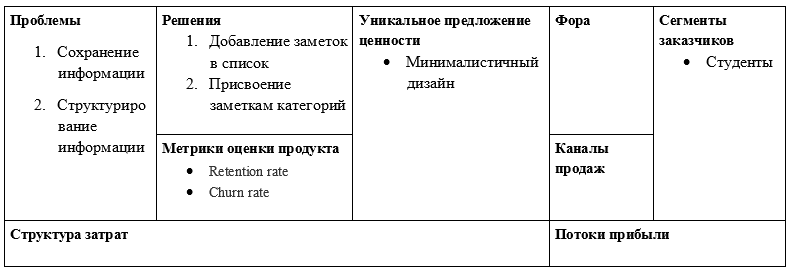


Рисунок 1 – Анализ идеи приложения

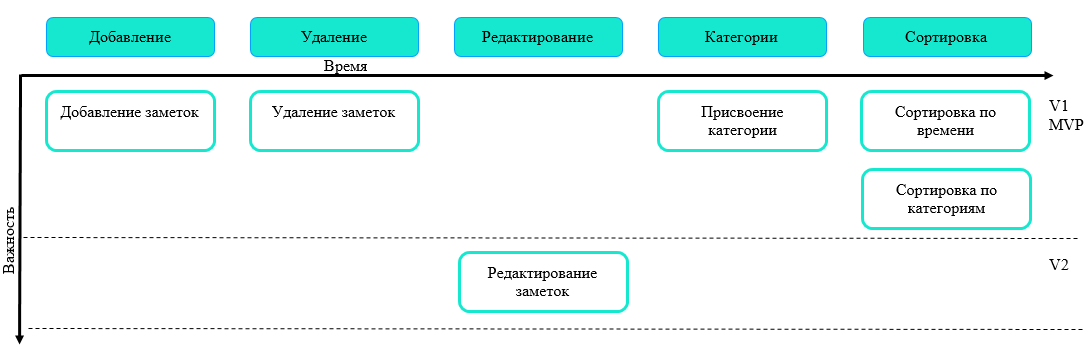
Для начала нужно решить, кому необходимо приложение с заметками, то есть определить сегменты заказчиков. Заметками пользуются все, для примера возьмём студентов. Проблемы, с которыми могут столкнуться наши заказчики, следующие:

* сохранение информации;
* структурирование информации.

Для того чтобы решить данные проблемы, в нашем веб приложении пользователи смогут создавать заметки, просматривать заметки отдельных категорий или за определённый день.

# Описание требуемых функциональных возможностей

Для описания требуемых функциональных возможностей веб-сайта и порядка их реализации составим User Story Map (USM), представленной на рисунке 2.

Рисунок 2 – User Story Map

На USM представлено две версии веб-сайта. Первая версия MVP – минимальный жизнеспособный продукт и вторая версия с добавлением новых функций. Мы разработаем первую версию. В неё входит следующие функции:

* добавление заметок;
* удаление заметок;
* присвоение категории;
* сортировка заметок по категориям;
* сортировка заметок по времени.

# Модель предметный области

Также составим модель предметной области, то есть создадим UML диаграмму, которые отражают различные взаимосвязи между сущностями предметной области, такими как объекты и подсистемы, а также описывают их внутреннюю структуру и типы отношений. Модель предметной области представлена на рисунке 3.

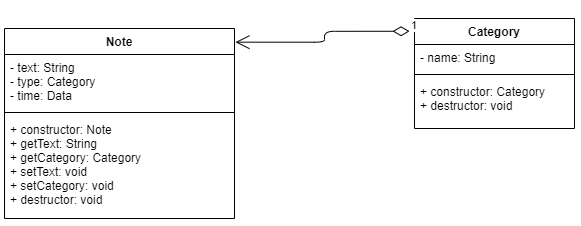


Рисунок 3 – UML-диаграмма предметной области

Основной класс – Note, объекты этого класса и есть сами заметки, у каждой из которых есть текст, категория и время создания. Категории вынесены в отдельный класс Category и имеют только название.

# Эскиз интерфейса

Переходим к разработке эскиза интерфейса. Интерфейс состоит из одной страницы, представленной на рисунке 4.

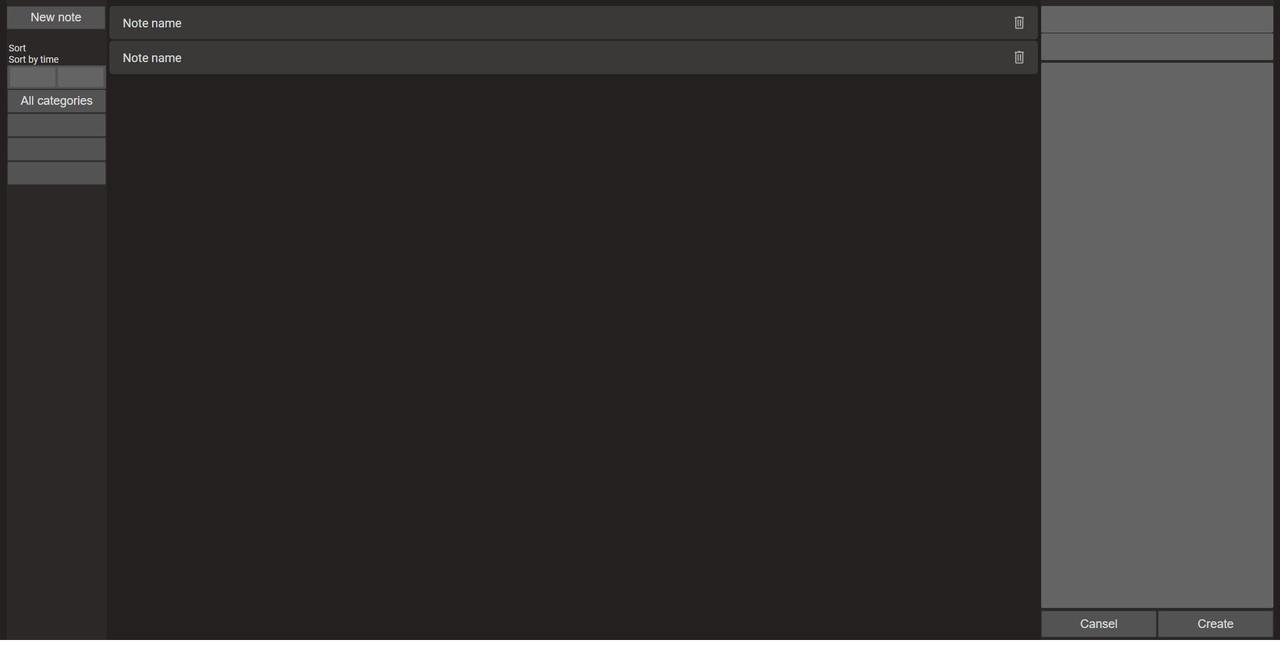


Рисунок 4 – Эскиз сайта

Слева страницы находится блок категорий и установки временных ограничений для отбора заметок. В центре находится список заметок, изменяемый при отборе по категориям или времени. Справа показывается содержание заметки в полях для ввода, при создании новой заметки эти поля очищаются и пользователь вводит данные.

# Варианты использования

Опишем варианты использования с помощью Use Cases.

1. Название: Создание заметки.

Цель/краткое описание: система создаёт новую заметку.

Начальное состояние: пользователем выбрана опция «Создать заметку».

Основной сценарий:

1. Система просит ввести данные: описание заметки, категорию.
2. Система создаёт заметку.
3. Название: Удаление заметки.

Цель/краткое описание: система удаляет заметку.

Начальное состояние: пользователем выбрана функция «Удалить заметку»

Основной сценарий:

1. Пользователь выбирает заметку.
2. Пользователь нажимает кнопку «Удалить заметку».
3. Система удаляет заметку.
4. Название: Выборка по времени.

Цель/краткое описание: система делает выборку заметок за определённый день.

Начальное состояние: пользователь находится на главной странице.

Основной сценарий:

1. Пользователь указывает день в специальном поле.
2. Пользователь нажимает кнопку «Применить».
3. Система выдает выборку заметок за указанный день.
4. Название: Выделение категории

Цель/краткое описание: система показывает заметки определённой категории.

Начальное состояние: пользователь находится на главной странице.

Основной сценарий:

1. Пользователь нажимает кнопку интересующей категории.
2. Система выдает список заметок по выбранной категории.

# Описание требуемых функциональных возможностей веб-сайта

Опишем функциональные возможности нашего веб-сайта. Для того, чтобы создать заметку пользователю нужно ввести данные в соответствующие поля справа и нажать на кнопку «Create», для отмены (очистки полей) – «Cancel». Новые категории создаются автоматически, если при создании заметки система не выявит совпадения выбранной категории с существующими. Для просмотра заметки нужно нажать на её название в списке. Для выборки заметок по категории нужно нажать на кнопку соответствующей категории. Для выборки за определённый день нужно выбрать день в специальном поле слева и нажать кнопку «Применить». Для возврата к общему списку заметок используется кнопка «Отменить». Для удаления заметки нужно нажать на кнопку в виде корзины на соответствующей заметке в списке.

# Используемые frontend-решения

Для реализации фронтенда сайта мы использовали следующий инструментарий:

* HTML как язык разметки;
* CSS как язык описания внешнего вида документа;
* JavaScript как язык программирования, в рамках фреймворка Vue.js для использования библиотеки axios для формирования запросов со стороны клиента к серверу.

Сайт содержит следующие функции:

* Создание заметки – используется для создания новой заметки, у которой есть текст, название, категория, дата создания.
* Отображение заметки – при выборе заметки, все её данные отображаются в полях справа.
* Отображение списка заметок – можно выбрать, как будут отображаться заметки на экране: по выбранной категории или по выбранной дате.
* Удаление заметки – используется для удаления выбранной заметки.

# Используемые backend-решения

Для реализации бэкенда сайта мы использовали следующий инструментарий:

* Python как основной язык программирования;
* Django как фреймворк для работы с бэкендом;
* Django Rest API для обработки запросов от клиента;
* SQLite как СУБД для работы с базой данных.

Мы использовали Django так как он содержит огромное количество функциональности для решения большинства задач веб-разработки, а также он безопасен из коробки и включает механизмы предотвращения распространенных атак вроде SQL-инъекций (XSS) и подделки межсайтовых запросов (CSRF).

Так как сам фреймворк является синхронным, мы использовали Rest API для асинхронной работы.

# Описание работы созданного веб-сайта

При входе на сайт пользователь попадает в приложение (Рисунок 5).

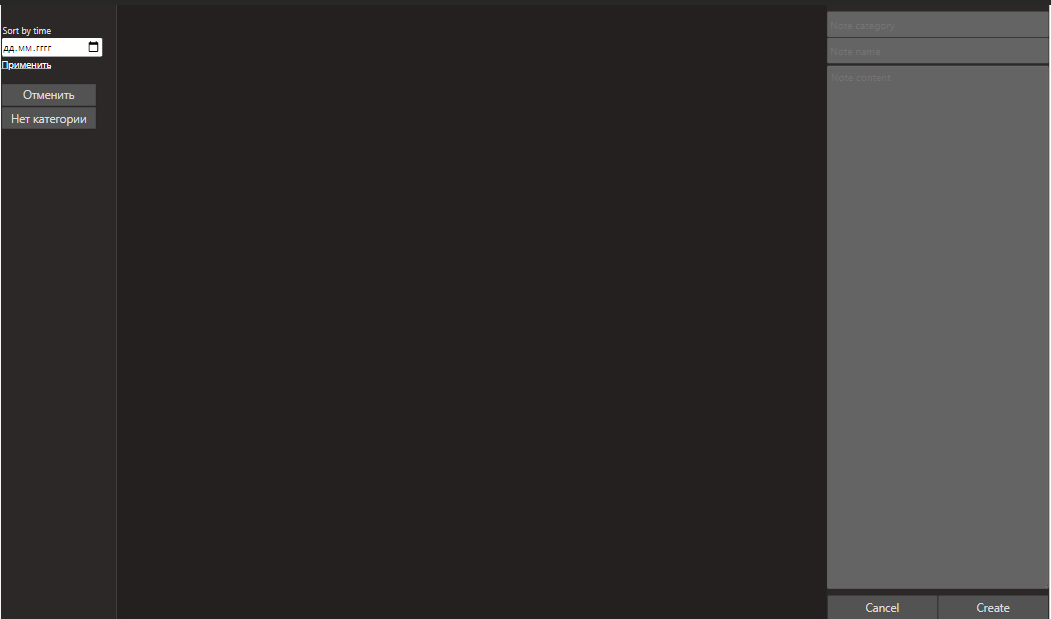


Рисунок 5 – Основная страница

Для создания заметки нужно заполнить поля справа и нажать кнопку «Create» (рисунок 6).

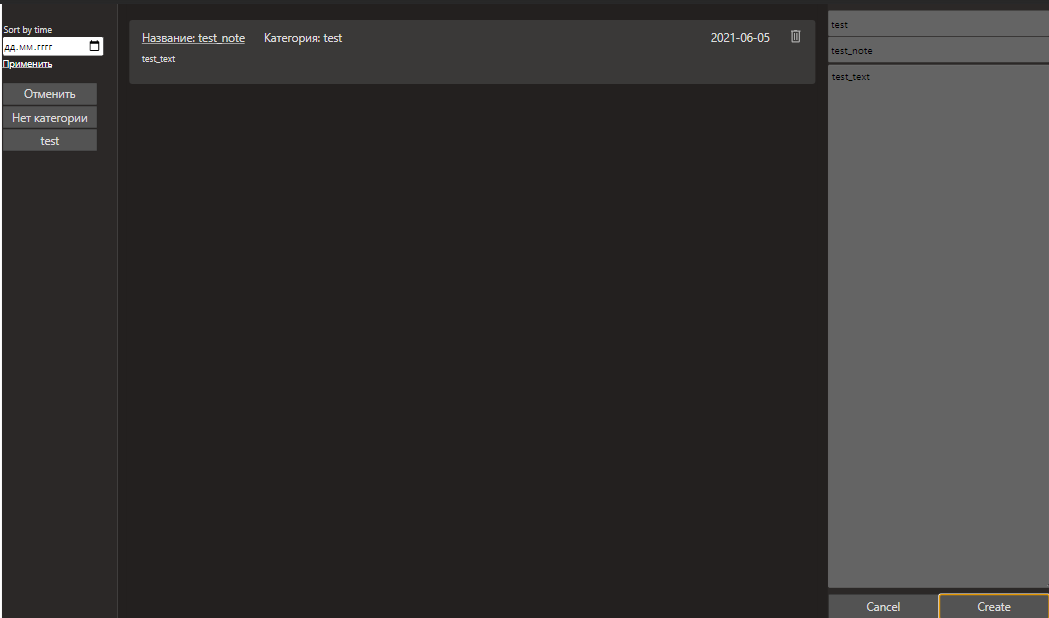


Рисунок 6 – Создание заметки

Для выборки по категории нужно нажать на кнопку с названием категории на панели слева (Рисунки 7, 8).

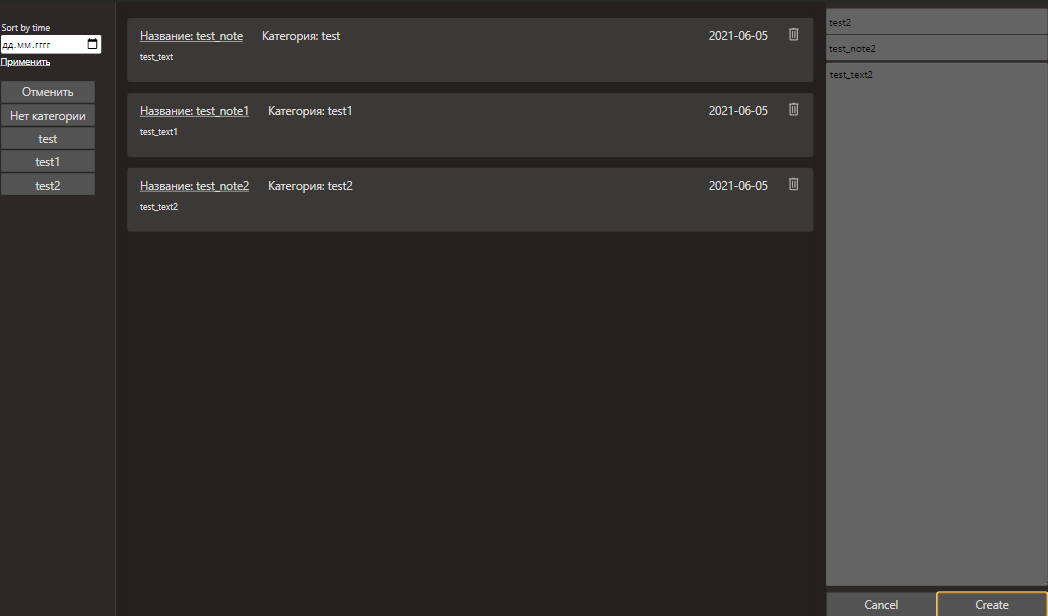


Рисунок 7 – Все заметки

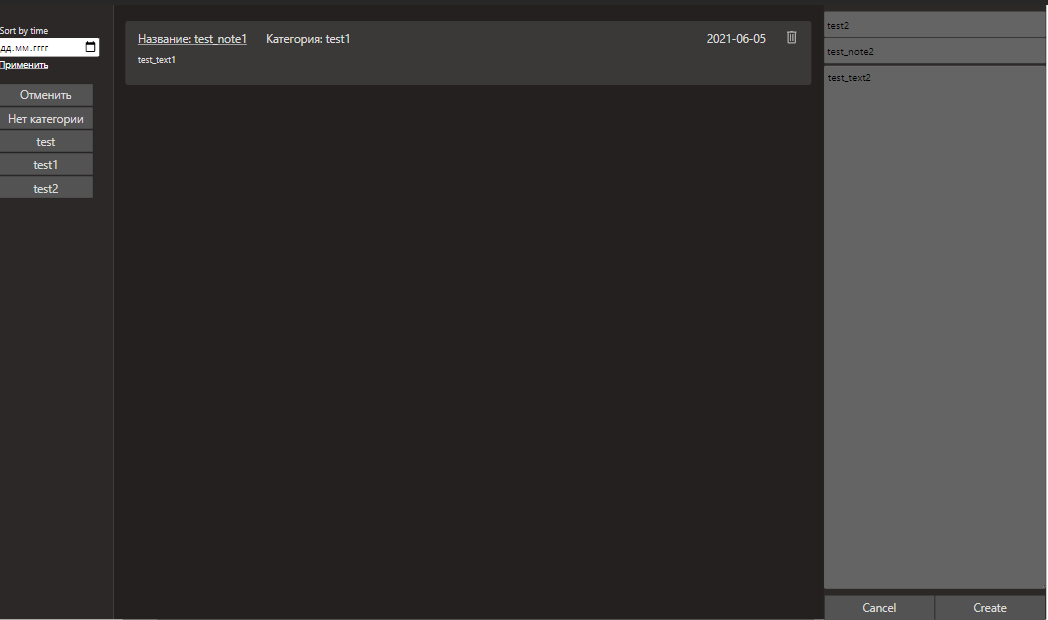


Рисунок 8 – Выборка по категории test1

Для отмены выборки жмём кнопку «Отменить», находящуюся слева (Рисунок 9).

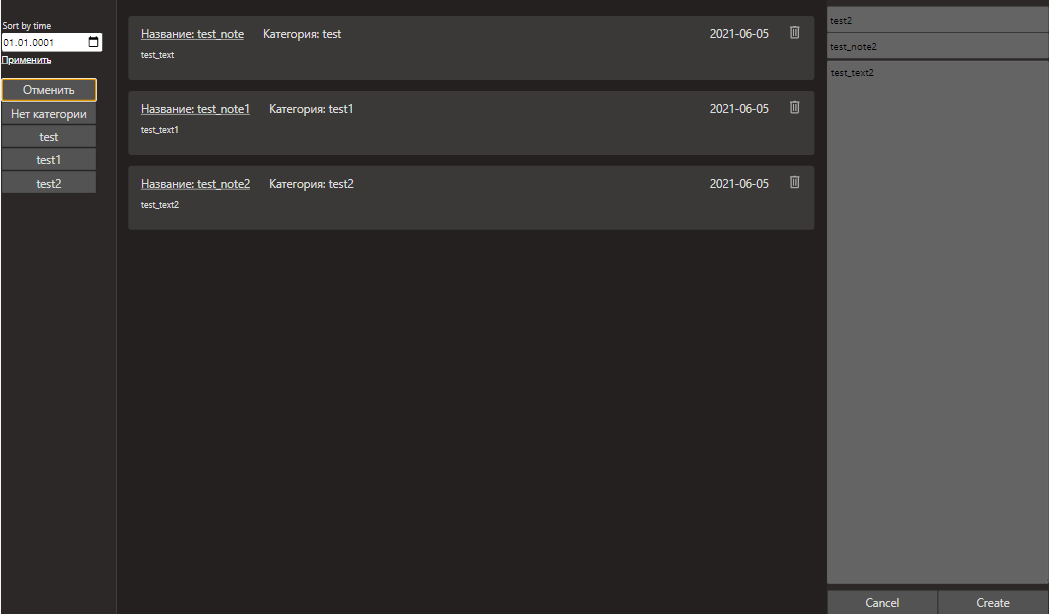


Рисунок 9 – Отмена выборки

Также выборку можно сделать за определённый день для этого вводим дату в специальное поле слева и жмём кнопку «Применить» (Рисунок 10).

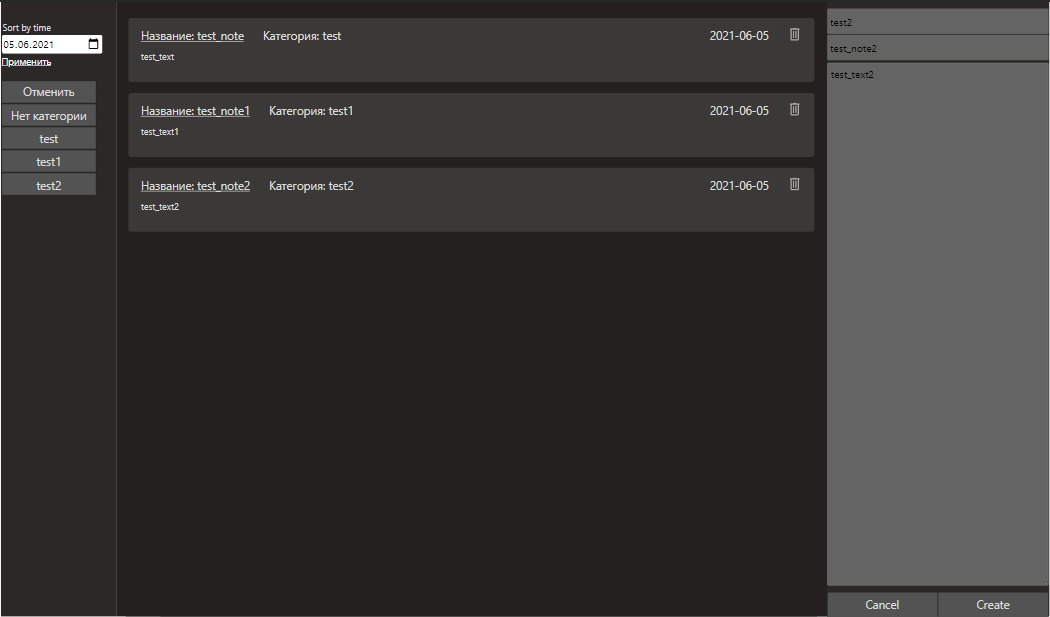


Рисунок 10 – Выборка по дате

Для удаления заметки жмём на кнопку в виде корзины на той заметке, которую хотим удалить (Рисунок 11).

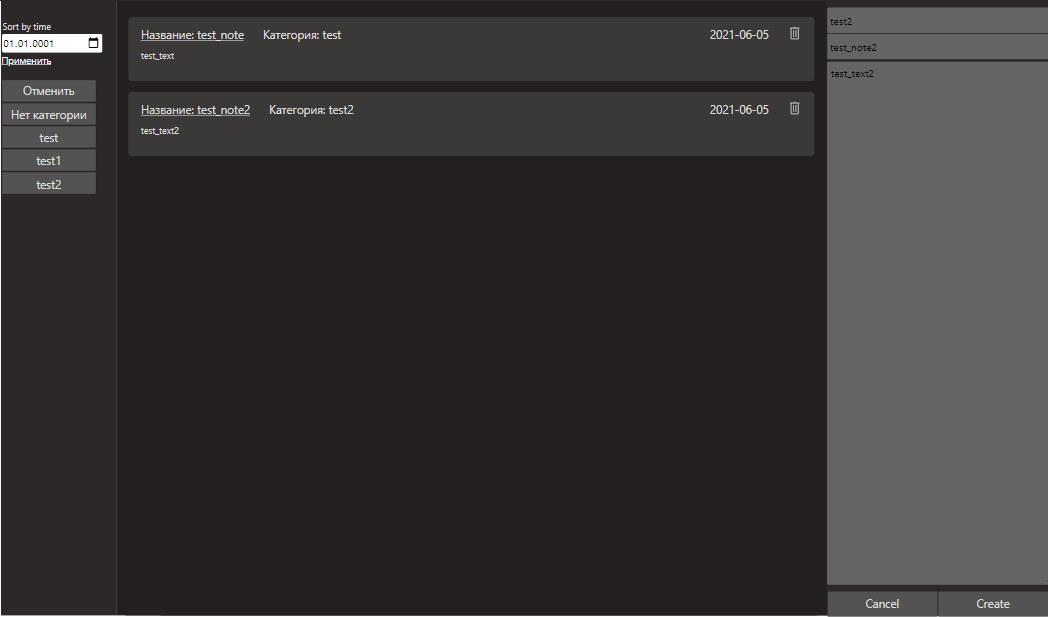


Рисунок 11 – Удаление заметки

# Заключение

В результате проекта было реализовано веб-приложение «Заметки».

В ходе выполнения проекта были изучены основы работы с фронтендом, а именно с языком разметки HTML, языком стилей CSS и языком программирования Javascript, фреймворком Vue.js и его библиотекой axios; с бэкендом, а именно с языком программирования Python, фреймворком Django, Django Rest API и базой данных SQLite. Был проведён анализ идеи, а также разработаны варианты использования и модель предметной области.