


**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**


Факультет компьютерных наук  
Образовательная программа «Программная инженерия»

**СОГЛАСОВАНО**

Руководитель, старший преподаватель  
департамента программной инженерии

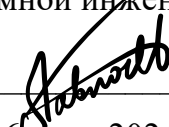
  
\_\_\_\_\_  
Н.А. Павлов  
«16» мая 2024 г.

Соруководитель, руководитель  
разработкой АО «СКБ Контур»

  
\_\_\_\_\_  
А.А. Кучук  
«16» мая 2024 г.

**УТВЕРЖДАЮ**

Академический руководитель  
образовательной программы  
«Программная инженерия», старший  
преподаватель департамента  
программной инженерии

  
\_\_\_\_\_  
Н.А. Павлов  
«16» мая 2024 г.

**Выпускная квалификационная работа  
(проектно-исследовательская)**


на тему:

**Сервис планирования подарков «Wishlist»**

по направлению подготовки 09.03.04 «Программная инженерия»

**ВЫПОЛНИЛ**

студент группы БПИ203  
образовательной программы  
09.03.04 «Программная инженерия»

  
\_\_\_\_\_  
С. А. Иванова  
«16» мая 2024 г.

**Москва – 2024**

## Реферат

Данная работа представляет собой сервис, предназначенный для планирования подарков.

Вишлисты – способ коммуникации между людьми, где один человек собирает идеи для подарков себе в один список, а его близкие могут не только выбирать, что подарить, но еще и развивать и интерпретировать в своем направлении. Кроме того, вишлисты используются как подборки идей подарков разным людям на различные события, благодаря чему можно найти подарки для тех, кто не составляет вишлисты.

Работа направлена на реализацию системы, где пользователи могут публиковать свои вишлисты для друзей, публичные вишлисты-подборки, списки подарков для себя, а также «резервировать» подарки друзей. Сервис позволяет планировать подарки для друзей и для себя.

**Работа содержит:** 40 страниц, 3 главы, 26 иллюстраций и 10 источников.

**Ключевые слова:** вишлисты, веб-приложение, клиент-серверная архитектура.

## **Abstract**

Current work is a service which is designed for gift planning.

Wishlists are the way of communication between people, where one person picks items that he wants to get or gift ideas, whereas his friends and relatives can not only choose what to give, but also develop and interpret in their own way. Moreover, wishlists are used as compilations of present ideas for different people and different occasions, that helps a lot in thinking of ideas for gifts for people without wishlists.

The aim of this work is at implementing the system, where people can publish their wishlists for friends, public wishlists, listings of gifts for themselves and also “reserve” presents of friends. The service allows plan gifts for friends and for yourself.

**The work contains:** 40 pages, 3 chapters, 26 figures and 10 sources.

**Keywords:** wishlists, web-application, client-server architecture.

## Основные определения, термины и сокращения

1. **API** (Application Programming Interface) – интерфейс программы, описание как взаимодействовать с программой.
2. **CAPTCHA** (Completely Automated Public Turing test to tell Computers and Humans Apart) – проверка, нацеленная на определение того, является ли пользователь человеком или компьютером.
3. **HTML** (HyperText Markup Language) – язык гипертекстовой разметки, предназначенный для представления веб-страниц.
4. **HTTP** (HyperText Transfer Protocol) – протокол передачи данных.
5. **JWT** (Json Web Token) – стандарт для защищенной передачи информации в формате JSON.
6. **MVC** (Model View Controller) – паттерн проектирования, декомпозирующий систему на три слоя: отображение данных, бизнес-логика и взаимодействие с пользователем.
7. **ORM** (Object Relation Mapping) – технология, связывающая сущности базы данных с сущностями объектно-ориентированного языка программирования.
8. **REST API** – архитектурный стиль проектирования компонентов системы, при котором серверная часть не хранит состояния запросов.
9. **SPA** (Single Page Application) – подход к проектированию клиентских приложений, в котором веб-сайт состоит из небольшого HTML/XML файла и программного файла, который ответственен за изменение интерфейса и логику взаимодействия с пользователем.
10. **UML (Unified Modeling Language)** – язык визуального моделирования.
11. **URL** (Uniform Resource Location) – адрес, идентифицирующий расположение ресурса в Интернете.
12. **Вишлист** – список подарков, которые бы хотел получить человек.
13. **Клиент-серверная архитектура** – вид архитектуры, состоящая из слоёв «сервер» и «клиент», где вся логика системы происходит в серверной части, а взаимодействие с пользователем – в клиентской части.
14. **Куки** (от англ. cookie) – информация, хранящаяся в клиентском приложении и передаваемая между клиентом и сервером.
15. **СУБД** – система управления базами данных.

## Содержание

Реферат.....	2
Abstract.....	3
Основные определения, термины и сокращения.....	4
Содержание.....	5
Введение.....	6
1. Глава 1. Предметная область и существующие решения.....	7
1.1. Описание предметной области.....	7
1.2. Описание существующих решений.....	7
1.2.1. Описание приложения «Oh My Wishes».....	7
1.2.2. Описание веб-сервиса «Подаркус».....	9
1.2.3. Описание веб-сервиса «MyWishBoard».....	10
1.2.4. Описание приложения «GiftBuster».....	11
1.3. Описание разрабатываемого решения.....	11
1.4. Анализ существующих решений.....	13
Выводы по главе.....	14
2. Глава 2. Проектирование веб-сервиса.....	16
2.1. Пользовательские сценарии.....	16
2.2. Архитектура приложения.....	20
2.2.1. Схема базы данных.....	21
2.2.2. Архитектура серверного приложения.....	22
2.2.3. Архитектура клиентского приложения.....	23
2.3. Выбор методов и средств реализации.....	25
2.3.1. База данных.....	25
2.3.2. Серверное приложение.....	25
2.3.3. Клиентское приложение.....	26
Выводы по главе.....	26
3. Глава 3. Программная реализация веб-сервиса.....	27
3.1. Система учетных записей.....	27
3.2. Система друзей.....	28
3.3. Система вишлистов и подарков.....	29
3.4. Система резервирования подарков.....	33
3.5. Система соавторства.....	36
3.6. Система уведомлений.....	37
Выводы по главе.....	38
Заключение.....	39
Список используемой литературы.....	40

## **Введение**

Люди часто сталкиваются с ситуациями, когда не знают, что подарить близким. Даже как бы хорошо они не знали своих друзей и любимых, процесс выбора подарка может быть довольно затруднительным.

С другой стороны, человек, которому предназначается подарок, будет рад оказанному вниманию, но если подарок не вписывается в образ жизни и желания человека, то эмоции могут немного уйти в негативную сторону от непонимания как использовать вещь в повседневной жизни или даже чувства вины из-за того, что она не используется.

Вишлисты, списки подарков, которые бы хотел получить человек, могут помочь обеим сторонам: получатель подарка сможет рассказать о своих желаниях без каких-либо мучений, благодаря чему вероятность получить нежелательный подарок становится низкой, а друзья и близкие получают лист вариантов и направление для идей, куда можно двигаться.

Сервис с вишлистами позволит не только делиться списками, но также и имеет функцию «резервирования» подарка, которая предназначена для того, чтобы друзья пользователя не повторялись с подарками. Система запомнит «резервирование» и отобразит другим, кроме самого пользователя, кто собирается подарить ту или иную вещь.

Целью данной работы является разработка сервиса планирования подарков, основанного на концепции вишлистов.

Для достижения цели поставлены следующие задачи:

1. Анализ существующих решений;
2. Формирование требований к сервису;
3. Выбор методов и средств разработки;
4. Проектирование архитектуры;
5. Программная реализация веб-сервиса;
6. Тестирование веб-сервиса;
7. Разработка технической документации;

## **1. Глава 1. Предметная область и существующие решения**

### **1.1. Описание предметной области**

Ежегодно люди празднуют немало событий: от календарных праздников, как Новый год, самый распространенный и популярный праздник в России, до личных празднований, куда входят дни рождения и другие знаменательные лично для человека события. Все эти праздники подразумевают дарение подарков как знак внимания и теплых чувств. Однако сделать этот процесс идеальным – выбрать самый подходящий подарок, чтобы принести только позитивные эмоции человеку – может быть очень трудным.

Одним из способов выбрать нужный и соответствующий личности получателя подарок являются вишлисты. Слово «вишлист» происходит от английского слова «wishlist», что в переводе означает список желаний. Фактически, вишлист – это список подарков или идей для них, которые бы хотел получить человек, составивший его.

Такой способ коммуникации между людьми во многом способствует решению проблемы выбора подарка для человека.

С точки зрения дарителя, вишлисты дают не только список опций, которые можно подарить, но и направление, куда можно двигаться и куда развивать идеи. Безусловно, можно выбрать подарок из списка и не прогадать – получатель точно будет рад, так как он сам это выбрал. Но помимо этого, с помощью вишлистов можно лучше узнать человека и его предпочтения, благодаря чему можно не ограничиваться рамками выбранного подарка и добавить что-то личное от себя. Например, привязать к подарку какие-то общие воспоминания или привнести что-то смешное.

Что касается получателя подарка, то он, конечно, хотел бы получить то, чем он бы пользовался в своей жизни или что доставляло бы позитивные эмоции. Однако, говорить прямо о своих желаниях не всегда может быть легко. В этом случае вишлисты служат как способ коммуникации, в котором человек просто скидывает ссылку на свой вишлист, сохраняя интригу и не обязывая других, ведь это просто список, от которого можно отталкиваться.

### **1.2. Описание существующих решений**

В последнее время вишлисты набирают популярность, и все больше и больше людей пользуются ими. Об этом говорит появление ряда решений, основанных на вишлистах: приложение «Oh My Wishes» [1], веб-сервис «Подаркус» [2], веб-сервис «MyWishBoard» [3], приложение «GiftBuster» [4].

#### **1.2.1. Описание приложения «Oh My Wishes»**

Приложение «Oh My Wishes» представляет собой мобильное приложение для создания вишлистов, но также имеет веб-сайт с подборками идей для подарков.

Веб-сервис имеет систему учетных записей и предоставляет возможность регистрации и авторизации с помощью электронной почты, учетной записи социальной сети «ВКонтакте» или учетной записи Apple ID.

В мобильном приложении пользователь может создавать отдельные вишлисты, но также у него есть список «Мои желания», в котором отображаются подарки из всех вишлистов. Чтобы создать вишлист, необходимо нажать на соответствующую кнопку и ввести название вишлиста, которое можно в последствии менять. При необходимости вишлист также можно удалить. Вишлисты доступны только друзьям.

У каждого пользователя также есть список «Секретные желания», куда пользователь может добавлять подарки, которые сможет видеть только он.

Добавление подарка в вишлист происходит при нажатии кнопки и заполнении всплывающей формы. Форма состоит из следующих полей: изображение, название, ссылка на подарок, описание, цена с указанием валюты, а также является ли подарок «Секретным желанием» и в какие вишлисты нужно его добавить. Обязательным полем для добавления карточки подарка является название.

При вводе ссылки в соответствующее поле приложение предлагает заполнить поля автоматически. После нажатия кнопки «Заполнить» сервис по возможности заполняет поля формы. Приложение не гарантирует корректное заполнение полей, поэтому оно предоставляет возможность отменить автозаполнение. Например, при вставке ссылки с маркетплейса «Яндекс.Маркет» приложение подтягивает иконку компании «Яндекс» и в поле названия подарка вставляет текст «Яндекс». Скорее всего, приложение не распознает сайты с динамическим содержанием и подтягивает информацию из тегов «meta».

Все вишлисты (кроме списка «Секретные желания») доступны друзьям пользователя. Помимо этого, сервис предоставляет возможность поделиться вишлистом по ссылке. Если открыть ссылку в веб-сайте, то пользователю будут доступны вишлисты с карточками подарка, однако посмотреть более подробную информацию или зарезервировать подарок без приложения он не сможет.

Приложение предоставляет функцию резервирования друзьям пользователя. При резервировании подарок добавляется в список «Хочу подарить», где отображаются все зарезервированные пользователем подарки.

Веб-сервис предоставляет бизнесам возможность продвижения. Для этого нужно зарегистрироваться как бизнес-аккаунт и создавать с этой учетной записи подборки своих товаров



как идеи для подарков. Для продвижения есть три платных варианта: размещение карточки идеи, размещение карточки бренда и размещение бренда на баннере.

Раздел «Идеи» доступен и в мобильном приложении, и на веб-сайте. Он состоит из четырех частей:

1. Баннер, который продвигает бизнес-аккаунты;
2. Раздел «Бренды», отображающий список бизнес-аккаунтов;
3. Раздел «Редакторские подборки», где можно найти подборку на какую-то тему от редакции приложения.
4. Раздел «Идеи подарков» с карточками подарков, которые можно добавить в свои вишлисты.

Таким образом, публиковать подборки идей для подарков могут только бизнес-аккаунты или редакция веб-сервиса.

### **1.2.2. Описание веб-сервиса «Подаркус»**

Веб-сервис «Подаркус» представляет собой веб-сайт для создания вишлистов.

Веб-сервис предоставляет возможность регистрации и авторизации с помощью электронной почты, учетной записью Google, социальной сети «Одноклассники», социальной сети «ВКонтакте» или учетной записью «Яндекс».

Для создания вишлиста пользователь должен перейти в раздел «Вишлисты» и нажать кнопку «Новый список». Перед пользователем появится форма со следующими полями: название списка, эмодзи, дата списка. По умолчанию вишлист доступен только друзьям пользователей, но в форме можно отметить, является ли вишлист приватный, т.е. доступен только самому пользователю и соавторам. В форме можно также добавить соавторов вишлиста – друзей пользователя, которые смогут добавлять подарки в вишлист. При необходимости вишлист можно редактировать или удалить.

Чтобы добавить подарок, нужно нажать на кнопку «Добавить подарок». Откроется форма со следующими полями: ссылка на подарок, название подарка, цена, изображение, а также вишлисты, куда добавить подарок. Обязательными полями для добавления подарка являются название подарка, ссылка на подарок и вишлист, куда его добавить. Так как подарок не обязательно должен быть товаром, наличие ссылки в карточке подарка может быть неудобным, хоть и решается простым заполнением любыми символами.

Сервис также предоставляет автоматическое заполнение полей. Он корректно заполняет информацию с маркетплейса «Яндекс.Маркет». Однако с других платформ электронной коммерции, как «Wildberries» или «Ozon», сервис не может заполнить информацию.

Если вишлист доступен друзьям, то им можно поделиться по ссылке. При переходе по ссылке пользователь сможет увидеть подарки в вишлисте.

Резервирование подарков доступно только авторизованным друзьям пользователя. При резервировании подарка можно отметить друзей, которые совместно дарят выбранный подарок.

У сервиса также есть календарь, где отмечены дни рождения друзей и который позволяет создавать свои события.

«Подаркус» предоставляет возможность регистрироваться бизнес-аккаунтам. Они делятся на два типа: «Профессиональный», позволяющий создавать публичные подборки, и «Коммерческий» для продвижения своих товаров и услуг.

Веб-сервис имеет раздел подборок и идей подарков, который состоит из 4 частей. Во-первых, у него есть лента идей подарков, которые можно добавлять в свои вишлисты. Далее идет раздел с подборками на различные темы. Затем есть раздел «Каталог подарков», в котором находятся товары бизнес-аккаунтов, и раздел «Магазины» с ними самими.

Кроме того, веб-сервис предлагает платную функцию создания открыток. Для этого пользователь должен выбрать иллюстрацию, написать пару строк и прикрепить свой вишлист. Таким образом, получится красивое приглашение на празднование события.

### **1.2.3. Описание веб-сервиса «MyWishBoard»**

Веб-сервис «MyWishBoard» представляет собой веб-сайт для создания коллекций подарков.

Веб-сервис предоставляет возможность регистрации и авторизации как с помощью электронной почты и пароля, так и предлагает большое разнообразие опций использования учетных записей других сервисов: «Telegram», «ВКонтакте», «Яндекс», «Apple ID», «Google» и еще 7 сервисов.

Подарки в веб-сервисе не привязаны к конкретным вишлистам. Пользователь может просто добавлять подарки в раздел «Желания». При необходимости можно создать коллекцию (то же самое, что и вишлист) и добавить желания туда.

Для создания подарка необходимо ввести название подарка в строку ввода и нажать на кнопку «Добавить». Появится форма со следующими полями: название, фотография или цветной фон, описание, цена с валютой, ссылка на подарок, степень желания от 0 до 9, теги. Кроме того, можно выбрать коллекцию, куда добавить подарок, и доступ: «Видно всем», «Друзьям» и «Только мне». Также сервис ищет подарки с похожим названием, чтобы вместо заполнения всей формы можно было просто одним кликом создать подарок.

На странице подарка есть кнопки «Найти подарок и купить», которые перенаправляют на страницы маркетплейсов, с введенным в строку поиска названием подарка. Помимо этого, сервис

предоставляет возможность добавления подарка в коллекцию, в список зарезервированных, в список исполненных или в список понравившихся. Также на странице подарка есть возможность оставлять комментарии.

Для создания коллекции нужно нажать кнопку «Создать новую коллекцию». Появится форма с полями названия коллекции, типом коллекции и доступом.

Веб-сервис имеет раздел «Идеи», где публикуются карточки подарков пользователей с настройкой доступа «Видно всем».

#### **1.2.4. Описание приложения «GiftBuster»**

Приложение «GiftBuster» представляет собой приложение для создания вишлистов.

Приложение позволяет авторизоваться или зарегистрироваться с помощью электронной почты или с помощью учетной записи сервисов: «Facebook», «Google», «X» или «Apple ID».

Чтобы создать вишлист, необходимо нажать на соответствующую кнопку. Перед пользователем отобразится форма с полями названия вишлиста и доступа к вишлисту («Приватный» или «Для друзей»). Пользователь имеет возможность редактировать или удалять вишлист, а также делиться им.

Для добавления подарка в вишлист можно добавить подарок по ссылке или заполнить информацию вручную. Автозаполнение приложения корректно заполняет фотографию и цену подарка. Пользователь может также вручную ввести информацию в соответствующие поля: название, описание, цена с валютой, вишлист, куда добавить подарок. Форма подарка также позволяет добавлять несколько фотографий подарка и поднять подарок вверх списка.

Резервирование подарков доступно друзьям пользователя, а зарезервированные подарки можно найти отдельно в соответствующем списке.

Приложение также предлагает промокоды для подарков из магазинов.

У сервиса есть напоминания дней рождений друзей.

В приложении есть раздел «Идеи» с рекламными баннерами магазинов и с лентой товаров, которые можно добавить в свои вишлисты.

### **1.3. Описание разрабатываемого решения**

Данный раздел представляет собой функциональные требования к разрабатываемому сервису.

Разрабатываемый сервис представляет собой приложение для планирования подарков и создания вишлистов.

Сервис должен иметь систему учетных записей и предоставлять возможность регистрации, авторизации с помощью адреса электронной почты или логина, управления своей учетной записью. Также сервис должен иметь систему друзей с возможностями добавления и удаления друзей.

Система должна предоставлять возможность создания вишлистов, изменения названия вишлиста и управление списком подарков вишлиста, включая добавление и удаление карточек подарков.

У вишлистов должны быть настройка доступа, которая может принимать следующие значения: «Публичный», «Для друзей», «Личный», «Для ограниченного круга лиц» (далее «Кастомный»).

Вишлисты с настройкой доступа «Публичный» – подборки подарков, которые могут создавать пользователи. С помощью публичных вишлистов пользователи смогут делиться идеями подарков на разные темы. Также это хорошая возможность для бизнесов продвигать свою продукцию, которая также может выступать в качестве подарков.

Вишлисты с настройкой доступа «Личный» могут выступать в качестве инструмента для планирования личных подарков и будущих покупок.

Настройка доступа «Кастомный» предполагает гибкую настройку доступа для друзей, чтобы пользователь имел возможность создавать отдельные вишлисты для отдельных друзей.

Карточки представляют собой объект со следующими полями: название, URL продукта, описание, цена и валюта, изображение подарка. Пользователь должен иметь возможность добавлять подарки в свои вишлисты, редактировать любые поля подарка и удалять карточки.

Система должна иметь функцию автозаполнения информации подарка по ссылке, указанной в поле «URL продукта». Система должна по возможности заполнять такие данные подарка как название, изображение, цена и валюта. Функция направлена на улучшение пользовательского опыта и упрощения работы пользователя.

Система должна иметь возможность добавления и удаления соавторов. Добавив в свой вишлист соавтора, пользователь предоставляет возможность другому добавлять, редактировать и удалять карточки подарков. Таким образом, пользователи смогут вместе составлять вишлисты на совместные празднования как свадьба или новоселье, а в случае с публичными подборками и бизнесами – реализовывать коллаборации.

Система должна позволять друзьям пользователя резервировать подарки и предоставлять им список зарезервированных подарков. Если пользователь разрешил доступ к вишлисту по ссылке, то подарки вишлиста должны быть доступны для резервирования как авторизованным, так и неавторизованным пользователям. С помощью резервирования пользователи смогут сообщить

другим о своем желании подарить конкретный подарок, всего лишь кликнув по кнопке, тем самым избежав повторных подарков.

Наличие возможности резервирования неавторизованным пользователем направлено на упрощение резервирования. Таким образом, человеку требуется только вписать свое имя. Это способствует тому, что у пользователя не пропадет желание зарезервировать подарок.

Кроме того, сервис должен предусматривать вариант, когда неавторизованный пользователь решает зарегистрироваться или аутентифицироваться. Система при авторизации пользователя должна перенести гостевые резервирования на учетную запись пользователя.

У сервиса «Wishlist» должен быть раздел с публичными вишлистами, где пользователи могут брать идеи для подарков. В этот раздел должны попасть все вишлисты с настройкой «Публичный». У пользователя должна быть возможность поиска нужных вишлистов.

Сервис «Wishlist» должен предоставлять друзьям пользователя подсказки для подарков. Пользователь при открытии страницы подарка, должен увидеть список подсказок, которые направлены на стимулирование вдохновения у него. На этапе разработки прототипа подсказки реализуются как список статичных фраз. При дальнейшей разработке они могут генерироваться системой.

Дополнительно, сервис должен отображать пользователю количество зарезервированных его друзьями подарков. Это поможет пользователю быть более уверенным относительно своего вишлиста и не испытывать смущения. Такая функция является реализацией паттерна «Социальное одобрение», описанного в книге «Разработка интерфейсов. Паттерны проектирования. 3-е изд.» [7].

#### 1.4. Анализ существующих решений

В Таблице 1 предоставлено сравнение существующих решений и разрабатываемого сервиса.

Таблица 1 – Сравнение существующих и разрабатываемого решений.

Аналоги Критерии	WishList	Oh My Wish	Подаркус	MyWishBoard	GiftBuster
	Создание вишлистов и подарков				
Создание вишлистов	Есть	Есть	Есть	Есть	Есть
Автозаполнение подарка	Есть	Есть	Есть	Нет	Есть

Продолжение Таблицы 1.

Соавторство	Есть	Нет	Есть	Нет	Нет
Антивишлист	Есть	Нет	Нет	Нет	Нет
Настройка доступа к вишлисту					
Приватный	Есть	Есть	Есть	Есть	Есть
Для друзей	Есть	Есть	Есть	Есть	Нет
Публичный	Есть	Нет	Нет	Есть	Есть
Кастомный (для разных друзей)	Есть	Нет	Нет	Нет	Нет
Резервирование					
Резервирование подарков друзьями	Есть	Есть	Есть	Есть	Есть
Резервирование неавторизованными пользователями	Есть	Нет	Нет	Нет	Нет
Планер с резервированными подарками	Есть	Есть	Есть	Нет	Нет
Остальное					
Раздел идей для подарков	Есть	Есть	Есть	Есть	Есть
Напоминание дней рождений	Есть	Нет	Есть	Нет	Нет
Подсказки для подарков	Есть	Нет	Нет	Нет	Нет
Обратная связь по резервированию	Есть	Нет	Нет	Нет	Нет

**Выводы по главе**

В данной главе рассматривается предметная область и описываются проблемы планирования подарков. Дается обзор проблемы с точки зрения получателя подарка и с точки зрения дарителя.

Сервис планирования подарков «Wishlist». Иванова С. А.

Также производится обзор существующих решений и их функционала, а затем – обзор разрабатываемого решения в виде функциональных требований и сравнительный анализ между ними.

Далее дается описание архитектуры разрабатываемой системы.

## 2. Глава 2. Проектирование веб-сервиса

### 2.1. Пользовательские сценарии

В текущем разделе представлены пользовательские сценарии разрабатываемого сервиса, которые описывают взаимодействие между системой и пользователем. Для удобства восприятия они изображены с помощью диаграммы прецедентов и языка моделирования UML [8], а также разбиты на несколько диаграмм по смысловым частям.

Основой веб-сервиса является система учетных записей, с помощью которой авторизуется большинство операций. На диаграмме (рис. 1) изображены пользовательские сценарии, связанные с системой учетных записей.

Пользователь имеет возможность зарегистрироваться в веб-сервисе, авторизоваться и, по необходимости, деавторизоваться. Он может управлять своей учетной записью: изменять email, пароль или логин, изменять персональную информацию, добавлять изображение своего профиля.

Система предлагает возможность поиска пользователей по логину и просмотр их профилей.



Рисунок 1 – Диаграмма прецедентов системы учетных записей.



Веб-сервис включает в себя систему дружбы, предназначенную для авторизации операций, которые могут выполнять только друзья пользователя (например, отображение вишлистов, доступных только друзьям пользователя, резервирование подарков и т.д.).

На диаграмме (рис. 2) изображены пользовательские сценарии для системы дружбы.

Пользователь может управлять списком своих друзей: отправлять заявку на дружбу, принимать и отклонять заявки, удалять друзей. Кроме того, система отображает информацию о друзьях пользователя, его подписчиках и подписках.

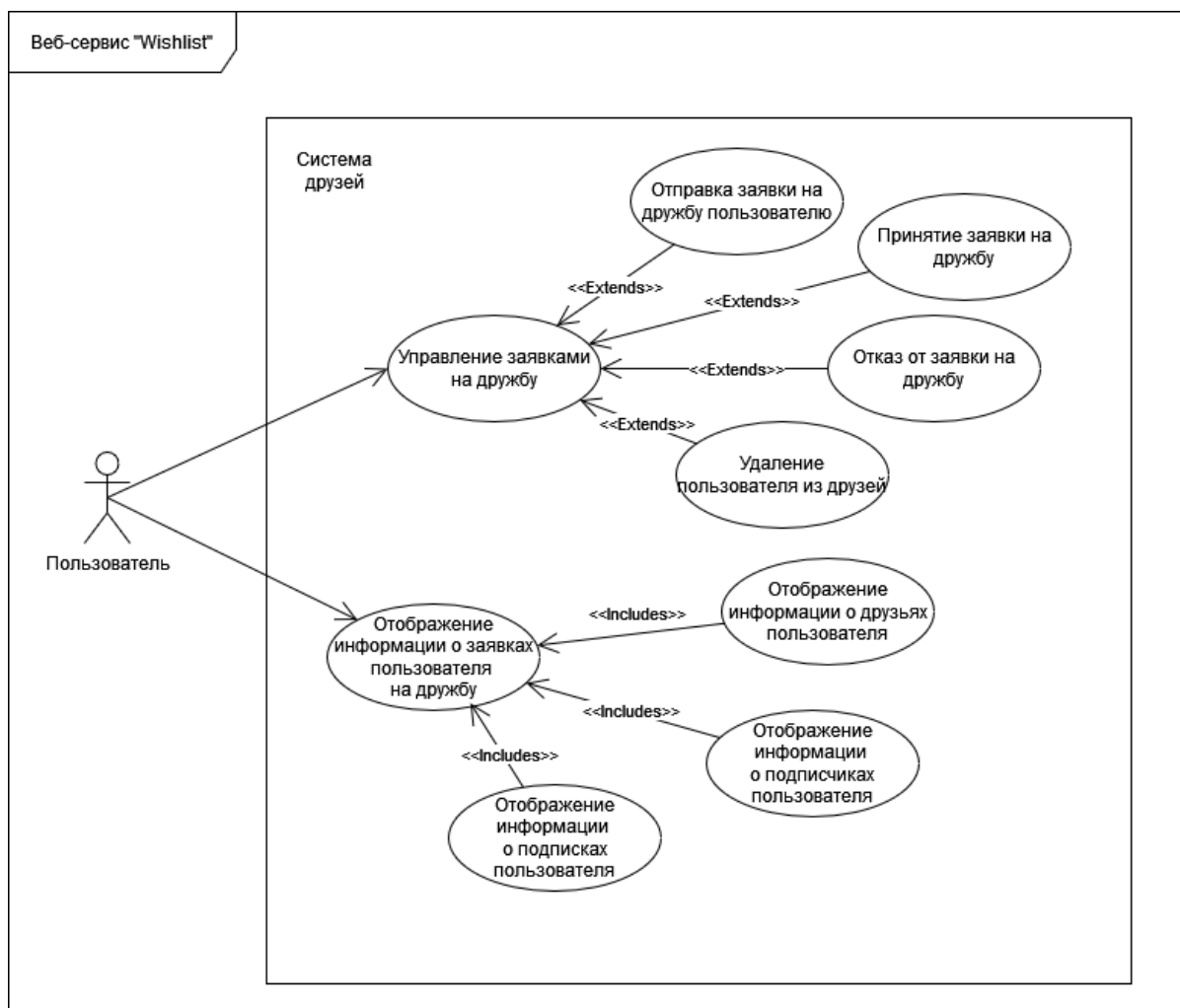


Рисунок 2 – Диаграмма прецедентов системы друзей.

На диаграмме (рис. 3) изображены пользовательские сценарии, связанные с системой вишлистов.

Пользователь может создавать и удалять вишлисты, а также изменять их: название, доступ видимости. При необходимости пользователь имеет возможность добавлять или удалять соавторов, а также изменять доступ к вишлисту по ссылке.

Вишлист – список подарков, поэтому система предлагает возможность управлять подарками вишлиста: пользователь может как создавать карточки подарков с нуля, так и добавлять подарки в

вишлист по ссылке из интернет-магазинов. Карточки подарков можно изменять, удалять и добавлять в другие вишлисты.

Сервис имеет раздел с идеями подарков в виде публичных вишлистов. Пользователь может искать нужные вишлисты, просматривать их, а те карточки подарков, которые ему понравились, – добавить в любой из своих вишлистов.

Если пользователю необходимо перенести вишлист в другой сервис или приложение, он может сделать это, скопировав подарки по соответствующей кнопке.

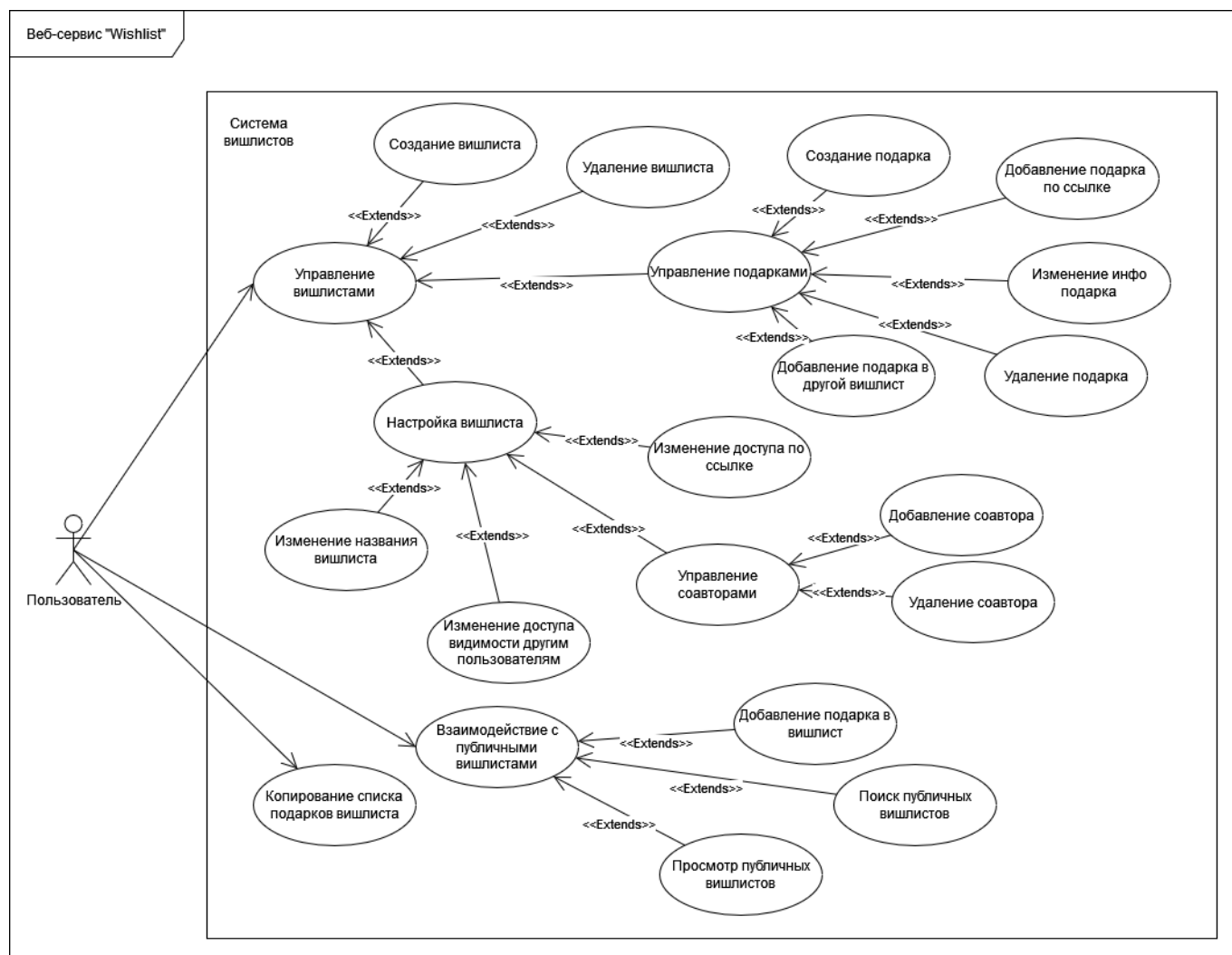


Рисунок 3 – Диаграмма прецедентов системы вишлистов.

Сервис предоставляет возможность резервировать подарки и отображает информацию о резервировании подарка другим пользователям. На диаграмме (рис. 4) представлены пользовательские сценарии, связанные с резервированием подарков.

Если пользователю доступен вишлист с доступом «Для друзей», то он может зарезервировать карточки подарков из вишлиста. При необходимости резервирование можно отменить. Все резервирования доступны в планере пользователя.

Если пользователь неавторизован, а вишлист доступен по ссылке, то он также может резервировать подарки и отменять резервирование. При этом ему необходимо указать свое имя.

Также друзьям пользователя доступна информация о резервировании конкретного подарка: количество зарезервировавших и список этих людей.

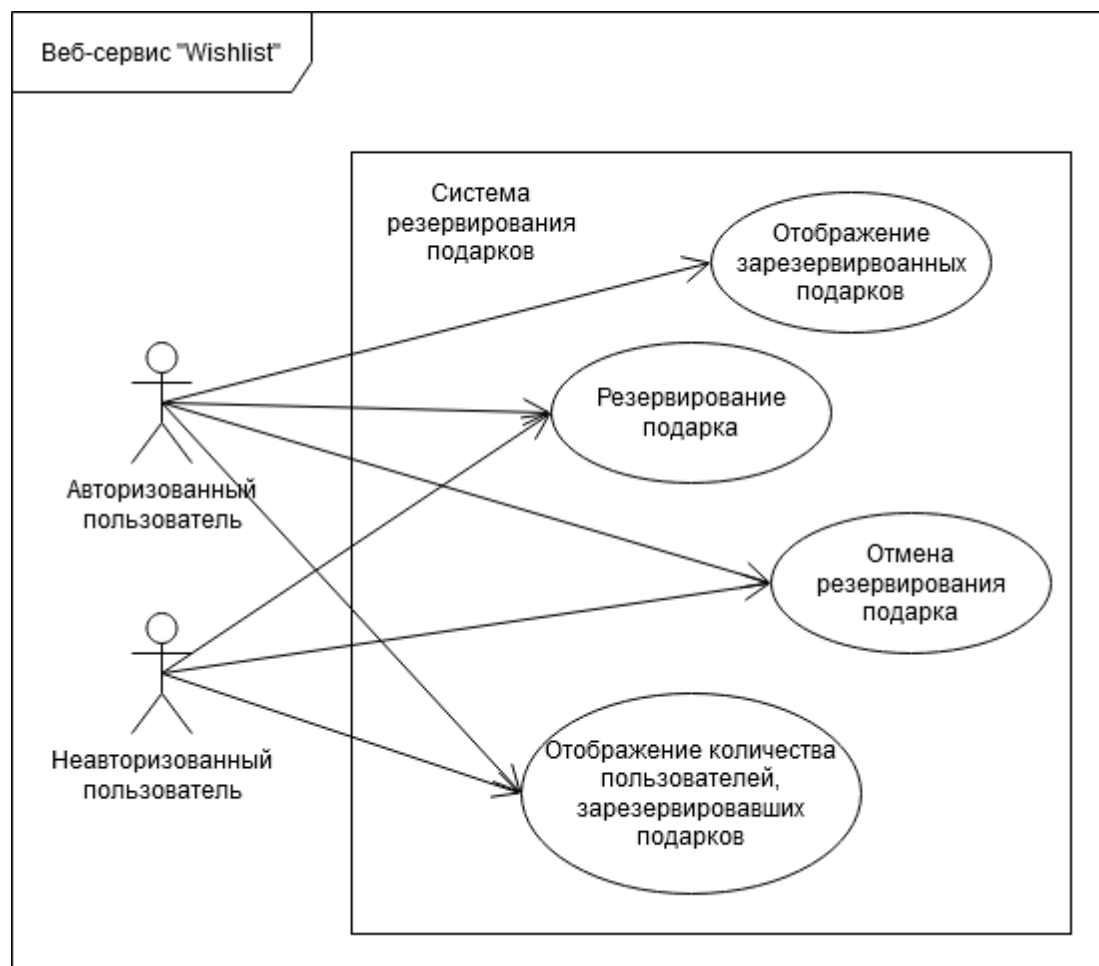


Рисунок 4 – Диаграмма прецедентов системы резервирования.

На диаграмме (рис. 5) изображены прецеденты системы уведомлений, которые предназначены для отображения входящих заявок в друзья, в соавторы, а также уведомлений о наступающих днях рождениях друзей.

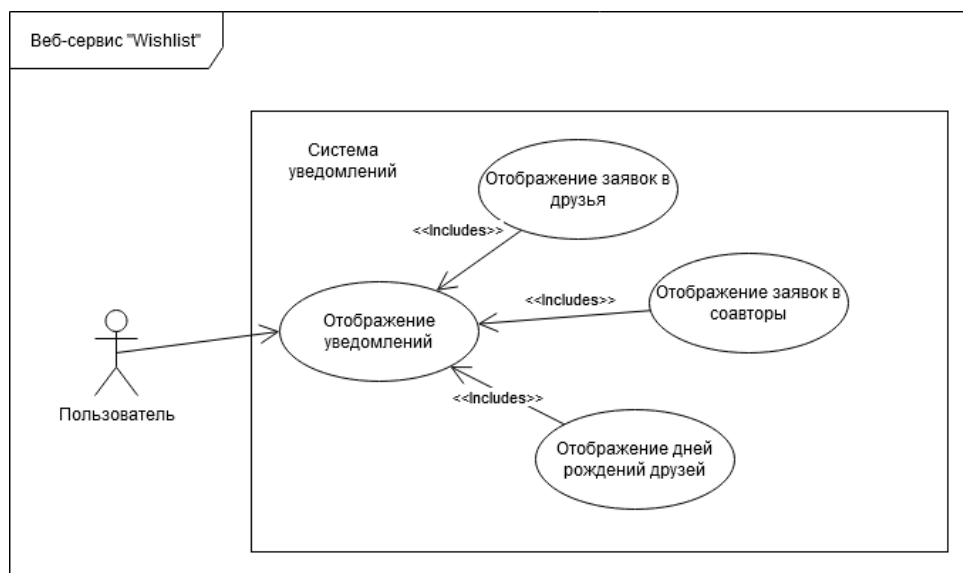


Рисунок 5 – Диаграмма прецедентов системы уведомлений.

## 2.2. Архитектура приложения

Раздел содержит архитектурные решения для разработки веб-сервиса, которые изображены с помощью UML и C4 [9].

Система реализует трехслойную клиент-серверную архитектуру (рис. 6). Клиентская часть – веб-сайт, с которым взаимодействует пользователь через браузер. Серверная часть содержит в себе серверное приложение, которое реализует в себе всю бизнес-логику, а также слой базы данных, где хранятся все данные системы.

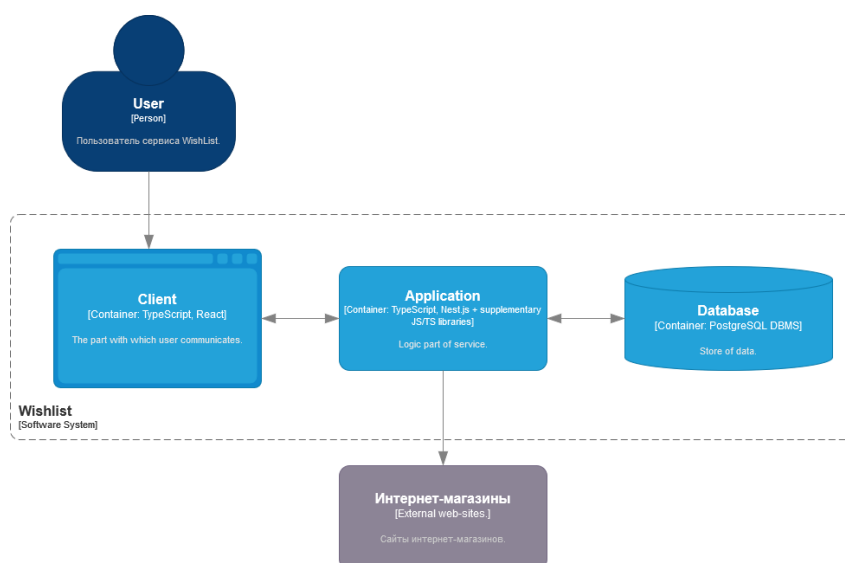


Рисунок 6 – Диаграмма C4 уровня «Контейнеры».

Главным преимуществом выбранного типа архитектуры является разделение ответственности между слоями, что обеспечивает сервису:

1. Гибкость. Пользователь свободен в выборе устройства, с которого ему будет удобнее взаимодействовать с сервисом, главное – наличие браузера на нем.
2. Масштабируемость. При необходимости сервис легко дополняется новыми слоями и компонентами – это могут быть как мобильные, так и дополнительные серверные приложения.
3. Обеспечение безопасности. Все данные хранятся в базе данных, которые пользователи так просто получить не смогут.

### 2.2.1. Схема базы данных

На диаграмме баз данных (рис. 7) изображены все сущности системы и отношения между ними.

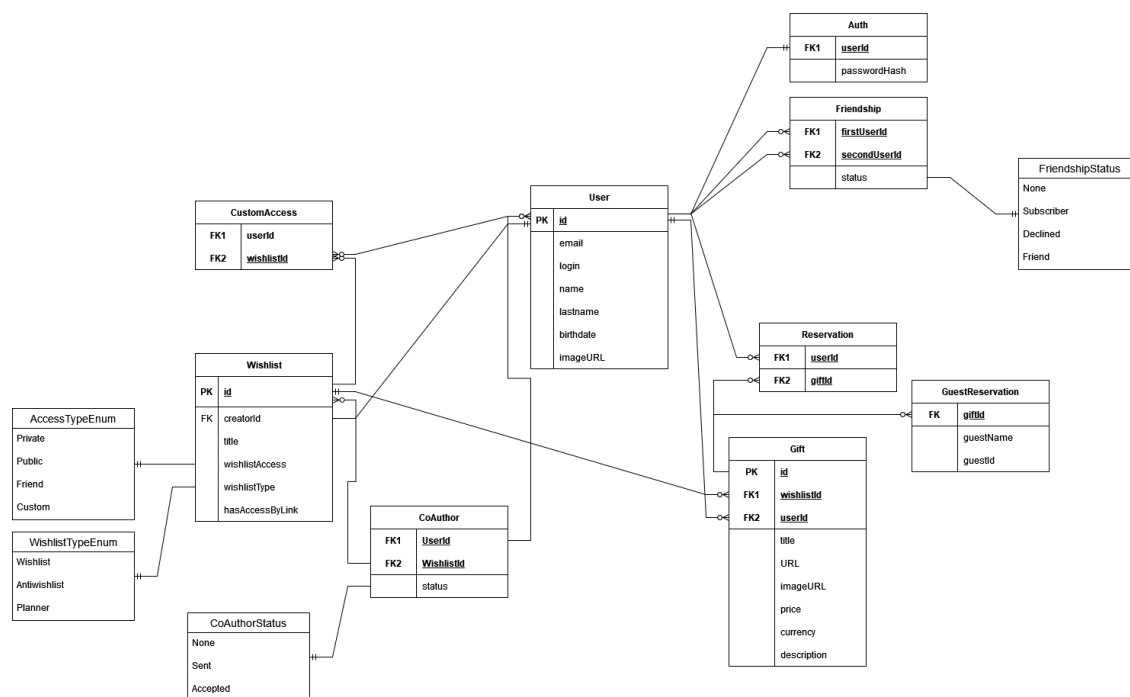


Рисунок 7 – Диаграмма базы данных.

Система имеет следующие сущности: User, Auth, Friendship, Wishlist, Gift, Reservation, GuestReservation, CustomAccess, CoAuthor.

Сущность User представляет собой учетную запись пользователя, в которой хранится его персональная информация. Для аутентификации используется сущность Auth, в которой хранится захешированный пароль.

Для хранения статуса дружбы между пользователями используется сущность `Friendship` со следующими полями: `firstUserId`, `secondUserId` и `status`. Поле `status` может принимать значения `None`, `Subscriber`, `Declined`, `Friend`. Если пользователи друзья, то `status` имеет значение `Friend`. Если пользователь с идентификатором является подписчиком другого пользователя, то `firstUserId` равен идентификатору первого пользователя, а `secondUserId` – второго. При этом поле `status` имеет значение `Subscriber`, если запрос на дружбу отправлен, или же `Declined`, если запрос отклонен.

Сущность `Wishlist` представляет собой вишлист с полями: `title` – название вишлиста, `creatorId` – идентификатор пользователя, создавший вишлист, `wishlistAccess` – доступ к вишлисту, `wishlistType` – тип вишлиста, `hasAccessByLink` – доступен ли вишлист по ссылке. Поле `wishlistAccess` может принимать следующие значения: `Private` (вишлист доступен только самому пользователю), `Public` (вишлист доступен всем пользователям), `ForFriends` (вишлист доступен только друзьям), `Custom` (вишлист доступен только тем друзьям, которых выбрал пользователь). Для хранения друзей, которым доступен вишлист с настройкой доступа `Custom`, используется сущность `CustomAccess`.

Для хранения данных о соавторах используется сущность `CoAuthor` с полями: `userId`, идентификатор, соответствующий пользователю-соавтору, и `status` – статус заявки на соавтора.

Сущность `Gift` представляет информацию о подарке. Поля `userId` и `wishlistId` обозначают пользователя, создавшего подарок, и вишлист, где хранится подарок, соответственно. Сущность подарка также хранит следующую информацию: название подарка (`title`), ссылка на товар (`URL`), ссылка на изображение подарка (`imageUrl`), описание (`description`), цена (`price`) и валюта (`currency`).

Для резервирования подарков используются сущности `Reservation` и `GuestReservation` для авторизованных и неавторизованных пользователей соответственно.

### 2.2.2. Архитектура серверного приложения

Серверное приложение (рис. 8) спроектировано как трехслойная архитектура:

1. Контроллеры – слой, отвечающий за принятие запросов с клиента и отправку ответов.
2. Бизнес-логика – слой, отвечающий за выполнение пользовательских сценариев.
3. Репозиторий – слой, взаимодействующий с базой данных.

Выбранный подход в архитектуре разграничивает ответственность между слоями, что способствует более простому добавлению функционала.

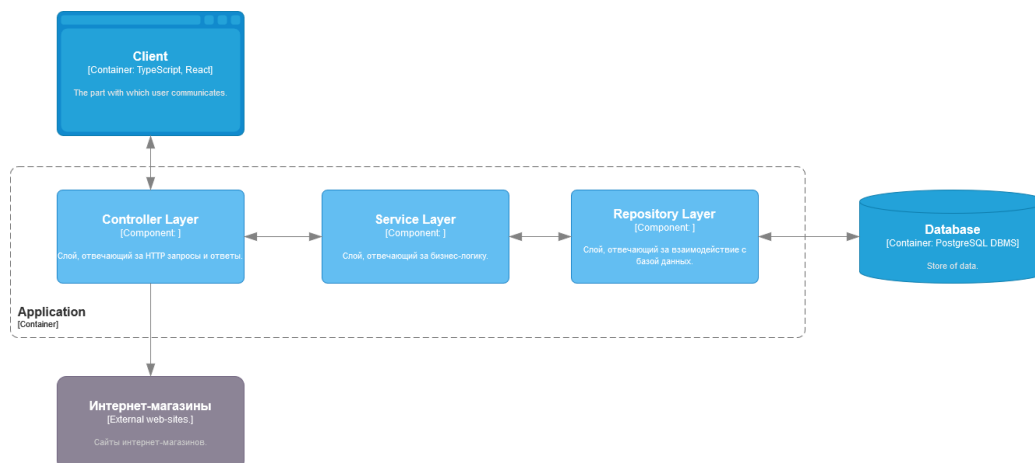


Рисунок 8 – Диаграмма C4 уровня «Component» для серверного приложения.

Серверная часть состоит из следующих подсистем: система учетных записей, система друзей, система вишлистов и подарков, система резервирования, система соавторства, система уведомлений, система файлов.

На диаграмме пакетов (рис. 9) изображены подсистемы и зависимости между ними.

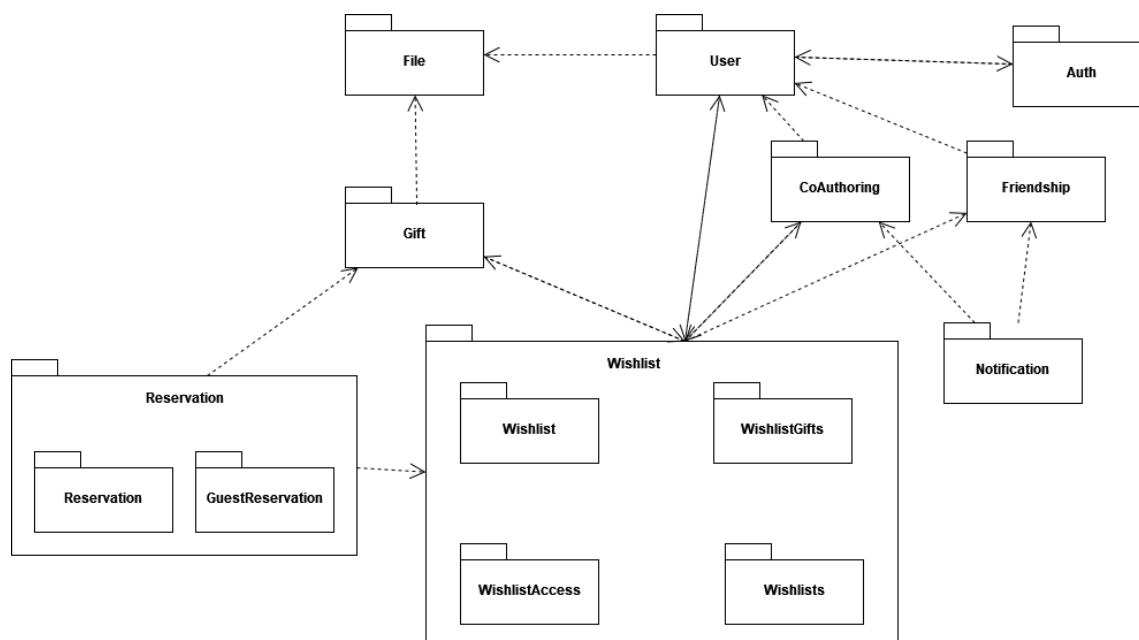


Рисунок 9 – Диаграмма пакетов серверного приложения.

### 2.2.3. Архитектура клиентского приложения

Клиентская часть представляет собой SPA приложение, спроектированное по Flux-архитектуре.

Flux-архитектура – модификация архитектуры MVC под специфику клиентских приложений, состоящая из следующих частей:

1. View – интерфейс, с которым взаимодействует пользователь.
2. Store – хранилище данных и состояния приложения (аналог Model из MVC).
3. Dispatch – часть, отвечающая за обновление данных в store.
4. Action – подобие контроллеров в MVC, взаимодействующие с API сервера или выполняющие какую-то логику над данными.

Такая архитектура направлена на построение одного потока данных как централизованного места, откуда компоненты должны брать необходимые данные.

На диаграмме (рис. 10) изображена архитектура клиентского приложения. Помимо основных компонентов Flux-архитектуры она имеет слой «API», который выполняет HTTP запросы к серверу.

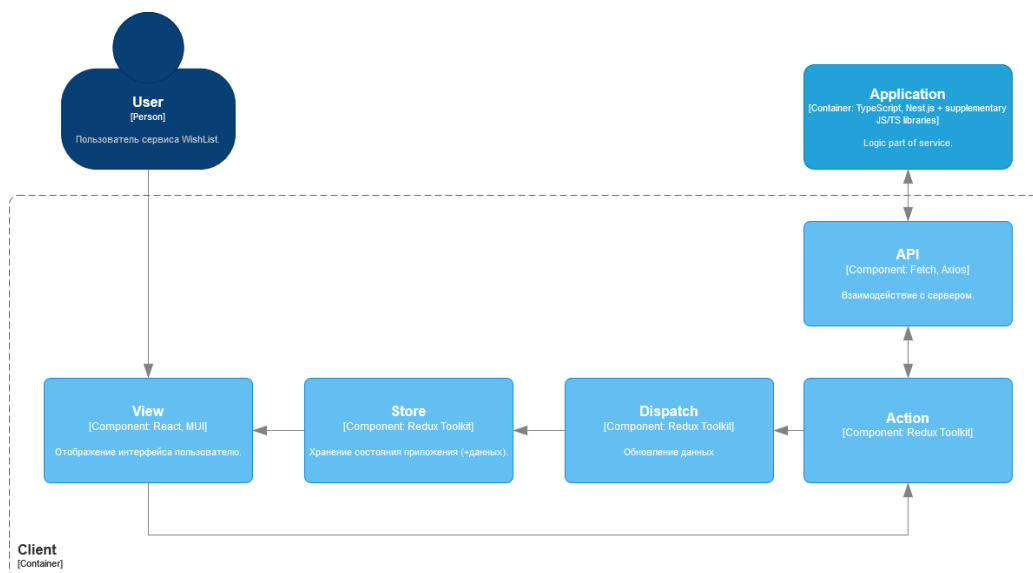


Рисунок 10 – Диаграмма C4 уровня «Component» для клиентского приложения.

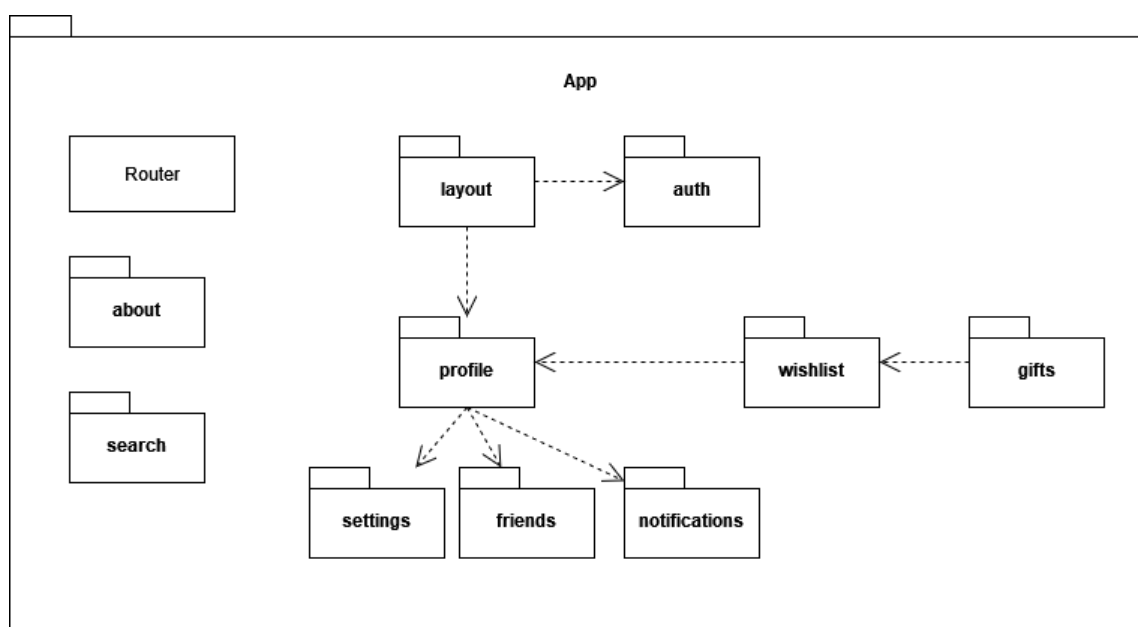


Рисунок 11 – Диаграмма пакетов клиентского приложения.



На диаграмме пакетов (рис. 11) изображены группы компонентов, относящиеся к одним и тем же сущностям, и связи между ними. Для упрощения на диаграмме не изображены связи между роутером (на диаграмме в виде объекта Router) и остальными пакетами, от которых роутер зависит. Роутер имеет зависимости от всех пакетов кроме «Notifications».

## **2.3. Выбор методов и средств реализации**

### **2.3.1. База данных**

Исходя из анализа требований и архитектуры базы данных была выбрана реляционная модель баз данных. Она предполагает строгую структуру сущностей и определенные отношения между ними. С помощью реляционной базы данных можно задать четкие взаимосвязи между сущностями системы, схема которых не склонна к изменениям.

Среди реляционных СУБД была выбрана PostgreSQL из-за наличия опыта работы с ней и отсутствия требований, предъявляемых к СУБД.

### **2.3.2. Серверное приложение**

Для реализации серверного приложения была выбрана платформа Node.js по следующим причинам:

1. TypeScript как язык реализации. Так как TypeScript совместим с JavaScript, то он предполагает гибкость в разработке, и в то же время благодаря статическому назначению типов согласует интерфейсы между функциями и модулями. Кроме того, написание клиентского и серверного приложения на одном языке значительно ускоряет разработку.
2. Наличие большого количество готовых решений для разных целей. С помощью пакетного менеджера можно установить любую JavaScript библиотеку.

Для построения структурированного и легко масштабируемого серверного приложения был выбран фреймворк Nest.js. В качестве ORM для работы с базой данных была выбрана библиотека Sequelize.

Для автоматизированного тестирования была выбрана библиотека Jest, так как она предлагает большой инструментарий и в то же время является простой в использовании.

Для веб-скрейпинга была выбрана библиотека Cheerio из-за легковесности, по сравнению с библиотекой Puppeteer, которая также предлагает большое количество излишнего для проекта функционала.

Для рассылки писем используется библиотека Nodemailer.

### **2.3.3. Клиентское приложение**

Для реализации клиентской части была выбрана библиотека React. Она является мощным и простым в использовании инструментом для создания пользовательских интерфейсов. Кроме того, React является популярным решением, благодаря чему существует большое количество решений под него.

В качестве языка программирования был выбран TypeScript.

Для управления состоянием клиентского приложения были выбраны библиотеки Redux Toolkit и RTK Query, которые предлагают не только стейт-менеджмент, но и взаимодействие с сервером и кеширование данных.

Дополнительно использовалась библиотека готовых UI компонент MaterialUI.

### **Выводы по главе**

Данная глава содержит пользовательские сценарии веб-сервиса, а также описывает архитектуру всей системы и ее частей.

### 3. Глава 3. Программная реализация веб-сервиса

В данной главе представлено описание программной реализации сервиса и итоговый интерфейс.

#### 3.1. Система учетных записей

Система учетных записей позволяет пользователю регистрироваться в системе, авторизоваться, управлять своей учетной записью.

Для авторизации пользователя в системе используются JWT токены [10], хранящиеся в httpOnly куках. В токенах содержится идентификатор пользователя, а также дата истечения срока годности токена. Токены необходимо передавать в большинстве запросов к серверу, поэтому они сразу хранятся в куках, а флаг httpOnly запрещает доступ клиентскому приложению чтение и изменение кук, что повышает безопасность сервиса.

Для авторизации к API сервера используется механизм Guard фреймворка Nest.js. По сути они представляют собой функции, имеющие доступ к объекту запроса и которые имеют возможность прерывать запрос, если он не удовлетворяет какому-то условию. В данном случае Guard верифицирует JWT токен, находящийся в куках, тем самым осуществляя авторизацию пользователя. Эта функция используется с помощью декоратора `@UseGuards()` и может быть применена как ко всему серверному приложению, так и к отдельным контроллерам или методам.

Токены выписываются при успешной регистрации или успешной аутентификации.

Для регистрации в сервисе пользователю необходимо заполнить форму регистрации в клиентском приложении. Для заполнения доступны поля адреса электронной почты, логина и пароля. При невалидных данных приложение отображает подсказки.

Для аутентификации в сервисе пользователю необходимо заполнить форму входа в систему. Пользователь может залогиниться как с помощью адреса электронной почты, так и с помощью логина, предоставив также пароль. При неуспешной аутентификации приложение отобразит подсказки.

Управление учетной записью доступно на странице настроек (рис. 12), куда можно перейти, нажав на кнопку «Редактировать профиль» на странице профиля пользователя. Пользователь имеет возможность изменять персональные данные, такие как имя, фамилию, дату рождения, адрес электронной почты, логин и пароль. Также приложение позволяет изменить изображение профиля.

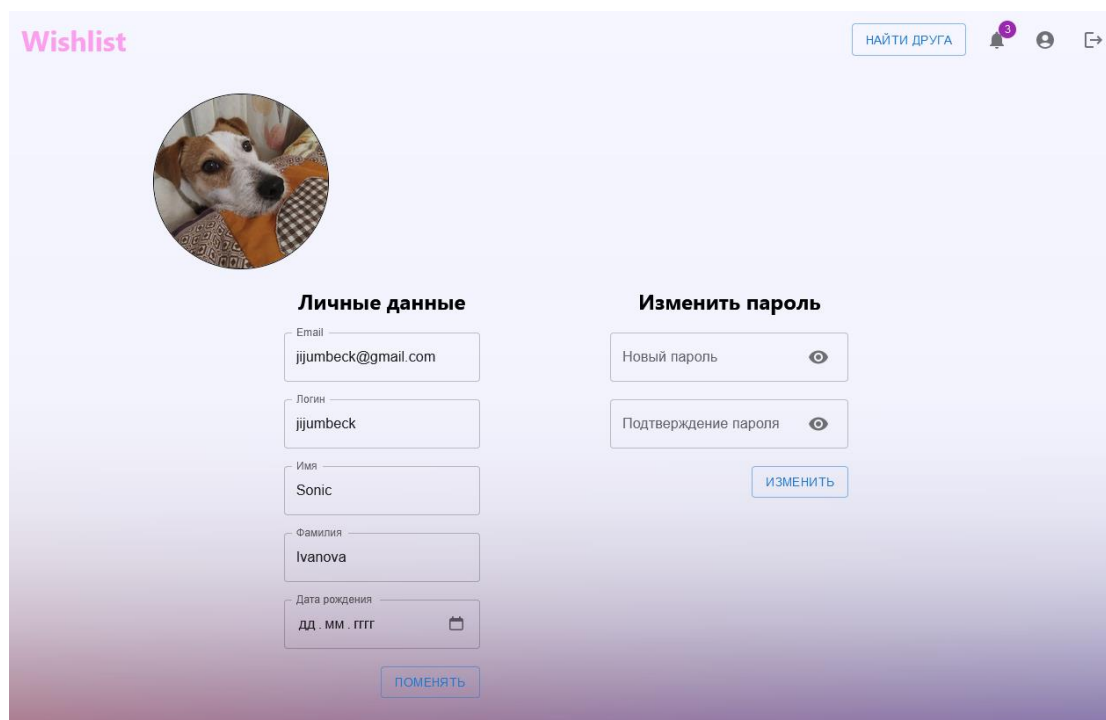


Рисунок 12 – Интерфейс страницы настроек.

Сервис предоставляет возможность поиска пользователя по логину.

### 3.2. Система друзей

Система друзей позволяет отправлять другим пользователям запросы на дружбу, принимать или отклонять запросы, удалять из друзей. Если пользователи друзья, то они могут резервировать подарки друг друга и быть соавторами вишлистов.

Управлять заявками на дружбу можно на странице поиска пользователей (рис. 13) или на странице профиля пользователя.

Чтобы добавить пользователя в друзья, необходимо нажать на кнопку «Добавить в друзья». После нажатия пользователю отправится запрос на дружбу, который он может либо отклонить, либо принять. Если пользователь принял запрос, то пользователи становятся друзьями, в противном случае пользователь, отправивший запрос, остается подписчиком второго.

Пользователь имеет возможность отменить свой запрос в друзья. Таким образом, если пользователи были друзьями, то один становится подписчиком другого. В случае, если пользователи не были друзьями, пользователи не являются ни друзьями, ни подписчиками.

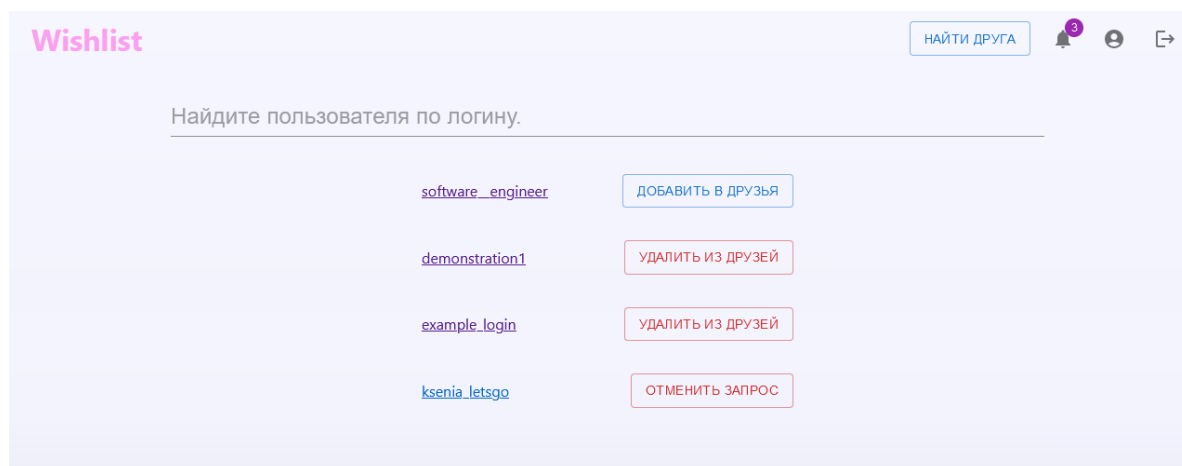


Рисунок 12 – Интерфейс поиска пользователей с демонстрацией функционала системы друзей.

### 3.3. Система вишлистов и подарков

Система вишлистов и подарков позволяет создавать вишлисты, определять их доступ для чтения другим пользователям, добавлять подарки, управлять вишлистами. Система подарков позволяет управлять своими подарками – получать, изменять и удалять их.

При регистрации пользователя система автоматически создает вишлисты со следующими настройками: вишлист типа «Антивишлист», доступный «Для друзей», вишлист с названием «Вишлист для друзей» с настройкой доступа «Для друзей», вишлист с названием «Подборка подарков» и настройкой доступа «Публичный», вишлист с названием «Личный вишлист» и настройкой доступа «Приватный». Это реализовано для того, чтобы пользователю было легче адаптироваться к сервису.

Для создания вишлиста пользователю необходимо перейти на страницу своего профиля, где находится кнопка создания вишлиста. По умолчанию создается вишлист с названием «Новый вишлист» и настройкой доступа «Приватный».

Чтобы редактировать сам вишлист или добавлять подарки, необходимо перейти на страницу вишлиста, нажав на соответствующую карточку.

На странице вишлиста (рис. 14) приложение отображает элементы для управления вишлистом.

Название можно изменить в строке с названием вишлиста. При изменении названия приложение отображает кнопку, при нажатии на которую сервис сохраняет новое название.

Настройку доступа можно изменить в элементе-переключателе, состоящем из 4 кнопок: кнопка с иконкой планеты для настройки доступа «Публичный», кнопка с иконкой людей для настройки доступа «Для друзей», кнопка с иконкой глаза для настройки доступа «Приватный» и

кнопка с шестеренкой для настройки доступа «Кастомный». Для каждой кнопки всплывает подсказка, объясняющая каждую кнопку. При нажатии на кнопку с шестеренкой приложение отобразит выпадающий список, чтобы пользователь мог выбрать друзей, кому будет доступен вишлист.

На странице также отображается элемент для управления соавторами вишлиста.

При нажатии на кнопку «Скопировать» приложение формирует список подарков в текст разметки Markdown. Эта разметка поддерживается немалым количеством сервисов, куда можно вставить текст, и он преобразится в список названий подарков.

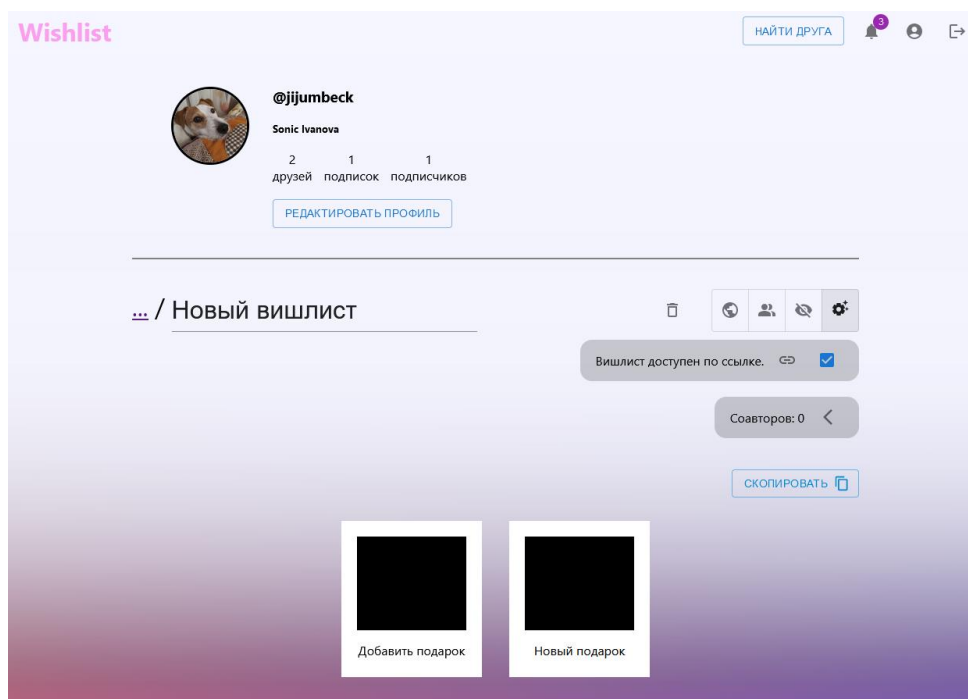


Рисунок 14 – Интерфейс страницы вишлиста.

При изменении настройки доступа на значение «Для друзей» приложение предоставляет доступ к функции изменения доступа к вишлисту по ссылке (рис. 15). Для открытия доступа необходимо отметить галочкой в элементе, для закрытия доступа – убрать. При нажатии на кнопку ссылки система помещает в буфер обмена устройства ссылку на вишлист, при открытии которой пользователю будет доступен список карточек подарков.

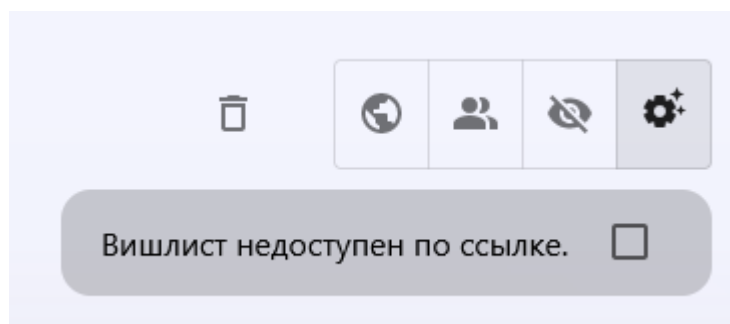


Рисунок 15 – Интерфейс для изменения доступа вишлиста по ссылке.

Для добавления карточки подарка в вишлист необходимо нажать на кнопку «Добавить подарок». После нажатия в вишлист добавиться карточка с названием «Новый подарок». Для изменения информации подарка необходимо перейти на страницу подарка (рис. 16).

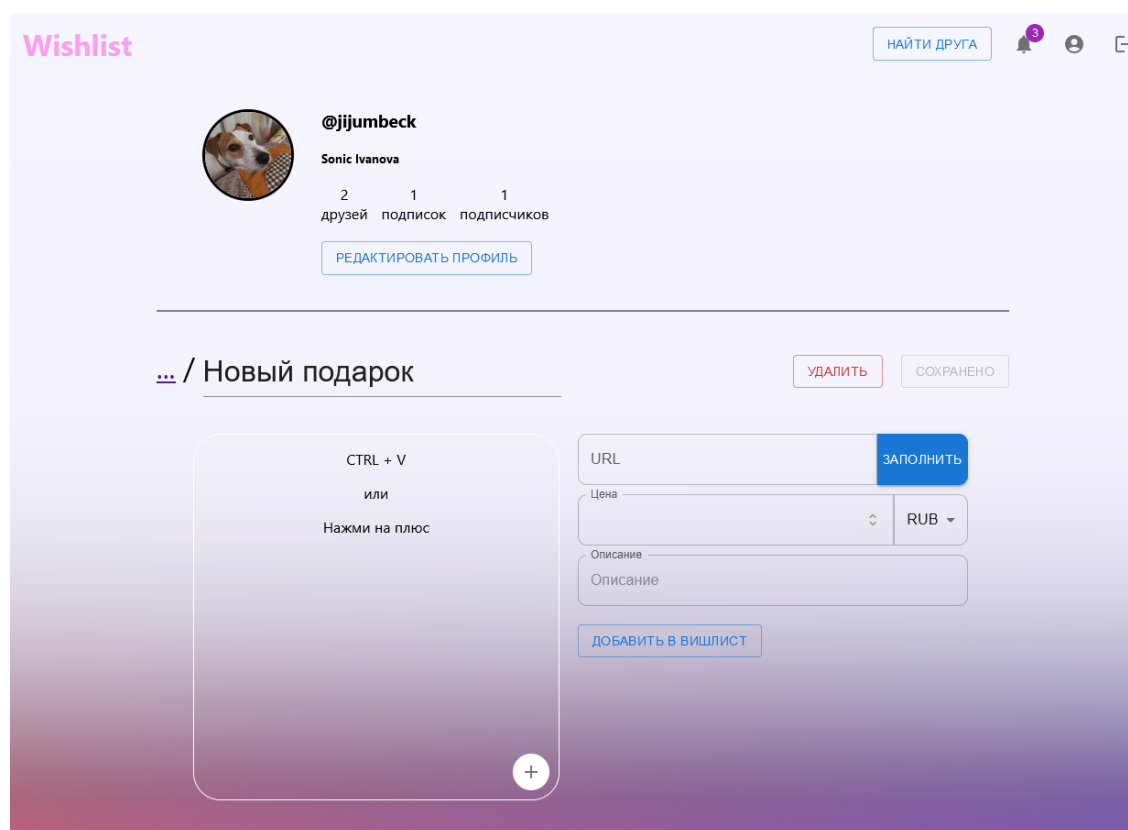


Рисунок 16 – Интерфейс страницы подарка.

Пользователю доступны для редактирования следующие поля: название, ссылка на продукт, цена и валюта, описание, изображение подарка. При изменении любой информации пользователю необходимо нажать на кнопку «Сохранить». Также при необходимости пользователь может удалить карточку подарка, нажав на кнопку «Удалить».

Для добавления изображения подарка пользователь может воспользоваться двумя способами. Первый – добавление картинки из файловой системы устройства. При нажатии на кнопку с плюсом приложение откроет файловый проводник и предложит пользователю выбрать любой файл с расширением «.png», «.jpg» или «.jpeg». После выбора картинка отобразится в рамке. Второй, более удобный в использовании, – вставка файла из буфера обмена. Пользователю необходимо только сделать скриншот и нажать сочетание кнопок для вызова события «paste».

Сервис также реализует автозаполнение полей карточки подарка по URL товара из интернет-магазина. Для этого необходимо вставить ссылку и нажать кнопку «Заполнить».

Функция автозаполнения имеет ограничение в виде проверки клиента на роботизированность. Система не сможет получить данные, если сайт имеет защиту как, например, CAPTCHA.

Для сбора информации о товаре система отправляет запрос на получение HTML страницы. В основном, система ориентируется на Open Graph микроразметку, которая предназначена для расширенного представления веб-страницы. Данная микроразметка направлена на преобразование простой ссылки в красивое представление сайта. Интернет-магазины часто используют ее для продвижения своих товаров – для каждого товара они заполняют краткую информацию и добавляют ее в мета теги документа.

Если HTML документ не имеет мета тегов с микроразметкой, то система парсит контент тега body для извлечения необходимой информации.

Веб-скрейпинг системы собирает такие данные, как название, изображение, цена и валюта. В отличие от остальных аналогов сервис не включает в сбор информации поле «Описание», так как в отличие от интернет-магазинов описание в разрабатываемом решении предназначено для комментариев пользователя нежели для описания товара.

Для всех авторизованных пользователей на странице карточки подарка доступна кнопка «Добавить в вишлист», при нажатии на которую открывается список вишлистов пользователя. Система предоставляет возможность добавление подарка в любой из этих вишлистов.

В приложении также есть раздел с идеями для подарков (рис. 17). Он содержит в себе вишлисты пользователей с настройкой доступа «Публичный». Пользователи могут искать подборку подарков по названию, а система отобразит искомые вишлисты.

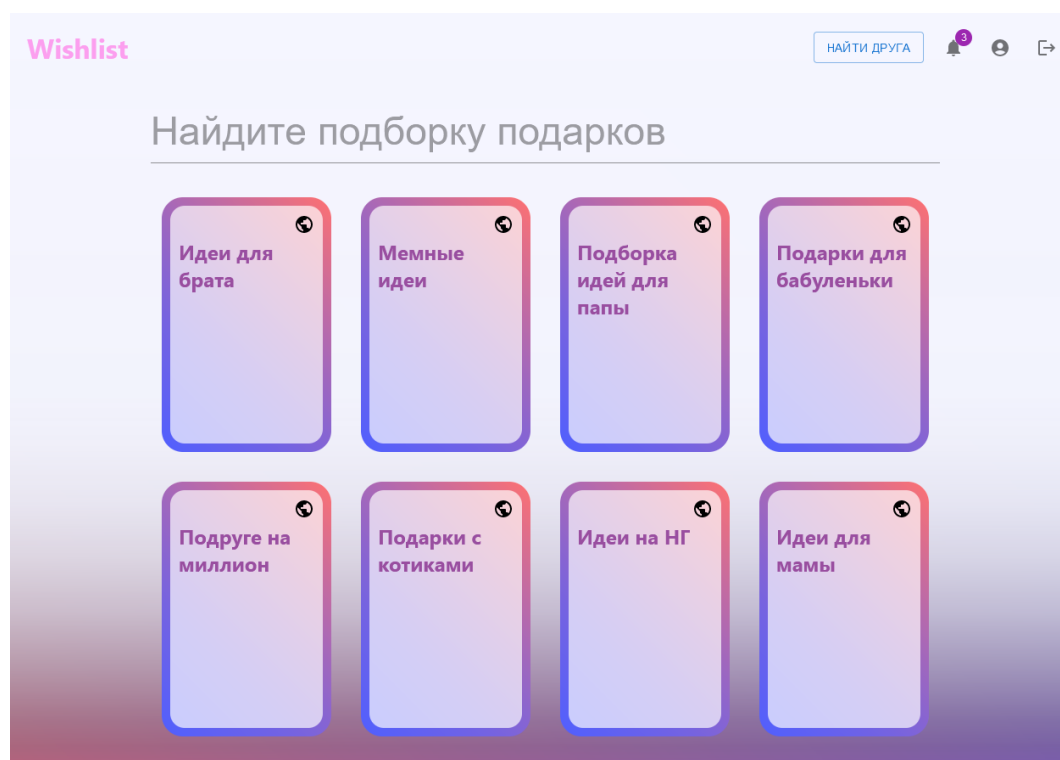


Рисунок 17 – Интерфейс раздела с идеями для подаркой.



### 3.4. Система резервирования подарков

Система резервирования подарков позволяет резервировать подарки из вишлистов с настройкой доступа «Для друзей», созданных другом пользователя. Дополнительно, пользователь имеет возможность отменить резервирование или просмотреть список зарезервированных им подарков.

Сервис предусматривает два возможных варианта: для авторизованного пользователя и для неавторизованного. Авторизация в сервисе предоставляет доступ ко всему функционалу резервирования.

Для того, чтобы зарезервировать подарок, необходимо перейти на страницу вишлиста, созданного пользователем, который является другом, и имеющим настройку доступа «Для друзей». Сервис отобразит список подарков вместе с элементами для бронирования (рис. 18). Под каждой карточкой подарка отобразит кнопку, при нажатии на которую пользователь сможет зарезервировать данный подарок, а также информацию о количестве пользователей, которые тоже зарезервировали его. Если пользователь зарезервировал подарок, то вместо кнопки резервирования, будет отображаться кнопка отмены.

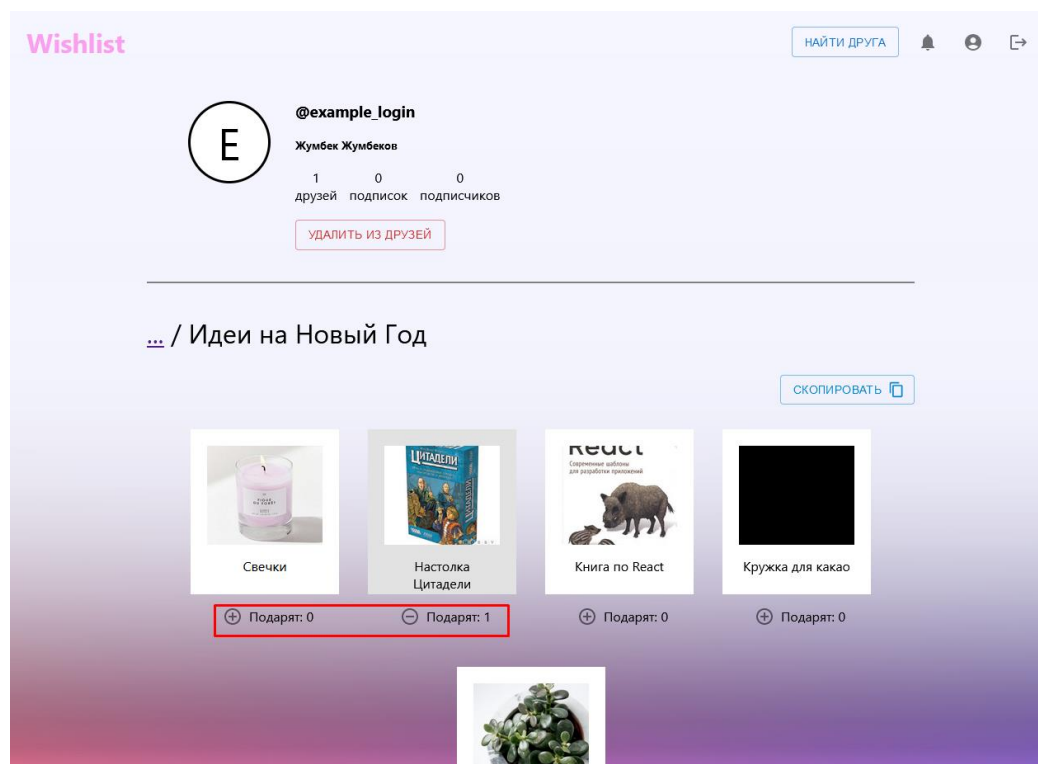


Рисунок 18 – Интерфейс отображения списка подарков с возможностью зарезервировать.

При открытии страницы подарка (рис. 19) пользователю также доступна функция резервирования и функция отмены. Дополнительно, кроме информации о количестве сервис отображает список пользователей, которые зарезервировали данный подарок. Если этот подарок

зарезервировал авторизованный пользователь, то в списке будет ссылка на страницу профиля этого пользователя.

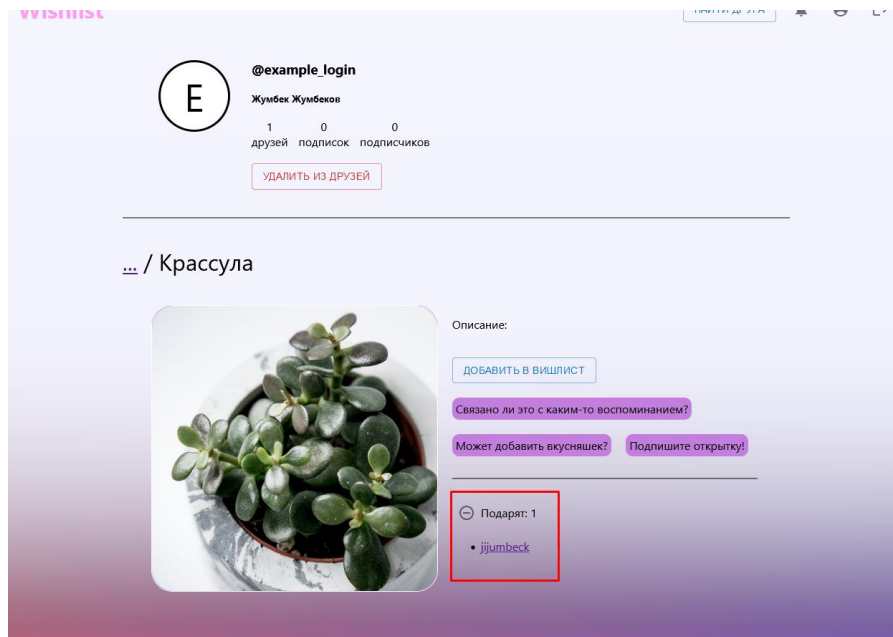


Рисунок 19 – Интерфейс отображения страницы подарка с функционалом для резервирования.

Функция резервирования доступна неавторизованным пользователям только в том случае, если создатель вишлиста открыл доступ по ссылке. Чтобы зарезервировать подарок, пользователь должен ввести свое имя в целях идентификации для других пользователей и нажать кнопку «Зарезервировать» (рис. 20). После этого, сервис отобразит имя пользователя в списке.

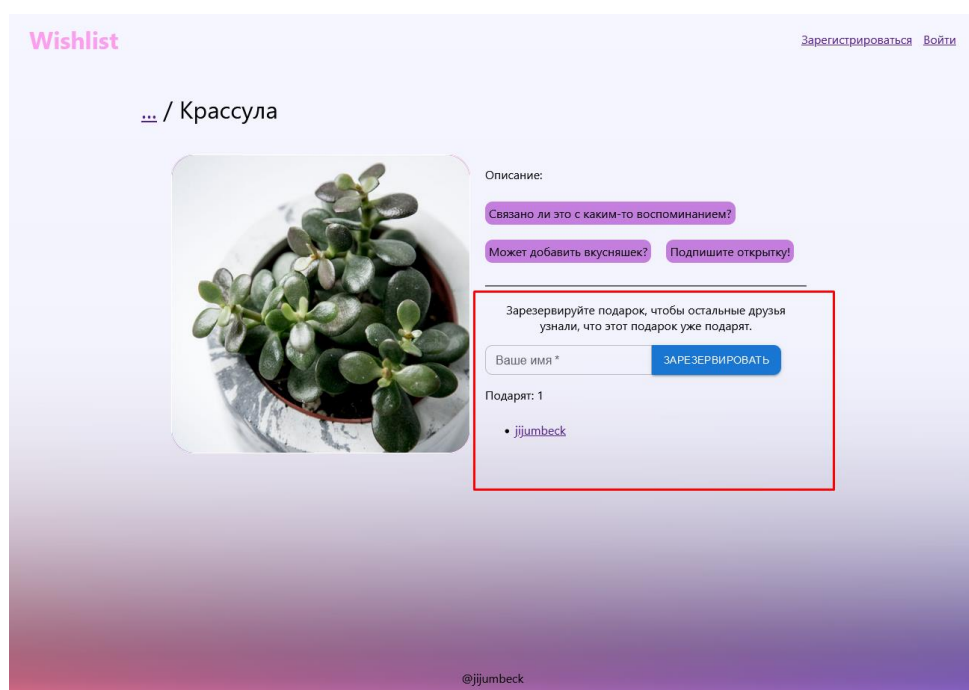


Рисунок 20 – Интерфейс резервирования подарка неавторизованным пользователем.

Неавторизованному пользователю также, доступна функция отмены резервирования по кнопке «Отменить резервирование» (рис. 21).

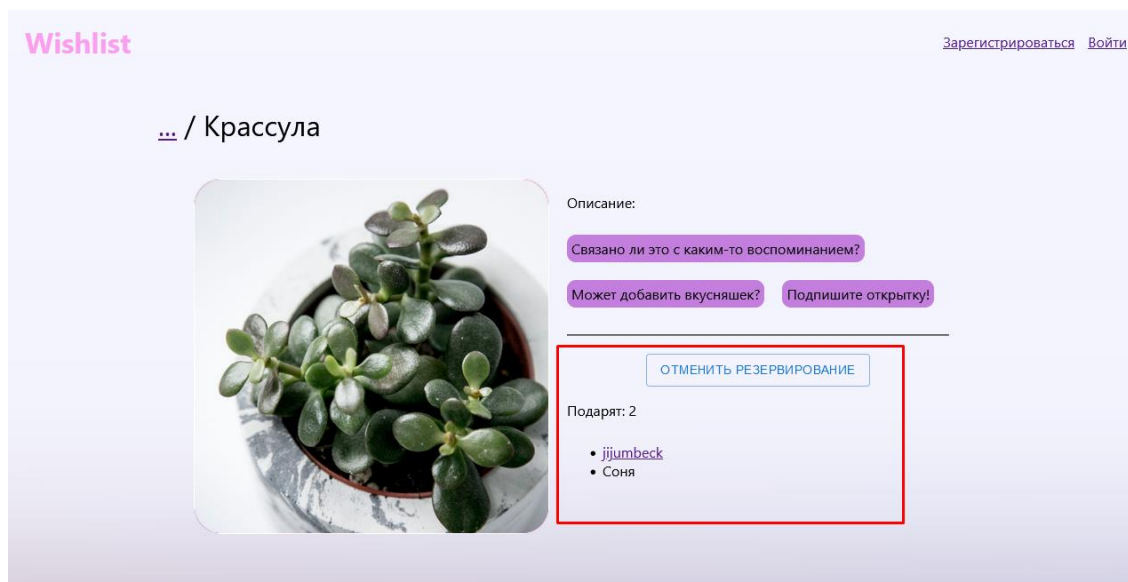


Рисунок 21 – Интерфейс отмены резервирования подарка неавторизованным пользователем.

При перезагрузке страницы приложение также отобразит актуальную информацию о зарезервированных подарках неавторизованного пользователя. Это возможно, так как при первом резервировании система генерирует идентификатор гостя, с которым связывает резервирования, и сохраняет его в локальном хранилище браузера. Сохраненный идентификатор используется в дальнейшем при резервировании.

Если пользователь решает зарегистрироваться или аутентифицироваться и при этом он резервировал подарки как гость, то система перенесет резервирования с гостевого идентификатора на учетную запись.

При резервировании подарка сервис присылает письма по электронной почте остальным пользователям, которые также зарезервировали этот подарок, в целях уведомления (рис. 22).

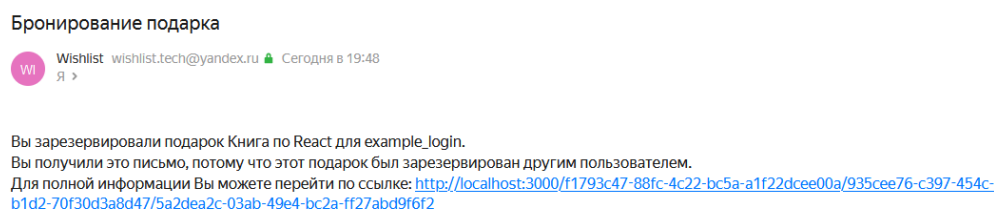


Рисунок 22 – Письмо о бронировании подарка другим пользователем.

Авторизованному пользователю доступен планер (рис. 23), где отображаются все подарки, забронированные пользователем. Чтобы увидеть список подарков, пользователю необходимо перейти на страницу профиля, где в начале списка вишлистов отобразится карточка «Планер». При

нажатии на нее приложение отобразит список резервирований, каждое из которых включает в себя: ссылку на страницу подарка, ссылку на страницу профиля друга, кнопку для отмены резервирований.

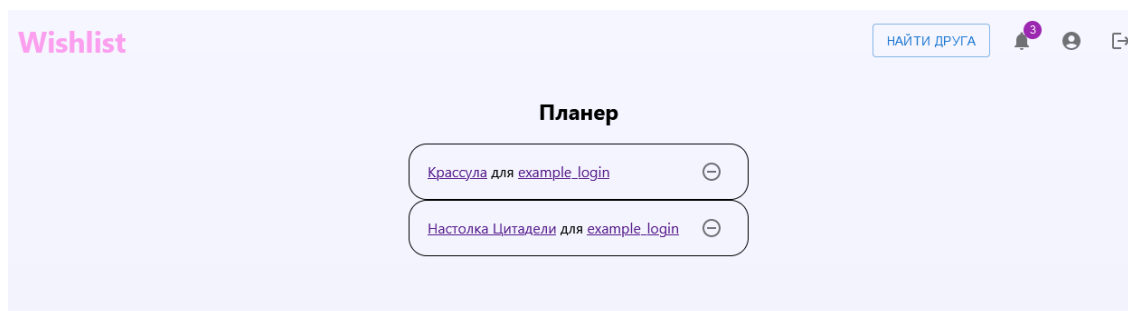


Рисунок 23 – Интерфейс планера зарезервированных подарков.

### 3.5. Система соавторства

Система соавторства позволяет пользователю делиться доступом к добавлению и удалению подарков в вишлисте.

Для добавления пользователя в соавторы вишлиста, он должен быть другом; вишлист может иметь любую настройку доступа. Чтобы добавить друга в соавторы, необходимо нажать на кнопку в виде стрелки и в выпадающем списке выбрать друга, нажав на кнопку с плюсом (рис. 24). Если друг уже является соавтором, вместо кнопки добавления будет кнопка удаления пользователя из соавторов.

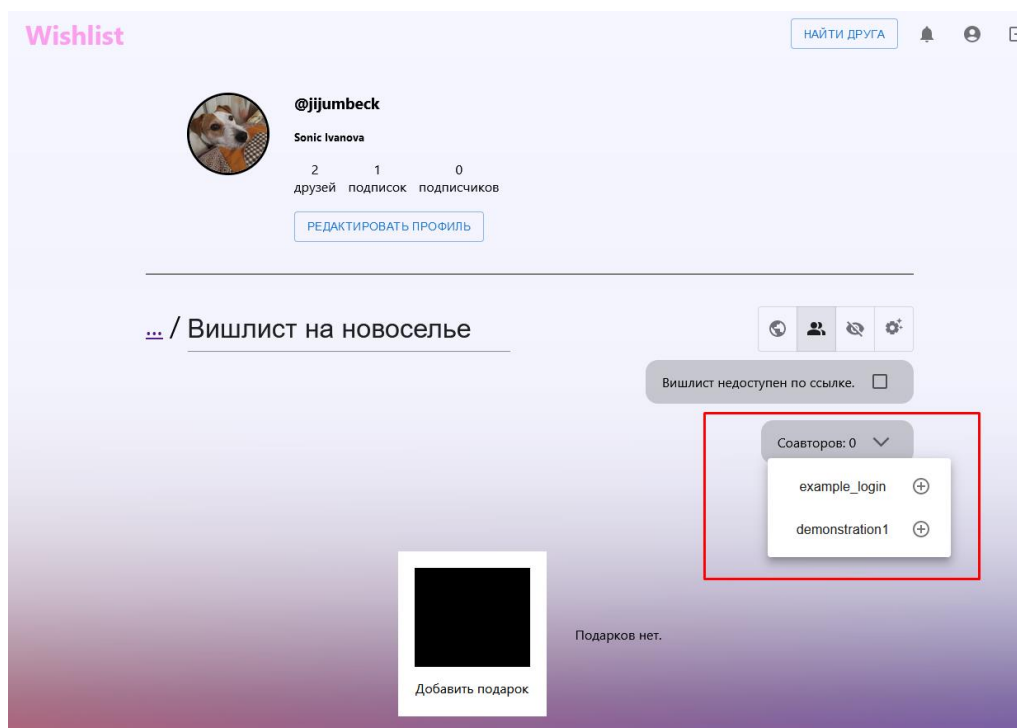


Рисунок 24 – Интерфейс для управления соавторами.

После этого друг получит уведомление о заявки на соавторство вишлиста, которую он может как принять, так и отклонить. Если пользователь принял заявку, то он становится соавтором вишлиста, а сам вишлист становится доступным в профиле пользователя. На странице вишлиста пользователю доступно добавление подарков в вишлист и кнопка «Покинуть вишлист», которая убирает пользователя из вишлиста (рис. 25).

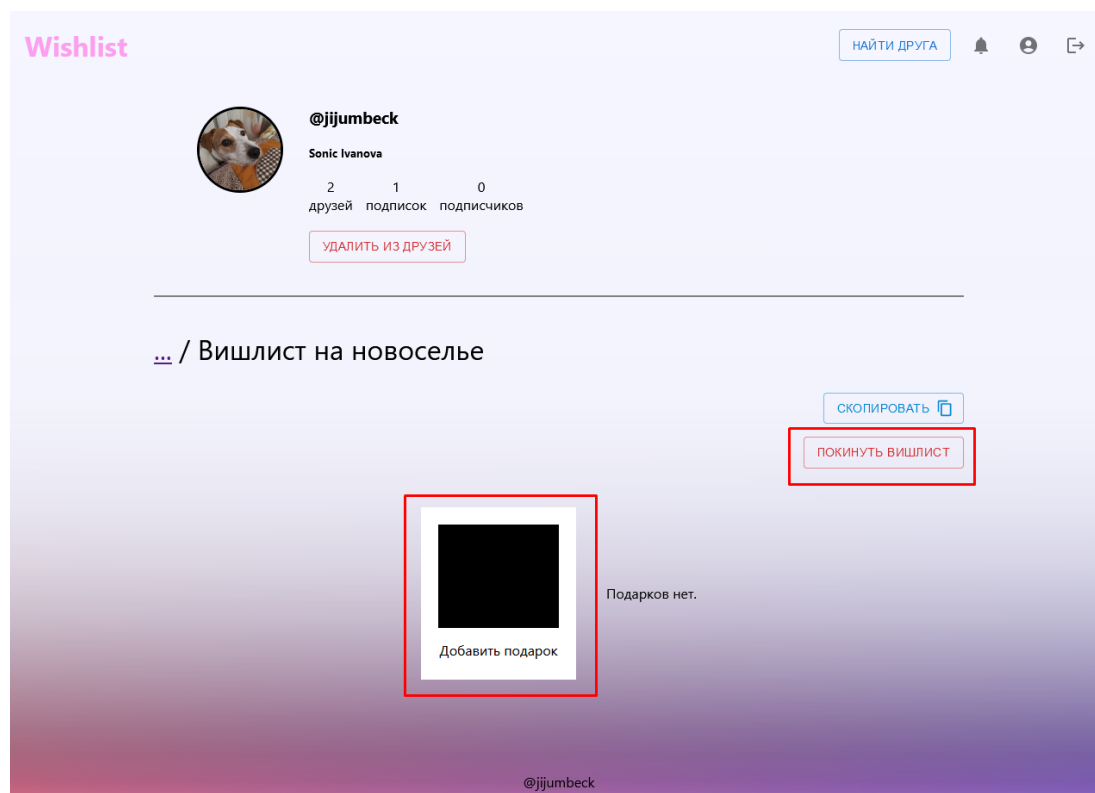


Рисунок 25 – Интерфейс вишлиста с кнопкой «Покинуть вишлист».

### 3.6. Система уведомлений

Система уведомлений предназначена для оповещения пользователя о входящих запросах на дружбу, запросах на соавторство, а также о днях рождениях друзей.

В клиентском приложении уведомления доступны авторизованному пользователю в верхнем правом углу при нажатии на кнопку с колокольчиком. В выпадающем списке пользователь может видеть все уведомления и взаимодействовать с ними (рис. 26).

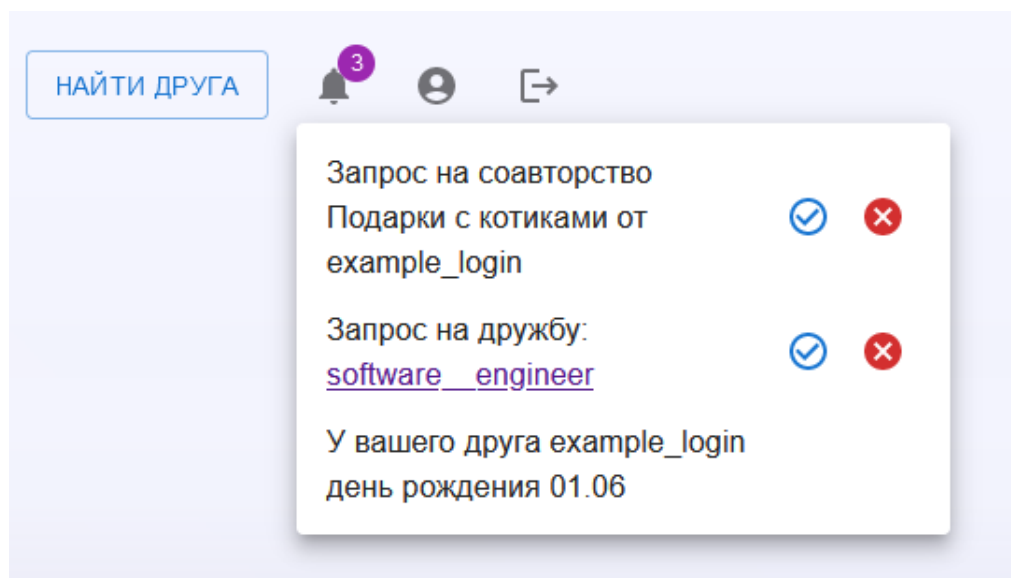


Рисунок 26 – Интерфейс уведомлений.

### Выводы по главе

Данная глава дает описание программной реализации сервиса планирования подарков «Wishlist». Разработанное решение реализует все заявленные функции.

## **Заключение**

Процесс придумывания подарков близким и друзьям зачастую бывает затруднительным, а из-за большого количества событий людям необходимо думать над подарками много раз ежегодно. Вишлисты могут выступать в качестве средства коммуникации, которое может облегчить как процесс выбора подарка, так и сообщения о своих желаниях другим.

В ходе проекта был реализован весь заявленный функционал. Цель была достигнута, а все задачи выполнены. Разработанный сервис представляет собой инструмент для планирования подарков, который предлагает гибкий функционал для создания вишлистов и планирования подарков друзьям, близким и даже себе. С помощью раздела идей для подарков пользователи могут помогать друг другу с выбором подходящих подарков. Для бизнеса сервис может стать платформой для продвижения своей продукции.

В качестве направления дальнейшего развития предполагается внедрение системы рекомендаций для раздела идей подарков и улучшение системы поиска вишлистов, а также разработка функционала для командного планирования подарков.

### Список используемой литературы

1. Сайт для скачивания приложения «Oh My Wishes» [Электронный ресурс] // Oh My Wishes. Режим доступа: <https://apps.apple.com/ru/app/ohmywishes/id1275470725> свободный. (дата обращения 22.04.2024)
2. Веб-сервис «Подаркус» [Электронный ресурс] // Подаркус. Режим доступа: <https://podarkus.ru/> свободный. (дата обращения 22.04.2024)
3. Веб-сервис «MyWishBoard» [Электронный ресурс] // MyWishBoard. Режим доступа: <https://mywishboard.com/> свободный. (дата обращения 22.04.2024)
4. Сайт для скачивания приложения «GiftBuster» [Электронный ресурс] // GiftBuster. Режим доступа: <https://apps.apple.com/us/app/wishlist-by-giftbuster/id1051348033> свободный. (дата обращения 22.04.2024)
5. Фреймворк NestJS [Электронный ресурс] // NestJS. Режим доступа: <https://docs.nestjs.com/> свободный. (дата обращения 22.04.2024)
6. Библиотека для создания интерфейсов React [Электронный ресурс] // React. Режим доступа: <https://react.dev/> свободный. (дата обращения 22.04.2024)
7. Тидвелл Дженифер, Брюэр Чарли «Разработка интерфейсов. Паттерны проектирования. 3-изд.» // O'Reilly, 2022.
8. Unified Modeling Language [Электронный ресурс] // OMG. Режим доступа: <https://www.omg.org/spec/UML/2.5.1/PDF> свободный. (дата обращение 14.05.2024)
9. C4 Architecture [Электронный ресурс] // C4. Режим доступа: <https://c4model.com/> свободный. (дата обращения: 14.05.2024)
10. JSON Web Token [Электронный ресурс] // JWT. Режим доступа: <https://jwt.io/introduction> свободный. (дата обращения 14.05.2024)