

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МОЭВМ**

**КУРСОВАЯ РАБОТА**  
**по дисциплине «Разработка приложений для мобильных платформ»**  
**Тема: Разработка приложения для проращивания рассады**

Студент гр. 8304

\_\_\_\_\_

Мешков М.А.

Студент гр. 8304

\_\_\_\_\_

Рыжиков А.В.

Преподаватель

\_\_\_\_\_

Заславский М.М.

Санкт-Петербург

2022

## ВВЕДЕНИЕ

Все подходят к проведению досуга по разному, одно из возможных занятий — выращивание небольших растений, это хороший метод освободиться от стресса или же перезагрузиться после работы. Актуальность выбранной темы заключается в том, что чтобы не забывать ухаживать за растениями и следить за ними, удобно использовать приложение.

Предлагаемое решение заключается в разработке мобильного приложения, которое позволит добавлять выращиваемые растения в него, настраивать частоту различного типа ухода, сохранять фотографию выращиваемого растения, предлагаемое решение будет уведомлять пользователя, когда необходимо осуществить действия по уходу за растениями, также приложение будет предоставлять пользователю просмотр календаря ухода как за отдельным растением, так и для всех добавленных.

Предлагаемое решение необходимо реализовывать как мобильное приложение поскольку люди практически всегда имеют под рукой свой телефон, это позволяет вовремя услышать уведомление, а также мобильное решение позволит прямо например в теплице отметить задачу по уходу за растением как выполненную.

Цель работы: разработать мобильное приложения для проращивания рассады.

Для достижения цели необходимо решить следующие задачи — разработка сценариев использования, макета, пользовательского интерфейса, создание тестов, оценка сложности пользовательского интерфейса.

## 1. СЦЕНАРИИ ИСПОЛЬЗОВАНИЯ

Были разработаны следующие сценарии использования:

*Сценарий использования - "Добавление растения"*

Действующее лицо: пользователь.

Основной сценарий:

1. Пользователь нажимает на кнопку с иконкой плюса.
2. Пользователь попадает в список видов растений.
3. Пользователь выбирает вид добавляемого растения.
4. Пользователь нажимает на место для фотографии растения.
5. Открывается страница с возможностью выбора картинки.
6. Пользователь выбирает картинку.
7. Пользователь нажимает на кнопку редактирования плана ухода.
8. Открывается страница с возможностью поменять параметры ухода (частота полива, удобрения, подстригания, а также нужно ли ставить растение в теплицу).
9. Пользователь редактирует параметры ухода и возвращается назад.
10. Пользователь нажимает на кнопку редактирования растения.
11. Открывается страница с возможностью указать комментарий для растения.
12. Пользователь вводит комментарий.
13. Пользователь подтверждает смену комментария нажатием кнопки "Edit plant comment".
14. Пользователь возвращается 2 раза назад - т.е. на главную страницу.

*Сценарий использования - "Просмотр списка растений"*

Действующее лицо: пользователь.

Основной сценарий:

1. Пользователь попадает на главный экран при открытии приложения.

2. Пользователь нажимает на кнопку сортировки и выбирает тип сортировки (напр. по дате создания в порядке от наиболее недавних).
3. Пользователь видит отсортированный список растений.
4. Пользователь вводит комментарий или название растение в поисковую строку.
5. На экране остаются только растения, удовлетворяющие поисковому критерию.

### *Сценарий использования - "Редактирование растения"*

Действующее лицо: пользователь.

Основной сценарий:

1. Пользователь нажимает по растению на главном экране приложение.
2. Открывается страница выбранного растения.
3. Пользователь нажимает на место для фотографии растения.
4. Открывается страница с возможностью выбора картинки.
5. Пользователь выбирает картинку.
6. Пользователь нажимает на кнопку редактирования плана ухода.
7. Открывается страница с возможностью поменять параметры ухода (частота полива, удобрения, подстригания, а также нужно ли ставить растение в теплицу).
8. Пользователь редактирует параметры ухода и возвращается назад.
9. Пользователь нажимает на кнопку редактирования растения.
10. Открывается страница с возможностью указать комментарий для растения.
11. Пользователь вводит комментарий.
12. Пользователь подтверждает смену комментария нажатием кнопки "Edit plant comment".
13. Пользователь возвращается 2 раза назад - т.е. на главную страницу.

### *Сценарий использования - "Удаление растения"*

Действующее лицо: пользователь.

Основной сценарий:

1. Пользователь нажимает по растению на главном экране приложение.
2. Открывается страница выбранного растения.
3. Пользователь нажимает на кнопку редактирования растения.
4. Открывается страница с возможностью указать комментарий для растения.
5. Пользователь нажимает на кнопку удаления растения.

### *Сценарий использования - "Просмотр расписания ухода для конкретного растения"*

Действующее лицо: пользователь.

Основной сценарий:

1. Пользователь нажимает по растению на главном экране приложение.
2. Открывается страница выбранного растения.
3. Пользователь нажимает на кнопку просмотра плана ухода за растением.
4. Открывается страница с расписанием ухода для конкретного растения.

### *Сценарий использования - "Просмотр расписания ухода для всех растений"*

Действующее лицо: пользователь.

Основной сценарий:

1. Пользователь нажимает на кнопку с иконкой календаря на главном экране.
2. Открывается страница с расписанием ухода для всех растений.

### *Сценарий использования - "Получение напоминания об уходе"*

Действующее лицо: пользователь.

Основной сценарий:

1. Пользователю приходит уведомление о необходимости ухода за растениями.
2. Пользователь нажимает по уведомлению.
3. Открывается страница с описанием действий по уходу.
4. Пользователь нажимает кнопку "Done" когда он выполнил уход.

## 2. ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС

### 2.1. Макет UI

Макет UI см. на рис. 1.

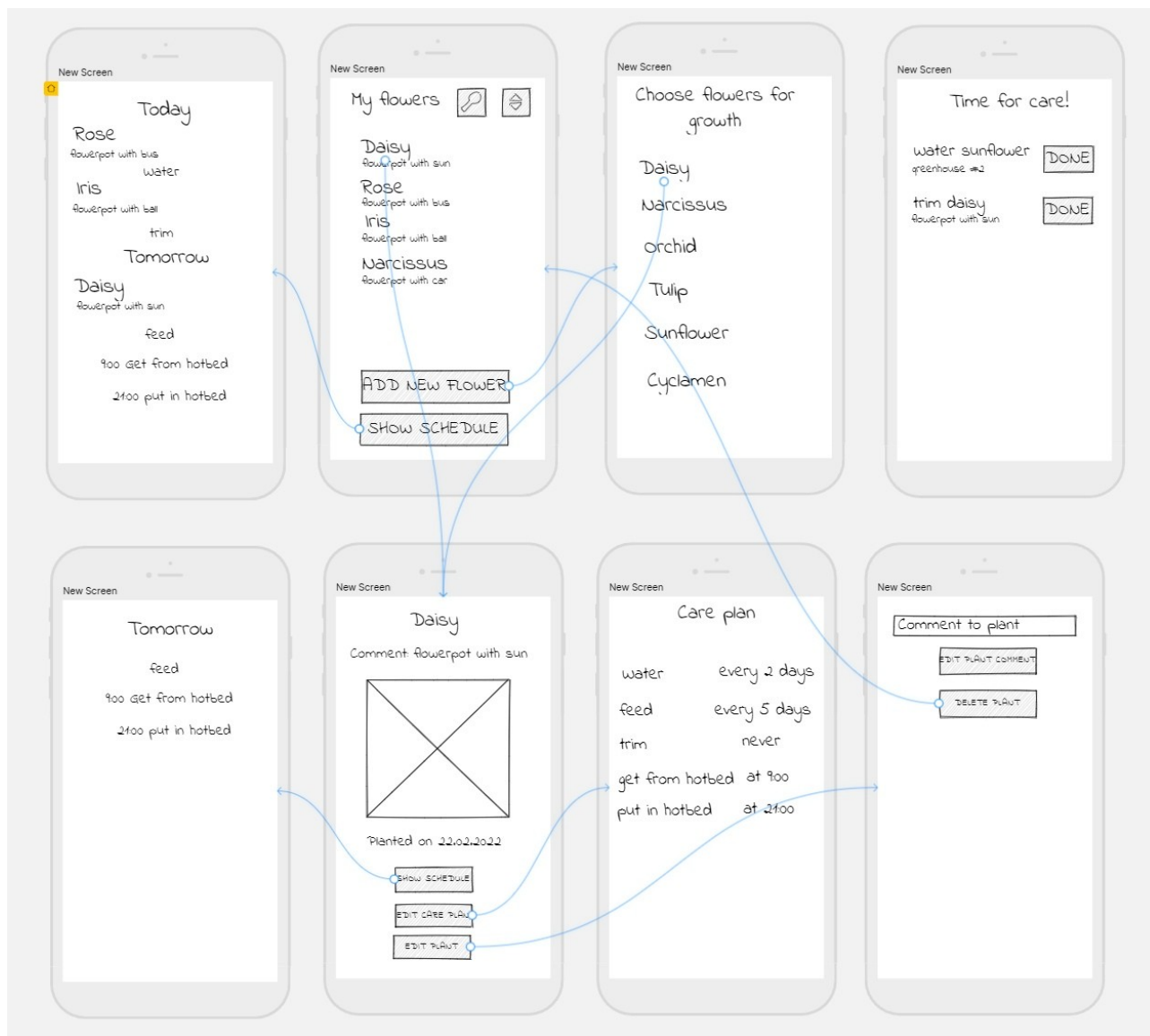


Рисунок 1 — Макет UI

### 2.2. Целевые устройства, обоснование требований и характеристики

В качестве целевых устройств были выбраны смартфоны так, как именно они доступны для использования почти в любой момент времени, а также имеют камеру и удобные средства ввода текста. Поддержка была ограничена версией 8.0 и выше, из-за требований используемых библиотек, это делает приложение доступным для 90% пользователей Android и позволяет использовать большинство возможностей ОС.

### 3. МОДЕЛЬ ДАННЫХ

Для хранения данных используется 2 таблицы — plants и tasks, их схемы представлены на рис. 2. Таблица plants используется для хранения созданных пользователем растений, таблица tasks используется для хранения действий по уходу за растениями.

plants	tasks
id: int	id: int
name: string	name: string
water: string	plantId: int
trim: string	action: string
feed: string	comment: string
getFromHotbed: string	taskDate: string
putInHotbed: string	isDone: boolean
comment: string	
createTime: string	
imagePath: string	
isHotbedUse: boolean	

Рисунок 2 — Схемы таблиц



## 4. РАЗРАБОТАННОЕ ПРИЛОЖЕНИЕ

### 4.1. Краткое описание

Приложение для выращивания рассады имеет возможности создания, просмотра и редактирования записей о растениях, а уведомляется пользователя о необходимых действиях по уходу за растениями.

### 4.2. Схема архитектуры

Архитектура приложения построена по принципу One Single Activity с Navigation Component и Fragments.

На рис. 3 представлен граф переходов между фрагментами.

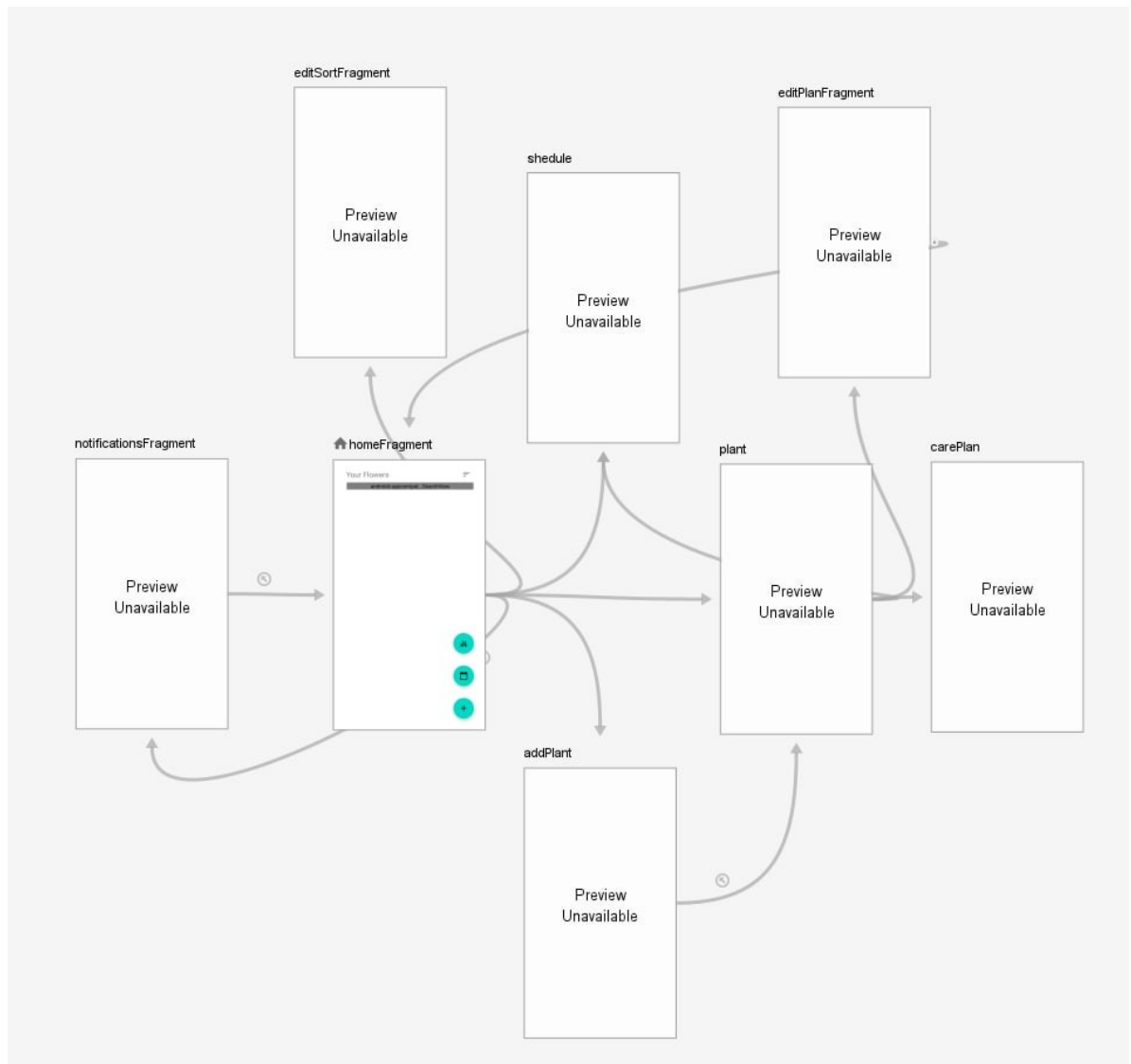


Рисунок 3 — Граф переходов между фрагментами (nav\_graph.xml)

На рис. 4 представлена архитектура работы с данными.

Entities: Plant, Task

Dao: PlantDao, TaskDao

RoomDatabase: PlantRoobDatabase

Repository: PlantRepository

ViewModel: PlantViewModel

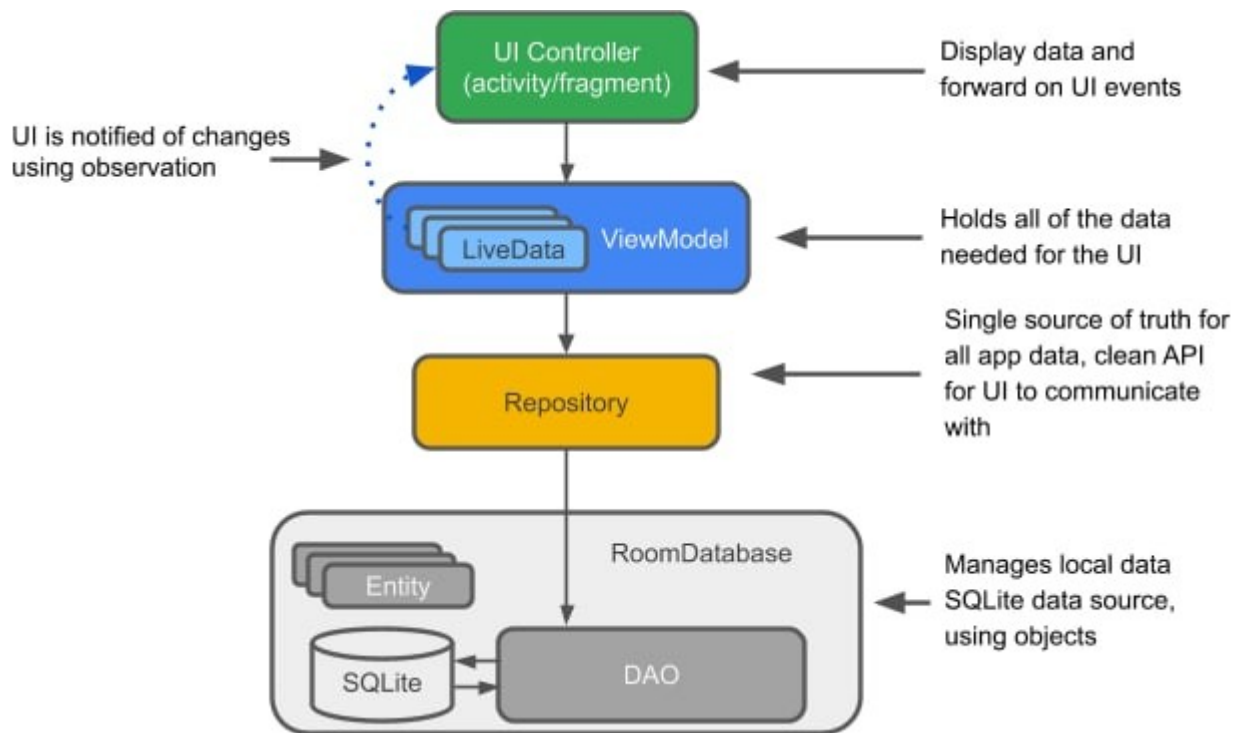


Рисунок 4 — Архитектура работы с данными

#### 4.3. Используемые технологии

- Junit
- Firebase

#### 4.4. Используемые библиотеки

- com.firebaseui:firebase-ui-firestore:7.1.1
- androidx.navigation:navigation-fragment-ktx
- androidx.navigation:navigation-ui-ktx
- androidx.room:room-ktx
- androidx.room:room-compiler

- androidx.room:room-testing
- androidx.lifecycle:lifecycle-viewmodel-ktx
- androidx.lifecycle:lifecycle-livedata-ktx
- androidx.lifecycle:lifecycle-common-java8
- org.jetbrains.kotlinx:kotlinx-coroutines-core
- org.jetbrains.kotlinx:kotlinx-coroutines-android
- com.squareup.picasso:picasso:2.71828

#### **4.5. Стратегия для обеспечения кросс-платформенности приложения**

Для обеспечения кросс-платформенности необходимо почти полностью переписать приложение, так как в текущей реализации не выделен слой логики, а работа с базой данных связана с использованием пакета `android.database.sqlite`. Так же потребуется переписать графическую составляющую, так как она реализована с помощью стандартных средств `android` и не использует каких-либо кросс-платформенных компонентов. Для разработки более кроссплатформенного приложения можно было использовать фреймворк `Flutter`.

#### **4.6. Ссылка на приложение**

<https://github.com/moevm/adfmp1h22-plant>

В приложении А и Б размещены инструкция для пользователя и снимки экрана приложения.

## 5. ПОСЛЕДОВАТЕЛЬНОСТЬ ДЕЙСТВИЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ СЦЕНАРИЕВ ИСПОЛЬЗОВАНИЯ

### 5.1. Измерение последовательности действий для осуществления сценариев использования

Были составлены изображения последовательностей взаимодействия:

1. Добавление растения — см. рис. 5
2. Просмотр списка растений — см. рис. 6
3. Редактирование растения — см. рис. 7
4. Удаление растения — см. рис. 8
5. Просмотр расписания ухода для конкретного растения — см. рис. 9
6. Просмотр расписания ухода для всех растений — см. рис. 10
7. Получение напоминания об уходе — см. рис. 11

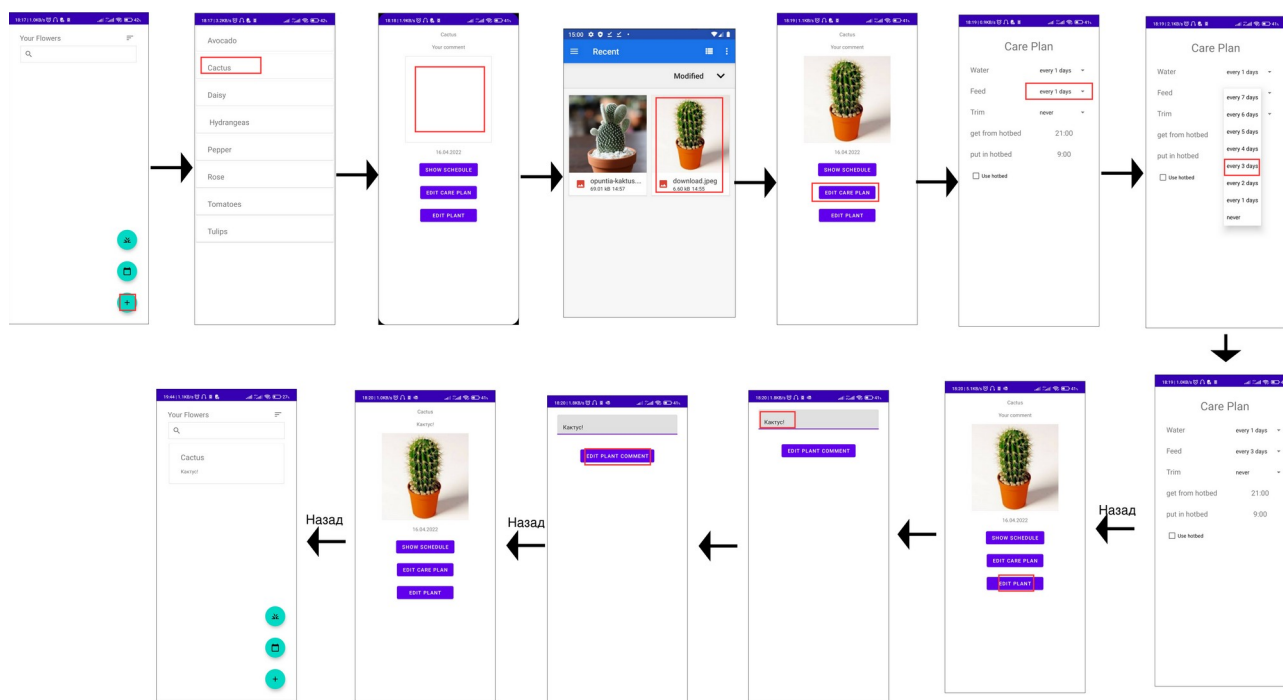


Рисунок 5 — Изображение последовательности взаимодействия 1

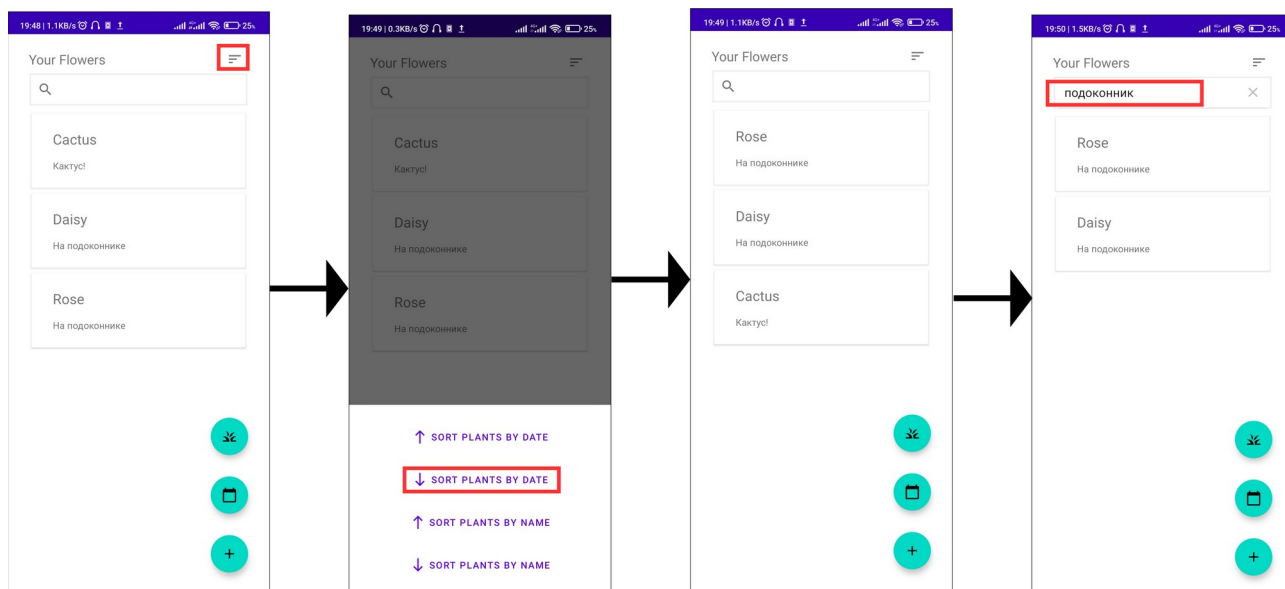


Рисунок 6 — Изображение последовательности взаимодействия 2

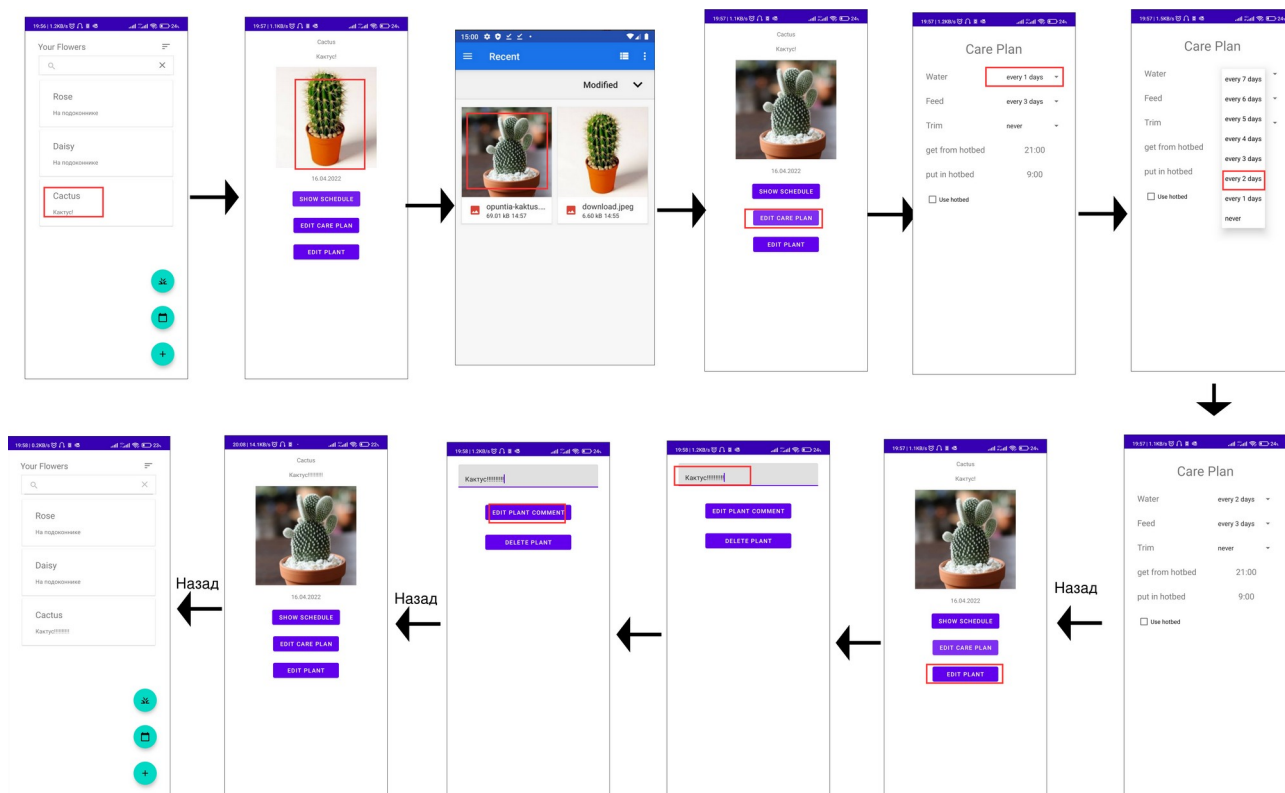


Рисунок 7 — Изображение последовательности взаимодействия 3

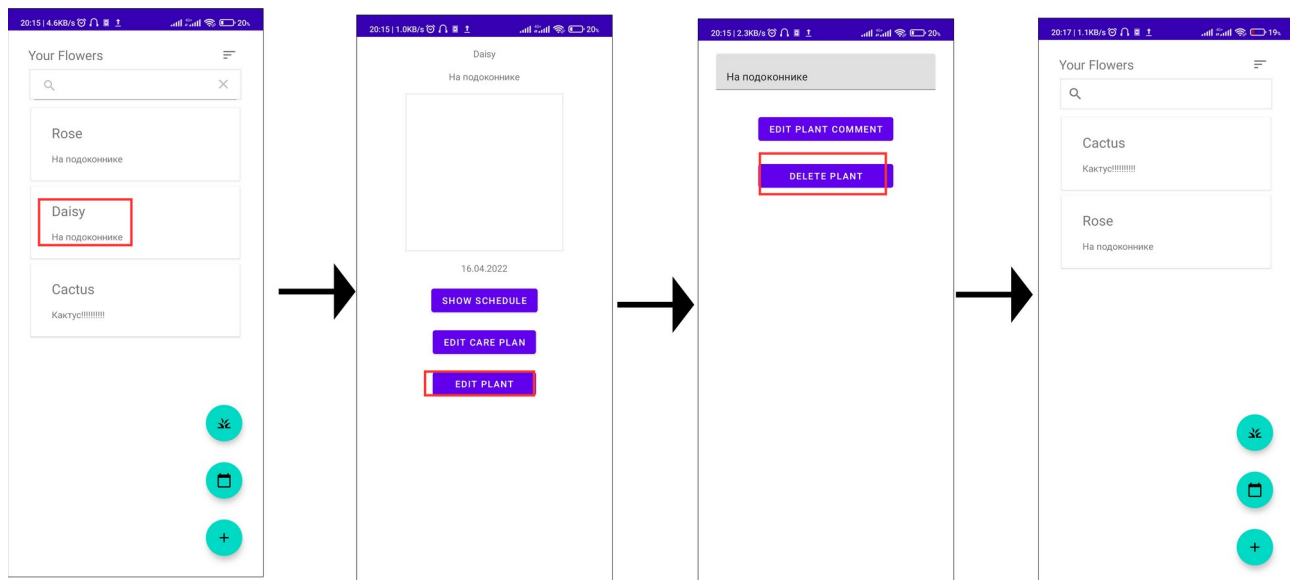


Рисунок 8 — Изображение последовательности взаимодействия 4

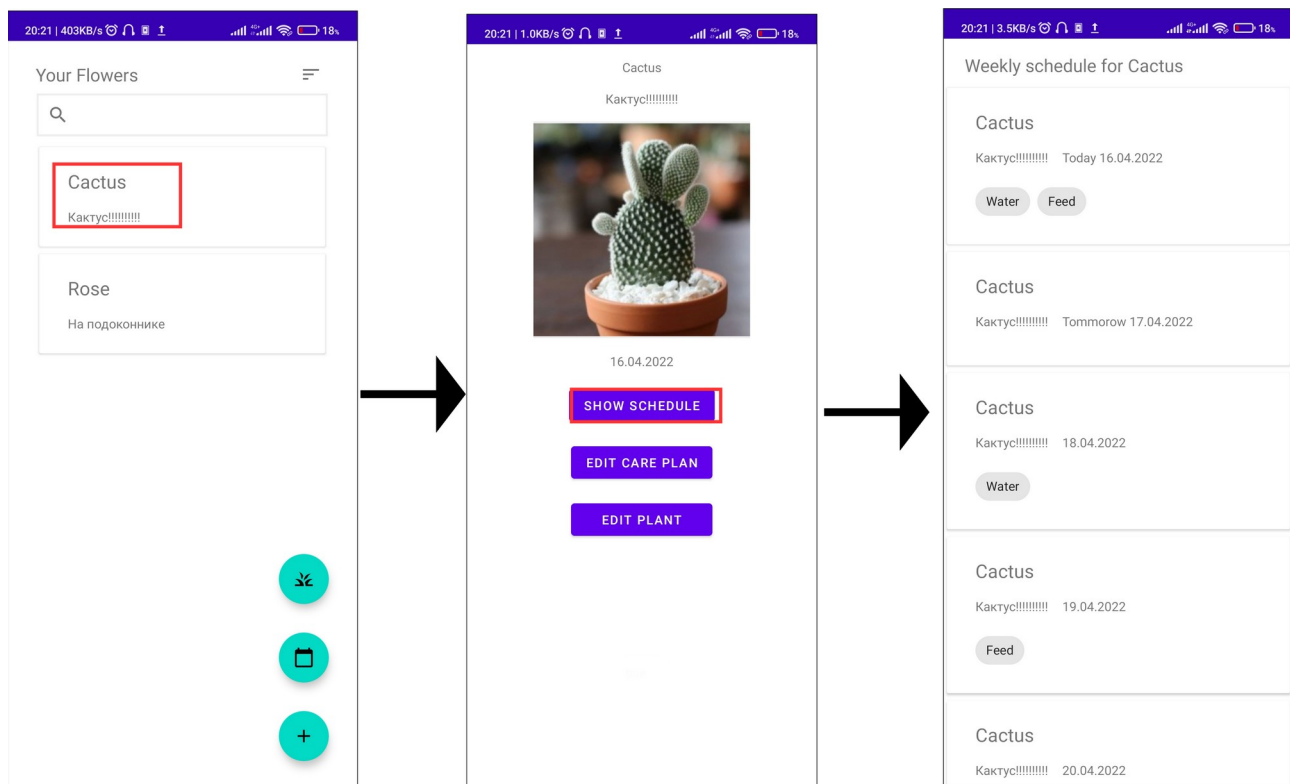


Рисунок 9 — Изображение последовательности взаимодействия 5

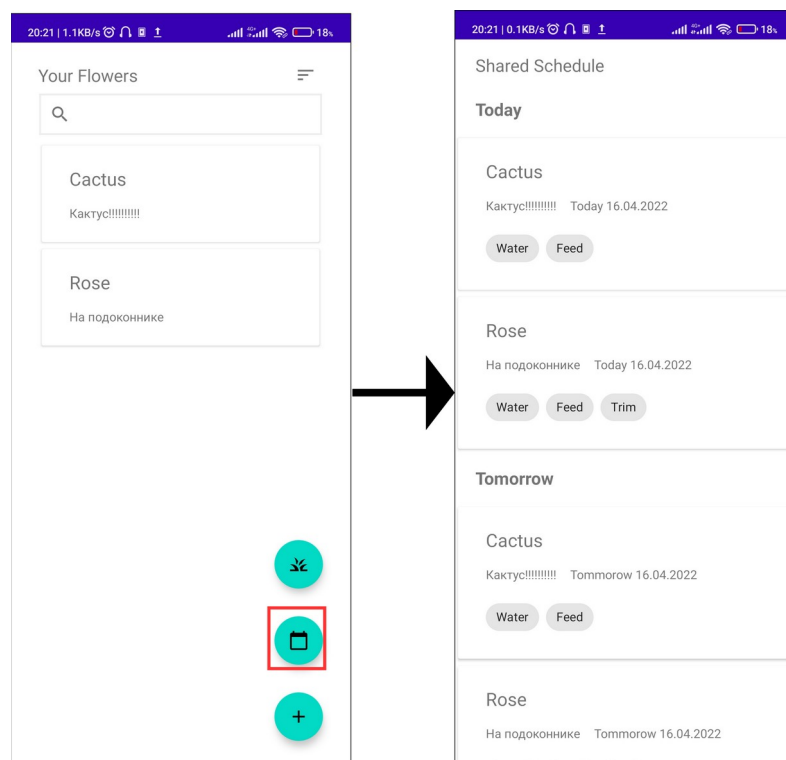


Рисунок 10 — Изображение последовательности взаимодействия 6

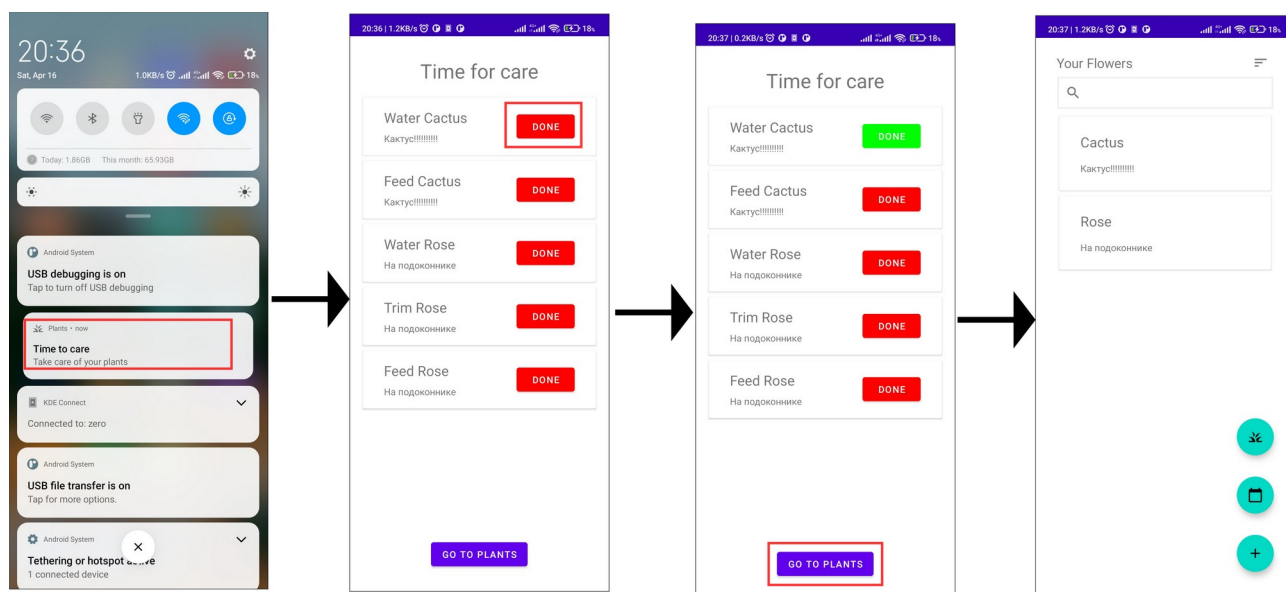


Рисунок 11 — Изображение последовательности взаимодействия 7

В таблице 1 представлено количество действий в последовательностях взаимодействия.

Таблица 1 - Количество действий в последовательностях взаимодействия

№ последовательности	Клик	Долгое нажатие	Ввод текста	Возвращение назад	Всего
1	9	0	1	3	13
2	2	0	1	0	3
3	8	0	1	3	12
4	3	0	0	0	3
5	2	0	0	0	2
6	1	0	0	0	1
7	3	0	0	0	3

## 5.2. Пути для сокращения последовательности

Упрощение изменения комментария к растению. Можно добавить возможность редактировать комментарий прямо на странице просмотра растения - т.е. сделать на странице просмотра растения поле с комментарием редактируемым. Это уменьшит количество действий необходимых для изменения комментария. На рис. 12 проиллюстрировано предлагаемое изменение.

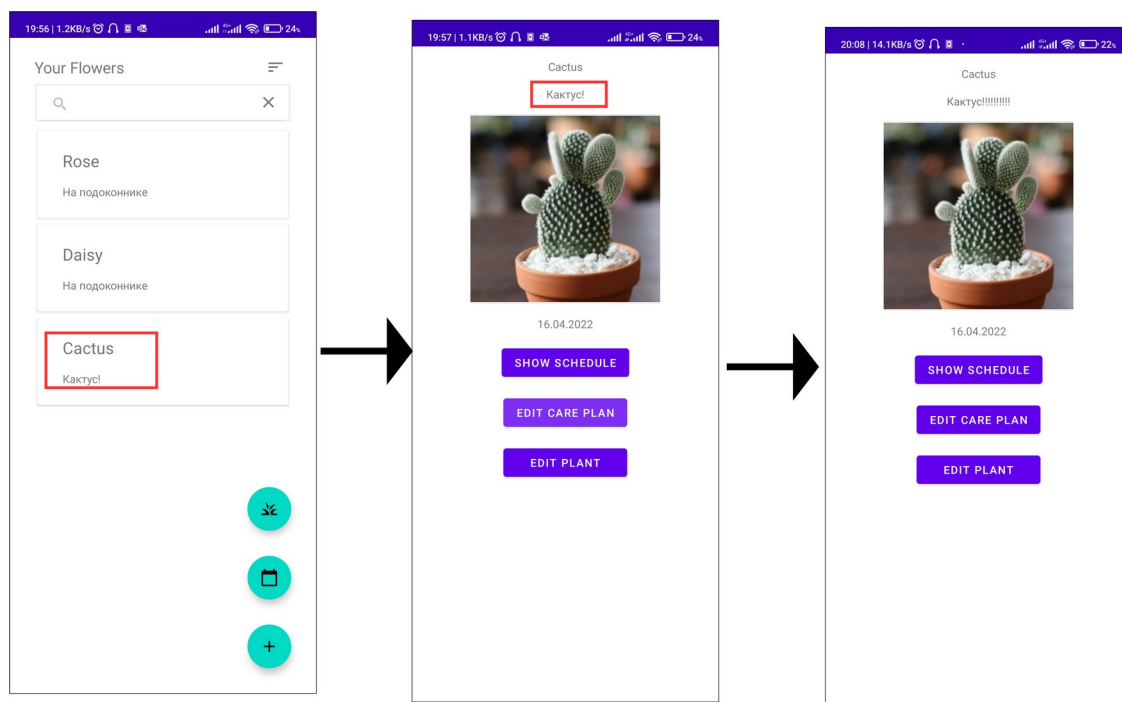


Рисунок 12 — Упрощение изменения комментария к растению



Также можно сделать удаление растения более похожим на то как обычно происходит удаление в Android - на удаляемом элементе делается долгое нажатие, появляется контекстное меню с действиями, от куда пользователь выбирает "Удалить", после чего появляется диалог с подтверждением и происходит удаление. На рис. 13 проиллюстрировано предлагаемое изменение.

На рис. 12 проиллюстрировано предлагаемое изменение.

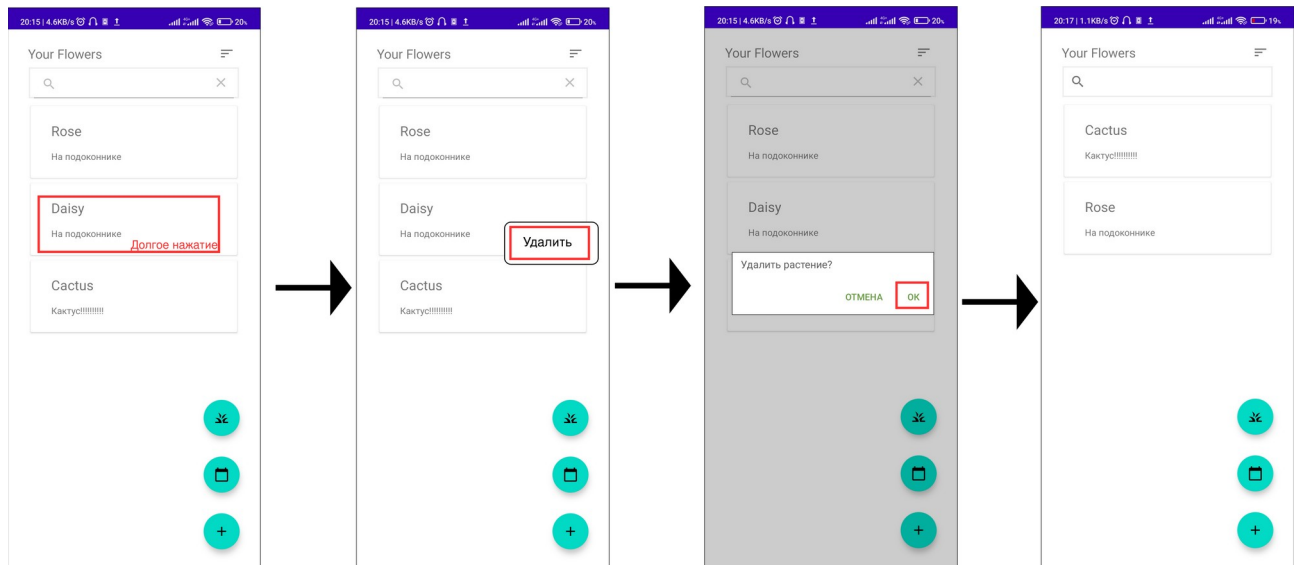


Рисунок 13 — Упрощение удаления растения

## **6. ДОПОЛНИТЕЛЬНЫЕ ЗАДАНИЯ**

Были выполнены 2 дополнительных задания: создание интеграционных тестов для приложения, материалы для публикации.

Интеграционное тестирование проводилось с помощью фреймворка Espresso. Тесты проверяют работу приложения преимущественно через его UI.

Материалы для публикации были размещены в каталог `play_market_publication`: описание, иконки, скриншоты.

## **ЗАКЛЮЧЕНИЕ**

В ходе выполнения работы было реализовано приложение для выращивания рассады. В нем доступно создание, просмотр и редактирование записей о растениях, приложение уведомляет пользователя о необходимых действиях по уходу за растениями, а также предоставляет возможность посмотреть календарь с действиями по уходу.

На текущий момент приложение позволяет выбрать фотографию растения только из галереи, т. е. не позволяет сделать фотографию. Также в ходе работы обнаружилось, что в готовом приложении есть возможности для сокращения количества нажатий пользователя в частых сценариях использования.

Будущее развитие приложения может заключаться в устранении недостатков описанных выше, а также размещение приложения в play market.

## **ИСПОЛЬЗОВАННЫЕ МАТЕРИАЛЫ**

1. Документация Kotlin URL: <https://kotlinlang.org/docs/home.html>
2. Документация Android URL: <https://developer.android.com/docs>
3. Репозиторий приложения URL: <https://github.com/moevm/adfmp1h22-plant>

## ПРИЛОЖЕНИЕ А. ИНСТРУКЦИЯ ДЛЯ ПОЛЬЗОВАТЕЛЯ

1. Для добавления растения необходимо на главном экране нажать кнопку с иконкой плюса, выбрать интересующий вид растения, затем выбрать фотографию для растения, нажав на большой пустой квадрат (placeholder для фотографии), далее нажав на кнопку edit care plan можно отредактировать план ухода за растением, а нажав на edit plant можно установить комментарий к растению. Добавление растения можно увидеть на рис. 5.

2. Просмотреть список растений можно на главном экране. Для сортировки растений можно использовать значек в верхнем правом углу. Для поиска растения по названию вида или по комментарию можно использовать поле для ввода в верхней части экрана. Просмотр списка растений можно увидеть на рис. 6.

3. Для редактирования растения необходимо на главном экране выбрать растение, кликнув по нему, затем можно выбрать новую фотографию для растения, нажав на текущую фотографию, далее нажав на кнопку edit care plan можно отредактировать план ухода за растением, а нажав на edit plant можно установить комментарий к растению. Редактирование растения можно увидеть на рис. 7.

4. Для удаления растения необходимо выбрать растение на главном экране (нажать по нему), затем нужно нажать кнопку edit plant и нажать на кнопку delete plant. Удаление растения можно увидеть на рис. 8.

5. Для того чтобы увидеть расписание ухода для конкретного растения нужно выбрать растение на главном экране (нажать по нему), затем нужно нажать на кнопку show schedule. Просмотр расписания ухода для конкретного растения можно увидеть на рис. 9.

6. Для того чтобы увидеть расписание ухода для всех растений нужно на главном экране нажать круглую кнопку с изображением календаря. Просмотр расписания ухода для всех растений можно увидеть на рис. 10.

7. После добавления растений, пользователю будут приходить напоминания о действиях по уходу за растениями. При получении уведомления нужно на него нажать, откроется окно приложения, в котором нужно по мере выполнения действий по уходу нужно нажимать кнопку Done для нужного растения и действия. Как выглядит работа с уведомлениями можно увидеть на рис. 11.

## ПРИЛОЖЕНИЕ Б. СНИМКИ ЭКРАНА ПРИЛОЖЕНИЯ

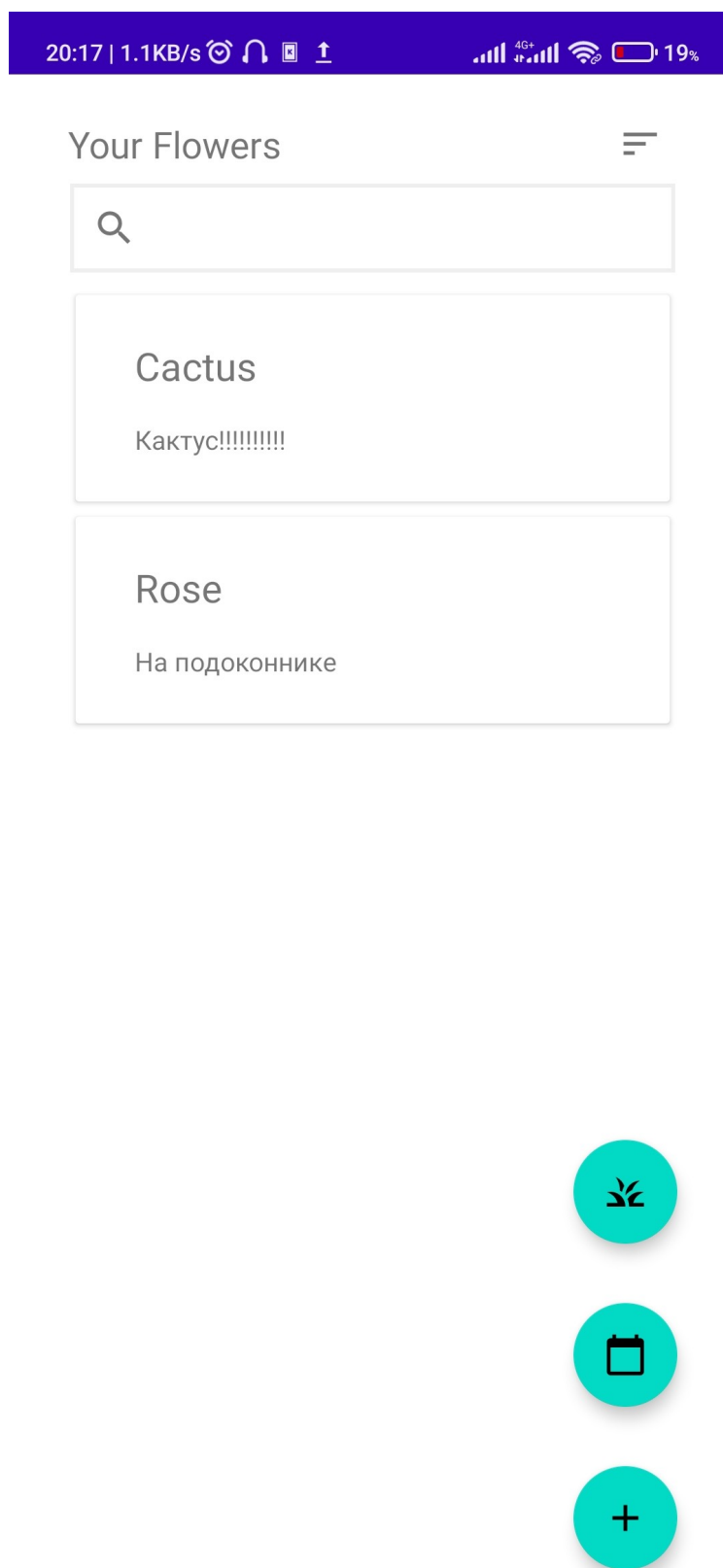


Рисунок 14 — Главный экран приложения

Cactus

Кактус!



16.04.2022

SHOW SCHEDULE

EDIT CARE PLAN

EDIT PLANT

Рисунок 15 — Экран просмотра растения



## Care Plan

Water every 2 days ▼

Feed every 3 days ▼

Trim never ▼

get from hotbed 21:00

put in hotbed 9:00

☐ Use hotbed

Рисунок 16 — Экран редактирования плана ухода за растением

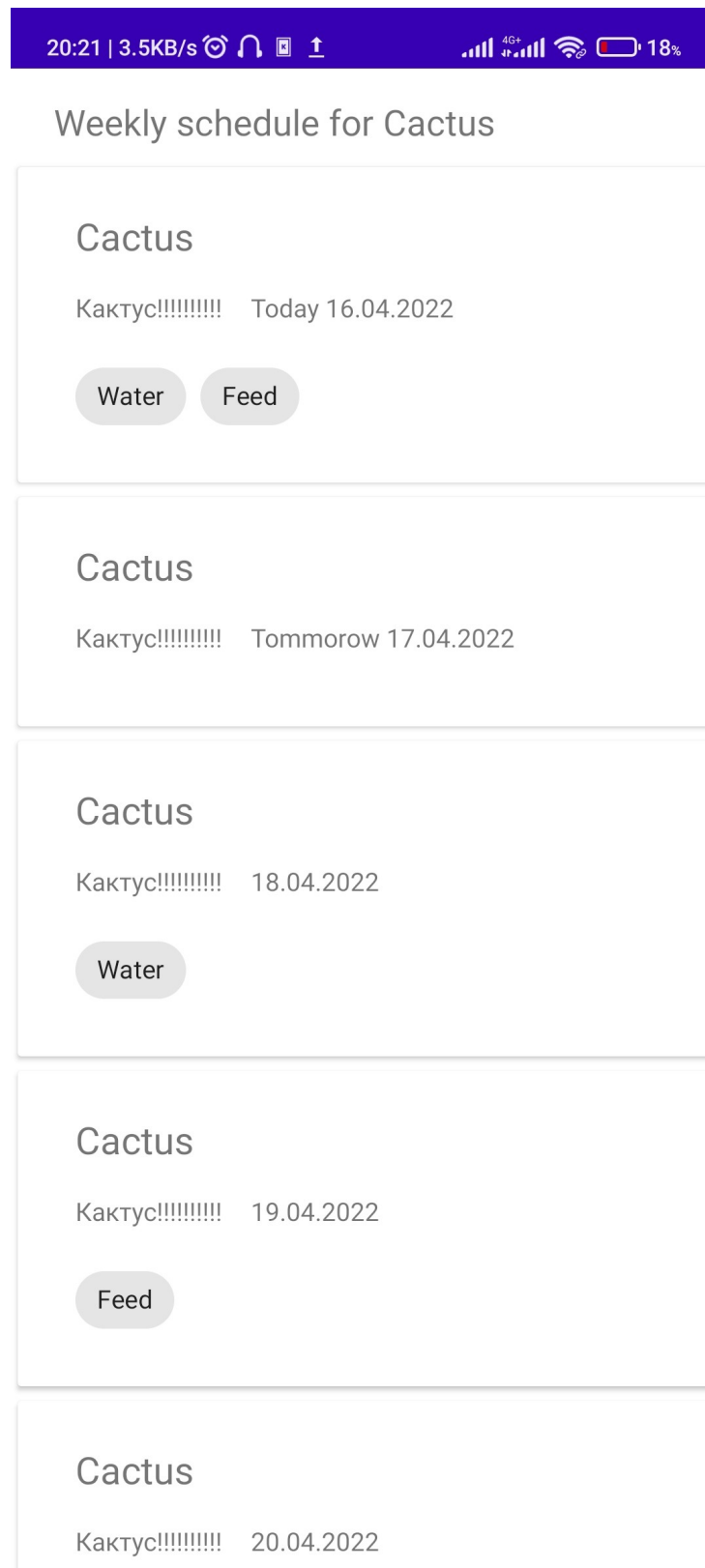


Рисунок 17 — Экран просмотра календаря ухода за растением

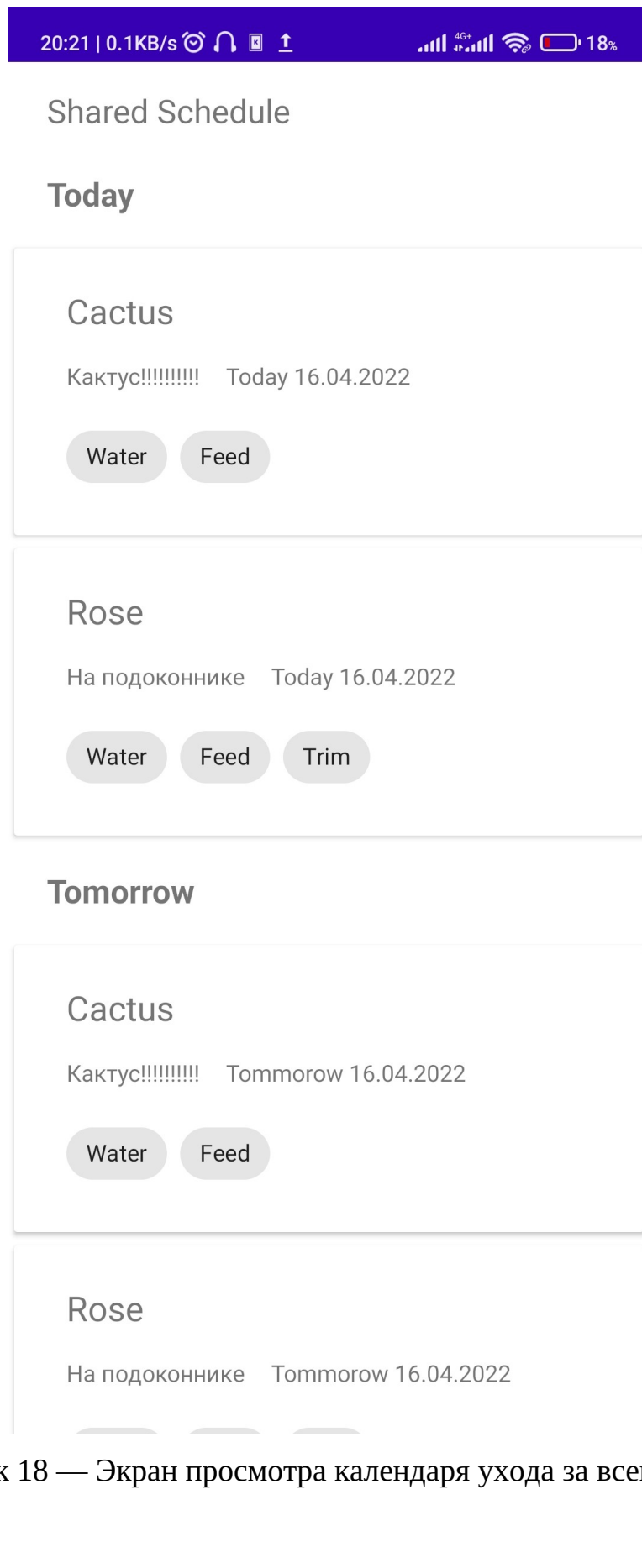


Рисунок 18 — Экран просмотра календаря ухода за всеми растениями

## Time for care

Water Cactus

Кактус!!!!!!!!!!

DONE

Feed Cactus

Кактус!!!!!!!!!!

DONE

Water Rose

На подоконнике

DONE

Trim Rose

На подоконнике

DONE

Feed Rose

На подоконнике

DONE

GO TO PLANTS

Рисунок 19 — Экран, на который перенаправляет уведомление

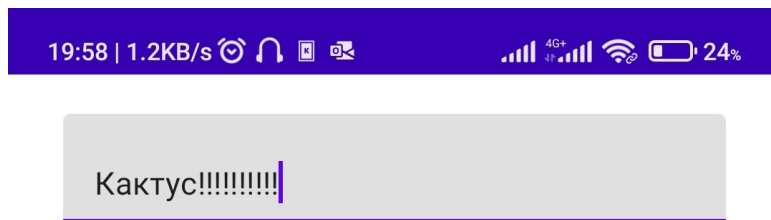


Рисунок 20 — Экран редактирования растения