

Пару слов о документе

Данное ТЗ является сборником артефактов, которые я когда-то продумывал во время первых попыток реализации проекта.

Цели проекта

Конечная цель проекта - создание мобильного приложения трекера, которое позволяет пользователю оптимизировать процесс ухода за домашними растениями.

Основные направления реализации

1. **Создания фронтэнд части.** То есть создание непосредственно самого приложения и функциональности для пользователей на платформах android и iOS.
2. **Создания бекэнд части.** Создание баз данных: цветов, аккаунтов, создание особого функционала для подписок, добавление возможности оплаты.
3. **Наполнение базы данных.** В соответствии с ботаникой и особенностями ухода за домашними растениями, необходимо заполнить базы данных уникальной информацией.
4. **Создание информационных ресурсов.** Необходимо подготовить площадку из аккаунтов в соц сетях, чтобы в будущем рекламировать проект и поддерживать связь с аудиторией.
5. **Бизнес аналитика (если это она).** Несмотря на то, что первичный анализ конкурентов уже был произведен, необходимо провести его повторно, анализируя также запросы ЦА, возможность монетизации и будущего масштабирования.
6. **Работа с нейросетями.** Потенциальная возможность определения вида растения по фотографии.

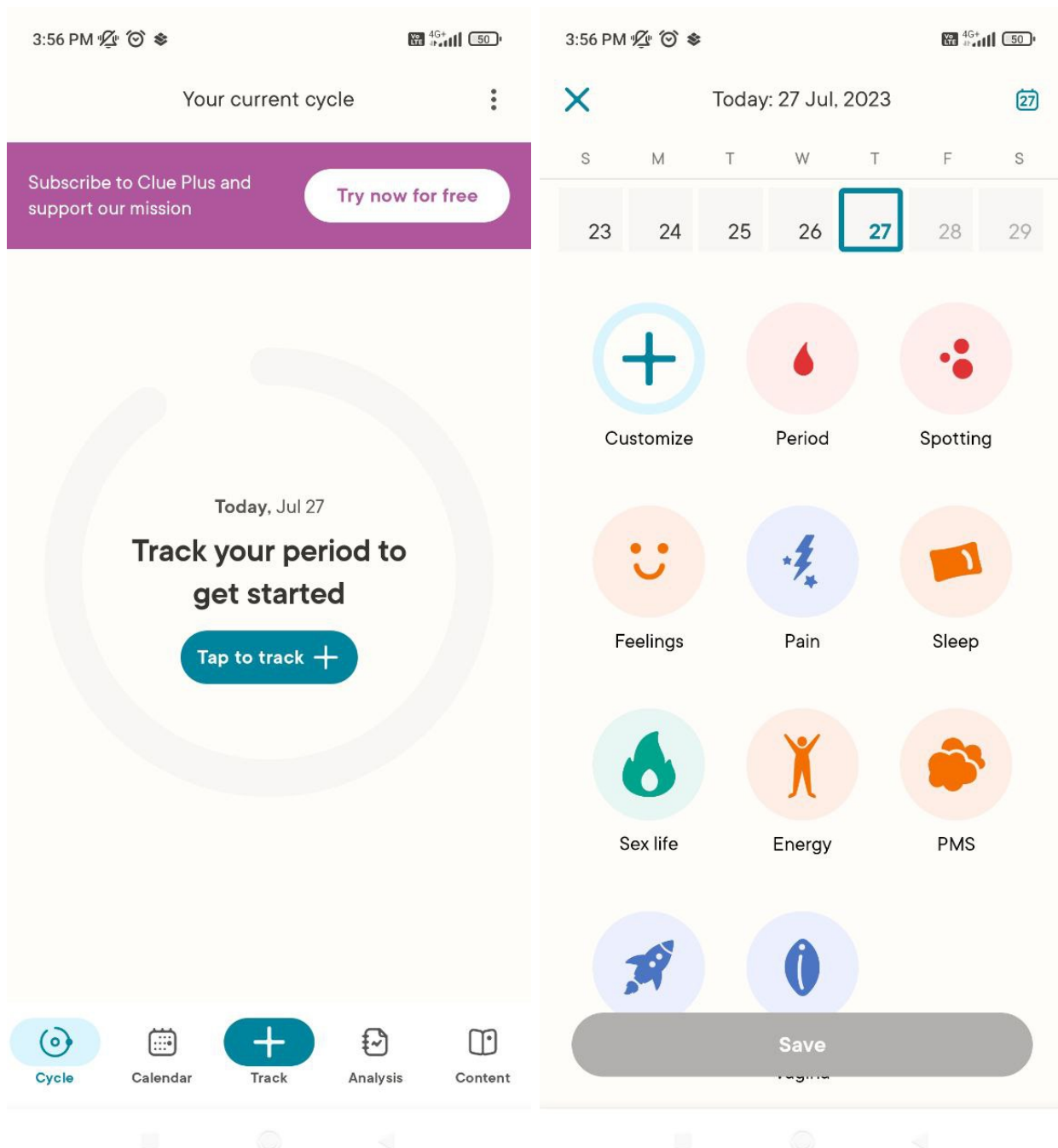
UI и UX

В данном пункте описываются всевозможные алгоритмы взаимодействия пользователя с приложением. На данный момент основной дизайн уже закончен и данная информация не очень актуальна, но я решил ее оставить.

Алгоритм добавления растения

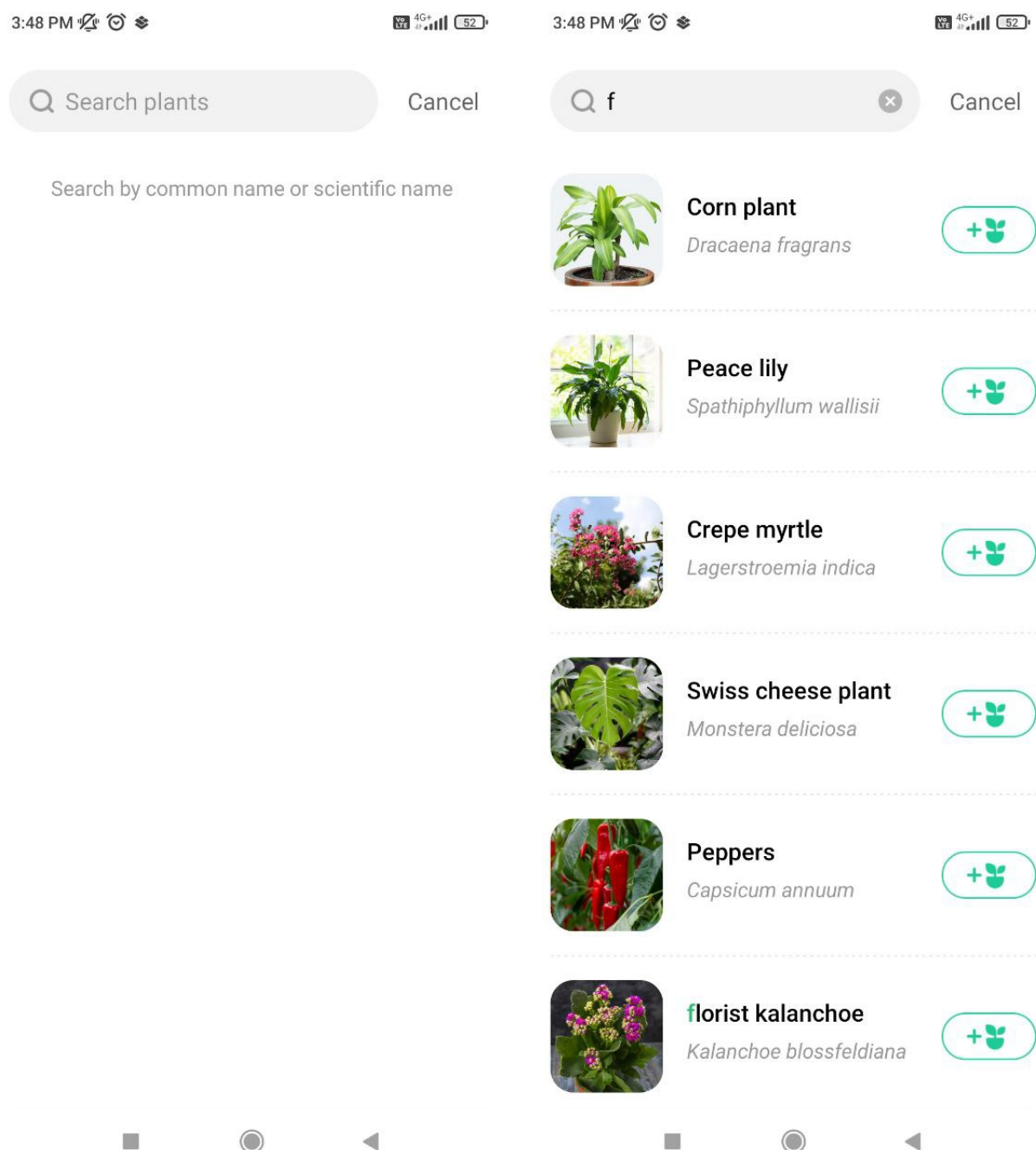
На данном шаге пользователь добавляет к себе в хранилище новый цветок, для этого он должен:

1. Нажать на кнопку добавления.



Пример того, как при нажатии кнопки посередине открывается окно

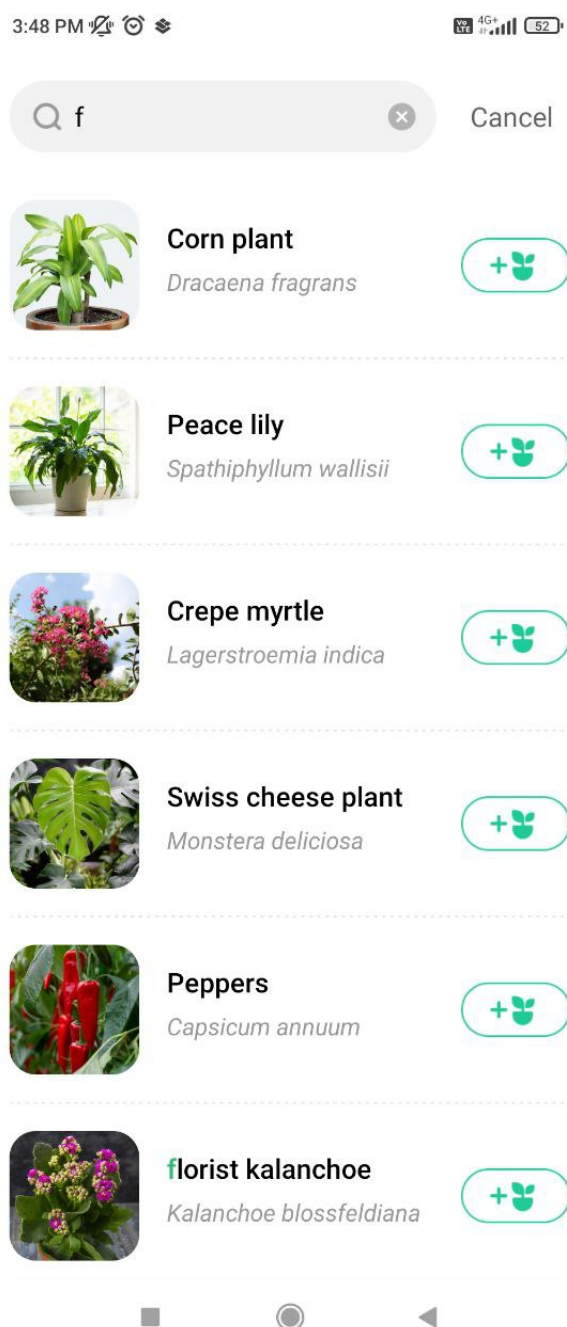
2. В открывшемся окне пользователь вводит название своего растения. По мере ввода, ему предлагаются соответствующие названию растения



Пример того, как устроена система поиска

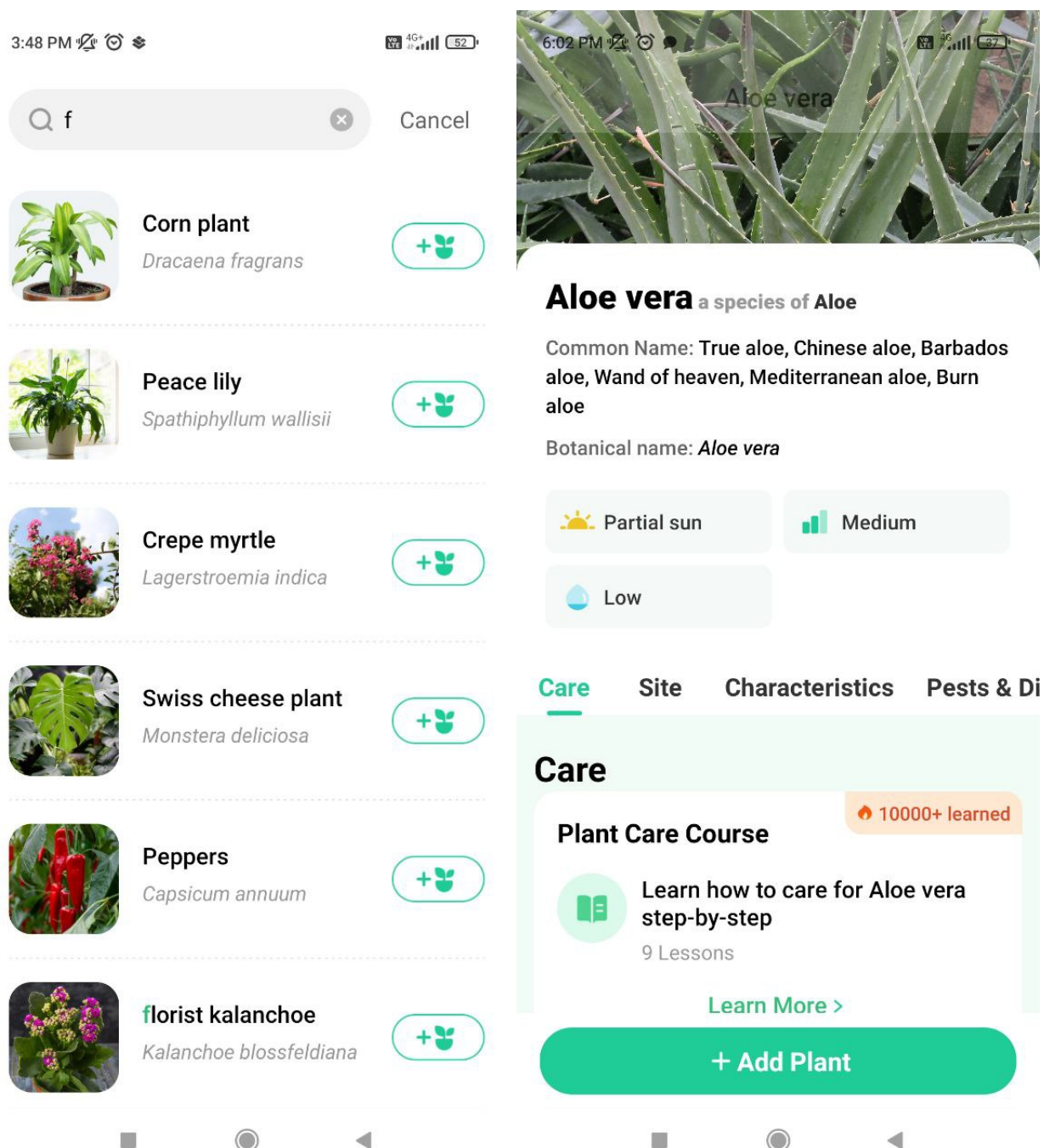
*(На первой вкладке, в рекомендациях, можно добавить наиболее частые растения, чтобы окно не было таким пустым)

3. После того, как пользователь нашел необходимое растение, он может сразу его добавить в свою коллекцию через соответствующую кнопку в строке.



Пример реализации добавления

4. Либо он может открыть вкладку с подробным описанием растения и добавить его в свою коллекцию (Пользователь также должен иметь возможность вернуться обратно на страницу поиска).

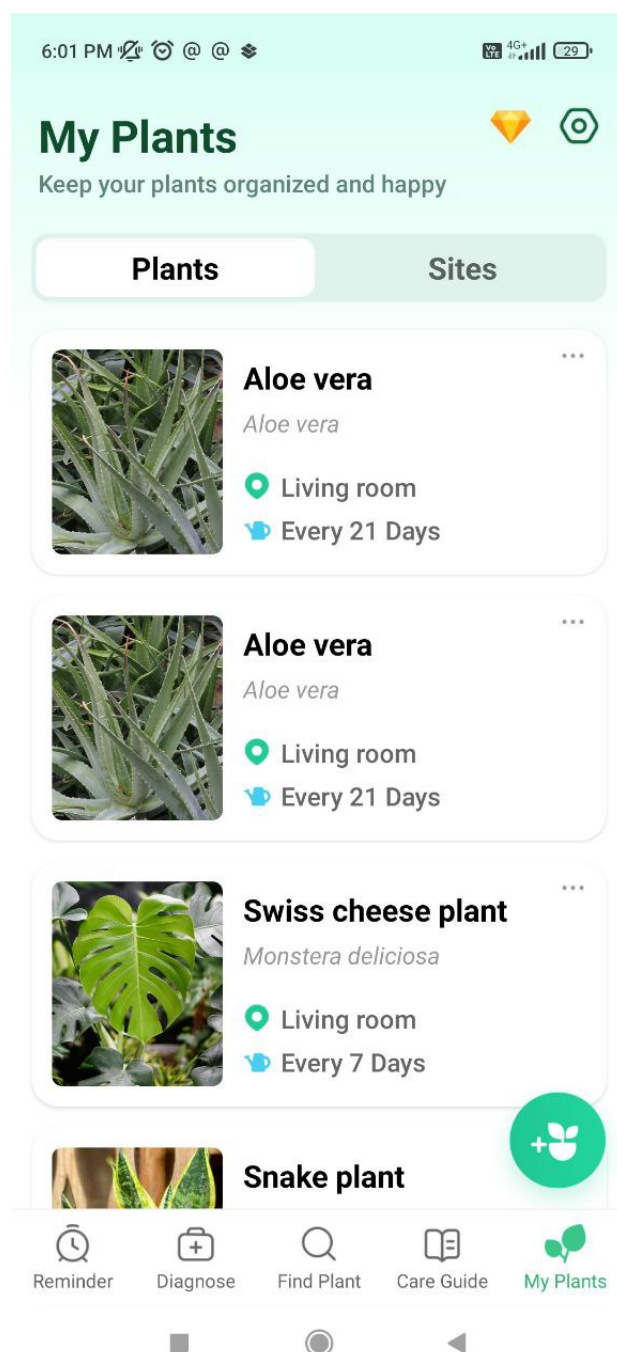


Пример вкладки с описанием растения

Алгоритм удаления растения.

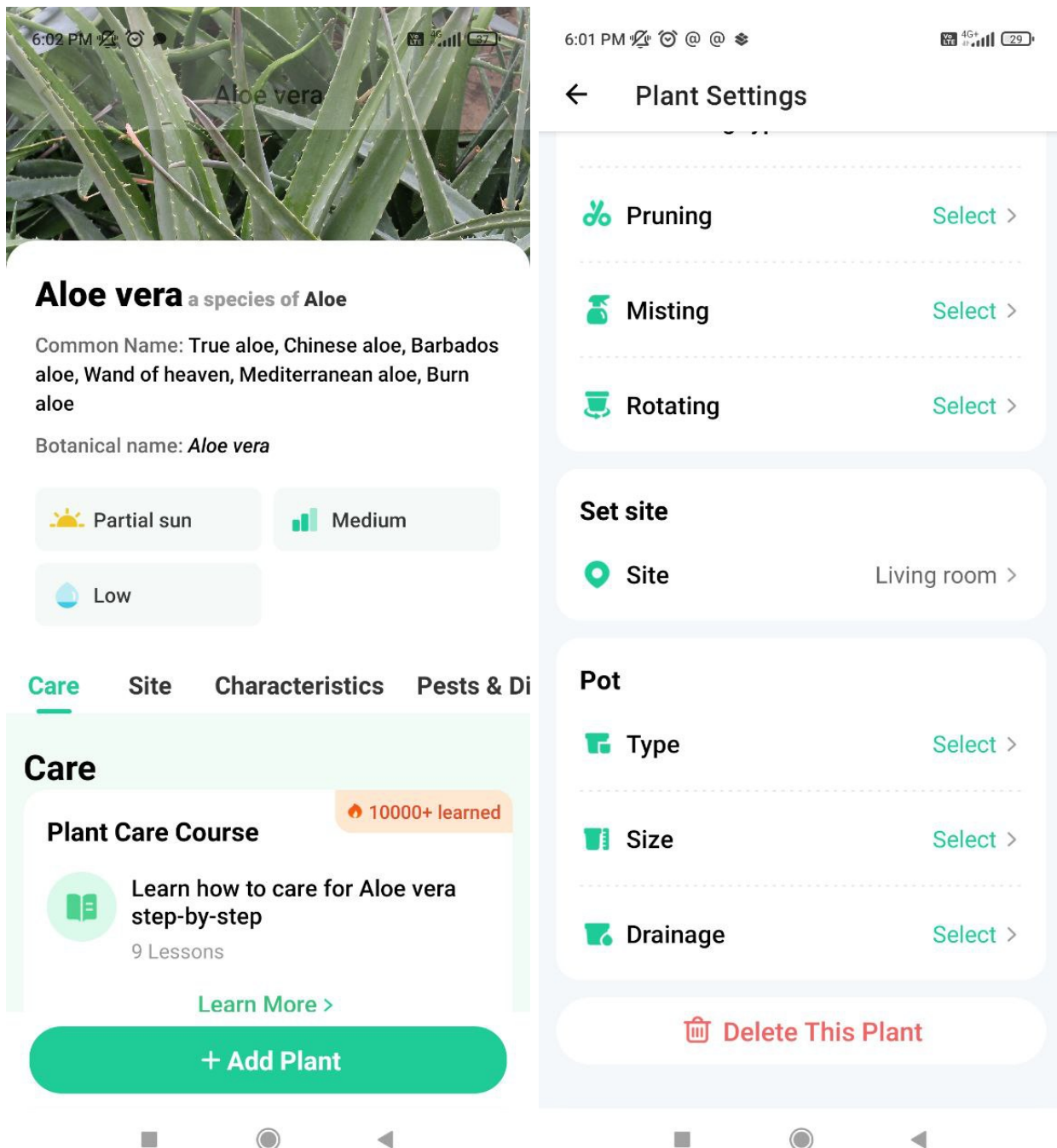
1. Удаление растения через коллекцию:

(a) Нажать на кнопку "Мои растения"



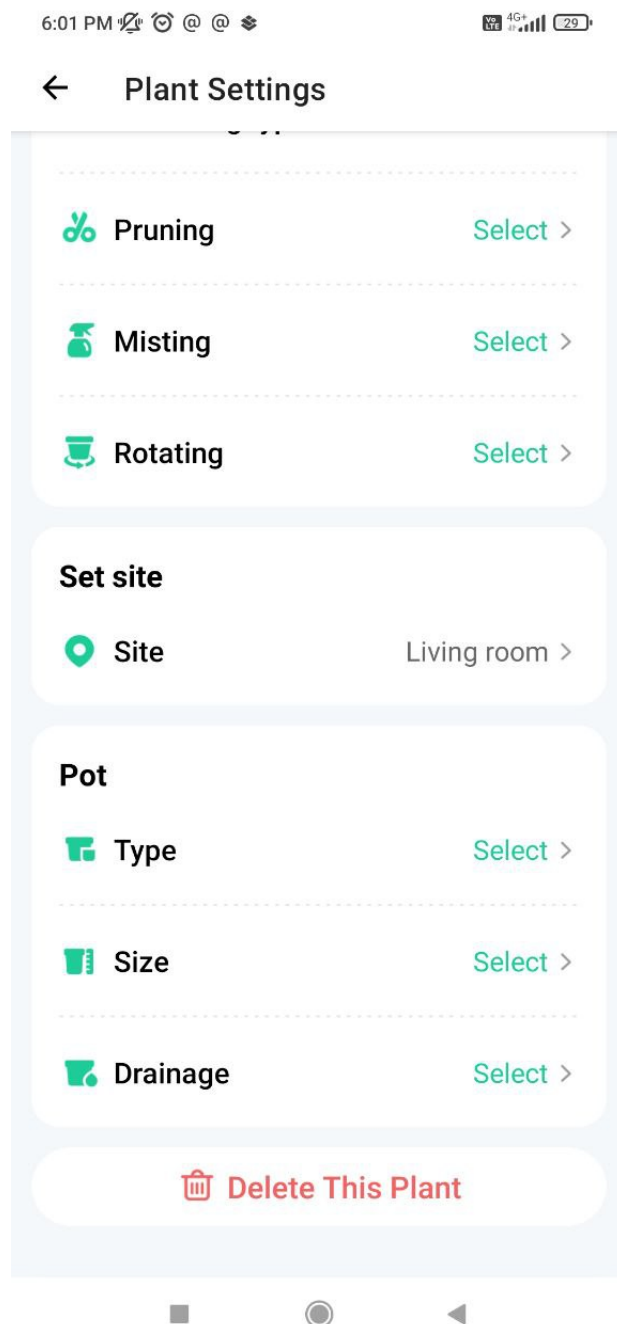
Пример того, как при нажатии кнопки справа открывается окно

- (b) В открывшемся окне пользователь нажимает на цветок, который он хочет удалить, после чего открывается специальное меню.



Пример того, как устроено меню

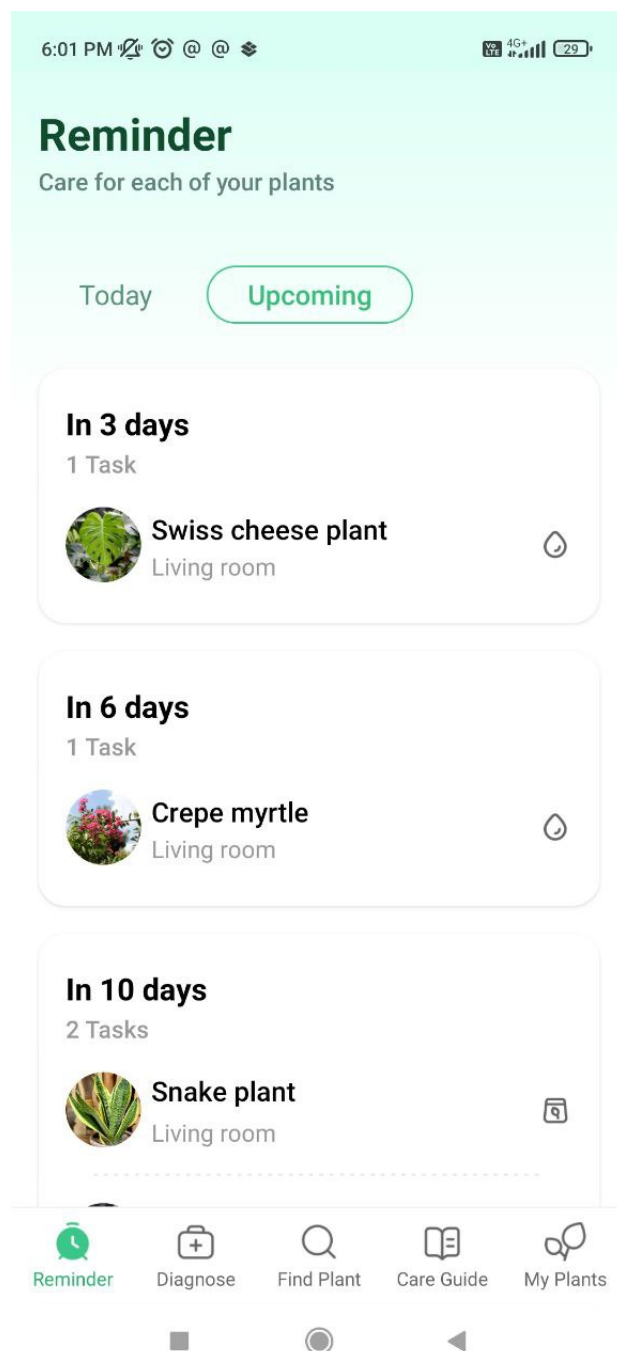
- (с) В открывшемся меню, в самом конце страницы, пользователь может нажать на кнопку удаления.



Пример реализации удаления

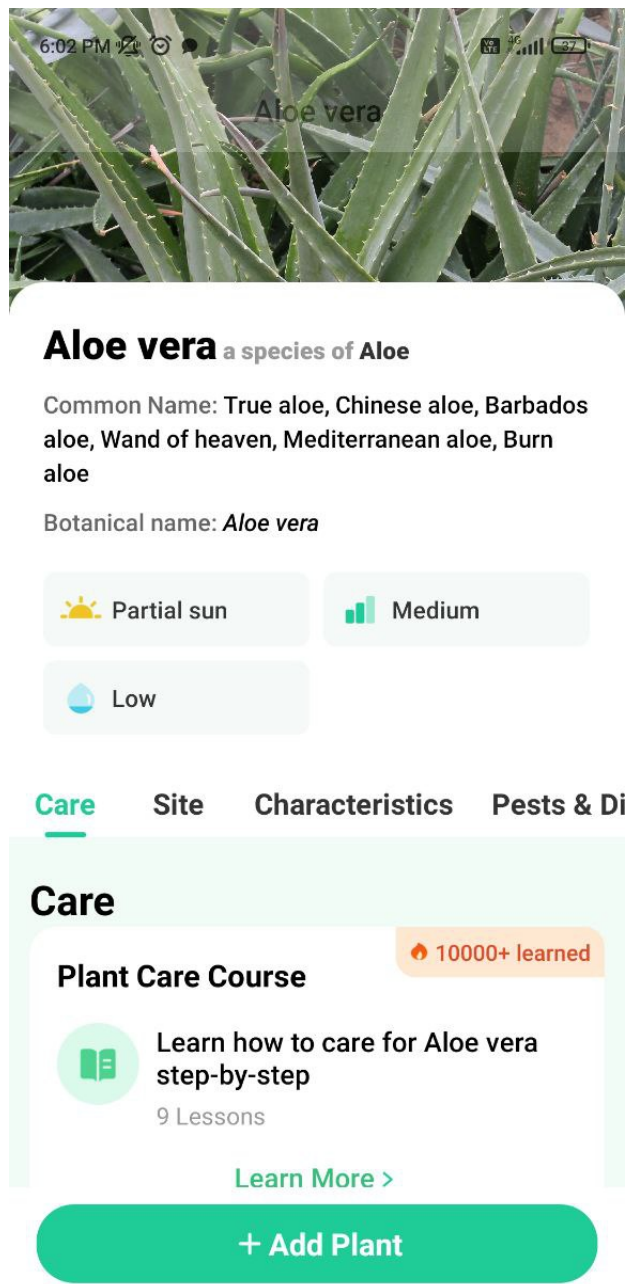
2. Удаление растения через напоминание:

(a) Нажать на кнопку "Календарь"



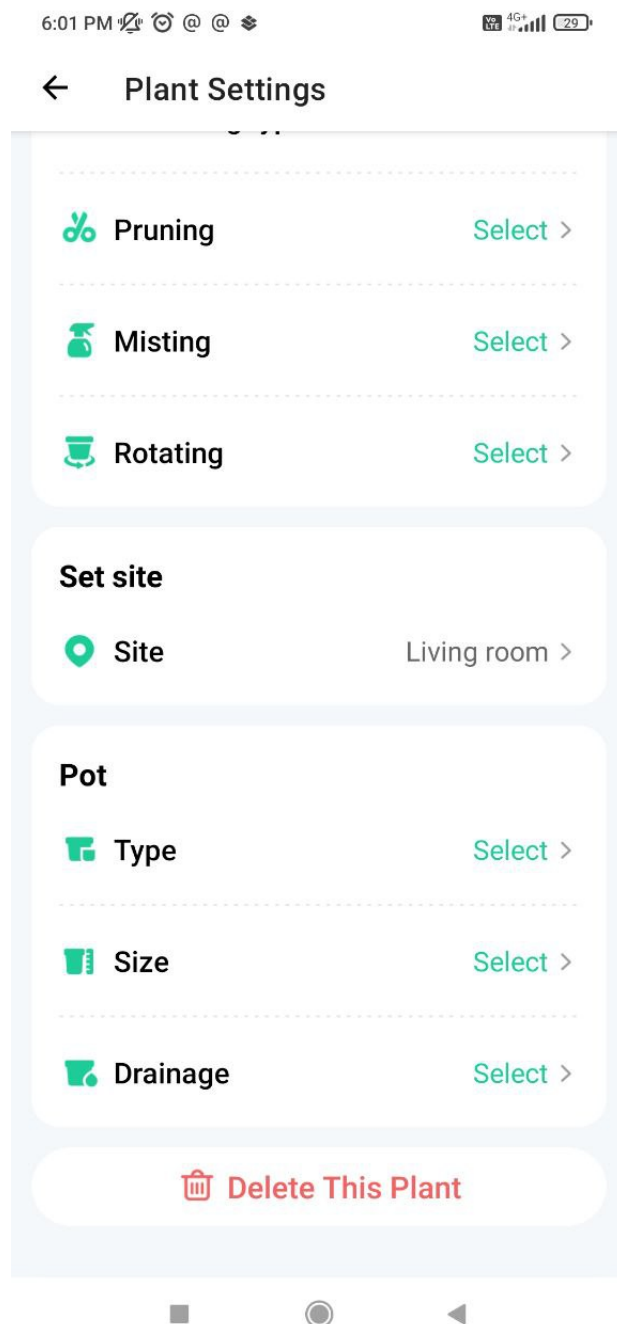
Пример того, как при нажатии открывается календарь

- (b) В открывшемся окне пользователь нажимает на цветок, который он хочет удалить, после чего открывается специальное меню.



Пример того, как устроена система поиска

- (с) В открывшемся меню, в самом конце страницы, пользователь может нажать на кнопку удаления.

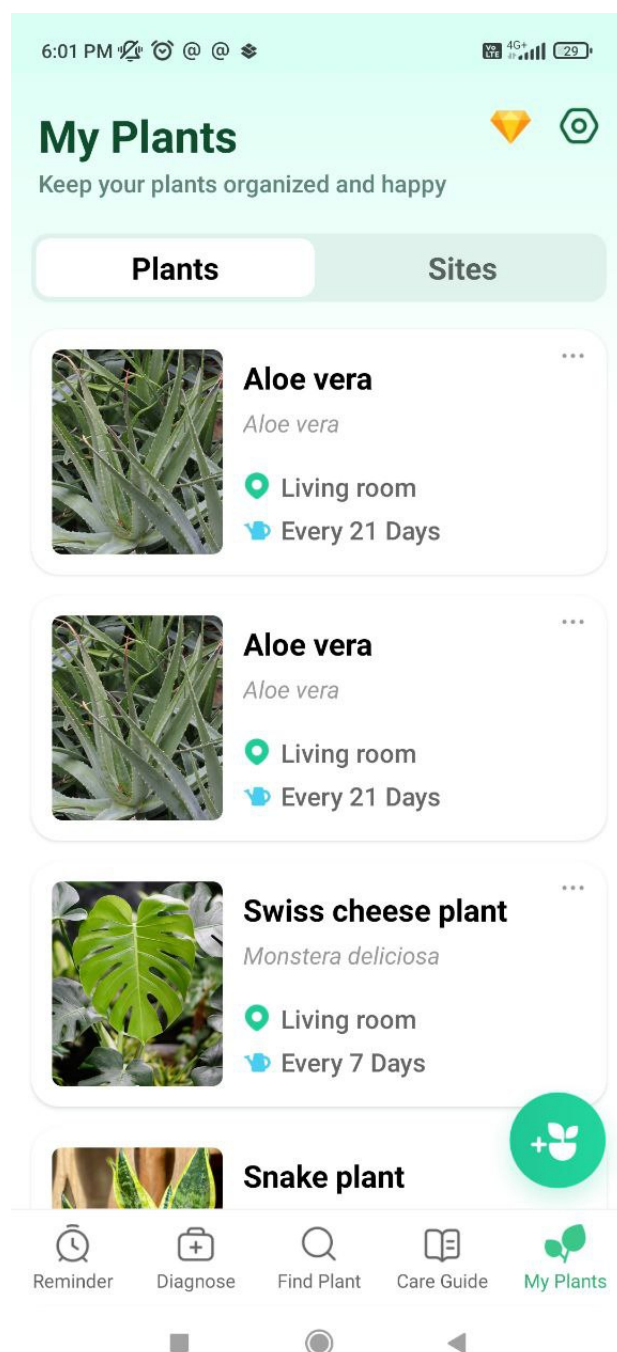


Пример реализации добавления

Алгоритм обновления описания растения.

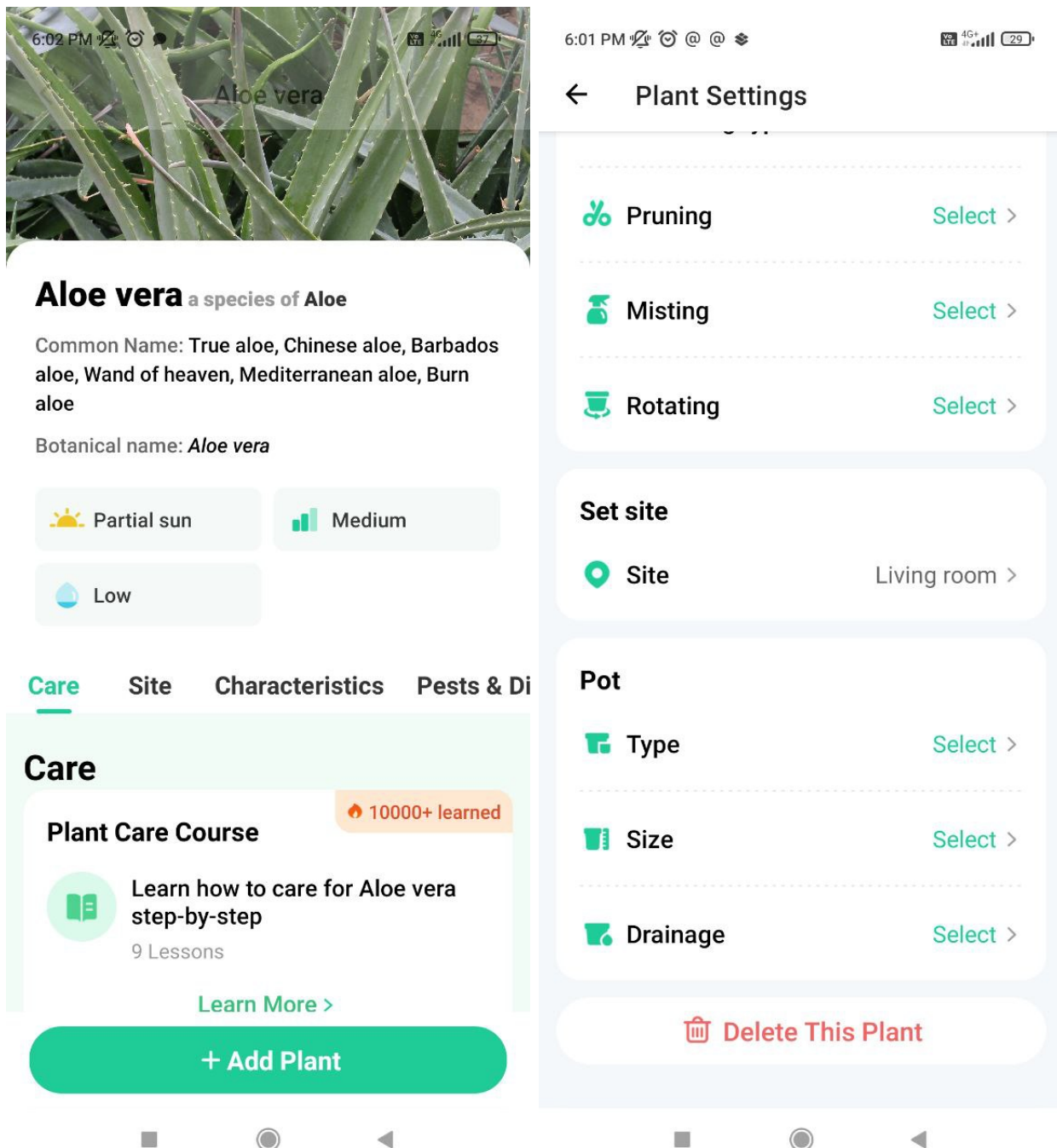
1. Удаление растения через коллекцию:

(a) Нажать на кнопку "Мои растения"



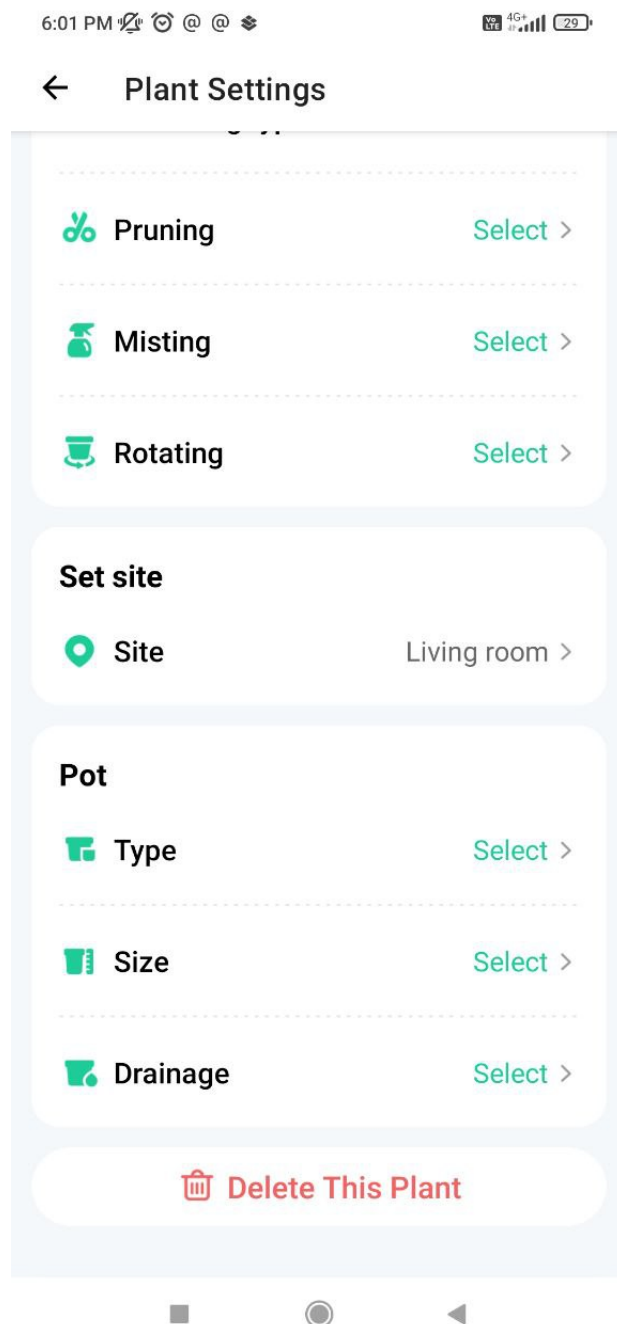
Пример того, как при нажатии кнопки справа открывается окно

- (b) В открывшемся окне пользователь нажимает на цветок, который он хочет удалить, после чего открывается специальное меню.



Пример того, как устроено меню

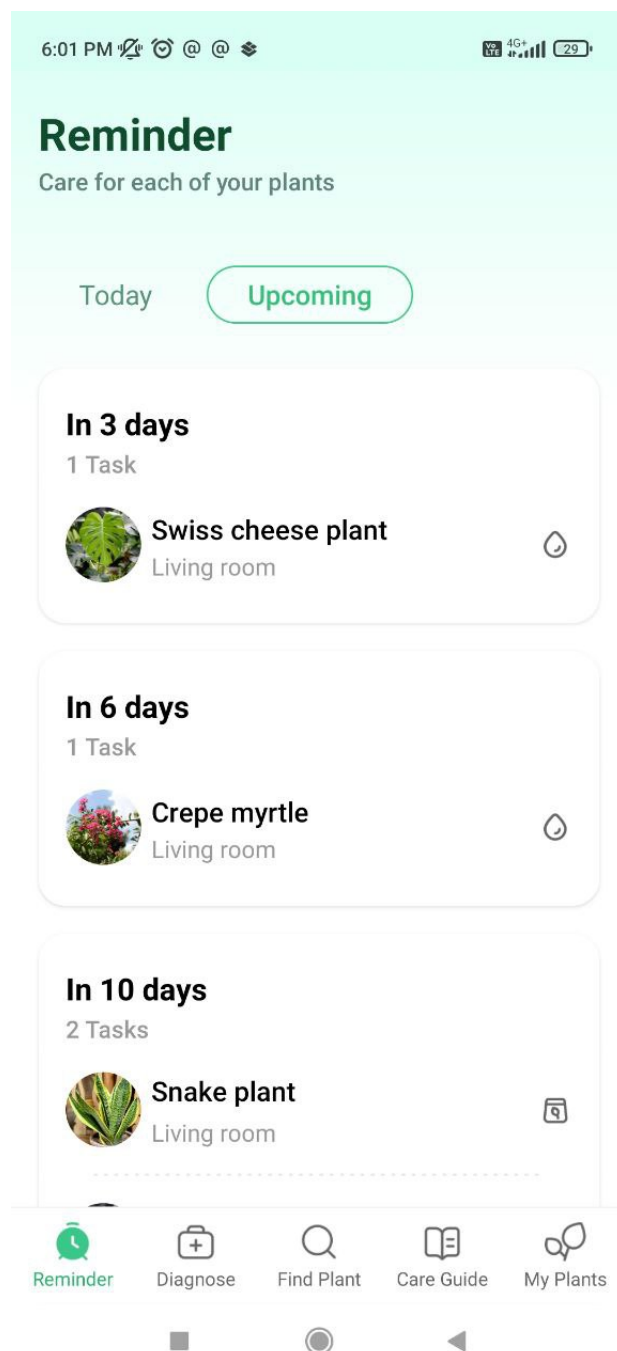
- (с) В открывшемся меню, в самом конце страницы, пользователь может нажать на кнопку удаления.



Пример реализации удаления

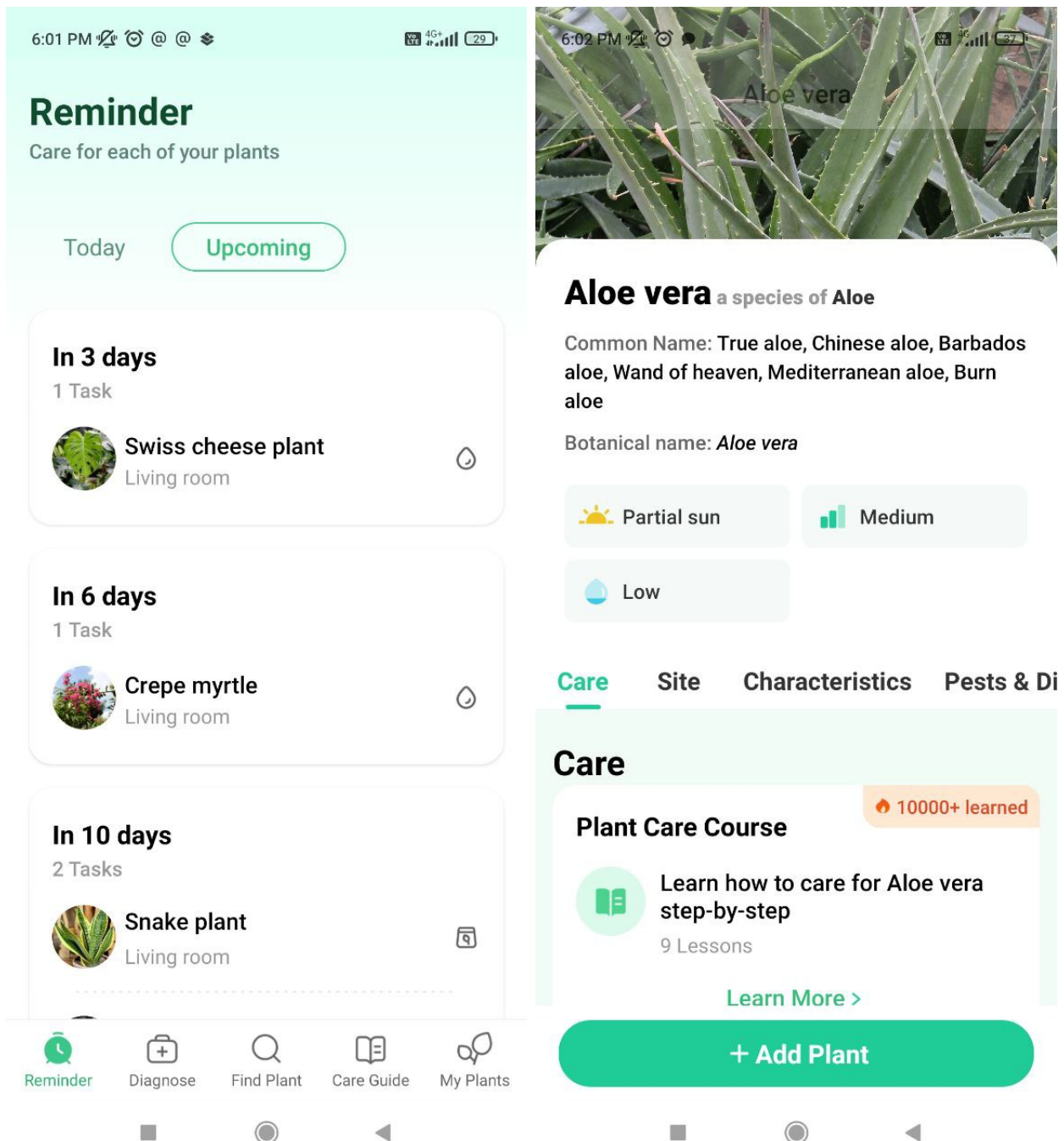
2. Удаление растения через напоминание:

(a) Нажать на кнопку "Календарь"



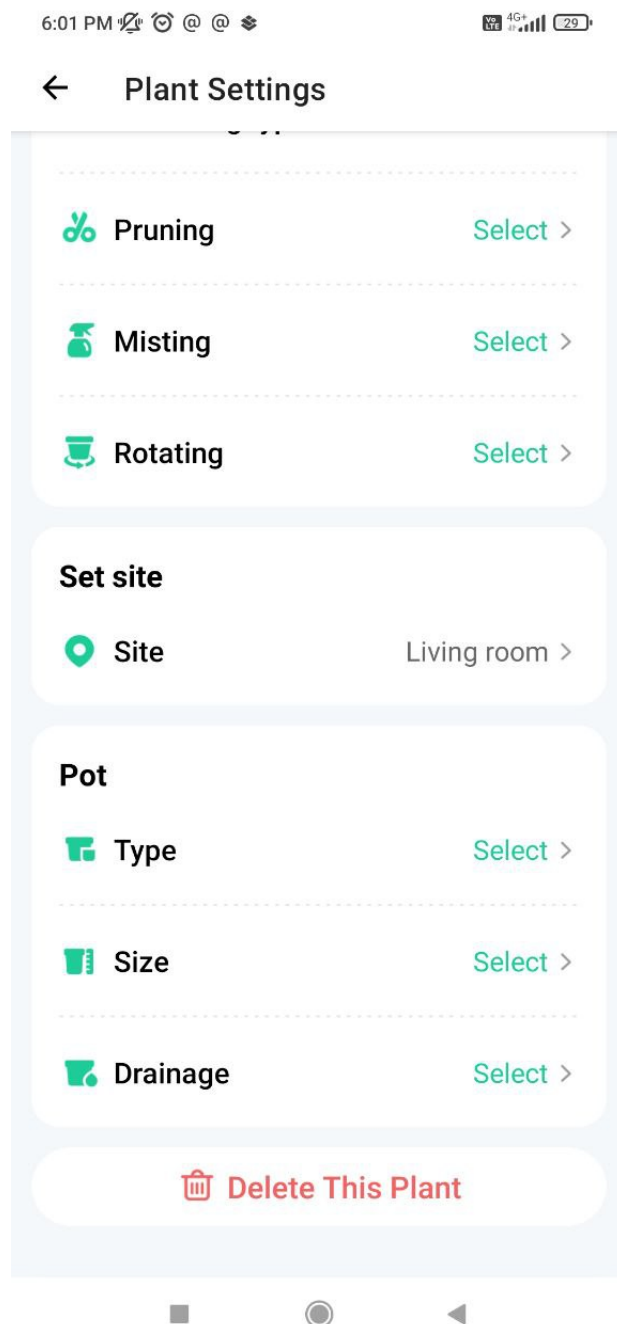
Пример того, как при нажатии открывается календарь

- (b) В открывшемся окне пользователь нажимает на цветок, который он хочет удалить, после чего открывается специальное меню.



Пример того, как устроена система поиска

- (с) В открывшемся меню, в самом конце страницы, пользователь может нажать на кнопку удаления.



Пример реализации добавления

Endpoint

В данном пункте описываются всевозможные взаимодействия клиента с сервером.
Первичная попытка описать главные точки в базах данных

1. Регистрация аккаунта.

Пользователь создает новый аккаунт, заполняя необходимую информацию.

2. Логин в приложение.

3. Добавление растения.

Пользователь добавляет новое растение, создавая его в базе данных.

4. Удаление растения.

Пользователь удаляет новое растение, создавая его в базе данных.

5. Выставление геолокации.

В отдельной вкладке пользователь выставляет свое нынешнее местоположение.

6. Подгрузка данных при заходе в приложение.

Во время того как пользователь заходит, с помощью токена он подгружает данные о добавленных растениях.

7. Обновление состояния растения.

Изменение названия и/или описания.

Дополнения по дизайну

Ссылка на figma проекта: <https://www.figma.com/design/IC9i4ipmpUfJSf7jsChAWF/Untitled?node-id=0-1&p=f&t=Csag07yLyDSHKjFm-0>

**Данный документ будет дополняться по мере
развития проекта**