

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

Факультет Компьютерных наук
Кафедра программирования и информационных технологий

Техническое задание
на разработку мобильного приложения
«Система для учета и анализа питания домашних животных, с рекомендациями
по кормлению и уходу»

Исполнители

_____ О.А. Блинова
_____ А.А. Семенов
_____ Е.Ю. Вдовикова

Заказчик

_____ В.С. Тарасов

Воронеж 2024

Содержание

Содержание.....	2
1 Термины и определения.....	4
2 Общие сведения.....	5
2.1 Полное наименование системы и ее условное обозначение.....	5
2.1.1 Полное наименование системы.....	5
2.1.2 Условное наименование системы.....	5
2.2 Разработчики и заказчик.....	5
2.3 Перечень документов, на основании которых создается приложение.....	5
2.4 Состав и содержание работ по созданию системы.....	5
2.5 Порядок контроля и приемки автоматизированной системы.....	6
2.6 Сведения об источниках и порядке финансирования работ.....	7
3 Цели и назначение автоматизированной системы.....	8
3.1 Назначение системы.....	8
3.2 Цели создания системы.....	8
3.3 Задачи, решаемые при помощи системы.....	8
4 Характеристика объекта автоматизации.....	9
5 Требования к системе.....	10
5.1 Требования к системе в целом.....	10
5.2 Требования к защите информации.....	10
5.3 Требования к функциям (задачам), выполняемым приложением.....	10
5.4 Требования к видам обеспечения.....	11
5.4.1 Требования к структуре.....	11
5.4.2 Требования к технологиям, используемым при написании приложения	12
5.4.3 Требования к программному обеспечению.....	13
5.4.4 Требования к техническому обеспечению.....	13
6 Языковые версии приложения.....	14
7 Группы пользователей.....	15
8 Дизайн приложения.....	16

8.1 Общие положения дизайна	16
8.2 Экран “Вход”	17
8.3 Экран “Регистрация”	17
8.4 Экран “Профиль”	18
8.5 Экран “Настройки”	19
8.6 Экран “Советы”	20
8.7 Экран “Еда”	21
8.8 Экран “Здоровье”	22
9 Навигация по приложению	24
10 Требования к составу и содержанию работ по подготовке объекта автоматизации.....	25
11 Требования к документированию	26
12 Источники разработки.....	27
13 Подписи сторон	28
ПРИЛОЖЕНИЕ А	29

1 Термины и определения

- СУБД – система управления базами данных. Комплекс программно-языковых средств, позволяющих создать базы данных и управлять данными.
- Фреймворк – это структура программного обеспечения, которая упрощает разработку приложений, предоставляя набор инструментов и правил для работы разработчиков.
- Регистрация – действия, направленные на создание личной учетной записи в приложении, с целью получения доступа к его полному функционалу.
- Пользователь – лицо, которое использует действующую систему для выполнения конкретной функции.
- Dart – язык программирования, разработанный Google, который используется для создания веб-приложений, мобильных приложений и серверных приложений.
- Flutter – это открытая среда разработки от Google для создания кроссплатформенных мобильных, веб- и настольных приложений с использованием одного кода.
- Python – это язык программирования, который широко используется в интернет-приложениях, разработке программного обеспечения, науке о данных и машинном обучении.
- Django REST framework – свободный фреймворк для разработки на языке Python.
- PostgreSQL – объективно-реляционная система управления базами данных.

2 Общие сведения

2.1 Полное наименование системы и ее условное обозначение

2.1.1 Полное наименование системы

Система для учета и анализа питания домашних животных, с рекомендациями по кормлению и уходу.

2.1.2 Условное наименование системы

PetFeed

2.2 Разработчики и заказчик

Заказчик: Старший преподаватель: Тарасов Вячеслав Сергеевич.
Воронежский Государственный Университет, Факультет Компьютерных Наук, кафедра Программирования и Информационных Технологий.

Разработчик: команда 6, группа 4

Состав команды разработчики:

- Блинова Оксана Александровна
- Семенов Алексей Алексеевич
- Вдовикова Елена Юрьевна

2.3 Перечень документов, на основании которых создается приложение

Данное приложение будет создаваться на основе следующих документов:

- федеральный закон “Об информации, информационных технологиях и о защите информации” от 27.07.2006 N 149-ФЗ (ред. от 12.12.2023);
- федеральный закон “О персональных данных” от 27.07.2006 N 152-ФЗ;
- настоящее техническое задание, составленное в соответствии с ГОСТ 34.602 - 2020.

2.4 Состав и содержание работ по созданию системы

Состав и содержание работ по созданию приложения включают в себя следующие этапы:

- Сбор необходимой информации, постановка целей, задач системы, которые в будущем должны быть реализованы 13.02.2024-20.02.2024;
- Анализ предметной области, анализ конкурентов и построение структуры требований, ведущих к решению поставленных задач и целей 21.02.2024-26.02.2024;
- Построение модели программы, описание спецификация данных, определение связей между сущностями, разработка модели БД 26.02.2024-06.03.2024;
- Разработка программной части 01.04.2024-25.05.2024;
- Тестирование функциональности приложения 01.04.01.-01.06.2024;
- Разработка курсового проекта 01.05.2024-01.06.2024;
- Сдача проекта 10.06.2024.

2.5 Порядок контроля и приемки автоматизированной системы

Предварительные отчеты по работе будут проводиться во время рубежных аттестаций:

- 1 аттестация (середина марта 2024) – создан репозиторий на GitHub, распределены задачи проекта в таск-менеджере YouTrack, создан проект Miro с общей логикой системы, создан дизайн-макет приложения в онлайн сервисе Figma, представлено готовое техническое задание;
- 2 аттестация (конец апреля 2024) – написана основополагающая часть кода приложения, разработаны экраны приложения и их взаимодействие между собой, реализована БД и ее взаимодействие с сервером;
- 3 аттестация (конец мая 2024) – проведена отладка и доработка

кода, проведено тестирование по работе системы, разработан курсовой проект, выполнены завершающие работы по доработке приложения, предоставлена готовая система.

2.6 Сведения об источниках и порядке финансирования работ

Финансирование работ осуществляется из личных средств Исполнителя.

3 Цели и назначение автоматизированной системы

3.1 Назначение системы

Приложение позволяет решать следующие задачи:

- получать рекомендации по питанию на основании особенностей питомца;
- получать информацию о наличии определенных пищевых элементах в различных кормах;
- просматривать статьи;
- вести динамику веса и активности питомца.

3.2 Цели создания системы

Целью данного приложения является предоставление удобного и информативного инструмента, который поможет владельцам животных следить за рационом кормления, анализировать состав кормов, получать рекомендации и улучшать заботу о питомцах.

3.3 Задачи, решаемые при помощи системы

Система решает следующие задачи:

- учет питания;
- анализ корма;
- мониторинг здоровья;
- получение полезной информации по теме домашних животных.

4 Характеристика объекта автоматизации

Объектом автоматизации является процесс контроля питания и здоровья домашних животных.

5 Требования к системе

5.1 Требования к системе в целом

- приложение должно устанавливаться и работать на мобильных устройствах версий Android 10 и выше, имеющих доступ к сети Интернет;
- приложение должно должно корректно выполнять все поставленные задачи.

У приложения есть следующие перспективы развития:

- добавление функции для поиска ближайших ветеринарных клиник;
- возможность оставлять комментарии к статьям;
- загрузка фотографий в профиль;
- удаление рекламы в приложении для премиум пользователей.

5.2 Требования к защите информации

- актуализация и защита БД: Регулярные обновления и патчи для баз данных, а также ограничение доступа к БД только необходимым пользователям;
- организация резервного копирования: Регулярное резервное копирование данных для обеспечения их безопасности и возможности восстановления в случае инцидентов.

5.3 Требования к функциям (задачам), выполняемым приложением

Для неавторизованного пользователя должны быть предоставлены следующие функции:

- авторизоваться в системе;

Для авторизованного пользователя:

- добавление нескольких питомцев (не более 2-х);
- внесение информации о питомце (кличка, пол, вес, возраст,

активность);

- добавление информации о здоровье питомца (стерилизация, беременность, аллергии и т.п.);
- указание количества и названия употребляемого корма;
- просмотр анализа корма и подходящих альтернатив;
- просмотр статей.

Для премиум пользователя:

- добавление нескольких питомцев (не более 2-х);
- внесение информации о питомце (кличка, пол, вес, возраст, активность);
- добавление информации о здоровье питомца (стерилизация, беременность, аллергии и т.п.);
- указание количества и названия употребляемого корма;
- просмотр статей.

5.4 Требования к видам обеспечения

5.4.1 Требования к структуре

Frontend:

Приложение должно быть реализовано в соответствии с подходом MVVM (Model-View-ViewModel). MVVM – это паттерн разработки, позволяющий разделить приложение на три функциональные части:

- Model – основная логика программы;
- View – вид или представление;
- ViewModel – модель представления, которая служит прослойкой между View и Model.

Шаблон MVVM помогает четко отделять бизнес-логику приложения и логику презентации от пользовательского интерфейса, что помогает решить многочисленные проблемы разработки и упрощает тестирование,

обслуживание и развитие приложения.

Backend:

Приложение должно быть реализовано в соответствии с подходом MVT (Model – View – Template) – паттерн разработки, разделяющий архитектуру приложения на три модуля:

- Model – этот компонент отвечает за обработку данных и взаимодействие с базой данных;
- View – обрабатывает запросы от клиента, взаимодействует с моделью и возвращает данные в виде ответа;
- Template – определяет структуру и внешний вид страницы, представления данных для пользователя.

5.4.2 Требования к технологиям, используемым при написании приложения

Для реализации серверной части приложения выбраны технологии:

- язык программирования Python 12-ой версии;
- фреймворк Django REST framework;
- СУБД PostgreSQL;
- инструмент для создания документации API Swagger;
- uWSGI.

Для реализации клиентской части приложения выбраны технологии:

- язык программирования Dart;
- фреймворк Flutter.

Для развертывания приложения будут использоваться следующие средства:

- клиент Certbot для создания и получения SSL сертификата;
- Docker для автоматизации развертывания приложения;
- Nginx – прокси-сервер с поддержкой SSL.

В качестве преимуществ выбранных технологий можно отметить следующее:

Для Python и Django:

- готовые решения для реализации RESTful архитектуры;
- удобные инструменты для работы с PostgreSQL.

Для PostgreSQL:

- предоставляет большой бесплатный функционал.

Для Flutter:

- мультиплатформенность;
- понятная и полная документация;
- возможность быстро проектировать мобильные приложения.

5.4.3 Требования к программному обеспечению

Требования к программному обеспечению клиентской части:

- клиентская часть приложения должна быть реализована на языке программирования Dart с использованием фреймворка Flutter;
- приложение должно устанавливаться и работать на устройстве под управлением операционной системы Android 10 и выше.

Требования к программному обеспечению серверной части:

- серверная часть приложения должна быть реализована на языке программирования Python 12-ой версии с использованием фреймворка Django REST framework;
- в качестве системы управления базами данных должна быть использована PostgreSQL.

5.4.4 Требования к техническому обеспечению

Требования к техническому обеспечению клиентской части:

Устройство под управлением ОС Android 10 должно иметь следующие характеристики:

- диагональ экрана 4,5 дюйма и больше;
- объем оперативной памяти 1,5 Гб и больше.

Требования к техническому обеспечению серверной части:

- оперативная память сервера 512 Мб и больше;
- постоянная память сервера 512 Мб и больше;
- тактовая частота процессора 1 ГГц и выше;
- количество ядер процессора 1 и более;
- возможность доступа в сеть Интернет.

6 Языковые версии приложения

Основной язык приложения русский.

7 Группы пользователей

В системе существует три группы пользователей: неавторизованные, авторизованные и премиум.

Неавторизованный пользователь имеет возможность только

авторизоваться в системе.

Авторизованный пользователь имеет возможность добавлять не более 2-х питомцев, вносить информацию о питомце и его здоровье, указывать количество и название корма для анализа и просмотра подходящих альтернатив.

Премиум пользователь имеет возможность добавлять неограниченное количество питомцев, вносить информацию о питомце и его здоровье, указывать количество и название корма для анализа и просмотра подходящих альтернатив, возможность использования приложения без рекламы.

8 Дизайн приложения

8.1 Общие положения дизайна

Приложение должно быть выполнено в едином стиле. Цветовая палитра

приложения должна содержать цвета: 4F3129, FFFFFFFF, E1BF11, 8E9326, EDF0E8, F6F6F6, 000000. Используются шрифты без засечек, всего не больше 3 шрифтов. Текст должен быть хорошо виден на экране. Для иконок должны быть выбраны понятные изображения без мелких деталей.

8.2 Экран “Вход”

Этот экран предназначен для ввода электронной почты и пароля пользователя для процедуры авторизации. После успешной авторизации происходит переход на экран “Профиль” для авторизованных пользователей. Также на экране присутствует кнопка “Забыли пароль” для возможности восстановления пароля. Пользователь также может выбрать кнопку “Регистрация” для перехода на экран регистрации.

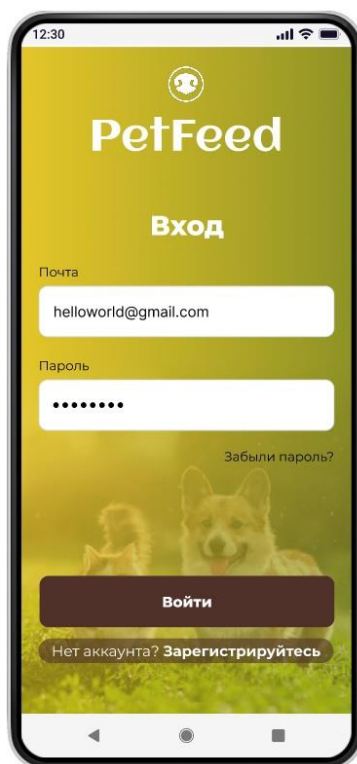


Рисунок 1 - Макет экрана "Вход"

8.3 Экран “Регистрация”

На данном экране предлагается ввести адрес электронной почты, придумать пароль и нажать кнопку “Регистрация”. Затем необходимо последовательно указать следующие данные:

- ваше имя;
- вид питомца;
- кличку питомца;
- пол питомца;
- дату рождения;
- уровень активности;
- вес;
- особенности здоровья;
- основной корм.

В процессе регистрации можно воспользоваться кнопками “<” и “Далее >” для навигации между разделами регистрации. После заполнения всех форм регистрации необходимо нажать кнопку “Завершение”.

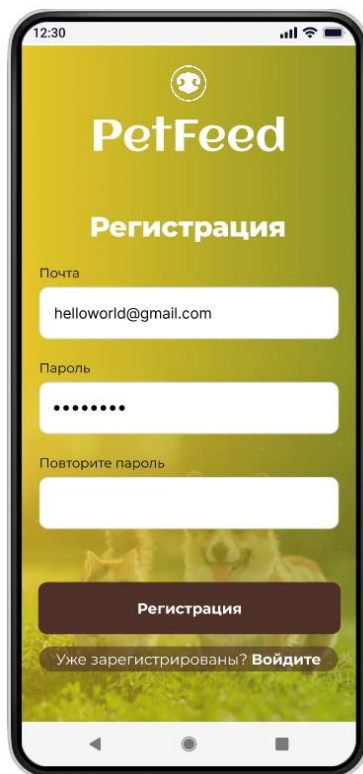


Рисунок 2 - Макет экрана “Регистрация”

8.4 Экран “Профиль”

Данный экран появляется при входе в приложение авторизованного пользователя или после успешной регистрации. Здесь содержится информация о питомце, его весе, активности и наличии заболеваний. В верхней части экрана находятся кнопки “Еда” и “Советы”, а в нижней части расположены кнопки “Питомцы”, “Главное” и “Настройки”.

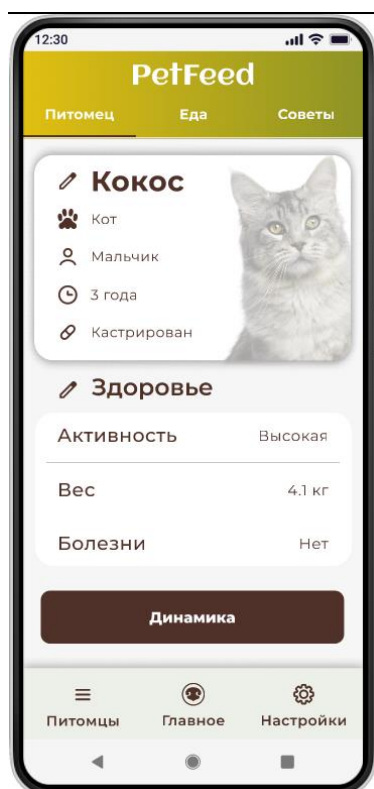


Рисунок 3 - Макет экран “Профиль”

8.5 Экран “Настройки”

На данном экране представлены кнопки для изменения имени, адреса электронной почты и пароля. Пользователь может выбрать светлую или темную тему оформления, а также изменить язык интерфейса. Предусмотрены опции выхода из аккаунта и удаление аккаунта.

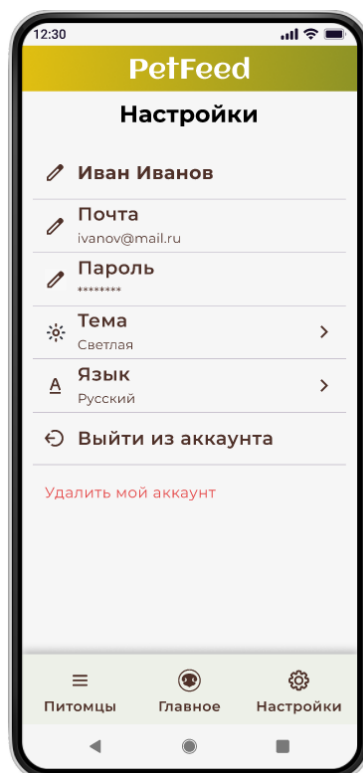


Рисунок 4 - Макет экрана "Настройки"

8.6 Экран “Советы”

На данном экране расположены кнопки для разделов “Еда”, “Здоровье”, “Дом и игры” и “Помощь”. В каждом разделе пользователь может найти статьи, предоставленные разработчиками из научных журналов и посвященные соответствующей тематике. Кроме того, статьи также могут быть получены из академических исследований, университетских изданий и других официальных источников. Пользователи могут добавлять статьи в избранное. Также имеется кнопка “Все статьи”, позволяющая просматривать содержимое всех разделов одновременно.

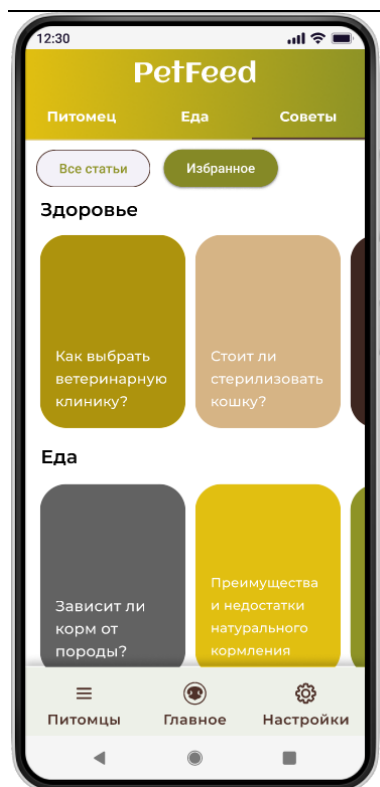


Рисунок 5 - Макет экрана "Советы"

8.7 Экран “Еда”

На данном экране предлагается ввести следующие данные о кормлении питомца:

- количество употребляемого корма;
- название корма;

Также на странице представлены:

- подходящие альтернативы, которые могут быть предложены в случае, если текущий корм не подходит питомцу;
- анализ состава корма, включающий количество белков, жиров, витаминов и других питательных элементов, а также минимальные рекомендуемые значения для каждого из них.

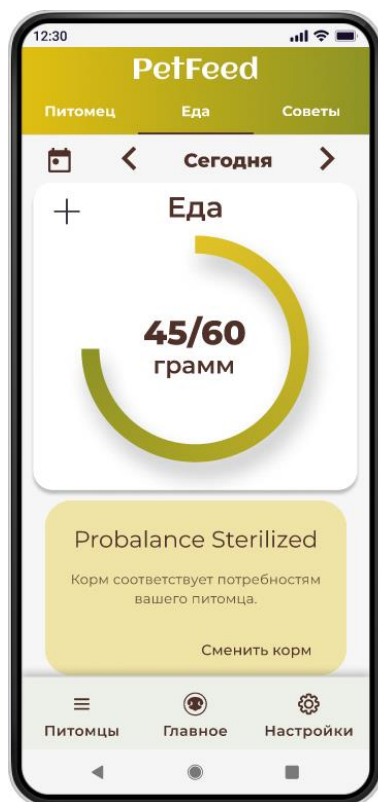


Рисунок 6 - Макет экрана "Еда"

8.8 Экран “Здоровье”

На данном экране предлагается отметить следующие особенности здоровья питомца:

- стерилизация;
- беременность;
- аллергии;
- чувствительный ЖКТ;
- болезни мочевыделительной системы;
- болезни почек;
- диабет;
- ожирение;

- проблемы с кожей;
- проблемы с печенью.

Это позволит выбрать соответствующие особенности здоровья, чтобы обеспечить правильный уход за питомцем.

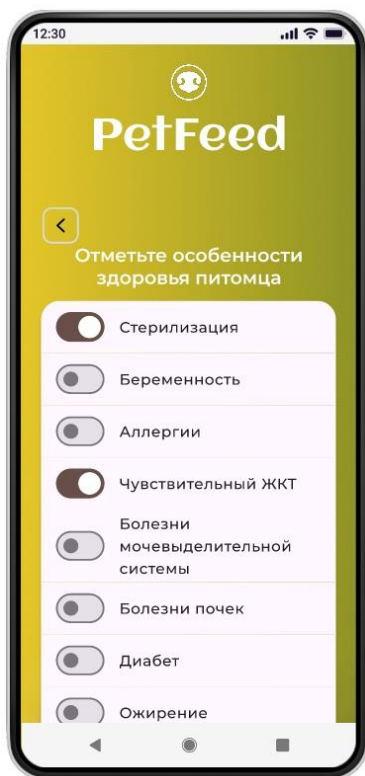


Рисунок 7 - Макет экрана "Здоровье"

9 Навигация по приложению

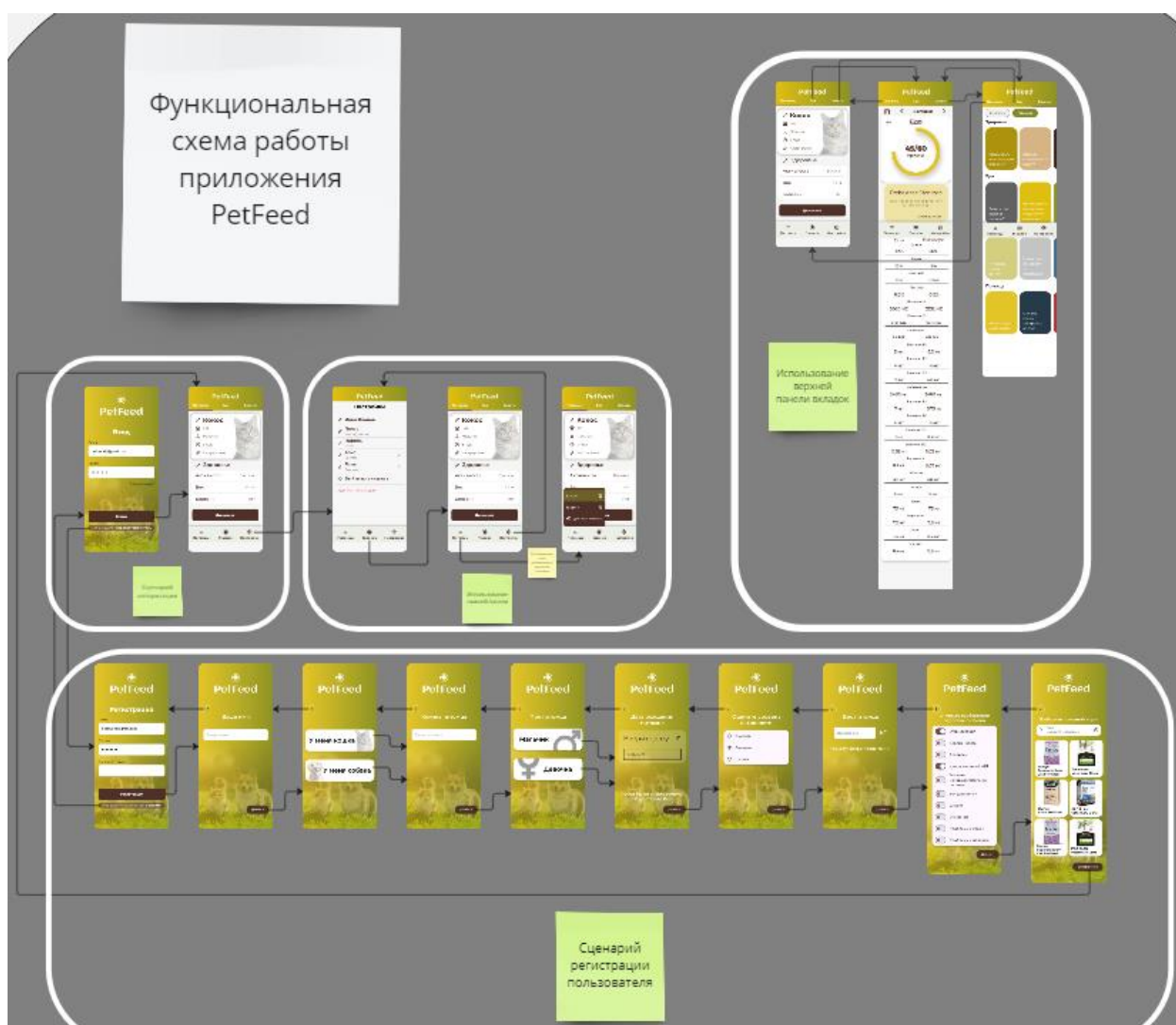
Основным навигационным меню должно выступать нижнее меню.

Нижнее меню должно включать в себя следующие кнопки:

- Питомцы;
- Главное;
- Настройки.

Также есть дополнительные разделы сверху экрана:

- Питомец;
- Еда;
- Советы.



10 Требования к составу и содержанию работ по подготовке объекта автоматизации

При подготовке объекта автоматизации к вводу автоматизированной системы в действие необходимо обеспечить выполнение требований к программно-техническим средствам, на которых должно быть развернуто программное обеспечение автоматизированной системы. Также необходимо провести опытную эксплуатацию автоматизированной системы.

11 Требования к документированию

Перечень документов, которые необходимо предоставить:

- Курсовая работа;
- ссылки на сервисы с документацией (GitHub, Trello, Miro, Figma).

12 Источники разработки

1. ГОСТ 34.602-2020. Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.
2. ГОСТ 34.601-90. Автоматизированные системы. Стадии создания.

13 Подписи сторон

ЗАКАЗЧИК:

Ст. преп. Тарасов В.С.

_____/_____/

«____»_____20____г.

ИСПОЛНИТЕЛЬ:

Рук. команды 4-6 Блинова О.А.

_____/_____/

Разработчик Семенов А.А.

_____/_____/

Разработчик Вдовикова Е.Ю.

_____/_____/

«____»_____20____г.

ПРИЛОЖЕНИЕ А

Диаграммы, иллюстрирующие работу системы:

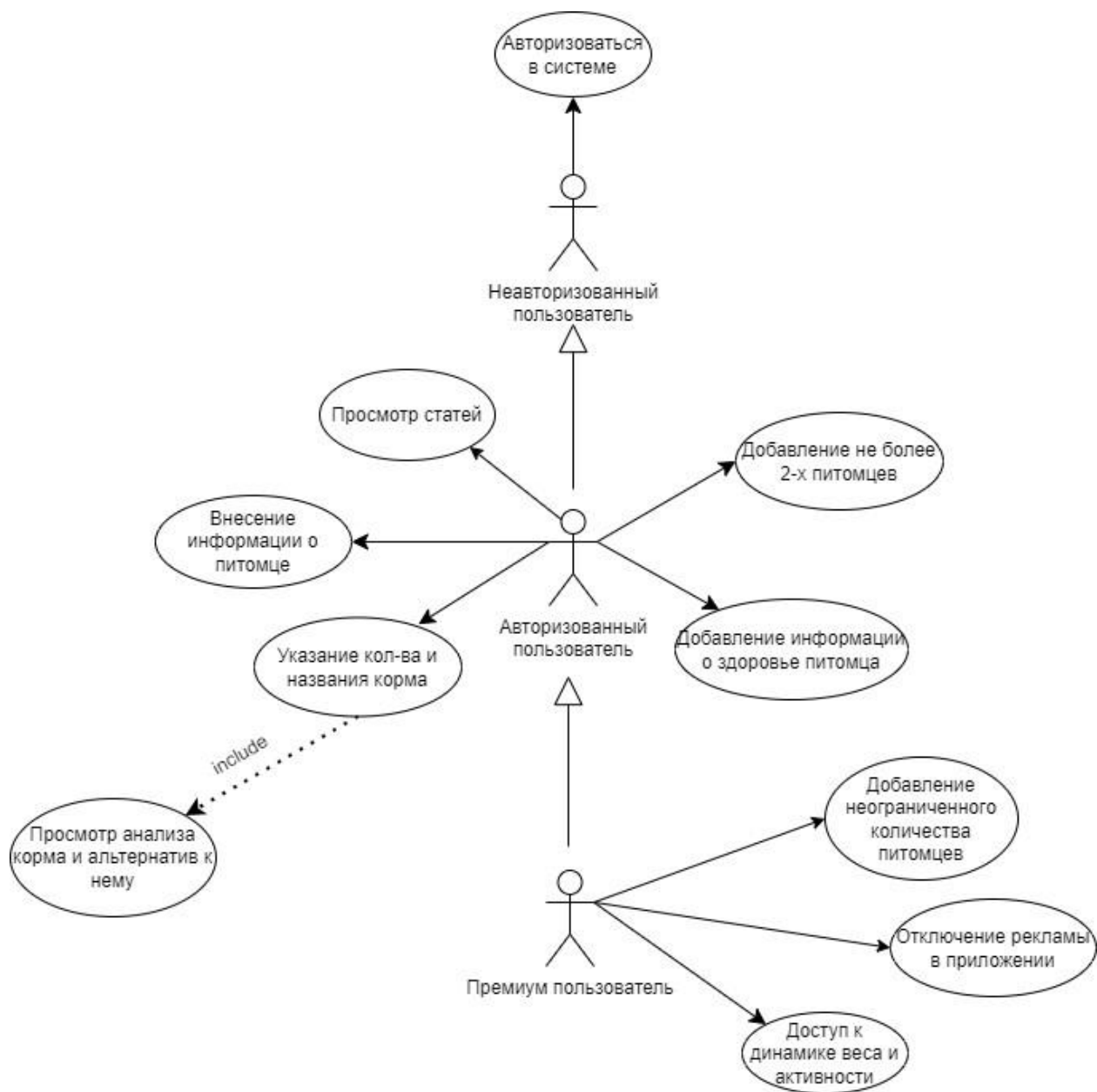


Рисунок 8 - Диаграмма прецедентов

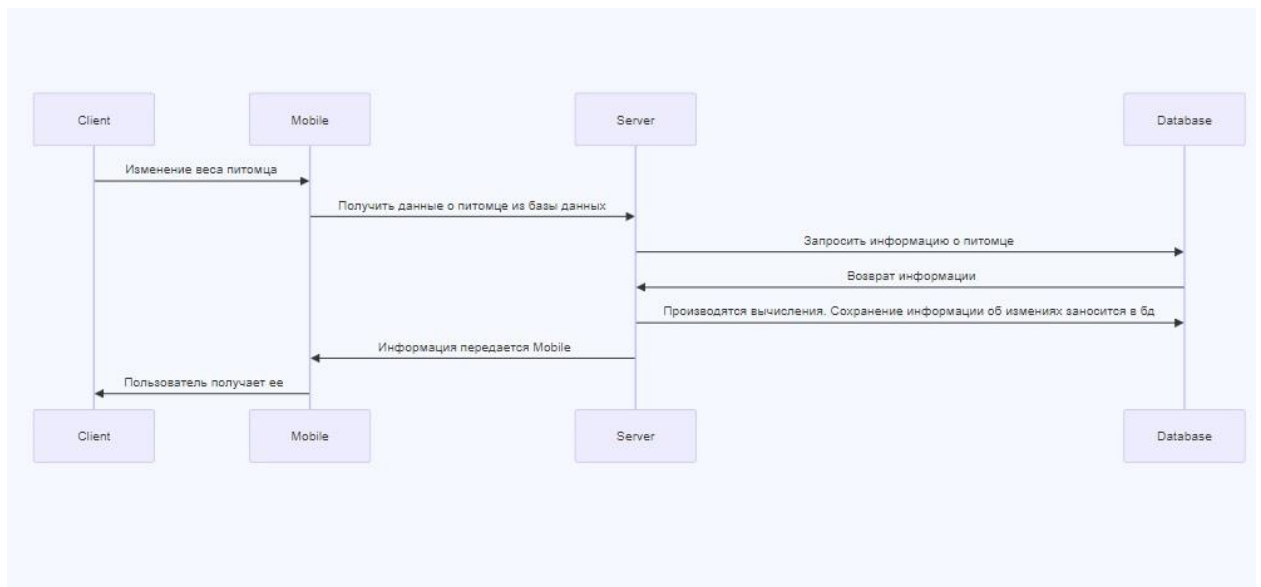


Рисунок 9 - Диаграмма последовательностей

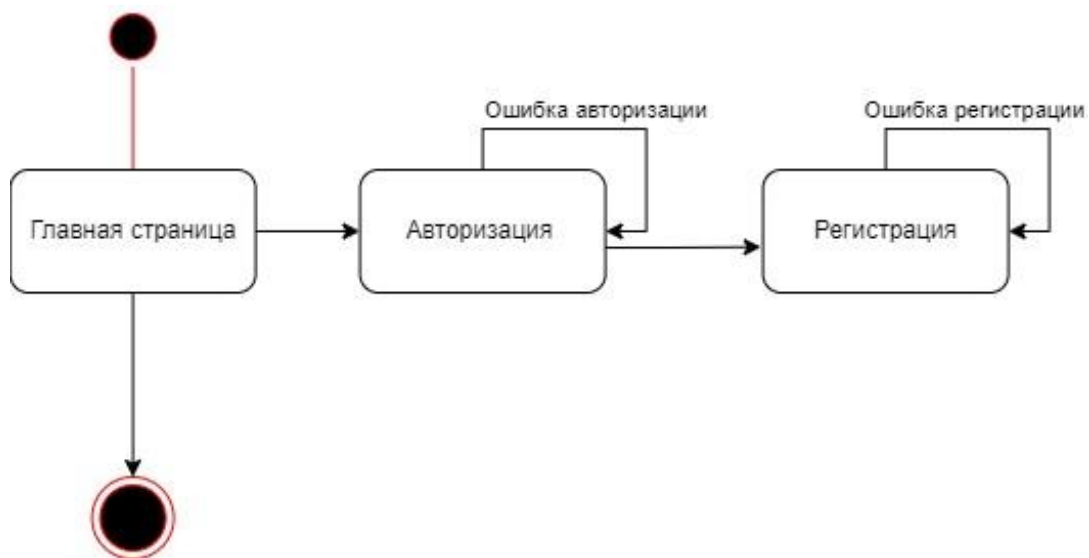


Рисунок 10 - Диаграмма состояний (для неавторизованного пользователя)

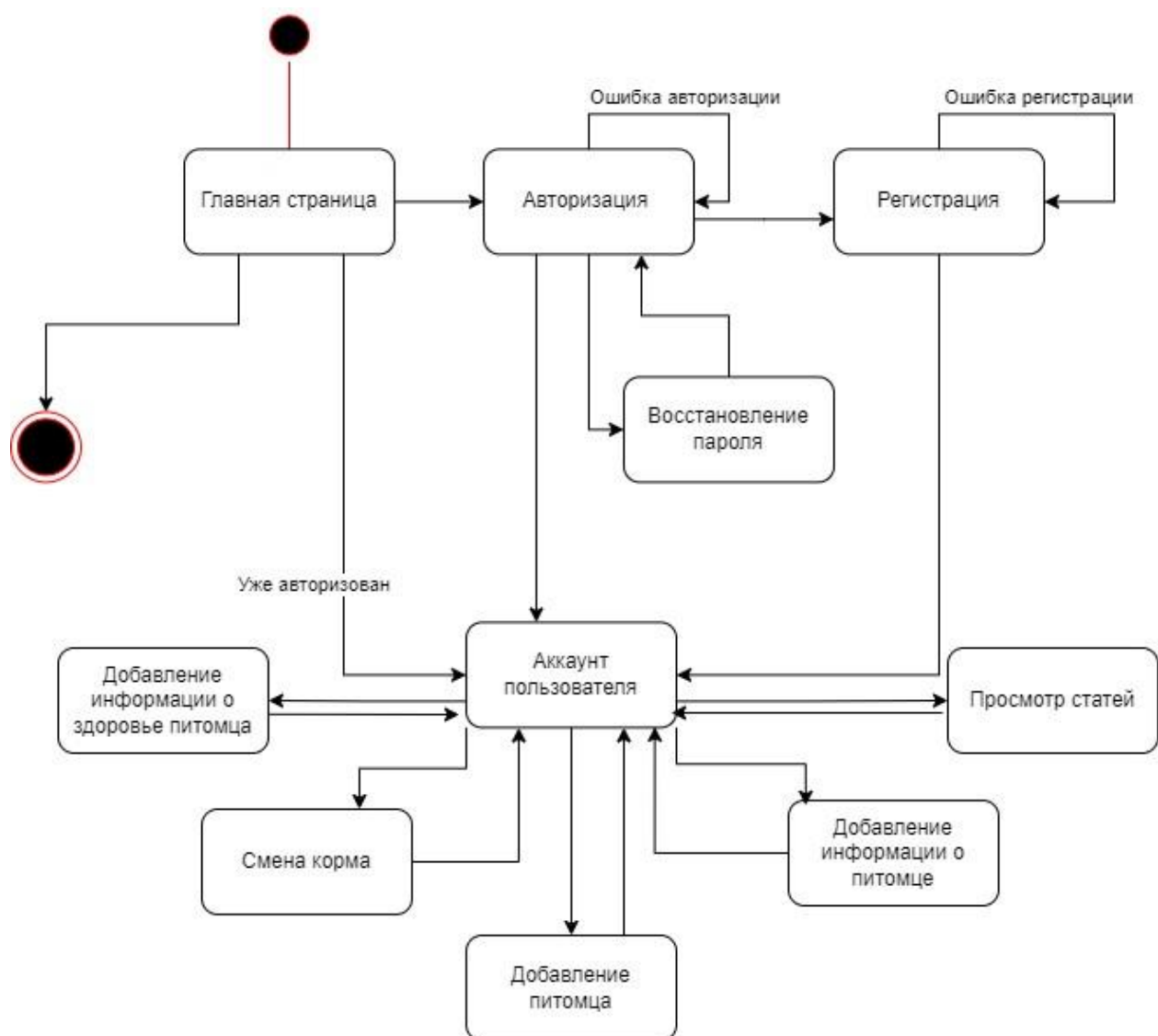


Рисунок 11 - Диаграмма состояний (для авторизованного пользователя)

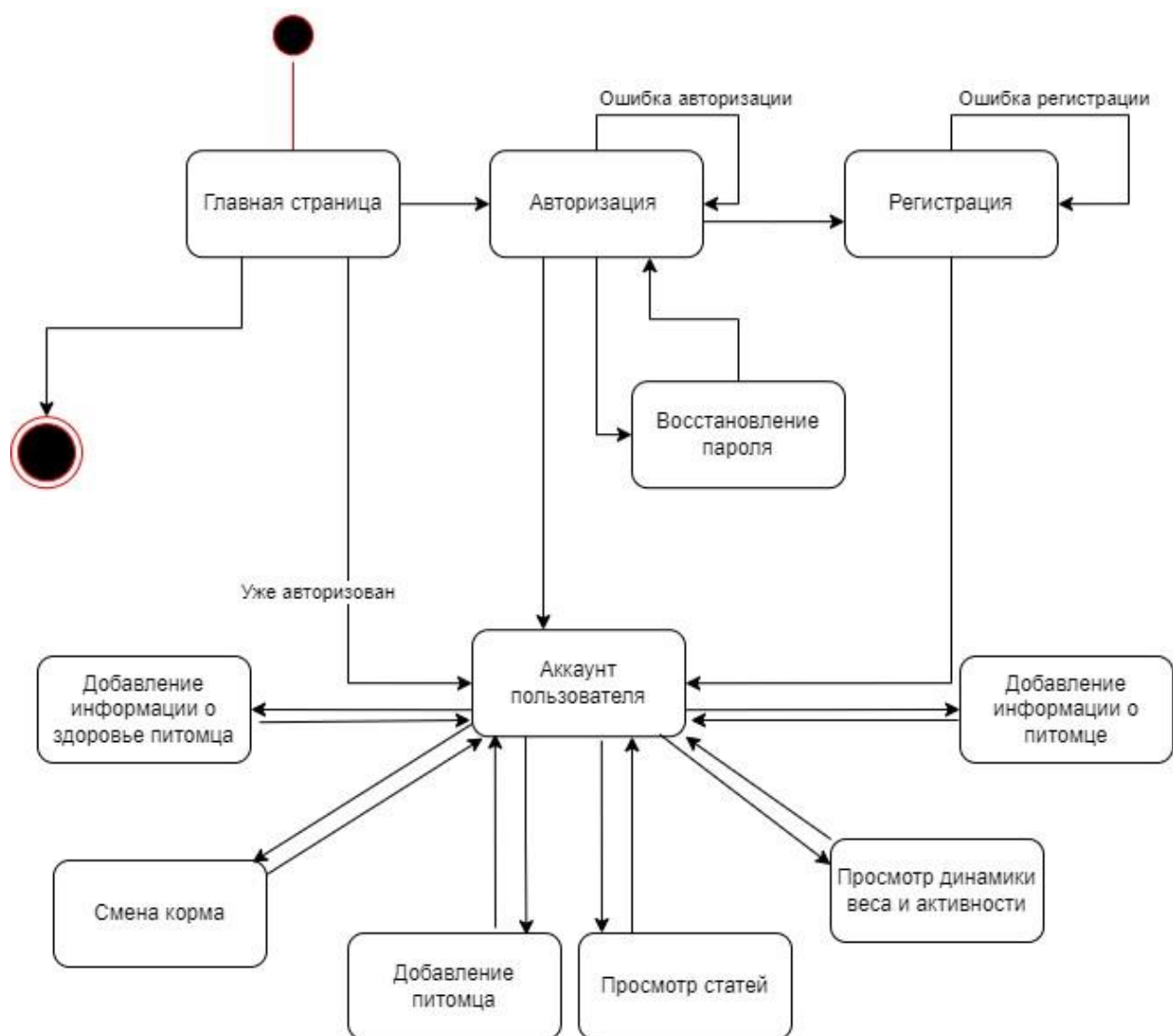


Рисунок 12 - Диаграмма состояний (для премиум пользователя)