**Аналитическая записка**

«Информационная система учета животных в приютах»

1. **Наименование проекта**

Информационная система учета животных в приютах. Данная информационная система предназначена для автоматизации управления приютом, повышения качества обслуживания, ухода за животными, предназначена для автоматизации хранения данных о животных, состоянии их здоровья, потребностей. Кроме того, информационная система может использоваться для поиска и выбора животных, подачи заявок на «усыновление» и получения информации о доступных питомцах.

1. **Целевая аудитория**

Основная целевая аудитория проекта:

* индивидуальные предприниматели;
* юридические лица;
* сотрудники приюта: директор, администраторы, кураторы животных, специалисты по уходу за животными, волонтеры, специалисты по работе с усыновителями;
* будущие владельцы животных, потенциальные хозяева.

1. **Назначение проекта**

Назначение разработки состоит в создании и внедрении информационной системы, предназначенной для автоматизации управления приютами для животных. Система позволит автоматизировать работу приюта, автоматизировать процессы учета животных, улучшить качество ухода и обслуживания животных, кроме того, система позволит хранить и отслеживать информацию о состоянии здоровья животного, его истории пребывания, его вакцинациях, документациях, связанных с его ветеринарными процедурами. Информационная система также позволит упростить процесс приема новых животных, контролировать запасы еды, состояние вольера, позволит оптимизировать поиск новых хозяев для животных, через интеграцию с внутренними базами данных, позволит выводить необходимую отчетность для отслеживания работы всего приюта.

1. **Описание взаимодействия с потенциальным пользователем**

Пользовательские истории представлены в таблице 1.

Таблица 1 – пользовательские истории

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Роль** | **Действие** | **Цель** |
| 1 | Директор приюта | Контролирует работу всей информационной системы в целом, просматривает сводные отчеты, добавляет в систему нового сотрудника, назначает права доступа | Отслеживание работоспособности приюта для оценки эффективности его работы, управление доступом к информации и функциям системы |
| 2 | Администратор приюта | Добавляет в систему нового сотрудника, назначает права доступа, формирует отчетность, просматривает сводные отчеты, добавляет информацию о вакцинах и вольерах | Управление доступом к информации и функциям системы, формирование отчетности для дальнейшей оценки |
| 3 | Ветеринар | Вносит в систему информацию о физическом и ментальном состоянии здоровья животного, вносит информацию о вакцинации, назначает процедуры, просматривает заявки на лечение от кураторов животных | Обеспечение актуальности данных о состоянии здоровья животных, их медицинской истории |
| 4 | Куратор животного | Вносит, отмечает информацию о кормлении, уборке вольера, создает заявку на лечение животного | Обеспечение актуальности данных о состоянии животного, вольера |
| 5 | Специалист по уходу за животными | Регистрирует новое поступление животного в системе, оказывает необходимый уход животному, просматривает состояние животных | Регистрация новых животных в системе, уход за животными, просмотр состояний животных |
| 6 | Волонтер | Вносит данные о состоянии и поведении животного в каталог животных, помогает с уходом специалисту по уходу за животными, помогает пристроить животное, просматривает состояние животных | Поддержание актуальной информации о животном, для улучшения условий его содержания, заполнение каталога животных с их описанием, просмотр состояний животных |
| 7 | Специалист по работе с усыновителями | Принимает и обрабатывает заявки на «усыновление» животных, формирует документ «усыновления» | Обеспечение быстрого и эффективного процесса обработки заявок, создание, генерация документации |
| 8 | Потенциальный хозяин | Просматривает каталог животных, используя фильтры, создает заявку на «усыновление» животного | Упрощение поиска животного по параметрам, упрощение процесса усыновления через электронные заявки |

Пользовательские сценарии представлены в таблице 2.

Таблица 2 – пользовательские сценарии

|  |  |  |
| --- | --- | --- |
| **№** | **Пользователь** | **Система** |
| 1 | Заполняет отчетность | Сохраняет отчетность в базу данных |
| 2 | Просматривает отчетность | Вывод необходимой отчетности из базы данных |
| 3 | Создает сотрудника, назначает права доступа | Сохраняет информацию, личные данные для входа в базу данных, шифрует необходимые данные, определяет права доступа |
| 4 | Заполняет состояние здоровья, вносит информацию о вакцинации | Сохраняет и систематизирует информацию в базе данных |
| 5 | Вносит информацию о кормлении, уборке вольера | Сохраняет информацию в базе данных: дата кормления, питомец, номер вольера, описание кормления, описание уборки |
| 6 | Регистрирует животное в системе | Сохраняет и систематизирует информацию в базе данных |
| 7 | Вносит данные о состоянии животного | Сохраняет и систематизирует информацию в базе данных |
| 8 | Принимает и обрабатывает заявки на «усыновление» животных | Сохраняет и систематизирует информацию в базе данных, выводит полученные заявки в отдельный каталог информационной системы, генерирует электронный документ «усыновления» животного |
| 9 | Просматривает каталог животных, создает заявку на «усыновление» животного | Выводит информацию из базы данных в отдельный каталог животных в информационной системе, отправляет заявку в базу данных, выводит заявку в каталог заявок приюта |

Диаграмма Use-Case представлена на рисунке 1.

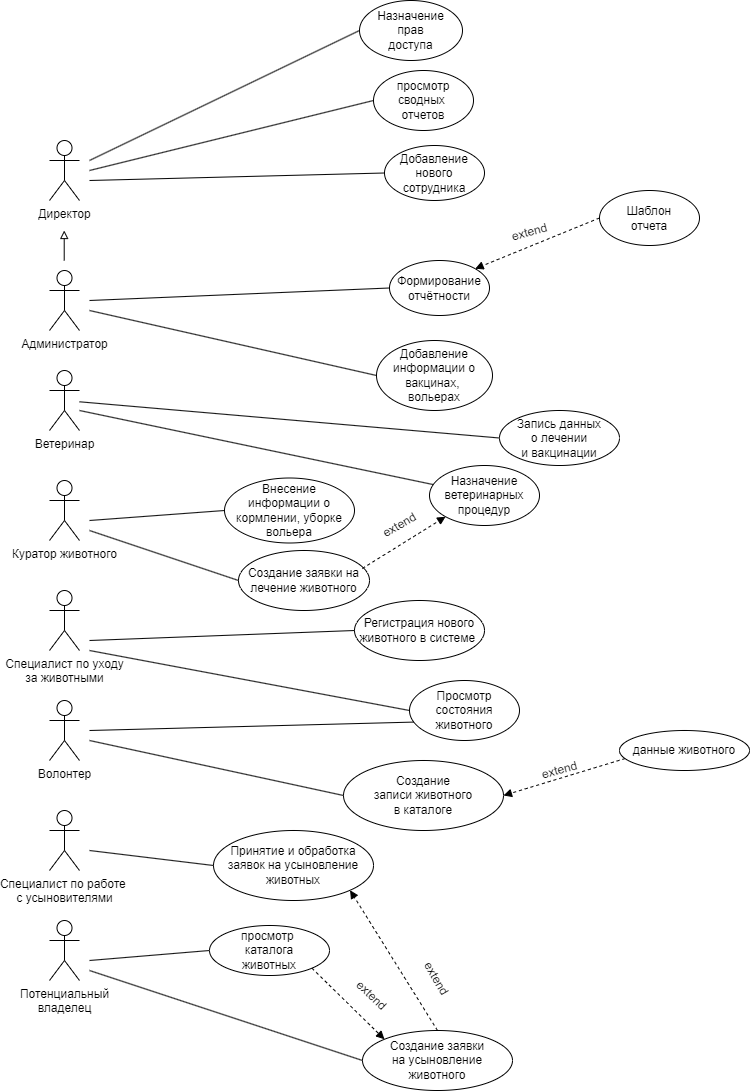


Рисунок 1 – диаграмма Use-Case

Диаграмма Use-Case описывает взаимодействие пользователей, акторов с функциональностью информационной системы учета животных, предназначенной для автоматизации работы приютов. Основные элементы диаграммы включают в себя акторов (пользователей системы), варианты использования и связи между ними.

Основные участники (акторы):

* Директор приюта – отвечает за управление приютом, контролирует работу всей информационной системы в целом, просматривает сводные отчеты, отвечает за добавление новых сотрудников, назначение прав доступа.
* Администратор – отвечает за добавление новых сотрудников, назначение прав доступа, формирует отчетность, просматривает отчетность, добавляет информацию о вольерах, вакцинах.
* Ветеринар – занимается отслеживанием состояния здоровья животного, записью информации о лечении и вакцинациях, назначением ветеринарных процедур.
* Куратор животного – осуществляет контроль за животными, следит за их состоянием, создает заявки на лечение животного, вносит информацию о кормлении и уборке вольера.
* Специалист по уходу за животными – занимается регистрацией нового животного в системе, оказывает необходимый уход, просматривает состояния животных.
* Волонтер – создает запись животного в каталоге, помогает пристроить животное, просматривает состояние животного.
* Специалист по работе с усыновителями – взаимодействует с потенциальными усыновителями, обрабатывает их заявки и организует встречи с животными.
* Потенциальный хозяин – ищет животное для «усыновления», подает заявки и взаимодействует со специалистами по усыновлению.

Основные варианты использования Use-Case:

Просмотр сводных отчетов – директор анализирует сводные данные о состоянии приюта, животных, ресурсах.

Добавление нового сотрудника – администратор и директор регистрируют новых сотрудников в системе.

Назначение прав доступа – администратор и директор и назначают права доступа сотрудникам в информационной системе.

Формирование отчетности – администратор формирует отчетность для анализа и контроля деятельности приюта при помощи шаблонов отчета.

Добавление информации о вакцинах, вольерах – администратор вносит в информационную систему данные о вакцинах и вольерах.

Запись данных о лечении и вакцинации – ветеринар вносит в базу данных информацию о назначенном лечении и произведенных вакцинациях.

Назначение ветеринарных процедур – ветеринар после получения заявки от куратора животного назначает ветеринарные процедуры.

Внесение информации о кормлении, уборке вольеров – куратор животного вносит в информационную систему информацию о кормлении, уборке вольера.

Создание заявки на лечение животного – куратор животного отслеживает ментальное и физическое состояние здоровья животного, при видимых отклонениях создает заявку на лечение животного. Заявка отправляется ветеринару на рассмотрение.

Регистрация нового животного в системе – специалист по уходу за животными регистрирует поступившее в приют животное в информационной системе.

Просмотр состояния животного – специалист по уходу за животными и волонтер получают информацию о состоянии животного для ознакомления с животным, создании записей в каталоге животных.

Создание записи животного в каталоге – волонтер создает информативную запись животного для заполнения каталога.

Принятие и обработка заявок на «усыновление» животных – специалист рассматривает заявку, проверяет данные и организует встречу усыновителя с животным.

Просмотр каталога животных – потенциальный хозяин просматривает каталог животных, читает описание и совершает выбор животного при помощи информационной системы.

Создание заявки на «усыновление» животного – потенциальный хозяин оставляет заявку на «усыновление» выбранного в каталоге животного, заполняет необходимые данные.

Акторы связаны с соответствующими случаями использования посредством линий, что отражает их взаимодействие с системой.

Диаграмма иллюстрирует, какие роли играют акторы, как они взаимодействуют с системой, и каковы ключевые функции, выполняемые системой для обеспечения эффективной работы приюта.

1. **Основной функционал (функциональные требования)**

Функциональные требования представлены в таблице 3.

Таблица 3 – Функциональные требования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Название функции** | **Входные параметры** | **Выходные параметры** | **Описание функции** |
| 1 | Вывод сводных отчетов | Данные из базы данных, | Вывод отчетности в формате «.pdf» | Функция предназначена для сохранения, вывода следующих отчетов в информационную систему: отчет о поступлении животных, отчет об «усыновлении» животных, отчет о назначенных процедурах, отчет о вакцинации, отчет о поступивших заявках на «усыновление» |
| 2 | Добавление нового пользователя в систему | Для потенциального хозяина: почта, пароль, для сотрудника: почта, пароль, паспорт, трудовая книжка, СНИЛС, документы воинского учёта, ФИО | Запись в базе данных в таблице «Пользователь» | Функция формирует запись в базе данных с данными потенциального хозяина, сотрудника, определяет роль сотрудника, ограничивает доступ в соответствии с ролью |
| 3 | Формирование отчетности | Необходимые данные для отчетности из базы данных, шаблоны в формате «.pdf» | Сформированный, сгенерированный отчет | Функция генерирует отчет, позволяет сохранить отчет в необходимую директорию |
| 4 | Внесение информации о состоянии здоровья животного | Данные о состоянии здоровья животного, информация о вакцинации | Запись в базе данных в таблице «Здоровье» | Функция формирует запись в базе данных о состоянии здоровья животного, вакцинации, необходимого лечения животного |
| 5 | Внесение информации о кормлении животного | Данные о дате кормления, номер животного, номер вольера, описание кормления | Запись в базе данных в таблице «Кормление» | Функция формирует запись в базе данных о кормлении животного |
| 6 | Внесение информации об уборке вольера | Данные о дате уборки, номер вольера, описание уборки | Запись в базе данных в таблице «Уборка» | Функция формирует запись в базе данных о проведении уборки в вольере |
| 7 | Назначение ветеринарных процедур | Дата начала лечения, номер питомца, описание лечения | Запись в базе данных в таблице «Назначение лечения» | Функция формирует запись в базе данных |
| 8 | Создание заявки на лечение животного | Дата заявки, наименование заявки, номер ветеринара, описание заявки | Запись в базе данных в таблице «Заявки на лечение» | Функция формирует запись в базе данных, отправляет уведомление ветеринару |
| 9 | Регистрация нового животного в системе | Данные животного, параметры | Запись в базе данных в таблице «Животное» | Функция формирует запись в базе данных о животном, включая описание, характеристики и т.д. |
| 10 | Создание записи животного в каталоге | Данные о поведении животного, его умственные способности, изображение, описание | Запись в базе данных в таблице «Каталог», формирование каталога с животными | Функция формирует запись в базе данных о животном с описанием его характера, изображением, повадками и т.д. |
| 11 | Обработка заявки на «усыновление» животного | Заявка от потенциального «усыновителя», с данными для связи | Запись в базе данных в таблице «Заявки на усыновление» | Функция формирует запись в базе данных |
| 12 | Вывод информации о животном в каталог | Данные о животном из базы данных | Отображение информации из базы данных в каталоге информационной системы | Функция выводит в каталог информационной системы записи с информацией о животных |
| 13 | Создание заявки на «усыновление» животного | Данные о животном из каталога (номер животного), данные потенциального «усыновителя» (ФИО, номер телефона или электронная почта) | Отправленная заявка в базу данных приюта, уведомление специалиста по работе с усыновителями | Функция формирует запись в базе данных, уведомляет специалиста по работе с усыновителями |

1. **Возможные аналоги решения**

Сайт «acits.ru» был зарегистрирован в 2020 году. Главная страница сайта – страница назначений. Страница назначений представлена на рисунке 2.

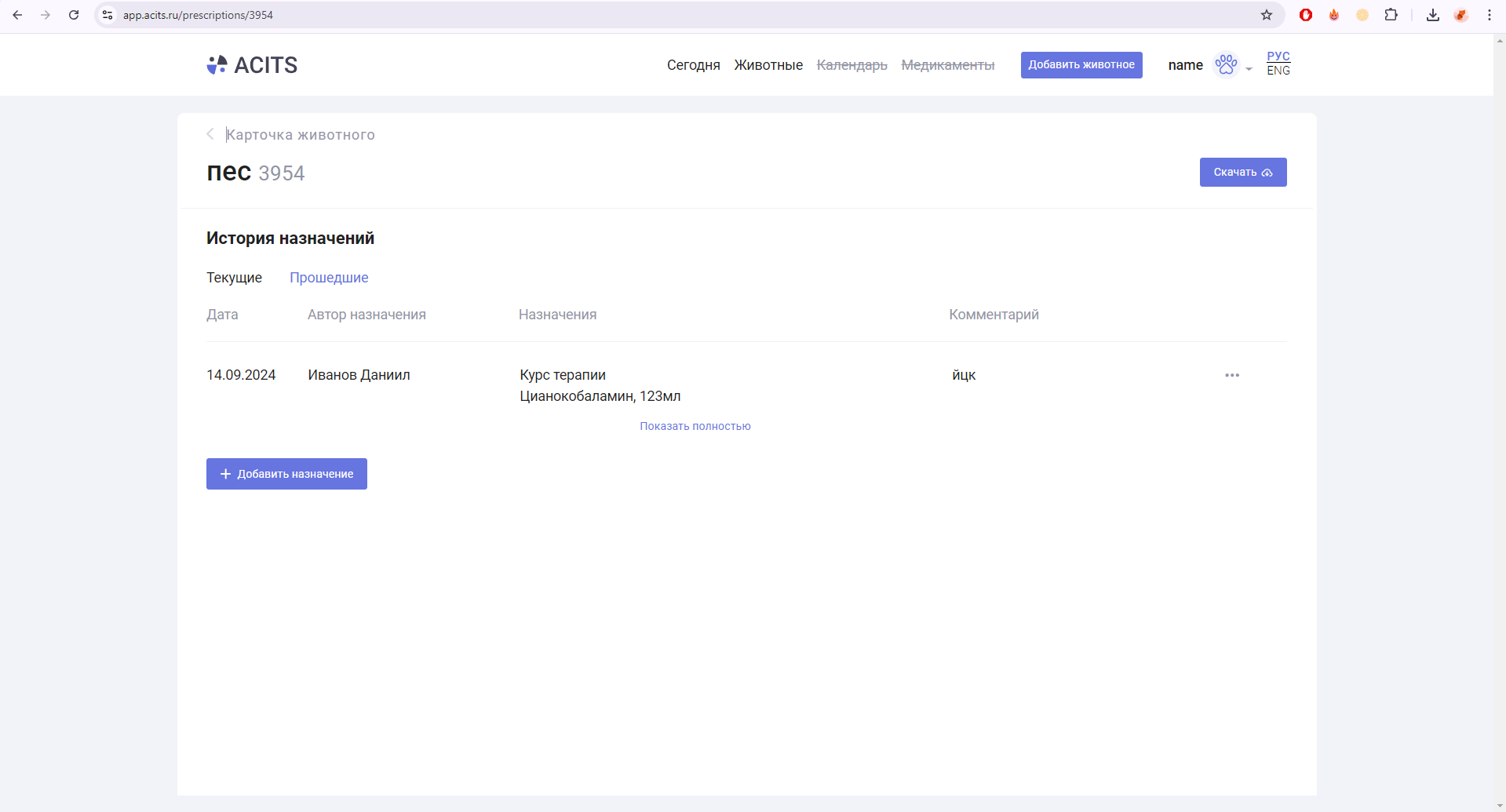


Рисунок 2 – Страница назначений

К преимуществам данного сайта можно отнести: простоту интерфейса, все необходимые вкладки располагаются сверху сайта, что упрощает работу для начинающих пользователей. Есть возможность добавления животного в базу данных, у каждого животного можно посмотреть статус (в приюте, передержка, пристроено, выпущено, погибло). У животного есть свой тип, описание, порода, время поступления в приют, дата чипирования, куратор, заявитель. Также в приложении можно редактировать свою учетную запись, выбирать необходимую организацию. Есть учет медикаментов, календарь учета животного.

К недостаткам данного сайта можно отнести: небольшой функционал, отсутствие взаимодействия с ветеринаром, при переходе на главную страницу, страницу назначений, не отображаются назначения. Назначения начинают отображаться только после добавления новой записи. Отсутствует вывод отчетности.

Приложение «USU software». При запуске приложения открывается главная страница с назначением животного на процедуру. Главная страница представлена на рисунке 3.

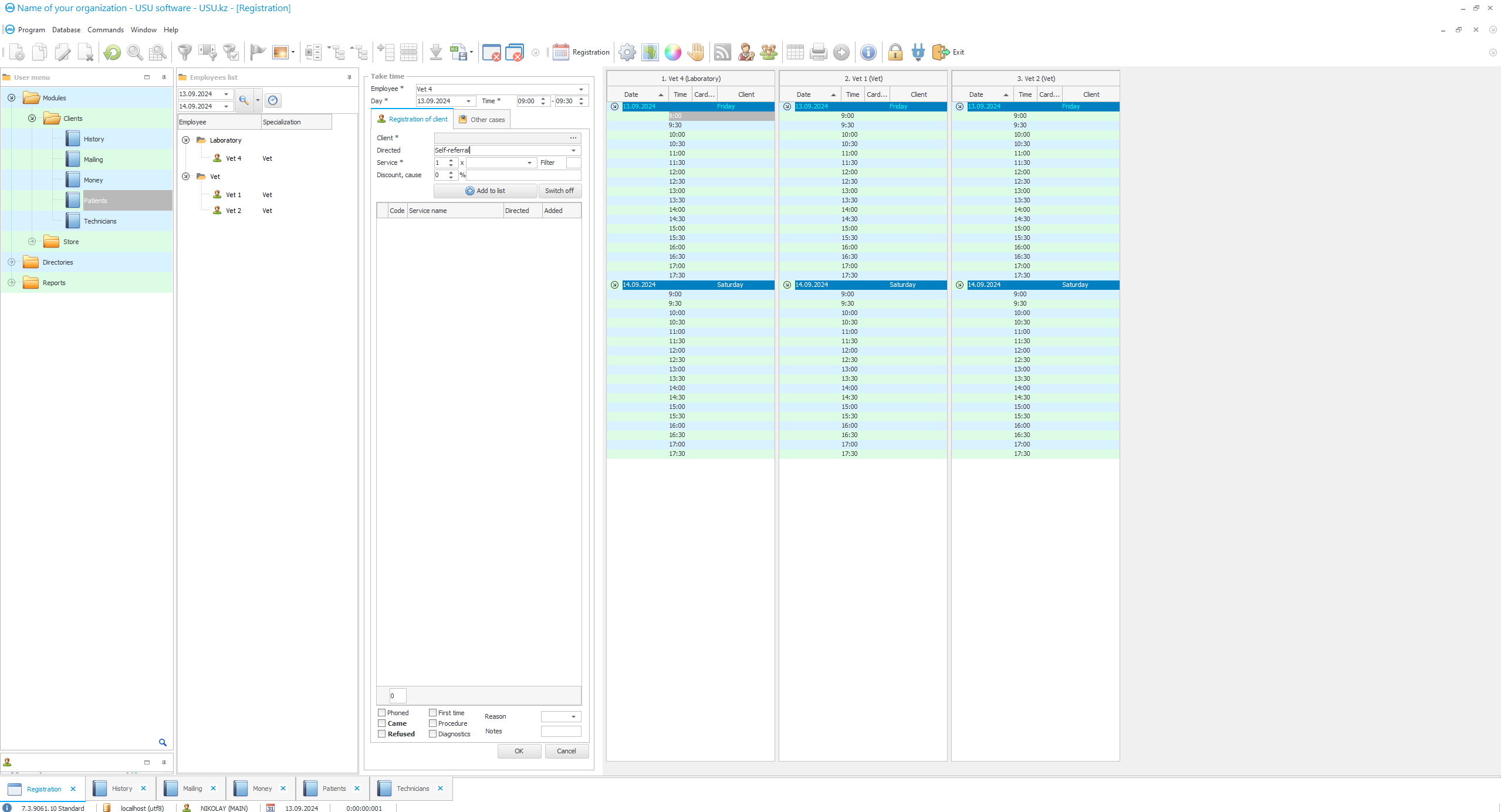


Рисунок 3 – Страница назначений

К преимуществам данного приложения можно отнести: перемещение, открепление, закрытие ненужных вкладок. Такой подход поможет пользователю сконцентрироваться только на своих задачах. Есть возможность добавлять питомца в базу данных, добавлять необходимых сотрудников, присутствует календарь приема и регистрации питомца, вывод отчетности. В приложении есть возможность поменять дизайн вкладок, цвет. Также имеется карта для просмотра места регистрации животного.

К недостаткам данного приложения можно отнести: слишком перегруженный интерфейс, начальному пользователю будет сложно разобраться, нет возможности учитывать вакцинацию питомцев, отсутствие возможности редактирования учетной записи пользователя, отсутствие возможности назначения необходимой роли пользователю информационной системы.

1. **Предлагаемые к использованию технологии и модели**

Для создания информационной системы учета животных в приютах можно использовать различные технологии и модели.

Пример технологий для базы данных: для хранения структурированных данных можно использовать реляционные базы данных, например, MySQL, PostgreSQL, для хранения неструктурированных данных, например, фотографий можно использовать MongoDB, CouchDB.

Пример моделей для базы данных: для представления структуры базы данных можно использовать ER-моделирование, ERD-диаграммы, JSON файлы.

Пример технологий для создания веб интерфейса: для создания пользовательского интерфейса можно использовать React, Angular, для обработки запросов и взаимодействия с базой данных можно использовать Node.js, Django, Flask.

Пример моделей для создания веб интерфейса: MVC моделирование.

Пример технологий для написания мобильного приложения: для использования всех возможностей устройства можно использовать Kotlin, Flutter, React Native.

Пример моделей для написания мобильного приложения: MVVM моделирование.

Пример технологий для проектирования интерфейса пользователя: для проектирования интерфейса приложения, сайта, можно использовать Figma, draw.io.

Модели для проектирования интерфейса пользователя: для определения основных задач и потребностей пользователя можно использовать пользовательские сценарии, для проверки удобства и функциональности интерфейса, прототипирование и тестирование.

Данные технологии и модели помогут создать эффективную и функциональную информационную систему. Выбор конкретных инструментов и подходов зависит от специфики задач.