МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное

учреждение высшего образования

«Казанский национальный исследовательский технический

университет им. А.Н. Туполева-КАИ»

(КНИТУ-КАИ)

Институт компьютерных технологий и защиты информации

(наименование института (факультета), филиала)

\_\_Кафедра Прикладной Математики и Информатики \_\_

(наименование кафедры)

\_\_\_\_\_\_\_\_09.03.01 «Информатика и вычислительная техника» \_\_\_\_

(шифр и наименование направления подготовки (специальности))

пояснительная записка

к Курсовой работе

по дисциплине:

«Технологии Веб-программирования»

на тему: «Клиент-серверное приложение “Ветеринарная клиника”»

Обучающийся 4310 Султанов А.Д.

(номер группы) (подпись, дата) (Ф.И.О.)

Руководитель доцент каф. ПМИ Тутубалин П.И.

(должность) (Ф.И.О.)

Курсовая работа зачтена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, дата)

Казань 2021

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 3](#_Toc74136709)

[1. ПОСТАНОВКА ЗАДАЧИ 4](#_Toc74136710)

[2. МОДЕЛЬ ДАННЫХ СИСТЕМЫ 5](#_Toc74136711)

[3. МАКЕТЫ HTML-СТРАНИЦ СИСТЕМЫ 6](#_Toc74136712)

[3.1 Макет главной страницы 6](#_Toc74136713)

[3.2 Макет страницы «Контакты» 7](#_Toc74136714)

[3.3 Макет страницы «Личный кабинет» 8](#_Toc74136715)

[4. СТРУКТУРНАЯ СХЕМА МОДУЛЕЙ ПО, СОСТАВЛЯЮЩИХ СИСТЕМУ 9](#_Toc74136716)

[4.1 Структурная модель приложения. Модель MVC 9](#_Toc74136717)

[4.2 Структурная схема HTML страниц на стороне клиента 10](#_Toc74136718)

[5. ДИАГРАММА КЛАССОВ РАЗРАБОТАННОЙ СИСТЕМЫ 11](#_Toc74136719)

[6. ОБЗОР ПО, ВЫБРАННОГО ДЛЯ РЕАЛИЗАЦИИ СИСТЕМЫ 12](#_Toc74136720)

[6.1 Среда разработки – Sublime Text 12](#_Toc74136721)

[6.2 Веб-браузер Google Chrome и инструменты разработчика 12](#_Toc74136722)

[6.3 Программная платформа Node.js 13](#_Toc74136723)

[6.4 СУБД MongoDB 13](#_Toc74136724)

[7. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 15](#_Toc74136725)

[7.1 Главная страница сайта 15](#_Toc74136726)

[7.2 Раздел «Услуги и цены» 17](#_Toc74136727)

[7.3 Раздел «Вопросы и Ответы» 18](#_Toc74136728)

[7.4 Раздел «Контакты» 19](#_Toc74136729)

[7.6 Вход в систему 21](#_Toc74136730)

[7.7 Личный кабинет пациента 22](#_Toc74136731)

[7.8 Личный кабинет врача 23](#_Toc74136732)

[7.9 Личный кабинет администратора 25](#_Toc74136733)

[ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ 27](#_Toc74136734)

[ПРИЛОЖЕНИЕ 1. Файл server.js 28](#_Toc74136735)

[ПРИЛОЖЕНИЕ 2. Файл index.html 29](#_Toc74136736)

[ПРИЛОЖЕНИЕ 3. Файл prices.html 33](#_Toc74136737)

[ПРИЛОЖЕНИЕ 4. Файл faq.html 36](#_Toc74136738)

[ПРИЛОЖЕНИЕ 5. Файл contacts.html 39](#_Toc74136739)

[ПРИЛОЖЕНИЕ 6. Файл doctor.html 43](#_Toc74136740)

[ПРИЛОЖЕНИЕ 7. Файл patient.html 44](#_Toc74136741)

[ПРИЛОЖЕНИЕ 8. Файл admin.html 45](#_Toc74136742)

[ПРИЛОЖЕНИЕ 9. Файл style.css 46](#_Toc74136743)

[ПРИЛОЖЕНИЕ 10. Файл accounts.css 54](#_Toc74136744)

[ПРИЛОЖЕНИЕ 11. Файл login.js 56](#_Toc74136745)

[ПРИЛОЖЕНИЕ 12. Файл register.js 57](#_Toc74136746)

[ПРИЛОЖЕНИЕ 13. Файл patient.js 59](#_Toc74136747)

[ПРИЛОЖЕНИЕ 14. Файл admin.js 65](#_Toc74136748)

[ПРИЛОЖЕНИЕ 15. Файл doctor.js 68](#_Toc74136749)

# ВВЕДЕНИЕ

Даже в крупных городах количество ветеринарных лечебниц нельзя назвать значительным по отношению к количеству потенциальных клиентов. Что касается небольших населённых пунктов, то там зачастую работает одна муниципальная ветеринарная клиника, уровень которой оставляет желать лучшего.

Но этот рынок не растёт стремительными темпами, так как заболевший человек почти наверняка обратится к врачу, но вот привезти своё захворавшее животное к ветеринару решится не каждый. Это связано в первую очередь с ценами, берегут своих питомцев только по-настоящему любящие хозяева. Однако животные нуждаются не только в лечении, но и в профилактике заболеваний не меньше людей, и наличие высококлассных профессионалов, которые в любое время могут оказать врачебную помощь просто необходимо в любом городе.

По данным ВЦИОМ, 76% россиян имеют домашних животных. Чаще всего — кошек, на втором месте — собаки, а на третьем — аквариумные рыбки. Животных заводят в дань моде, для спасения от одиночества, по просьбам детей, да и просто — ради удовольствия. Для любимых зверушек хозяева желают самого лучшего: качественных кормов, разнообразных игрушек и — конечно — медицинского сервиса на высшем уровне. Поэтому ветеринарный бизнес в нашей стране приобретает все бόльшую популярность и становится довольно прибыльным делом. Его развитие можно отследить по трем основным направлениям: ветеринарные клиники, ветаптеки и ветмагазины.

В век стремительного развития Интернет-технологий, практически каждое предприятие имеет свой Интернет-ресурс, благодаря которому потенциальный клиент может ознакомиться, например, с перечнем предоставляемых услуг, списком цен, отзывами. Кроме того, web-сервисы позволяют автоматизировать перечень некоторых необходимых действий, что позволит значительно ускорить и упростить процесс получения необходимой услуги. Всё это позволяет реализовать технология клиент-серверного взаимодействия.

# 1. ПОСТАНОВКА ЗАДАЧИ

Целью курсовой работы является разработка и программная реализация клиент-серверного приложения, а также закрепления знаний, полученных при изучении дисциплины и получение практических навыков программирования информационных систем с использованием современных технологий и инструментальных средств, и оформления сопровождающей документации программного обеспечения.

Задача курсовой работы состоит в разработке клиент-серверного приложения «Ветеринарная клиника» (web-клиента и web-сервера). Для выполнения этой задачи, были определены следующие требования:

* произвольная среда разработки – удобный современный текстовый редактор;
* применение системы контроля версий – git;
* опора на представление «Model-View-Controller»;
* разработка web-клиента с применением html, css, JavaScript, jQuery;
* разработка web-сервера с применением JSON, AJAX, Node.js, Express и NPM;
* использование базы данных (БД) на стороне web-сервера с применением СУБД – MongoDB;
* разворачивание ПО web-сервера либо на отдельном хосте, либо в среде виртуальной машины VirtualBox.

Актуальность работы заключается в том, что на технологии клиент-серверного взаимодействия строится практически вся сеть Интернет. Технология позволяет передавать информацию от серверного устройства к устройствам-клиентам на расстоянии.

Объектом моей курсовой работы является создание клиент-серверного приложения «Ветеринарная клиника».

Предметом работы являются языки программирования для разработки приложений web-сервера и web-клиента.

# 2. МОДЕЛЬ ДАННЫХ СИСТЕМЫ

В данной работе используется уже ставшая классической модель Web-приложений на базе AJAX (см. рис. 1). Основная идея AJAX – возможность для приложения получать информацию и отдавать ее другим компьютерам без перезагрузки веб-страницы.



Рис. 1. Модель Web-приложений на базе AJAX

Страница посредством JavaScript в браузере пользователя, по какому-либо событию с помощью объекта XMLHttpRequest подает запрос серверу, обращаясь к некоему серверному обработчику, передавая ему некоторые параметры, например, значения, введенные пользователем в поле «Имя пользователя» на форме регистрации. Обработчик принимает этот запрос и обрабатывает его. По завершению обработки данных, обработчик выдает эти данные обратно браузеру пользователя. Браузер принимает эти данные и использует для отображения на загруженной странице. Например, рядом с полем «Имя пользователя» появится надпись «имя свободно» или «имя занято».

# 3. МАКЕТЫ HTML-СТРАНИЦ СИСТЕМЫ

## 3.1 Макет главной страницы

Макет главной страницы разрабатываемого Интернет-ресурса ветеринарной клиники, выполненный в приложении “paint.exe”, изображен на рис. 2:

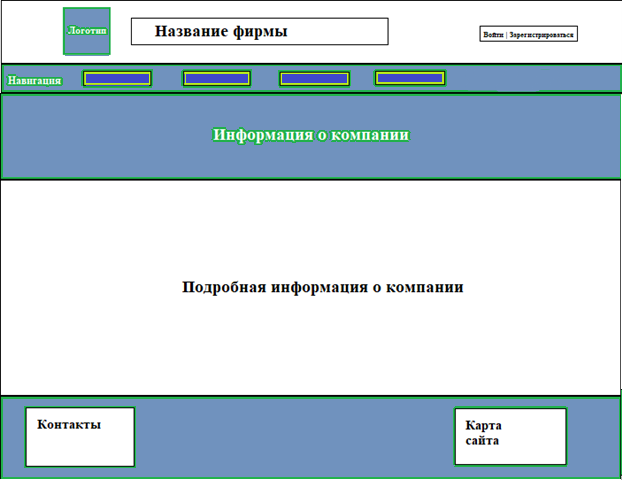


Рис. 2. Макет главной страницы

## 3.2 Макет страницы «Контакты»

Макет страницы «Контакты» разрабатываемого Интернет-ресурса ветеринарной клиники, выполненный в приложении “paint.exe”, изображен на рис. 3:

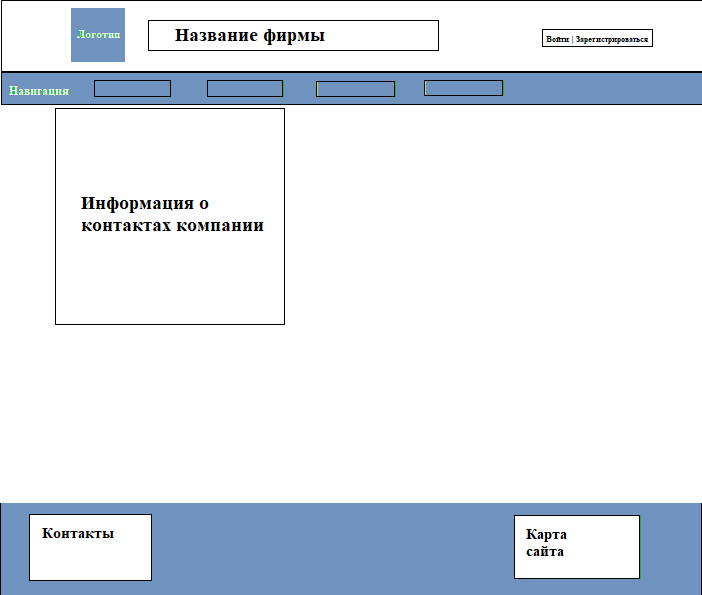


Рис. 3. Макет страницы «Контакты»

## 3.3 Макет страницы «Личный кабинет»

Макет страницы «Личный кабинет» разрабатываемого Интернет-ресурса ветеринарной клиники, выполненный в приложении “paint.exe”, изображен на рис. 4:

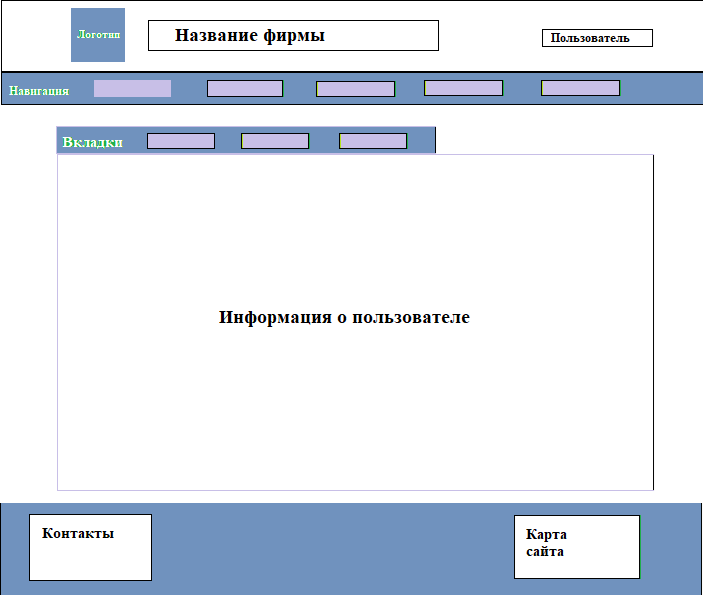


Рис. 4. Макет страницы «Личный кабинет»

# 4. СТРУКТУРНАЯ СХЕМА МОДУЛЕЙ ПО, СОСТАВЛЯЮЩИХ СИСТЕМУ

## 4.1 Структурная модель приложения. Модель MVC

Наша система использует шаблон «модель – представление – контроллер» (Model – View – Controller (MVC)) (см. рис. 5).

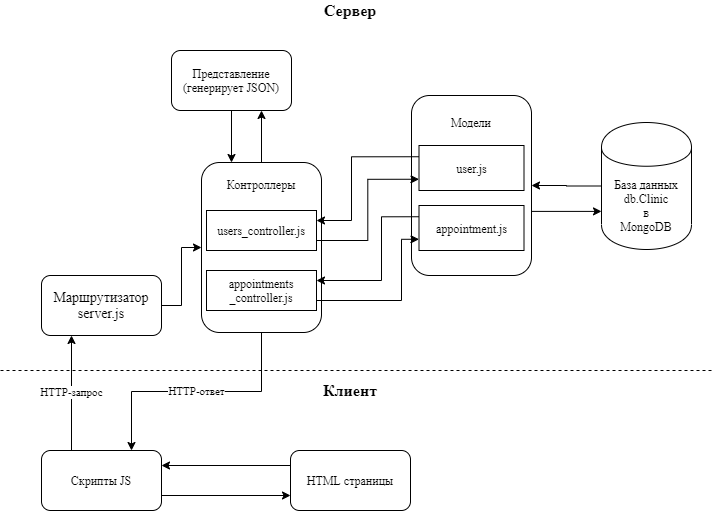


Рис. 5. Структурная схема приложения

Скрипты JS клиента (app.js, login.js, register.js, admin.js, patient.js, doctor.js) в зависимости от действий пользователя генерируют HTTP запросы.

Маршрутизатор – обрабатывает HTTP запрос с соответствующим действием контроллера.

Контроллер – преобразует запрос к серверу в действие. Как правило, соотносится с действием базы данных через модель. После этого отправляет ответ через представление.

Модель – это объектная абстракция элементов в базе данных, с ней взаимодействует контроллер.

Представление – в нашем случае это код HTML и CSS на клиентской стороне. Контроллер отправляет данные в виде JSON клиенту, а клиент решает, как их представить.

## 4.2 Структурная схема HTML страниц на стороне клиента

Структурная схема клиента, представленная HTML страницами, представлена на рис. 6

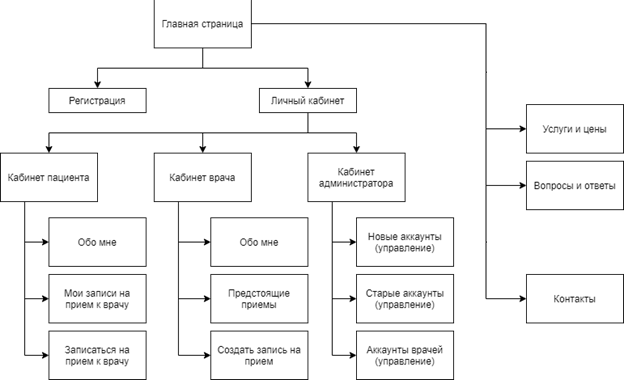


Рис. 6. Структурная схема клиента

# 5. ДИАГРАММА КЛАССОВ РАЗРАБОТАННОЙ СИСТЕМЫ

Диаграмма классов разработанной системы изображена на рис. 7:

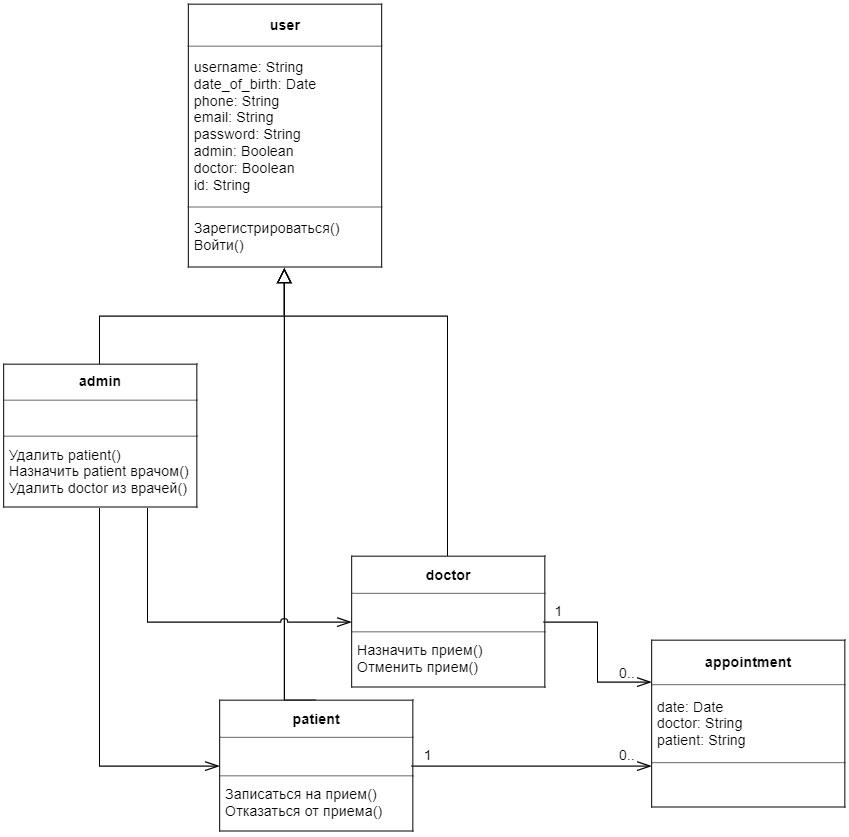


Рис. 7. Диаграмма классов разработанной системы

# 6. ОБЗОР ПО, ВЫБРАННОГО ДЛЯ РЕАЛИЗАЦИИ СИСТЕМЫ

## 6.1 Среда разработки – Sublime Text

Sublime Text — проприетарный текстовый редактор. Поддерживает плагины на языке программирования Python.

Разработчик позволяет бесплатно и без ограничений ознакомиться с продуктом, однако программа уведомляет о необходимости приобретения лицензии.

Некоторые возможности:

* Быстрая навигация (Goto Anything)
* Командная палитра (Command Palette)
* API плагинов на Python
* Одновременное редактирование (Split Editing)
* Высокая степень настраиваемости (Customize Anything)
* Поддержка языков

Sublime Text поддерживает большое количество языков программирования и имеет возможность подсветки синтаксиса для C, C++, C#, CSS, D, Dylan, Erlang, HTML, Groovy, Haskell, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, MATLAB, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL и XML.

В дополнение к тем языкам программирования, которые включены по умолчанию, пользователи имеют возможность загружать плагины для поддержки других языков.

## 6.2 Веб-браузер Google Chrome и инструменты разработчика

Google Chrome — браузер, разрабатываемый компанией Google на основе свободного браузера Chromium и движка Blink (до апреля 2013 года использовался WebKit).

Google Chrome направлен на повышение безопасности, скорости и стабильности. А 9 июня 2009 года разработчики Google Chrome в своём интервью для неофициального сообщества пользователей обозначили стратегию дальнейшего развития своего продукта, которая заключается в том, что браузер будет иметь максимально высокую скорость работы и минимально допустимую функциональность, а все дополнительные функции планируется внедрять при помощи расширений.

Инструменты разработчика Chrome позволяют быстро анализировать содержание и ресурсы веб-страницы. Они полезны для проверки тегов Менеджера кампаний.

## 6.3 Программная платформа Node.js

Node или Node.js — программная платформа, основанная на движке V8 (транслирующем JavaScript в машинный код), превращающая JavaScript из узкоспециализированного языка в язык общего назначения. Node.js добавляет возможность JavaScript взаимодействовать с устройствами ввода-вывода через свой API (написанный на C++), подключать другие внешние библиотеки, написанные на разных языках, обеспечивая вызовы к ним из JavaScript-кода. Node.js применяется преимущественно на сервере, выполняя роль веб-сервера, но есть возможность разрабатывать на Node.js и десктопные оконные приложения (при помощи NW.js, AppJS или Electron для Linux, Windows и macOS) и даже программировать микроконтроллеры (например, tessel, low.js и espruino). В основе Node.js лежит событийно-ориентированное и асинхронное (или реактивное) программирование с неблокирующим вводом/выводом.

В состав Node.js входит собственный установщик пакетов npm.

## 6.4 СУБД MongoDB

MongoDB — документоориентированная система управления базами данных с открытым исходным кодом, не требующая описания схемы таблиц. Классифицирована как NoSQL, использует JSON-подобные документы и схему базы данных. Написана на языке C++. Используется в веб-разработке, в частности, в рамках JavaScript-ориентированного стека MEAN.

Система поддерживает ad-hoc-запросы: они могут возвращать конкретные поля документов и пользовательские JavaScript-функции. Поддерживается поиск по регулярным выражениям. Также можно настроить запрос на возвращение случайного набора результатов.

Имеется поддержка индексов.

Система может работать с набором реплик, то есть содержать две или более копии данных на различных узлах. Каждый экземпляр набора реплик может в любой момент выступать в роли основной или вспомогательной реплики. Все операции записи и чтения по умолчанию осуществляются с основной репликой. Вспомогательные реплики поддерживают в актуальном состоянии копии данных. В случае, когда основная реплика дает сбой, набор реплик проводит выбор, которая из реплик должна стать основной. Второстепенные реплики могут дополнительно являться источником для операций чтения.

Система масштабируется горизонтально, используя технику сегментирования (англ. sharding) объектов баз данных — распределение их частей по различным узлам кластера. Администратор выбирает ключ сегментирования, который определяет, по какому критерию данные будут разнесены по узлам (в зависимости от значений хэша ключа сегментирования). Благодаря тому, что каждый узел кластера может принимать запросы, обеспечивается балансировка нагрузки.

Система может быть использована в качестве файлового хранилища с балансировкой нагрузки и репликацией данных (функция Grid File System; поставляется вместе с драйверами MongoDB). Предоставляются программные средства для работы с файлами и их содержимым. GridFS используется в плагинах для Nginx и lighttpd. GridFS разделяет файл на части и хранит каждую часть как отдельный документ.

Может работать в соответствии с парадигмой MapReduce. Во фреймворке для агрегации есть аналог SQL-инструкции GROUP BY. Операторы агрегации могут быть связаны в конвейер подобно UNIX-конвейрам. Фреймворк так же имеет оператор $lookup для связки документов при выгрузке и статистические операции такие как среднеквадратическое отклонение.

Поддерживается JavaScript в запросах, функциях агрегации (например, в MapReduce).

MongoDB поддерживает коллекции с фиксированным размером. Такие коллекции сохраняют порядок вставки и по достижении заданного размера ведут себя как кольцевой буфер.

В июне 2018 года (в версии 4.0) добавлена поддержка транзакций, удовлетворяющих требованиям ACID.

# 7. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

## 7.1 Главная страница сайта

Для запуска программы необходимо в адресной строке веб-браузера необходимо ввести адрес сайта «Ветеринарная клиника “Фе́нрир”» и нажать клавишу Enter. Будет запущена главная страница сайта (см. рис. 8).

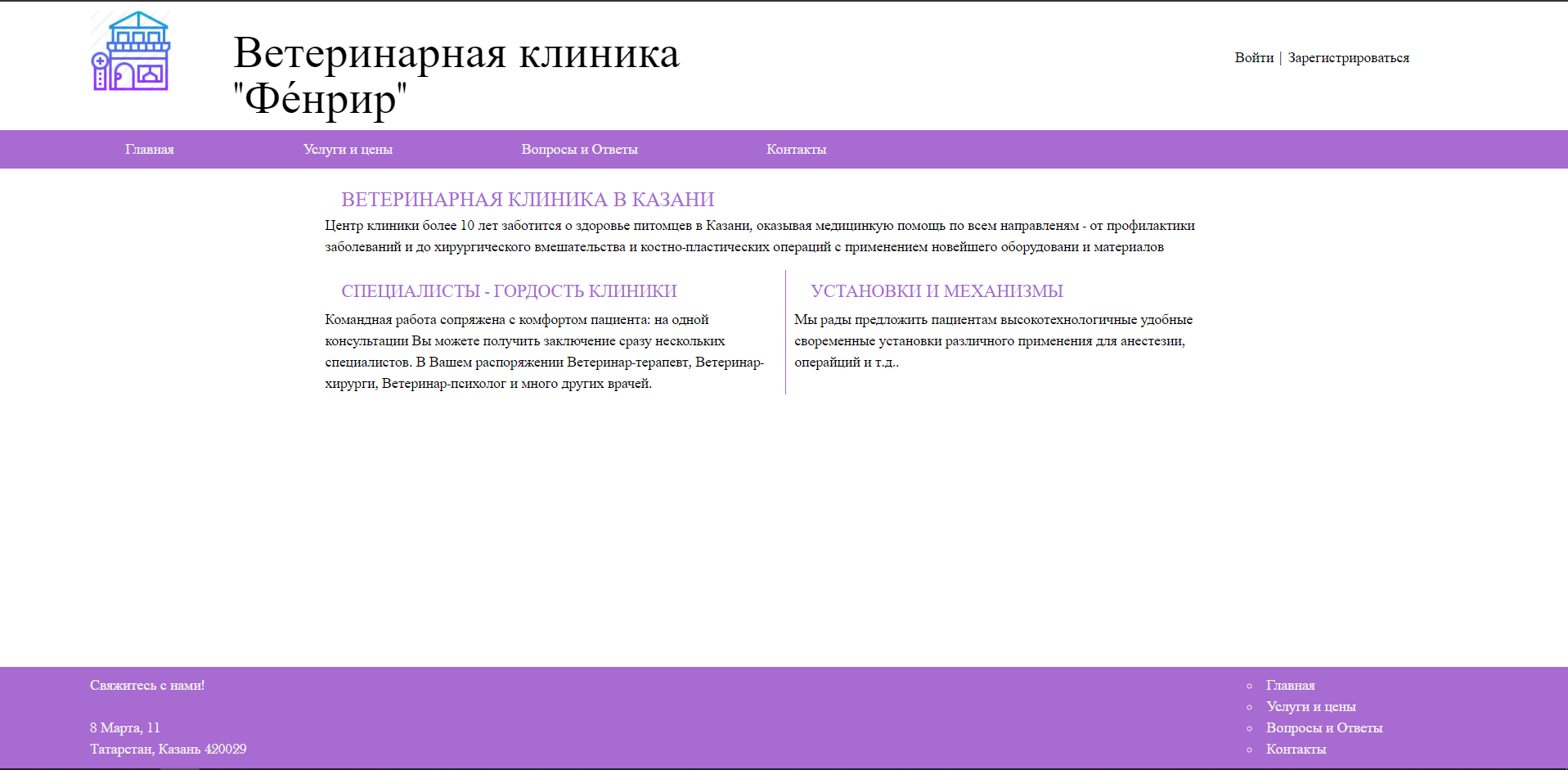


Рис. 8. Главная страница сайта «Ветеринарная клиника “Фе́нрир”»

* В «шапке» страницы изображена эмблема клиники, её название, а также кнопки «Войти» и «Зарегистрироваться», функции которых описаны в п/п 7.8 и 7.7 соответственно;
* В строке меню (см. рис. 9) Вы можете перейти в другие разделы сайта: «Услуги и цены», «Вопросы и Ответы», «Контакты».



Рис. 9. Строка меню

* В средней части страницы Вы наблюдаете общую (рекламную) информацию о клинике, которая представляет из себя статичный текст.
* В «подвале» страницы (см. рис. 10) слева Вы наблюдаете контакты клиники, а справа – ссылки на разделы сайта (дублируют ссылки из верхнего меню)



Рис. 10. «Подвал» сайта

## 7.2 Раздел «Услуги и цены»

Данный раздел (см. рис. 11) сайта содержит общую информацию о предоставляемых клиникой услугах, и их цен.

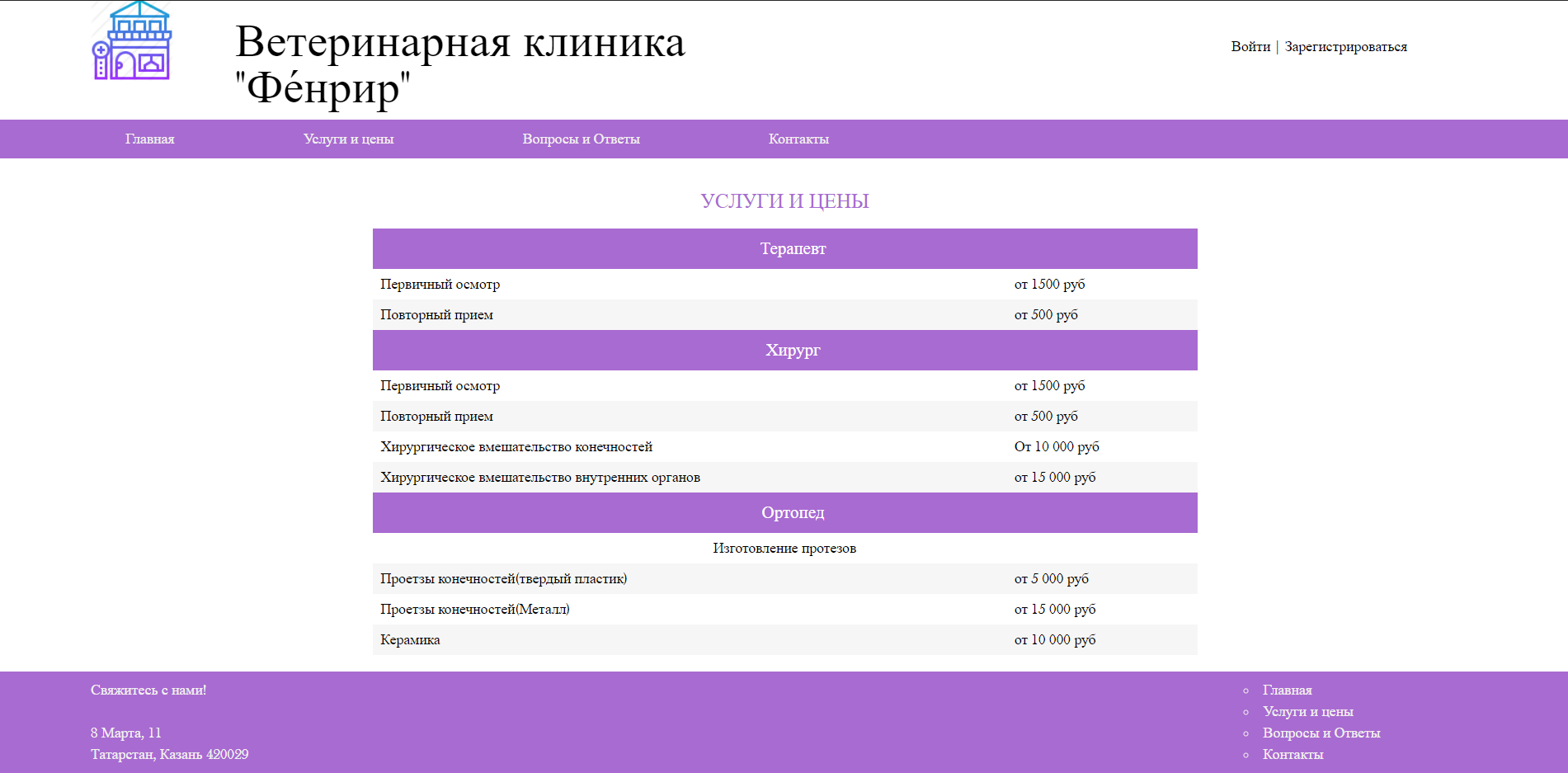


Рис. 11. Раздел «Услуги и цены»

## 7.3 Раздел «Вопросы и Ответы»

Данный раздел (см. рис. 12) сайта отвечает на самые часто задаваемые вопросы пациентов клиники.

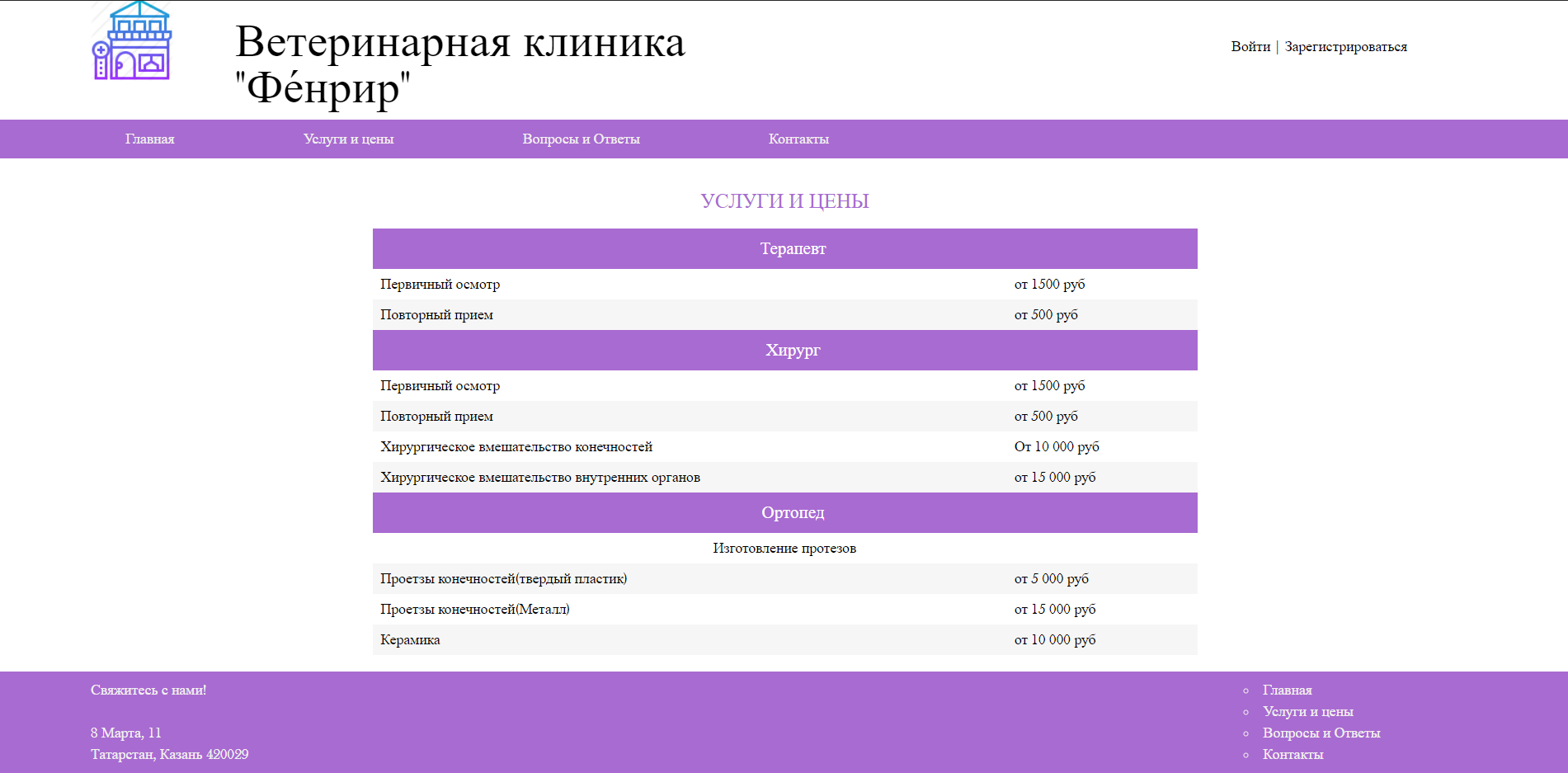


Рис. 12. Раздел «Вопросы и Ответа»

## 7.4 Раздел «Контакты»

В этом разделе сайта (см. рис. 13) можно узнать такую важную информацию о клинике, как её адрес. Можно ознакомиться с интерактивной картой расположения клиники, а также увидеть на интерактивной панораме место входа в клинику.

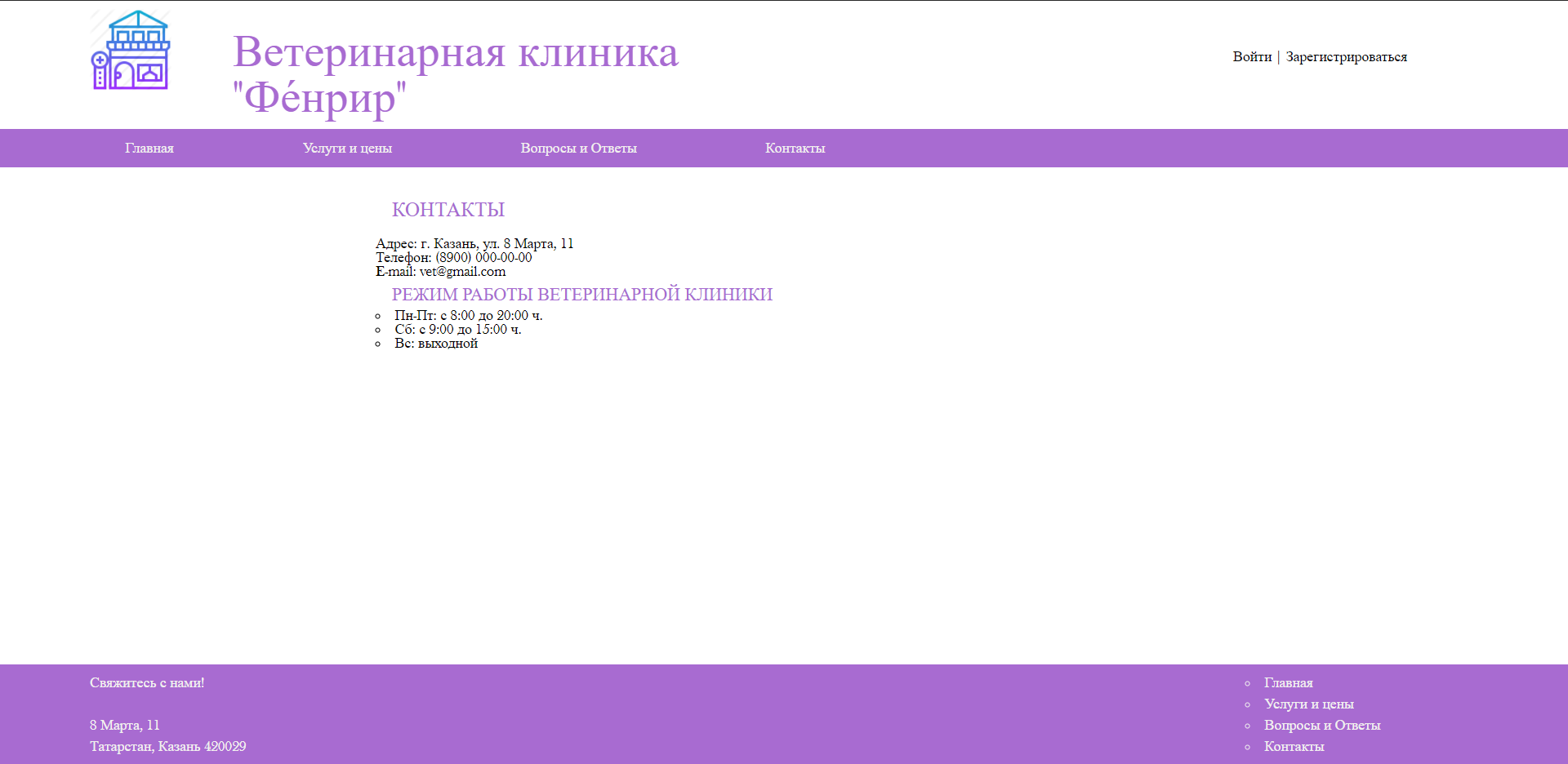


Рис. 13. Раздел «Контакты»

7.5 Форма регистрации пользователя в системе

Для того, чтобы воспользоваться всеми возможностями данной информационной системы, для начала необходимо в ней зарегистрироваться. Для регистрации в системе, нажмите по ссылке «Зарегистрироваться» в шапке сайта (см. рис. 14.):



Рис. 14. Ссылка «Зарегистрироваться»

После нажатия на ссылку, вы увидите форму регистрации пользователя (см. рис. 15):

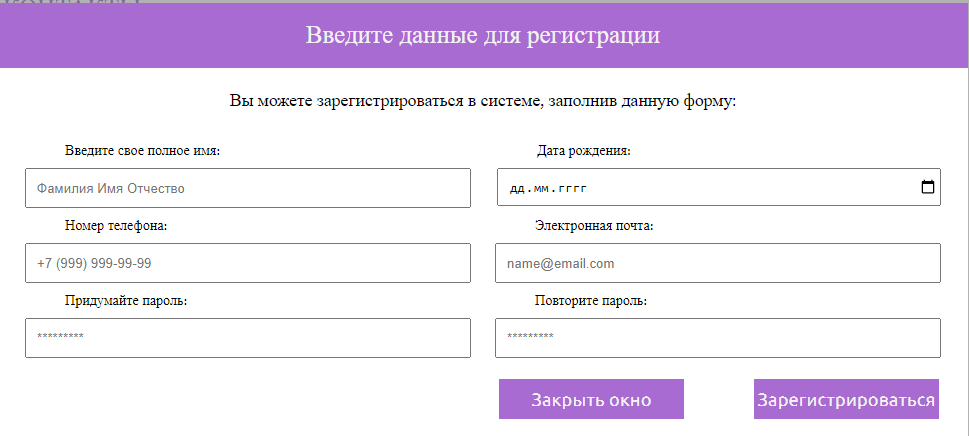


Рис. 15. Форма регистрации в системе

В данной форме необходимо заполнить все поля ввода данными (ФИО, дата рождения, номер телефона, электронная почта, пароль). После заполнения всех полей, необходимо нажать на кнопку «Зарегистрироваться» в правом нижнем углу формы.

Если все действия были выполнены верно, Вы увидите сообщение об успешной регистрации в системе.

Если Вы увидели ошибку – следуя инструкции, повторите ввод.

## 7.6 Вход в систему

Если Вы еще не зарегистрированы в системе, перейдите к пункту 7.5. Для использования информационной системы, необходимо в неё войти под своей электронной почтой и паролем. Для этого нажмите ссылку «Войти» в шапке сайта (см. рис. 16):

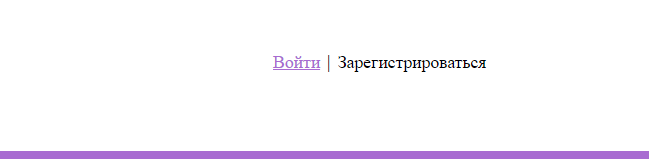


Рис. 16. Ссылка «Войти»

После нажатия на ссылку, вы увидите форму для входа пользователя в систему (см. рис. 17):

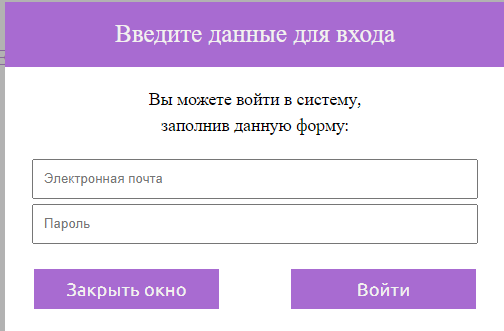


Рис. 17. Форма для входа пользователя в систему

В данной форме необходимо заполнить все поля ввода данными (электронная почта и пароль). После заполнения всех полей, необходимо нажать на кнопку «Войти» в правом нижнем углу формы.

Если все действия были выполнены верно, Вы автоматически перейдете в личный кабинет пользователя (пациента, врача или администратора), о нём см. п/п 7.7.

Если Вы увидели ошибку – проверьте вводимые данные и повторите ввод.

## 7.7 Личный кабинет пациента

Если Вы обычный клиент ветеринарной клиники, то, при входе в систему, вы попадете в личный кабинет пациента (см. рис. 18):

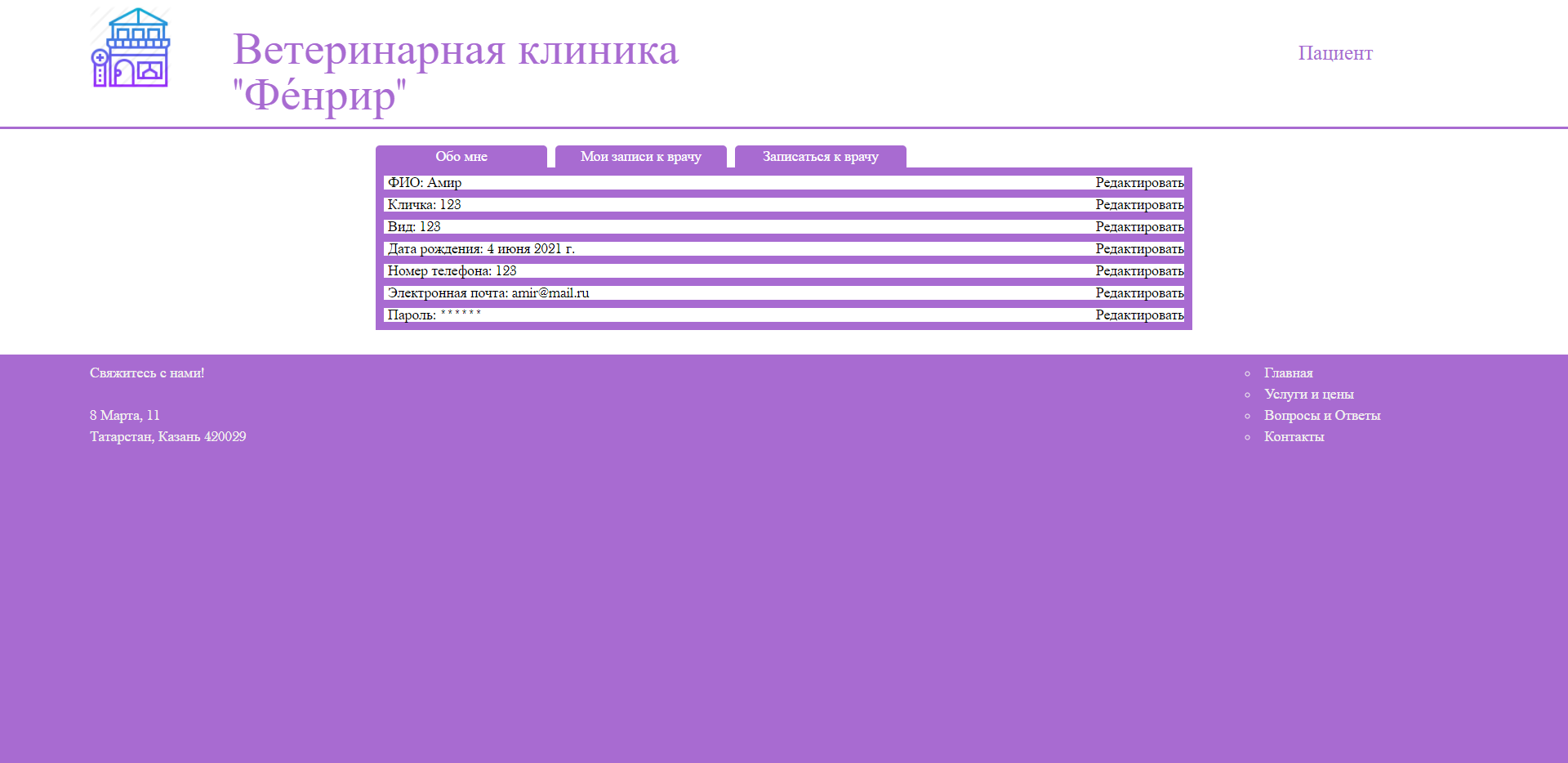


Рис. 18. Личный кабинет пациента

Личный кабинет имеет три вкладки: «Обо мне» - открыта по умолчанию, «Мои записи к врачу» и «Записаться к врачу».

Вкладка «Обо мне» содержит информацию о владельце личного кабинета: ФИО, дату рождения, номер телефона, электронную почту и пароль (скрыт). Любые из названных данных можно изменить, нажав на ссылку «Редактировать» в соответствующем поле. При нажатии «Редактировать» появится всплывающее окно для ввода новых данных (см. рис. 19), где Вы можете ввести новое значение поля и, нажав кнопку «ОК», сохранить его.

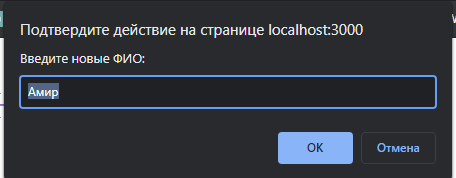


Рис. 19. Всплывающее окно редактирования

Вкладка «Мои записи к врачу» (см. рис. 20) содержит таблицу, в которой записаны все предстоящие приемы у врача (учитывается время приема, просроченные приемы в список не выводятся). При первом входе список пуст (Вы еще не записались ни на один прием), для его пополнения нужно перейти во вкладку «Записаться к врачу» (читайте об этом далее). Если вы записались на прием по ошибке, либо ваши планы изменились, Вы всегда можете отменить свою запись, нажав кнопку «Удалить», и подтвердив это во всплывающем диалоговом окне.

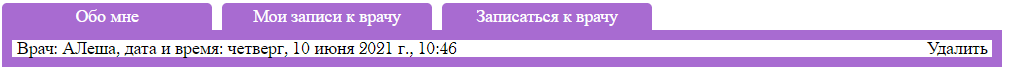


Рис. 20. Список предстоящих приемов у пациента

Вкладка «Записаться к врачу» создана для того, чтобы Вы могли записаться к тому врачу, к которому Вы хотите, и на определенное, удобное для Вас время. Перейдя в эту вкладку Вы увидите (см. рис. 21) поле с выбором интересующего Вас врача. Нажав на поле, выберите врача. При этом станут доступны еще одно поле и кнопка. Во втором поле Вы можете выбрать удобное для приема время. В этом поле показываются доступные приемы именно для того врача, которого вы выбрали. Подтвердите свою запись, нажав кнопку «Записаться». Во всплывающем диалоговом окне подтвердите свое действие.

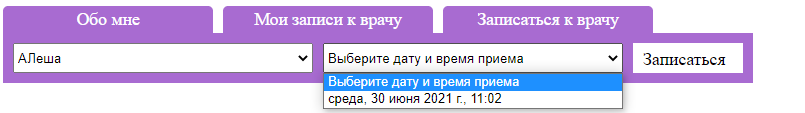


Рис. 21. Вкладка «Записаться к врачу»

## 7.8 Личный кабинет врача

Если Вы являетесь врачом ветеринарной клиники, то, при входе в систему, вы попадете в личный кабинет врача (см. рис. 22):

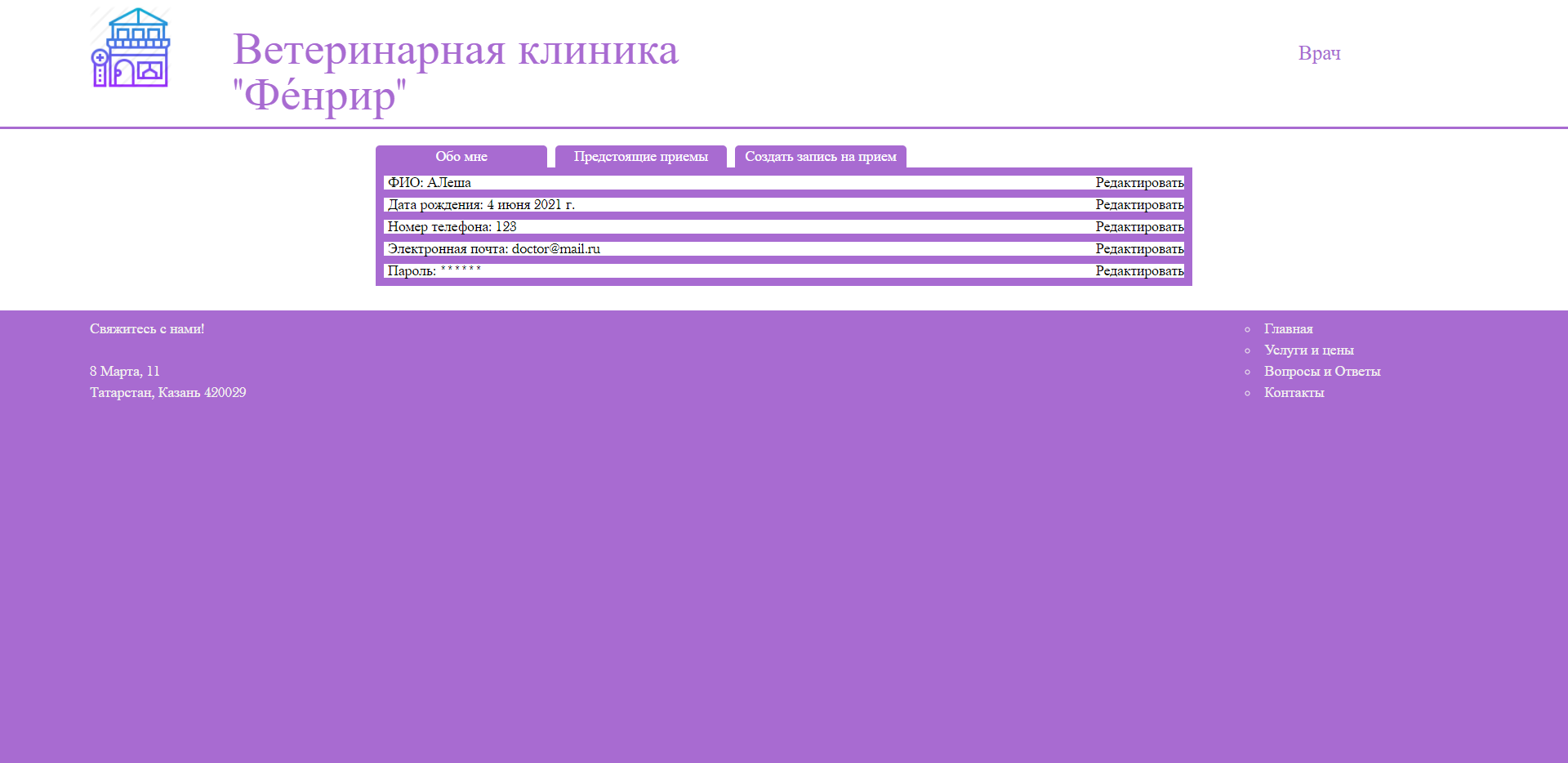


Рис. 22. Личный кабинет врача

Личный кабинет врача имеет три вкладки: «Обо мне» - открыта по умолчанию, «Предстоящие приемы», «Создать запись на прием».

Вкладка «Обо мне» содержит информацию о владельце личного кабинета: ФИО, дату рождения, номер телефона, электронную почту и пароль (скрыт). Любые из названных данных можно изменить, нажав на ссылку «Редактировать» в соответствующем поле. При нажатии «Редактировать» появится всплывающее окно для ввода новых данных (см. рис. 19), где Вы можете ввести новое значение поля и, нажав кнопку «ОК», сохранить его.

Вкладка «Предстоящие приемы» (см. рис. 23) содержит таблицу, в которой записаны все назначенные Вами приемы (учитывается время приема, просроченные приемы в список не выводятся). При первом входе список пуст (Вы еще не назначили ни один прием), для его пополнения нужно перейти во вкладку «Создать запись на прием» (читайте об этом далее). Если вы создали запись на прием по ошибке, либо ваши планы изменились, Вы всегда можете отменить свою запись, нажав кнопку «Удалить», и подтвердив это во всплывающем диалоговом окне. Если кто-то уже записался на Ваш прием, то в списке будет указана ФИО пациента.

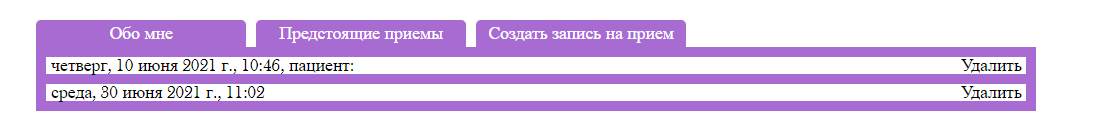


Рис. 23. Вкладка «Предстоящие приемы»

Перейдя во вкладку «Создать запись на прием» (см. рис. 25), Вы увидите поле для ввода даты и времени, а также кнопку подтверждения создания записи. Выберите нужную дату и время приема, нажмите кнопку «Создать запись» и подтвердите действие во всплывающем окне. Если всё было сделано верно, Вам об этом будет сообщено. При этом у Вас появится новая запись во вкладке «Предстоящие приемы», а также пациенты смогут записаться на это время через свой личный кабинет.

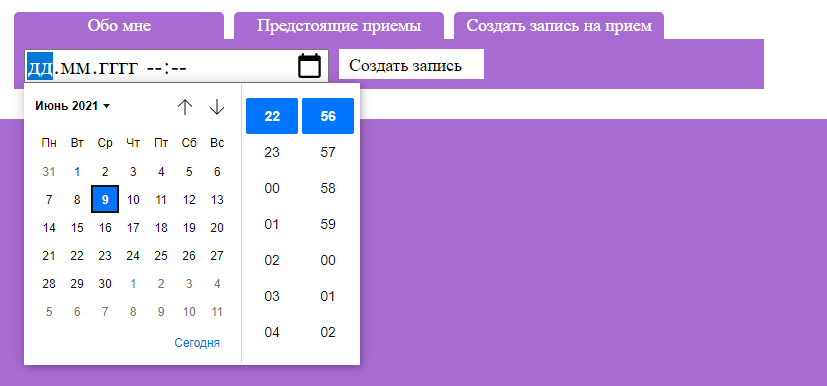


Рис. 25. Вкладка «Создание записи на прием»

## 7.9 Личный кабинет администратора

Если Вы являетесь администратором ветеринарной клиники, то, при входе в систему, вы попадете в личный кабинет администратора (см. рис. 26):

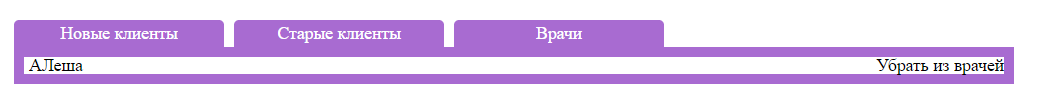


Рис. 26. Личный кабинет администратора

Личный кабинет администратора содержит три вкладки: «Новые клиенты» - открыта по умолчанию, «Старые клиенты» и «Врачи».

Во вкладке «Новые клиенты» показан список всех зарегистрированных пациентов клиники в порядке от вновь зарегистрировавшихся к давно зарегистрировавшимся. Если зарегистрированный пациент на самом деле зарегистрировался с целью работы в системе в роли врача, имеется возможность перевести аккаунт из разряда «пациент» в разряд «врач», нажав ссылку «Сделать врачом» в соответствующей строке с пациентом. Необходимо подтвердить свои действия во всплывающем диалоговом окне. Кроме того, имеется возможность удаления пациентов из системы нажатием ссылки «Удалить» в соответствующей строке списка. Это действие необходимо подтвердить во всплывающем диалоговом окне.

Во вкладке «Старые клиенты» (см. рис. 27) показан список всех зарегистрированных пациентов клиники в порядке от давно зарегистрировавшихся к вновь зарегистрировавшимся. Если зарегистрированный пациент на самом деле зарегистрировался с целью работы в системе в роли врача, имеется возможность перевести аккаунт из разряда «пациент» в разряд «врач», нажав ссылку «Сделать врачом» в соответствующей строке с пациентом. Необходимо подтвердить свои действия во всплывающем диалоговом окне. Кроме того, имеется возможность удаления пациентов из системы нажатием ссылки «Удалить» в соответствующей строке списка. Это действие необходимо подтвердить во всплывающем диалоговом окне.

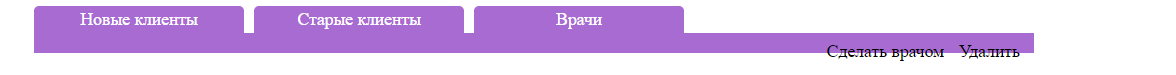


Рис. 27. Вкладка «Старые клиенты»

Во вкладке «Врачи» (см рис. 28) показан список всех врачей, зарегистрированных в системе ветеринарной клинике. При этом имеется возможность перевода аккаунта из разряда «врач» в разряд «пациенты» нажатием кнопки «Убрать из врачей» в соответствующей строке списка врачей. Это действие требует подтверждения во всплывающем диалоговом окне. Функция предназначена на случай ошибочных действий администратора, либо при увольнении специалиста с должности врача. При этом все предстоящие приемы удаляются из системы, кроме тех, которые уже были проведены (остаются в системе для отчетности)

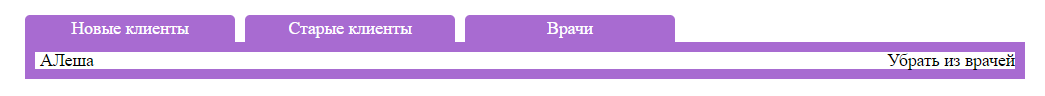


Рис. 28. Вкладка «Врачи»

# ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ

1. Справочник по HTML. // HTMLBOOK [Электронный ресурс]. – Режим доступа: http://htmlbook.ru. – (Дата обращения: 14.05.2020).

2. Флэнаган Д. JavaScript. Подробное руководство. – Пер. с англ. – СПб: Символ-Плюс, 2008. – 992 с., ил.

3. Кириченко А.В. HTML5+CSS3. Основы современного WEB-дизайна. – СПб: «Наука и Техника», 2018 г. – 352 с., ил.

4. Онлайн-руководство по MongoDB. // METANIT – сайт о программировании [Электронный ресурс]. – Режим доступа: https://metanit.com/nosql/mongodb/. – (Дата обращения: 26.05.2020)

5. JQuery // Википедия [Электронный ресурс]. Дата обновления: 29.01.2020. – Режим доступа: https://ru.wikipedia.org/?oldid=104831452 (дата обращения: 01.06.2020).

# ПРИЛОЖЕНИЕ 1. Файл server.js

|  |
| --- |
| var express = require("express"), |
|  | http = require("http"), |
|  | app = express(), |
|  | mongoose = require("mongoose"), |
|  | UsersController = require("./controllers/users\_controller.js"); |
|  | AppoController = require("./controllers/appointments\_controller.js"); |
|  |  |
|  | app.use('/',express.static(\_\_dirname + "/client")); |
|  |  |
|  | http.createServer(app).listen(3000); |
|  | // объекты JSON |
|  | app.use(express.urlencoded({ extended: true })); |
|  | // подключаемся к хранилищу данных Amazeriffic в Mongo |
|  | mongoose.connect('mongodb://localhost/clinic', { |
|  | useNewUrlParser: true, |
|  | useCreateIndex: true, |
|  | useUnifiedTopology: true |
|  | }).then(res => { |
|  | console.log("DB Connected!") |
|  | }).catch(err => { |
|  | console.log(Error, err.message); |
|  | }); |
|  |  |
|  | app.get("/users.json", UsersController.index); |
|  | app.post("/users", UsersController.create); |
|  | app.post("/accounts", UsersController.login); |
|  | app.get("/users/:id", UsersController.show); |
|  | app.get("/users/:id/account", UsersController.getaccount); |
|  | app.put("/users/:id", UsersController.update); |
|  | app.delete("/users/:id", UsersController.destroy); |
|  |  |
|  | app.get("/appoint.json", AppoController.index); |
|  |  |
|  | app.get("/users/:id/appoint.json", AppoController.index); |
|  | app.post("/users/:id/appoint", AppoController.create); |
|  | app.put("/users/:id/appoint/:idappo", AppoController.update); |
|  | app.delete("/users/:id/appoint/:idappo", AppoController.destroy); |

# ПРИЛОЖЕНИЕ 2. Файл index.html

<!doctype html>

<html>

<head>

<title>Ветеринарная клиника "Фе́нрир"</title>

<link rel="shortcut icon" href="images/i.svg" type="image/x-icon">

<link rel="stylesheet" type="text/css" href="css/reset.css">

<link rel="stylesheet" type="text/css" href="css/style.css">

<link rel="stylesheet" type="text/css" href="fonts/Droid-Sans.css">

<link rel="stylesheet" type="text/css" href="fonts/Ubuntu.css">

<script src="https://yastatic.net/jquery/3.3.1/jquery.min.js"></script>

<script type="text/javascript" src="js/login.js"></script>

<script type="text/javascript" src="js/register.js"></script>

</head>

<body>

<header>

<a href="index.html"><img src="images/icon1.png" alt="Логотип сайта" width="100px"></a>

<div class="container">

<a href="index.html"><h1>Ветеринарная клиника "Фе́нрир"</h1> </a>

<nav>

<a href="#login\_window">Войти</a> |

<a href="#register\_window">Зарегистрироваться</a>

</nav>

</div>

</header>

<div class="block\_menu">

<ul class="content\_block\_main">

<li><a href="index.html">Главная</a></li>

<li><a href="prices.html">Услуги и цены</a></li>

<li><a href="faq.html">Вопросы и Ответы</a></li>

<li><a href="contacts.html">Контакты</a></li>

</ul>

</div>

<main>

<div class="top-main" >

<h2>ВЕТЕРИНАРНАЯ КЛИНИКА В КАЗАНИ</h2>

<p>

Центр клиники более 10 лет заботится о здоровье питомцев в Казани, оказывая медицинкую помощь по всем направленям - от профилактики заболеваний и до хирургического вмешательства и костно-пластических операций с применением новейшего оборудовани и материалов

</p>

<div class="left-main">

<h3>СПЕЦИАЛИСТЫ - ГОРДОСТЬ КЛИНИКИ</h3>

Командная работа сопряжена с комфортом пациента: на одной консультации Вы можете получить заключение сразу нескольких специалистов. В Вашем распоряжении Ветеринар-терапевт, Ветеринар-хирурги, Ветеринар-психолог и много других врачей.

</div>

<div class="right-main">

<h3>УСТАНОВКИ И МЕХАНИЗМЫ</h3>

Мы рады предложить пациентам высокотехнологичные удобные своременные установки различного применения для анестезии, операйций и т.д..

</div>

</div>

<div style="margin-bottom: 330px"></div>

<!--Всплывающее окно входа-->

<div id="login\_window">

<div id="okno\_login">

<h2 class="h2\_request">

Введите данные для входа

</h2>

<div class="form\_request">

<p>Вы можете войти в систему,<br>заполнив данную форму:</p>

<input type="login" name="login" placeholder="Электронная почта">

<input type="password" name="password" placeholder="Пароль">

</div>

<a href="#" class="close">Закрыть окно</a>

<a class="submit">Войти</a> <!--Пока что тоже только закрывает-->

</div>

</div>

<!---->

<!--Всплывающее окно регистрации-->

<div id="register\_window">

<div id="okno\_register">

<h2 class="h2\_register">

Введите данные для регистрации

</h2>

<div class="form\_request">

<p>Вы можете зарегистрироваться в системе, заполнив данную форму:</p>

<div class="inputbox\_register\_left">

<label for="full\_name">Введите свое полное имя:</label>

<input type="full\_name" name="full\_name" placeholder="Фамилия Имя Отчество">

</div>

<div class="inputbox\_register\_right">

<label for="date">Дата рождения:</label>

<input type="date" name="date" placeholder="Дата рождения">

</div>

<div class="inputbox\_register\_left">

<label for="tel">Номер телефона:</label>

<input type="tel" name="tel" placeholder="+7 (999) 999-99-99">

</div>

<div class="inputbox\_register\_right">

<label for="email">Электронная почта:</label>

<input type="email" name="email" placeholder="name@email.com">

</div>

<div class="inputbox\_register\_left">

<label for="password">Придумайте пароль:</label>

<input type="password" name="password" placeholder="\*\*\*\*\*\*\*\*\*">

</div>

<div class="inputbox\_register\_right">

<label for="password">Повторите пароль:</label>

<input type="password" name="repeat\_password" placeholder="\*\*\*\*\*\*\*\*\*">

</div>

</div>

<a href="#" class="submit\_register">Зарегистрироваться</a> <!--Пока что тоже только закрывает-->

<a href="#" class="close\_register">Закрыть окно</a>

</div>

</div>

<!---->

</main>

<footer>

<div class="container">

<div class="contact">

<h5>Свяжитесь с нами!</h5> <br>

<p>8 Марта, 11</p>

<p>Татарстан, Казань 420029</p>

</div>

<div class="sitemap">

<ul>

<li><a href="index.html">Главная</a></li>

<li><a href="prices.html">Услуги и цены</a></li>

<li><a href="faq.html">Вопросы и Ответы</a></li>

<li><a href="contacts.html">Контакты</a></li>

</ul>

</div>

</div>

</footer>

</body>

</html>

# ПРИЛОЖЕНИЕ 3. Файл prices.html

<!doctype html>

<html>

<head>

<title> Услуги и цены</title>

<link rel="shortcut icon" href="images/i.svg" type="image/x-icon">

<link rel="stylesheet" type="text/css" href="css/reset.css">

<link rel="stylesheet" type="text/css" href="css/style.css">

<link rel="stylesheet" type="text/css" href="fonts/Droid-Sans.css">

<link rel="stylesheet" type="text/css" href="fonts/Ubuntu.css">

<script src="https://yastatic.net/jquery/3.3.1/jquery.min.js"></script>

<script type="text/javascript" src="js/login.js"></script>

<script type="text/javascript" src="js/register.js"></script>

</head>

<body>

<header>

<a href="index.html"><img src="images/icon1.png" alt="Логотип сайта" width="100px"></a>

<div class="container">

<a href="index.html"><h1>Ветеринарная клиника "Фе́нрир"</h1> </a>

<nav>

<a href="#login\_window">Войти</a> |

<a href="#register\_window">Зарегистрироваться</a>

</nav>

</div>

</header>

<div class="block\_menu">

<ul class="content\_block\_main">

<li><a href="index.html">Главная</a></li>

<li><a href="prices.html">Услуги и цены</a></li>

<li><a href="faq.html">Вопросы и Ответы</a></li>

<li><a href="contacts.html">Контакты</a></li>

</ul>

</div>

<main>

<div class="container">

<h2 class='header' align="center">УСЛУГИ И ЦЕНЫ</h2>

<table class="price\_table" rules="rows">

<!-- Терапевт -->

<tr>

<td class="td-name" colspan="2">

<h3>Терапевт</h3>

</td>

</tr>

<tr>

<td> <div>Первичный осмотр</div> </td>

<td> <div> от 1500 руб</div> </td>

</tr>

<tr style="background-color:#F6F6F6;">

<td> <div>Повторный прием</div> </td>

<td> <div> от 500 руб</div> </td>

</tr>

<!-- Хирург -->

<tr>

<td class="td-name" colspan="2">

<h3>Хирург</h3>

</td>

</tr>

<tr>

<td> <div>Первичный осмотр</div> </td>

<td> <div> от 1500 руб</div> </td>

</tr>

<tr style="background-color:#F6F6F6;">

<td> <div>Повторный прием</div> </td>

<td> <div> от 500 руб</div> </td>

</tr>

<tr>

<td> <div>Хирургическое вмешательство конечностей</div> </td>

<td> <div>От 10 000 руб</div> </td>

</tr>

<tr style="background-color:#F6F6F6;">

<td> <div>Хирургическое вмешательство внутренних органов</div> </td>

<td> <div> от 15 000 руб</div> </td>

</tr>

<!-- Ортопед -->

<tr>

<td class="td-name" colspan="2">

<h3>Ортопед</h3>

</td>

</tr>

<tr>

<td colspan="2">

<p>Изготовление протезов</p>

</td>

</tr>

<tr style="background-color:#F6F6F6;">

<td> <div>Проетзы конечностей(твердый пластик)</div> </td>

<td> <div> от 5 000 руб</div> </td>

</tr>

<tr>

<td> <div>Проетзы конечностей(Металл)</div> </td>

<td> <div> от 15 000 руб</div> </td>

</tr>

<tr style="background-color:#F6F6F6;">

<td> <div>Керамика</div> </td>

<td> <div> от 10 000 руб</div> </td>

</tr>

</table>

</div>

<!--Всплывающее окно входа-->

<div id="login\_window">

<div id="okno\_login">

<h2 class="h2\_request">

Введите данные для входа

</h2>

<div class="form\_request">

<p>Вы можете войти в систему,<br>заполнив данную форму:</p>

<input type="login" name="login" placeholder="Электронная почта">

<input type="password" name="password" placeholder="Пароль">

</div>

<a href="#" class="close">Закрыть окно</a>

<a class="submit">Войти</a>

</div>

</div>

<!---->

<!--Всплывающее окно регистрации-->

<div id="register\_window">

<div id="okno\_register">

<h2 class="h2\_register">

Введите данные для регистрации

</h2>

<div class="form\_request">

<p>Вы можете зарегистрироваться в системе, заполнив данную форму:</p>

<div class="inputbox\_register\_left">

<label for="full\_name">Введите свое полное имя:</label>

<input type="full\_name" name="full\_name" placeholder="Фамилия Имя Отчество">

</div>

<div class="inputbox\_register\_right">

<label for="date">Дата рождения:</label>

<input type="date" name="date" placeholder="Дата рождения">

</div>

<div class="inputbox\_register\_left">

<label for="tel">Номер телефона:</label>

<input type="tel" name="tel" placeholder="+7 (999) 999-99-99">

</div>

<div class="inputbox\_register\_right">

<label for="email">Электронная почта:</label>

<input type="email" name="email" placeholder="name@email.com">

</div>

<div class="inputbox\_register\_left">

<label for="password">Придумайте пароль:</label>

<input type="password" name="password" placeholder="\*\*\*\*\*\*\*\*\*">

</div>

<div class="inputbox\_register\_right">

<label for="password">Повторите пароль:</label>

<input type="password" name="repeat\_password" placeholder="\*\*\*\*\*\*\*\*\*">

</div>

</div>

<a href="#" class="submit\_register">Зарегистрироваться</a>

<a href="#" class="close\_register">Закрыть окно</a>

</div>

</div>

<!---->

</main>

<footer>

<div class="container">

<div class="contact">

<h5>Свяжитесь с нами!</h5> <br>

<p>8 Марта, 11</p>

<p>Татарстан, Казань 420029</p>

</div>

<div class="sitemap">

<ul>

<li><a href="index.html">Главная</a></li>

<li><a href="prices.html">Услуги и цены</a></li>

<li><a href="faq.html">Вопросы и Ответы</a></li>

<li><a href="contacts.html">Контакты</a></li>

</ul>

</div>

</div>

</footer>

</body>

</html>

# ПРИЛОЖЕНИЕ 4. Файл faq.html

<!doctype html>

<html>

<head>

<title> Вопросы и ответы</title>

<link rel="shortcut icon" href="images/i.svg" type="image/x-icon">

<link rel="stylesheet" type="text/css" href="css/reset.css">

<link rel="stylesheet" type="text/css" href="css/style.css">

<link rel="stylesheet" type="text/css" href="fonts/Droid-Sans.css">

<link rel="stylesheet" type="text/css" href="fonts/Ubuntu.css">

<script src="https://yastatic.net/jquery/3.3.1/jquery.min.js"></script>

<script type="text/javascript" src="js/login.js"></script>

<script type="text/javascript" src="js/register.js"></script>

</head>

<body>

<header>

<a href="index.html"><img src="images/icon1.png" alt="Логотип сайта" width="100px"></a>

<div class="container">

<a href="index.html"><h1>Ветеринарная клиника "Фе́нрир"</h1> </a>

<nav>

<a href="#login\_window">Войти</a> |

<a href="#register\_window">Зарегистрироваться</a>

</nav>

</div>

</header>

<div class="block\_menu">

<ul class="content\_block\_main">

<li><a href="index.html">Главная</a></li>

<li><a href="prices.html">Услуги и цены</a></li>

<li><a href="faq.html">Вопросы и Ответы</a></li>

<li><a href="contacts.html">Контакты</a></li>

</ul>

</div>

<main>

<div class="container">

<h2 align="center">ВОПРОСЫ И ОТВЕТЫ</h2>

<div class="faq">

<h3>&#9675; КАК УЗНАТЬ РАСПИСАНИЕ ПРИЕМА ПАЦИЕНТОВ?</h3>

<p>Для просмотра режима работы и расписания приема пациентов, перейдите в раздел "<a href="contacts.html" class="href-faq">Контакты</a>".</p>

<h3>&#9675; КАК ЗАПИСАТЬСЯ НА ПРИЕМ?</h3>

<p>Для записи на прием, <a href="#register\_window" class="href-faq">зарегистрируйтесь</a> в системе. Если Вы уже были зарегистрированы, необходимо <a href="#login\_window" class="href-faq">войти</a> в систему. После чего, перейдите в личный кабинет, там Вы сможете выбрать удобное время для приема.</p>

<h3>&#9675; КАК ПРОХОДИТ УСТАНОВКА ПРОТЕЗА?</h3>

<p>Во время первого визита терапев делает первичный осмотр. Далее направляет на ренген. Далее создается необходимого размера протез. Хирург под общим наркозом устанавливает протез.</p>

<h3>&#9675; Я собираюсь со своим животным прийти на прием к врачу. Какие мне надо сдать анализы?</h3>

<p>Вам нужно сначала прийти к врачу. Доктор осмотрит животное, и назначит анализы по результатам осмотра. Если у Вас уже сделаны какие-нибудь исследования, надо взять их с собой.</p>

<h3>&#9675; Сдача анализов</h3>

<p>Для сдачи крови на анализ животное должно быть голодным (8 – 10 часов). Никаких вкусняшек. У животного всегда должен быть доступ к воде. Биоматериал для анализов принимается в специальных контейнерах (баночках) для сбора мочи/ кала. Контейнер можно приобрести в клинике, или в обычной аптеке. Моча и кал на анализ должны быть свежесобранные, и доставлены в клинику до 16 часов того же дня (чтобы анализ был сделан в тот же день и результаты были достоверными) Если кот/ кошка принципиально ходит только в туалет с наполнителем, Вы можете приобрести специальный наполнитель для сбора мочи (пакетик с пластиковыми гранулами, которые не впитывают мочу)</p>

<h3>&#9675; Подскажите, пожалуйста, примерную стоимость постановки брекетов с расходными материалами. / Мы живем в другом городе и хотим поставить брекеты нашей собаке. Сколько это будет стоить?</h3>

<p>Сначала Вам нужно подъехать на консультацию с собакой, чтобы Полина Сергеевна Альберт посмотрела, возможна ли постановка брекет-системы животному. После осмотра она сможет сказать примерную стоимость. Владелец собаки должен ясно понимать: Брекеты ставятся на длительное время (около 6 месяцев или дольше). Все это время за собакой нужен тщательный контроль и уход. Надо каждый день чистить зубы, и следить за тем, чтобы собака ничего не грызла. Необходимо будет приезжать примерно раз в 2 недели показывать животное (если все в порядке). Если собака как-нибудь испортила конструкцию, надо приехать, чтобы ее исправить (даже если только вчера/сегодня утром вы были на приеме). Если в какой-нибудь момент владельцу будет «не до этого», или просто надоест, то результаты нескольких месяцев труда (владельца, врача и собаки) будут испорчены, и придется начинать все сначала (если это будет возможно). Соответственно, если собака живет в другом городе, то все это крайне затруднительно.</p>

<h3>&#9675; Я собираюсь поехать со своим любимцем на поезде/самолете. Какие мне нужны для этого документы?/ Я хочу пойти со своим любимцем на выставку собак/кошек</h3>

<p>Собака/кошка/хорек должны быть чипированы. Чип должен быть поставлен до того, как была сделана вакцинация. У животного должен быть ветеринарный паспорт с действующей вакцинацией против бешенства и инфекционных заболеваний (для каждого вида). Вакцинация должна быть сделана не раньше, чем год назад, и не позже, чем месяц назад. Перед отъездом, или выставкой, надо получить ветеринарную справку, которая действительна 5 дней. Справки выдает только государственная ветеринарная служба. Справка выдается только при наличии животного. Если вы собираетесь ехать за границу, то для получения справки возьмите с собой распечатанные требования страны, в которую вы собираетесь, к ввозу животных. (Требования ЕС все знают, а с другими странами могут быть проблемы). Также обратите внимание на требования к даче глистогонных препаратов.</p>

</div>

</div>

<!--Всплывающее окно входа-->

<div id="login\_window">

<div id="okno\_login">

<h2 class="h2\_request">

Введите данные для входа

</h2>

<div class="form\_request">

<p>Вы можете войти в систему,<br>заполнив данную форму:</p>

<input type="login" name="login" placeholder="Электронная почта">

<input type="password" name="password" placeholder="Пароль">

</div>

<a href="#" class="close">Закрыть окно</a>

<a class="submit">Войти</a> <!--Пока что тоже только закрывает-->

</div>

</div>

<!---->

<!--Всплывающее окно регистрации-->

<div id="register\_window">

<div id="okno\_register">

<h2 class="h2\_register">

Введите данные для регистрации

</h2>

<div class="form\_request">

<p>Вы можете зарегистрироваться в системе, заполнив данную форму:</p>

<div class="inputbox\_register\_left">

<label for="date">Введите свое полное имя:</label>

<input type="full\_name" name="Фамилия Имя Отчество" placeholder="Фамилия Имя Отчество">

</div>

<div class="inputbox\_register\_right">

<label for="date">Дата рождения:</label>

<input type="date" name="date" placeholder="Дата рождения">

</div>

<div class="inputbox\_register\_left">

<label for="tel">Номер телефона:</label>

<input type="tel" name="Номер телефона" placeholder="+7 (999) 999-99-99">

</div>

<div class="inputbox\_register\_right">

<label for="email">Электронная почта:</label>

<input type="email" name="Электронная почти" placeholder="name@email.com">

</div>

<div class="inputbox\_register\_left">

<label for="password">Придумайте пароль:</label>

<input type="password" name="password" placeholder="\*\*\*\*\*\*\*\*\*">

</div>

<div class="inputbox\_register\_right">

<label for="password">Повторите пароль:</label>

<input type="password" name="repeat\_password" placeholder="\*\*\*\*\*\*\*\*\*">

</div>

</div>

<a href="#" class="submit\_register">Зарегистрироваться</a>

<a href="#" class="close\_register">Закрыть окно</a>

</div>

</div>

<!---->

</main>

<footer>

<div class="container">

<div class="contact">

<h5>Свяжитесь с нами!</h5> <br>

<p>8 Марта, 11</p>

<p>Татарстан, Казань 420029</p>

</div>

<div class="sitemap">

<ul>

<li><a href="index.html">Главная</a></li>

<li><a href="prices.html">Услуги и цены</a></li>

<li><a href="faq.html">Вопросы и Ответы</a></li>

<li><a href="contacts.html">Контакты</a></li>

</ul>

</div>

</div>

</footer>

</body>

</html>

# ПРИЛОЖЕНИЕ 5. Файл contacts.html

<!doctype html>

<html>

<head>

<title> Контакты </title>

<link rel="shortcut icon" href="images/i.svg" type="image/x-icon">

<link rel="stylesheet" type="text/css" href="css/reset.css">

<link rel="stylesheet" type="text/css" href="css/style.css">

<link rel="stylesheet" type="text/css" href="fonts/Droid-Sans.css">

<link rel="stylesheet" type="text/css" href="fonts/Ubuntu.css">

<script src="https://yastatic.net/jquery/3.3.1/jquery.min.js"></script>

<script type="text/javascript" src="js/login.js"></script>

<script type="text/javascript" src="js/register.js"></script>

</head>

<body>

<header>

<a href="index.html"><img src="images/icon1.png" alt="Логотип сайта" width="100px"></a>

<div class="container">

<a href="index.html"><h1>Ветеринарная клиника "Фе́нрир"</h1> </a>

<nav>

<a href="#login\_window">Войти</a> |

<a href="#register\_window">Зарегистрироваться</a>

</nav>

</div>

</header>

<div class="block\_menu">

<ul class="content\_block\_main">

<li><a href="index.html">Главная</a></li>

<li><a href="prices.html">Услуги и цены</a></li>

<li><a href="faq.html">Вопросы и Ответы</a></li>

<li><a href="contacts.html">Контакты</a></li>

</ul>

</div>

<main>

<div class="container">

<div class="main\_page\_info">

<h2 class='header'>КОНТАКТЫ</h2>

<p> Адрес: г. Казань, ул. 8 Марта, 11</p>

<p> Телефон: (8900) 000-00-00</p>

<p> E-mail: vet@gmail.com</p>

<h3> РЕЖИМ РАБОТЫ ВЕТЕРИНАРНОЙ КЛИНИКИ</h3>

<ul>

<li> Пн-Пт: с 8:00 до 20:00 ч.</li>

<li> Сб: с 9:00 до 15:00 ч.</li>

<li> Вс: выходной</li>

</ul>

</div>

<div class="map1" style="display: flex; margin-bottom: 350px" >

</div>

</div>

</main>

<!--Всплывающее окно входа-->

<div id="login\_window">

<div id="okno\_login">

<h2 class="h2\_request">

Введите данные для входа

</h2>

<div class="form\_request">

<p>Вы можете войти в систему,<br>заполнив данную форму:</p>

<input type="login" name="login" placeholder="Электронная почта">

<input type="password" name="password" placeholder="Пароль">

</div>

<a href="#" class="close">Закрыть окно</a>

<a class="submit">Войти</a>

</div>

</div>

<!---->

<!--Всплывающее окно регистрации-->

<div id="register\_window">

<div id="okno\_register">

<h2 class="h2\_register">

Введите данные для регистрации

</h2>

<div class="form\_request">

<p>Вы можете зарегистрироваться в системе, заполнив данную форму:</p>

<div class="inputbox\_register\_left">

<label for="full\_name">Введите свое полное имя:</label>

<input type="full\_name" name="full\_name" placeholder="Фамилия Имя Отчество">

</div>

<div class="inputbox\_register\_right">

<label for="date">Дата рождения:</label>

<input type="date" name="date" placeholder="Дата рождения">

</div>

<div class="inputbox\_register\_left">

<label for="tel">Номер телефона:</label>

<input type="tel" name="tel" placeholder="+7 (999) 999-99-99">

</div>

<div class="inputbox\_register\_right">

<label for="email">Электронная почта:</label>

<input type="email" name="email" placeholder="name@email.com">

</div>

<div class="inputbox\_register\_left">

<label for="password">Придумайте пароль:</label>

<input type="password" name="password" placeholder="\*\*\*\*\*\*\*\*\*">

</div>

<div class="inputbox\_register\_right">

<label for="password">Повторите пароль:</label>

<input type="password" name="repeat\_password" placeholder="\*\*\*\*\*\*\*\*\*">

</div>

</div>

<a href="#" class="submit\_register">Зарегистрироваться</a>

<a href="#" class="close\_register">Закрыть окно</a>

</div>

</div>

<!---->

<footer>

<div class="container">

<div class="contact">

<h5>Свяжитесь с нами!</h5> <br>

<p>8 Марта, 11</p>

<p>Татарстан, Казань 420029</p>

</div>

<div class="sitemap">

<ul>

<li><a href="index.html">Главная</a></li>

<li><a href="prices.html">Услуги и цены</a></li>

<li><a href="faq.html">Вопросы и Ответы</a></li>

<li><a href="contacts.html">Контакты</a></li>

</ul>

</div>

</div>

</footer>

</body>

</html>

# ПРИЛОЖЕНИЕ 6. Файл doctor.html

<!doctype html>

<html>

<head>

<title>Личный кабинет</title>

<link rel="shortcut icon" href="images/i.svg" type="/image/x-icon">

<link rel="stylesheet" type="text/css" href="/css/reset.css">

<link rel="stylesheet" type="text/css" href="/css/style.css">

<link rel="stylesheet" type="text/css" href="/css/accounts.css">

<link rel="stylesheet" type="text/css" href="/fonts/Droid-Sans.css">

<link rel="stylesheet" type="text/css" href="/fonts/Ubuntu.css">

<script src="https://yastatic.net/jquery/3.3.1/jquery.min.js"></script>

<script type="text/javascript" src="/js/doctor.js"></script>

</head>

<body>

<header>

<a href="index.html"><img src="/images/icon1.png" alt="Логотип сайта" width="100px"></a>

<div class="container">

<h1>Ветеринарная клиника "Фе́нрир"</h1>

<nav>

<h2>Врач</h2>

</nav>

</div>

</header>

<main>

<div class="container">

<div class="tabs"></div>

<div class="content"></div>

</div>

</main>

<footer>

<div class="container">

<div class="contact">

<h5>Свяжитесь с нами!</h5> <br>

<p>8 Марта, 11</p>

<p>Татарстан, Казань 420029</p>

</div>

<div class="sitemap">

<ul>

<li><a href="/index.html">Главная</a></li>

<li><a href="/prices.html">Услуги и цены</a></li>

<li><a href="/faq.html">Вопросы и Ответы</a></li>

<li><a href="/contacts.html">Контакты</a></li>

</ul>

</div>

</div>

</footer>

</body>

</html>

# ПРИЛОЖЕНИЕ 7. Файл patient.html

<!doctype html>

<html>

<head>

<title>Личный кабинет</title>

<link rel="shortcut icon" href="images/i.svg" type="/image/x-icon">

<link rel="stylesheet" type="text/css" href="/css/reset.css">

<link rel="stylesheet" type="text/css" href="/css/style.css">

<link rel="stylesheet" type="text/css" href="/css/accounts.css">

<link rel="stylesheet" type="text/css" href="/fonts/Droid-Sans.css">

<link rel="stylesheet" type="text/css" href="/fonts/Ubuntu.css">

<script src="https://yastatic.net/jquery/3.3.1/jquery.min.js"></script>

<script type="text/javascript" src="/js/patient.js"></script>

</head>

<body>

<header>

<a href="/index.html"><img src="/images/icon1.png" alt="Логотип сайта" width="100px"></a>

<div class="container">

<h1>Ветеринарная клиника "Фе́нрир"</h1>

<nav>

<h2>Пациент</h2>

</nav>

</div>

</header>

<main>

<div class="container">

<div class="tabs"></div>

<div class="content"></div>

</div>

</main>

<footer>

<div class="container">

<div class="contact">

<h5>Свяжитесь с нами!</h5> <br>

<p>8 Марта, 11</p>

<p>Татарстан, Казань 420029</p>

</div>

<div class="sitemap">

<ul>

<li><a href="/index.html">Главная</a></li>

<li><a href="/prices.html">Услуги и цены</a></li>

<li><a href="/faq.html">Вопросы и Ответы</a></li>

<li><a href="/contacts.html">Контакты</a></li>

</ul>

</div>

</div>

</footer>

</body>

</html>

# ПРИЛОЖЕНИЕ 8. Файл admin.html

<!doctype html>

<html>

<head>

<title>Личный кабинет</title>

<link rel="shortcut icon" href="images/i.svg" type="/image/x-icon">

<link rel="stylesheet" type="text/css" href="/css/reset.css">

<link rel="stylesheet" type="text/css" href="/css/style.css">

<link rel="stylesheet" type="text/css" href="/css/accounts.css">

<link rel="stylesheet" type="text/css" href="/fonts/Droid-Sans.css">

<link rel="stylesheet" type="text/css" href="/fonts/Ubuntu.css">

<script src="https://yastatic.net/jquery/3.3.1/jquery.min.js"></script>

<script type="text/javascript" src="/js/admin.js"></script>

</head>

<body>

<header>

<a href="index.html"><img src="/images/icon1.png" alt="Логотип сайта" width="100px"></a>

<div class="container">

<h1>Ветеринарная клиника "Фе́нрир"</h1>

<nav>

<h2>Администратор</h2>

</nav>

</div>

</header>

<main class="admin">

<div class="container">

<div class="tabs">

</div>

<div class="content">

</div>

</div>

</main>

<footer>

<div class="container">

<div class="contact">

<h5>Свяжитесь с нами!</h5> <br>

<p>8 Марта, 11</p>

<p>Татарстан, Казань 420029</p>

</div>

<div class="sitemap">

<ul>

<li><a href="/index.html">Главная</a></li>

<li><a href="/prices.html">Услуги и цены</a></li>

<li><a href="/faq.html">Вопросы и Ответы</a></li>

<li><a href="/contacts.html">Контакты</a></li>

</ul>

</div>

</div>

</footer>

</body>

</html>

# ПРИЛОЖЕНИЕ 9. Файл style.css

body {

/\*background: #2cc0d3; \*/

font-family: "Baskerville Old Face";

font-size: 110%;

margin: 0; /\* Убираем отступы \*/

}

header, main, footer {

min-width: 855px;

}

/\*ссылки\*/

a {

text-decoration: none;

color: black;

}

a:hover { /\* псевдокласс для подчеркивания при наведении\*/

text-decoration: underline;

color: #a86bd1;

}

h3, h4 {

font-family: "Baskerville Old Face";

font-size: 125%;

}

h2 {

font-size:25px;

padding: 20px;

color: #a86bd1;

}

.image h2 {

font-size:25px;

padding: 20px;

color: #ffffff;

}

h3 {

padding-left: 20px;

}

header {

background: white;

overflow: auto;

padding: 10px;

}

header img {

float: left;

padding-left: 100px;

}

header h1 {

float: left;

width: 650px;

font-family: "Baskerville Old Face";

font-size: 320%;

text-align: left;

padding-top: 25px;

padding-left: 75px;

}

header nav {

float: right;

width: 300px;

padding-top: 50px;

padding-right: 100px;

}

.block\_menu {

margin-right : 0.1%;

width: 100%;

float:left;

background: #a86bd1;

padding-top: 15px;

padding-bottom: 15px;

}

main {

width: 100%;

/\*float:left;\*/

background: white;

overflow: auto;

padding-top: 20px;

padding-bottom: 20px;

}

main h3 {

margin-top: 10px;

margin-bottom: 5px;

}

main .container {

max-width: 1000px;

margin: auto;

}

main .main\_page\_info {

float: left;

}

main .top-main {

max-width: 1125px;

margin: 0 auto; /\* по центу \*/

padding-bottom: 150px;

line-height: 1.5;

}

main .bottom-main {

max-width: 1125px;

margin: auto; /\* по центу \*/

padding-top: 50px;

line-height: 1.5;

}

main .top-main h2, h3 {

color: #a86bd1;

padding-top: 0px;

padding-bottom: 0px;

}

main .left-main {

margin-top: 15px;

border-right: solid 1px #a86bd1;

width: 50%;

float:left;

/\*padding-bottom: 10px;\*/

}

main .right-main {

margin-top: 15px;

width: 49%;

float:right;

}

main .map1 {

margin-top: 15px;

width: 60%;

float:right;

}

.map2 {

position:relative;

padding-bottom: 10px;

max-width: 1000px;

margin: auto; /\* по центу \*/

}

ul.content\_block\_main li {

display: inline; /\* Отображать как строчный элемент \*/

list-style-type: none;

padding-left: 8%;

}

ul.content\_block\_main li a {

color: #F2F1EF;

}

ul.content\_block\_main li a:hover {

text-decoration: underline;

}

/\* для слайдшоу \*/

.image ul, .image li {

padding-left: 0;

list-style-type: none;

}

/\*----------Затемнение-----------\*/

.blackout {

width: 100%;

top: 690px;

height: 180px;

position: absolute;

background: #a86bd1;

color: #F2F1EF;

opacity: 0.8;

}

/\*кнопки\*/

.right\_btn1\_2 {

right: 1%;

font-size: 24px;

text-align: right;

}

.left\_btn1\_2 {

left: 1%;

float: left;

text-align: left;

}

.right\_btn1\_2, .left\_btn1\_2 {

position: absolute;

background: #a86bd1;

padding: 14px 16px;

font-size: 24px;

color: black;

outline: none; /\*убирает выделение синего цвета при нажатии на кнопку\*/

cursor: pointer;

border: none;

border-radius: 55px;

margin-top:360px;

}

button.right\_btn1\_2, button.left\_btn1\_2 {

color: #F2F1EF;

}

button.right\_btn1\_2:hover, button.left\_btn1\_2:hover {

opacity: 1;

color: #F2F1EF;

}

.image li:hover button {

opacity: 1;

transition: opacity .5s;

}

.image li:not(:hover) button {

opacity: 0;

transition: opacity .5s;

}

/\*-----\*/

/\* footer \*/

ul {

list-style: circle;

list-style-position: inside;

}

footer {

background: #a86bd1;

overflow: auto;

padding: 10px;

color: #F2F1EF;

line-height: 1.5;

}

footer a {

color:#F2F1EF;

}

footer a:hover {

text-decoration: underline;

color:#F2F1EF;

}

footer .contact {

float: left;

width: 570px;

padding-left: 100px;

}

footer .sitemap {

float: right;

width: 285px;

padding-right: 100px;

}

.tabs a span {

display: inline-block;

border-radius: 5px 5px 0 0;

width: 120px;

margin-right: 10px;

text-align: center;

background: #ccc;

padding: 5px;

}

.tabs a span.active {

background: #ddd;

}

.content {

background: white;

padding-bottom: 10px;

}

.content li{

list-style: none;

margin: 10px;

padding-left: 5px;

box-shadow: 0 0 0 10px #ddd;

}

.content input {

width: 85%;

height: 36px;

margin: 10px;

font-family: Jura;

text-align: justify;

font-size: 24px;

box-shadow: 0 0 0 10px #ddd;

}

.content button {

height: 40px;

width: 11%;

min-width: 90px;

font-size: 16px;

margin:10px;

font-family: Ubuntu;

background: white;

box-shadow: 0 -3px 0 11px #ddd;

}

/\* Дополнения для faq.html \*/

.href-faq {

text-decoration: underline;

}

.faq {

line-height: 1.5;

}

.faq h3 {

margin-bottom: 0px;

}

.faq p {

margin-bottom: 10px;

}

/\*----\*/

/\* Дополнения для vacancy.html \*/

.vacancy-left {

max-width: 480px;

height: 530px;

float: left;

line-height: 1.5;

}

.vacancy-right {

max-width: 480px;

height: 530px;

float: right;

line-height: 1.5;

}

.button\_request { /\* для кнопки "Оставить заявку" \*/

display: block;

position: relative; /\*расположение \*/

left: 50%; /\* по \*/

transform: translate(-50%, 0); /\* центру\*/

width: 185px;

height: 34px; /\* уменьшил ширину с 40px для того, чтоб по высоте текст выравнивался (костыль, другие методы не работают) \*/

background: #a86bd1;

border: 0;

text-align: center;

padding-top: 6px; /\* продолжение костыля \*/

outline: none; /\*убирает выделение синего цвета при нажатии на кнопку\*/

margin-top: 20px;

font-family: Ubuntu;

font-size: 105%;

color: #F2F1EF;

cursor: pointer;

}

.button\_request:hover {

color: #F2F1EF;

text-decoration: underline;

}

.h2\_request {

display: inline-block;

width: 460px;

background: #a86bd1;

color: #F2F1EF;

}

.form\_request p {

line-height: 1.5;

margin-top: 20px;

margin-bottom: 20px;

}

.form\_request input {

display: inline-block;

width: 430px;

height: 34px;

margin-bottom: 5px;

padding-left: 10px;

outline: none;

}

.form\_request label {

display: block;

text-align: left;

margin-top: 10px;

margin-bottom: 10px;

padding-left: 40px;

font-size: 80%;

}

/\*----\*/

/\*Всплывающее окно для заявки\*/

#zatemnenie {

background: rgba(102, 102, 102, 0.5);

width: 100%;

height: 100%;

position: absolute;

top: 0;

left: 0;

display: none;

position: fixed;

}

#okno {

width: 500px;

height: 405px;

text-align: center;

position: absolute;

top: 0;

right: 0;

bottom: 0;

left: 0;

margin: auto;

background: white;

}

#zatemnenie:target {display: block;}

.close {

display: block-inline;

/\*position: relative; /\*расположение \*/

/\*left: 50%; /\* по \*/

/\*transform: translate(-50%, 0); /\* центру\*/

float: left;

width: 185px;

height: 29px; /\* уменьшил ширину с 40px для того, чтоб по высоте текст выравнивался (костыль, другие методы не работают) \*/

background: #a86bd1;

border: 0;

text-align: center;

padding-top: 11px; /\* продолжение костыля \*/

outline: none; /\*убирает выделение синего цвета при нажатии на кнопку\*/

margin-left: 29px;

margin-top: 20px;

font-family: Ubuntu;

font-size: 105%;

color: #F2F1EF;

cursor: pointer;

}

.close:hover {

color: #F2F1EF;

text-decoration: underline;

}

.submit, .submit\_register {

display: block-inline;

/\*position: relative; /\*расположение \*/

/\*left: 50%; /\* по \*/

/\*transform: translate(-50%, 0); /\* центру\*/

float: right;

width: 185px;

height: 29px; /\* уменьшил ширину с 40px для того, чтоб по высоте текст выравнивался (костыль, другие методы не работают) \*/

background: #a86bd1;

border: 0;

text-align: center;

padding-top: 11px; /\* продолжение костыля \*/

outline: none; /\*убирает выделение синего цвета при нажатии на кнопку\*/

margin-right: 29px;

margin-top: 20px;

font-family: Ubuntu;

font-size: 105%;

color: #F2F1EF;

cursor: pointer;

}

.submit:hover, .submit\_register:hover {

color: #F2F1EF;

text-decoration: underline;

}

/\*----\*/

/\*Вспылвающее окно для входа\*/

#login\_window {

background: rgba(102, 102, 102, 0.5);

width: 100%;

height: 100%;

position: absolute;

top: 0;

left: 0;

display: none;

position: fixed;

}

#login\_window:target {display: block;}

#okno\_login {

width: 500px;

height: 330px;

text-align: center;

position: absolute;

top: 0;

right: 0;

bottom: 0;

left: 0;

margin: auto;

background: white;

}

/\*----\*/

/\*Всплывающее окно для регистрации\*/

#register\_window {

background: rgba(102, 102, 102, 0.5);

width: 100%;

height: 100%;

position: absolute;

top: 0;

left: 0;

display: none;

position: fixed;

}

#register\_window:target {display: block;}

#okno\_register {

width: 970px;

height: 435px;

text-align: center;

position: absolute;

top: 0;

right: 0;

bottom: 0;

left: 0;

margin: auto;

background: white;

}

.h2\_register {

display: inline-block;

width: 930px;

background: #a86bd1;

color: #F2F1EF;

}

.form\_register p {

line-height: 1.5;

margin-top: 20px;

margin-bottom: 20px;

}

.inputbox\_register\_left {

float: left;

margin-left: 27px;

height: 75px;

}

.inputbox\_register\_right {

float: right;

margin-right: 27px;

height: 75px;

}

.close\_register {

display: block-inline;

/\*position: relative; /\*расположение \*/

/\*left: 50%; /\* по \*/

/\*transform: translate(-50%, 0); /\* центру\*/

float: right;

width: 185px;

height: 29px; /\* уменьшил ширину с 40px для того, чтоб по высоте текст выравнивался (костыль, другие методы не работают) \*/

background: #a86bd1;

border: 0;

text-align: center;

padding-top: 11px; /\* продолжение костыля \*/

outline: none; /\*убирает выделение синего цвета при нажатии на кнопку\*/

margin-right: 70px;

margin-top: 20px;

font-family: Ubuntu;

font-size: 105%;

color: #F2F1EF;

cursor: pointer;

}

.close\_register:hover {

color: #F2F1EF;

text-decoration: underline;

}

/\*----\*/

/\*Таблица Услуги и Цены\*/

table.price\_table {

width:100%;

border: 1;

border-color: black;

}

table.price\_table td{

padding: 10px;

vertical-align: middle;

}

table.price\_table td.td-name {

background-color:#a86bd1;

}

table.price\_table td h3 {

margin-bottom:4px !important;

margin-top:4px !important;

font-size:21px !important;

text-align: center;

color:#ffffff;

}

table.price\_table td p {

text-align: center;

}

/\*----\*/

# ПРИЛОЖЕНИЕ 10. Файл accounts.css

body {

background: #a86bd1;

}

main {

background: white;

margin-top: 3px;

}

h1 {

color: #a86bd1;

}

.tabs a span {

display: inline-block;

border-radius: 5px 5px 0 0;

width: 200px;

margin-right: 10px;

text-align: center;

color:white;

background: #a86bd1;

padding: 5px;

}

.tabs a span.active {

background: #a86bd1

}

.tabs a span:hover {

color: white;

text-decoration: underline;

}

.content {

background: white;

padding-bottom: 10px;

}

.content li{

list-style: none;

margin: 10px;

padding-left: 5px;

box-shadow: 0 0 0 10px #a86bd1;

}

.content input {

float: left;

width: 300px;

height: 30px;

margin-right: 10px;

margin: 0px;

font-family: Jura;

text-align: justify;

font-size: 24px;

box-shadow: none;

}

.addAppo, .newAppo {

background: #a86bd1;

height: 30px;

width: 730px;

padding: 10px;

}

.addAppo select {

float: left;

width: 300px;

height: 30px;

margin-right: 10px;

}

.addAppo a, .newAppo a {

float: left;

width: 100px;

height: 22px;

padding-top: 8px;

padding-left: 10px;

background: white;

}

.addAppo a.disabled {

pointer-events: none; /\* делаем ссылку некликабельной \*/

cursor: default; /\* устанавливаем курсор в виде стрелки \*/

background: #b189cc; /\* цвет текста для нективной ссылки \*/

color: #888;

}

.newAppo a {

width: 135px;

}

.newAppo input {

margin-right: 10px;

}

/\*

.content button {

height: 40px;

width: 11%;

min-width: 90px;

font-size: 16px;

margin:10px;

font-family: Ubuntu;

background: white;

box-shadow: 0 -3px 0 11px #2cc0d3;

}

\*/

.link {

display: inline-block;

float: right;

width: auto;

height: 16px;

}

.linkRemove {

display: inline-block;

float: right;

width: 65px;

height: 16px;

padding-left: 15px;

}

.linkRemove:hover {

background: red;

color: white;

text-decoration: none;

}

header nav {

float: right;

width: 300px;

padding-top: 25px;

padding-right: 40px;

}

# ПРИЛОЖЕНИЕ 11. Файл login.js

var main = function () {

"use strict";

function loginfunc() {

var login = $('#login\_window .form\_request input[name="login"]').val();

var password = $('#login\_window .form\_request input[name="password"]').val();

if (login !== null && login.trim() !== "") {

console.log('email: ' + login + ' password: ' + password);

$.ajax({

url: '/accounts',

type: 'POST',

data: { 'email': login,

'password': password }

}).done(function(responde) {

window.location.replace('users/' + responde.\_id + '/');

}).fail(function(jqXHR, textStatus, error) {

console.log(error);

alert("Пользователь не найден.\nПроверьте верность введенных данных и повторите попытку");

$('#login\_window .form\_request input[name="password"]').val("");

});

}

}

$(".submit").on("click", function() {

loginfunc()

});

$('input[name="password"]').on('keydown', function(e) {

if (e.which === 13) {

loginfunc();

}

});

console.log("login.js выполнен");

}

$(document).ready(main);

# ПРИЛОЖЕНИЕ 12. Файл register.js

var main = function () {

"use strict";

function registerfunc() {

var full\_name = $('#register\_window input[name="full\_name"]').val(),

date = $('#register\_window input[name="date"]').val(),

tel = $('#register\_window input[name="tel"]').val(),

email = $('#register\_window input[name="email"]').val(),

password = $('#register\_window input[name="password"]').val(),

repeat\_password = $('#register\_window input[name="repeat\_password"]').val();

if (repeat\_password !== password) {

alert("Пароли не совпадают!\nВведите пароли снова и повторите попытку");

$('#register\_window input[name="password"]').val("");

$('#register\_window input[name="repeat\_password"]').val("");

} else if ((full\_name !== null && full\_name.trim() !== "") && (date !== null && date.trim() !== "")) {

if ((tel !== null && tel.trim() !== "") && (email !== null && email.trim() !== "") && (password !== null && password.trim() !== "") && !email.includes('doctor') && !email.includes('admin')) {

$.ajax({

url: '/users',

type: 'POST',

data: { 'username': full\_name,

'date\_of\_birth': date,

'phone': tel,

'email': email,

'password': password }

}).done(function(responde) {

console.log(responde);

alert('Аккаунт успешно создан!');

}).fail(function(jqXHR, textStatus, error) {

console.log(error);

if (jqXHR.status === 501) {

alert("Такой пользователь уже существует!\nИзмените логин и повторите "

+ "попытку");

} else {

alert("Произошла ошибка!\n"+jqXHR.status + " " + jqXHR.textStatus);

}

});

} else if (email !== null && email.trim() !== "" && email.includes('doctor')) {

$.ajax({

url: '/users',

type: 'POST',

data: { 'username': full\_name,

'date\_of\_birth': date,

'phone': tel,

'email': email,

'password': password,

'doctor': true,}

}).done(function(responde) {

console.log(responde);

alert('Аккаунт успешно создан!');

}).fail(function(jqXHR, textStatus, error) {

console.log(error);

if (jqXHR.status === 501) {

alert("Такой пользователь уже существует!\nИзмените логин и повторите "

+ "попытку");

} else {

alert("Произошла ошибка!\n"+jqXHR.status + " " + jqXHR.textStatus);

}

});

} else if (email !== null && email.trim() !== "" && email.includes('admin')) {

$.ajax({

url: '/users',

type: 'POST',

data: { 'username': full\_name,

'date\_of\_birth': date,

'phone': tel,

'email': email,

'password': password,

'admin': true,}

}).done(function(responde) {

console.log(responde);

alert('Аккаунт успешно создан!');

}).fail(function(jqXHR, textStatus, error) {

console.log(error);

if (jqXHR.status === 501) {

alert("Такой пользователь уже существует!\nИзмените логин и повторите "

+ "попытку");

} else {

alert("Произошла ошибка!\n"+jqXHR.status + " " + jqXHR.textStatus);

}

});

}

}

}

$(".submit\_register").on("click", function() {

registerfunc();

});

$('input[name="repeat\_password"]').on('keydown', function(e) {

if (e.which === 13) {

registerfunc();

}

});

}

$(document).ready(main);

# ПРИЛОЖЕНИЕ 13. Файл patient.js

var EditUsernameOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("ФИО: " + user.username);

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newUsername = prompt ("Введите новые ФИО: ", user.username);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { username: newUsername }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit username!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditNamePetOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("Кличка: " + user.name\_pet);

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newUsername = prompt ("Введите новую кличку: ", user.name\_pet);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { name\_pet: newUsername }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit username!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditTypePetOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("Вид: " + user.type\_pet);

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newUsername = prompt ("Введите вид питомца: ", user.type\_pet);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { type\_pet: newUsername }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit username!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditDateOnClick = function (user, callback) {

var date = new Date(user.date\_of\_birth);

var optionsDate = {

year: 'numeric',

month: 'long',

day: 'numeric',

};

var $usersListItem = $("<li>").text("Дата рождения: " + date.toLocaleString("ru", optionsDate));

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newDate = prompt ("Введите дату рождения: ", user.date\_of\_birth);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { date\_of\_birth: newDate }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit date!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditPhoneOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("Номер телефона: " + user.phone);

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newPhone = prompt ("Введите номер телефона: ", user.phone);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { phone: newPhone }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit phone!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditEmailOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("Электронная почта: " + user.email);

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newEmail = prompt ("Введите электронную почту: ", user.email);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { email: newEmail }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit email!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditPasswordOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("Пароль: \*\*\*\*\*\*");

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newPassword = prompt ("Введите электронную почту: ", user.password);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { password: newPassword }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit password!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var liaWithDeleteOnClick = function (appo, callback) {

var $apposListItem = $("<li>");

$.ajax ({

url: "/users/" + appo.doctor + "/account",

type: "GET"

}).done(function(responde) {

var date = new Date(appo.date);

var optionsDate = {

year: 'numeric',

month: 'long',

day: 'numeric',

weekday: 'long',

timezone: 'UTC',

hour: 'numeric',

minute: 'numeric',

};

var str = "Врач: " + responde.username + ", дата и время: " + date.toLocaleString("ru", optionsDate);

$apposListItem.text(str);

var $apposRemoveLink = $("<a>").attr("href", "#");

$apposRemoveLink.addClass("linkRemove");

$apposRemoveLink.text("Удалить");

$apposRemoveLink.on("click", function () {

if (confirm("Вы действительно хотите удалить запись на прием на " + appo.date + "?")) {

$.ajax({

url: "appoint/" + appo.\_id,

type: "PUT"

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on delete 'appo'!");

});

return false;

}

});

$apposListItem.append($apposRemoveLink);

}).fail(function (err) {

console.log("Ошибка: " + err);

});

return $apposListItem;

}

var main = function () {

"use strict";

// создание пустого массива с вкладками

var tabs = [];

// добавляем вкладку Обо мне

tabs.push({

"name": "Обо мне",

// создаем функцию content

// так, что она принимает обратный вызов

"content": function(callback) {

$.get("account/", function(user) {

var $content = $("<ul>");

$content.append(EditUsernameOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditNamePetOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditTypePetOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditDateOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditPhoneOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditEmailOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditPasswordOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

callback(null, $content);

}).fail(function (jqXHR, textStatus, error) {

callback(error, null);

});

}

});

// добавляем вкладку Мои записи к врачу

tabs.push({

"name": "Мои записи к врачу",

"content": function(callback) {

var $content = $("<ul>");

$content.append($("<a>").text(""));

$.getJSON("appoint.json", function (appoObjects) {

var i;

for (i = 0; i < appoObjects.length; i++) {

var $apposListItem = liaWithDeleteOnClick(appoObjects[i], function() {

alert("Вы успешно отказались от приема!");

$(".tabs a:nth-child(2) span").trigger("click");

});

$content.append($apposListItem);

}

callback(null, $content);

}).fail(function(jqXHR, textStatus, error) {

callback(error, null);

});

}

});

tabs.push({

"name": "Записаться к врачу",

"content": function(callback) {

$.getJSON("/users.json", function (usersObjects) {

var $content = $("<div>").addClass("addAppo"),

$selDoctor = $("<select>"),

$selDate = $("<select>"),

$btAdd = $("<a>").text("Записаться");

// $btAdd = $("<button>").text("Записаться");

$btAdd.attr("href", "#");

$btAdd.addClass("disabled");

$selDoctor.append($("<option>").text("Выберите врача"));

$selDate.append($("<option>").text("Выберите дату и время приема"));

$selDate.prop("disabled", true);

$btAdd.prop("disabled", true);

// заполняем список с врачами:

var i;

for (i = 0; i < usersObjects.length; i++) {

if (usersObjects[i].doctor) {

var $option = $("<option>").text(usersObjects[i].username);

$option.attr("value", usersObjects[i].\_id); // сохраняю \_id врача в value на будущее

$selDoctor.append($option);

}

}

$selDoctor.change(function() {

$selDate.find('option').remove();

$selDate.append($("<option>").text("Выберите дату и время приема"));

$selDate.prop("disabled", true);

$btAdd.attr("class", "disabled");

if ($(this).val() != 0) {

// заполняем список с приемами

$.ajax({

url: "/users/"+$selDoctor.val()+"/appoint.json",

type: "GET"

}).done(function(appoObjects) {

console.log(appoObjects);

$selDate.find('option').remove();

$selDate.append($("<option>").text("Выберите дату и время приема"));

var i, j = 0;

for (i = 0; i < appoObjects.length; i++) {

if ((appoObjects[i].doctor === $selDoctor.val()) // если у записи выбранный врач

&& (appoObjects[i].patient == undefined) // если нет пациента

&& (new Date(appoObjects[i].date) > new Date())) { // если дата больше текущей

var date = new Date(appoObjects[i].date);

var optionsDate = {

year: 'numeric',

month: 'long',

day: 'numeric',

weekday: 'long',

timezone: 'UTC',

hour: 'numeric',

minute: 'numeric',

};

var $option = $("<option>").text(date.toLocaleString("ru", optionsDate));

$option.attr("value", appoObjects[i].\_id); // сохраняю \_id врача в value на будущее

$selDate.append($option);

j++;

}

}

if (j > 0) {

$selDate.prop("disabled", false);

//$btAdd.prop("disabled", false);

$btAdd.attr("class", "");

} else {

alert("Приемов у этого врача не найдено");

$selDate.prop("disabled", true);

$btAdd.attr("class", "disabled");

}

})

}

});

$btAdd.on("click", function() {

if (confirm("Вы действительно хотите записаться на прием?")) {

$.ajax({

url: "appoint/" + $selDate.val(),

type: "PUT"

}).done(function(responde) {

alert("Вы успешно записались на прием!");

$(".tabs a:nth-child(2) span").trigger("click");

}).fail(function(jqXHR, textStatus, error) {

alert("Ошибка!\n" + jqXHR.status);

})

}

});

$content.append($selDoctor, $selDate, $btAdd);

callback(null, $content);

}).fail(function(jqXHR,textStatus, error) {

callback(error, null);

});

}

});

tabs.forEach(function (tab) {

var $aElement = $("<a>").attr("href",""),

$spanElement = $("<span>").text(tab.name);

$aElement.append($spanElement);

$("main .tabs").append($aElement);

$spanElement.on("click", function () {

var $content;

$(".tabs a span").removeClass("active");

$spanElement.addClass("active");

$("main .content").empty();

tab.content(function (err, $content) {

if (err !== null) {

alert ("Возникла проблема при обработке запроса: " + err);

} else {

$("main .content").append($content);

}

});

return false;

});

});

$(".tabs a:first-child span").trigger("click");

}

$(document).ready(function() {

main();

});

# ПРИЛОЖЕНИЕ 14. Файл admin.js

var liaWithDeleteOnClick = function (user, callback) {

var $usersListItem = $("<li>").text(user.username);

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("linkRemove");

$usersRemoveLink.text("Удалить");

$usersRemoveLink.on("click", function () {

if (confirm("Вы действительно хотите удалить пользователя " + user.username + "?")) {

$.ajax({

url: "/users/" + user.\_id,

type: "DELETE"

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on delete 'user'!");

});

return false;

}

});

if (!user.doctor) $usersListItem.append($usersRemoveLink);

if (!user.doctor) {

var $toTheDoctorLink = $("<a>").attr("href", "#");

$toTheDoctorLink.addClass("link");

$toTheDoctorLink.text("Сделать врачом");

$toTheDoctorLink.on("click", function() {

if (confirm("Вы действительно хотите пользователя " + user.username + " назначить врачом?")) {

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { "doctor": true }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error! " + err);

});

return false;

}

});

$usersListItem.append($toTheDoctorLink);

} else {

var $toTheDoctorLink = $("<a>").attr("href", "#");

$toTheDoctorLink.addClass("link");

$toTheDoctorLink.text("Убрать из врачей");

$toTheDoctorLink.on("click", function() {

if (confirm("Вы действительно хотите пользователя " + user.username + " убрать из врачей?")) {

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { "doctor": false }

}).done(function (responde) {

$.getJSON("/appoint.json", function(appoObjects) {

var i;

for (i = 0; i < appoObjects.length; i++) {

if ((appoObjects[i].doctor === user.\_id) && (new Date(appoObjects[i].date) >= new Date())) {

$.ajax({

url: "appoint/" + appoObjects[i].\_id,

type: "DELETE"

}).fail(function(err) {

console.log("Ошибка в удалении appoint: " + err);

});

}

}

callback();

}).fail(function (err) {

console.log("Ошибка в getJSON(appoint.json): " + err);

});

}).fail(function (err) {

console.log("error! " + err);

});

return false;

}

});

$usersListItem.append($toTheDoctorLink);

}

return $usersListItem;

}

var main = function () {

"use strict";

// создание пустого массива с вкладками

var tabs = [];

// добавляем вкладку Новые

tabs.push({

"name": "Новые клиенты",

// создаем функцию content

// так, что она принимает обратный вызов

"content": function(callback) {

$.getJSON("/users.json", function (usersObjects) {

var $content,

i;

$content = $("<ul>");

for (i = usersObjects.length-1; i>=0; i--) {

if (!usersObjects[i].admin && !usersObjects[i].doctor) {

var $usersListItem = liaWithDeleteOnClick(usersObjects[i], function() {

$(".tabs a:first-child span").trigger("click");

});

$content.append($usersListItem);

}

}

callback(null, $content);

}).fail(function (jqXHR, textStatus, error) {

callback(error, null);

});

}

});

// добавляем вкладку Старые

tabs.push({

"name": "Старые клиенты",

"content": function(callback) {

$.getJSON("/users.json", function (usersObjects) {

var $content,

i;

$content = $("<ul>");

for (i = 0; i < usersObjects.length; i++) {

if (!usersObjects[i].admin && !usersObjects[i].doctor) {

var $usersListItem = liaWithDeleteOnClick(usersObjects[i], function() {

$(".tabs a:nth-child(2) span").trigger("click");

});

$content.append($usersListItem);

}

}

callback(null, $content);

}).fail(function(jqXHR, textStatus, error) {

callback(error, null);

});

}

});

// добавляем вкладку Врачи

tabs.push({

"name": "Врачи",

"content":function (callback) {

$.getJSON("/users.json", function (usersObjects) {

var $content,

i;

$content = $("<ul>");

for (i = 0; i < usersObjects.length; i++) {

if (usersObjects[i].doctor) {

var $usersListItem = liaWithDeleteOnClick(usersObjects[i], function() {

$(".tabs a:nth-child(3) span").trigger("click");

});

$content.append($usersListItem);

}

}

callback(null, $content);

}).fail(function(jqXHR, textStatus, error) {

callback(error, null);

});

}

});

tabs.forEach(function (tab) {

var $aElement = $("<a>").attr("href",""),

$spanElement = $("<span>").text(tab.name);

$aElement.append($spanElement);

$("main .tabs").append($aElement);

$spanElement.on("click", function () {

var $content;

$(".tabs a span").removeClass("active");

$spanElement.addClass("active");

$("main .content").empty();

tab.content(function (err, $content) {

if (err !== null) {

alert ("Возникла проблема при обработке запроса: " + err);

} else {

$("main .content").append($content);

}

});

return false;

});

});

$(".tabs a:first-child span").trigger("click");

}

$(document).ready(function() {

main();

});

# ПРИЛОЖЕНИЕ 15. Файл doctor.js

var EditUsernameOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("ФИО: " + user.username);

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newUsername = prompt ("Введите новые ФИО: ", user.username);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { username: newUsername }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit username!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditDateOnClick = function (user, callback) {

var date = new Date(user.date\_of\_birth);

var optionsDate = {

year: 'numeric',

month: 'long',

day: 'numeric',

};

var $usersListItem = $("<li>").text("Дата рождения: " + date.toLocaleString("ru", optionsDate));

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newDate = prompt ("Введите дату рождения: ", user.date\_of\_birth);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { date\_of\_birth: newDate }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit date!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditPhoneOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("Номер телефона: " + user.phone);

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newPhone = prompt ("Введите номер телефона: ", user.phone);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { phone: newPhone }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit phone!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditEmailOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("Электронная почта: " + user.email);

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newEmail = prompt ("Введите электронную почту: ", user.email);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { email: newEmail }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit email!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var EditPasswordOnClick = function (user, callback) {

var $usersListItem = $("<li>").text("Пароль: \*\*\*\*\*\*");

var $usersRemoveLink = $("<a>").attr("href", "#");

$usersRemoveLink.addClass("link");

$usersRemoveLink.text("Редактировать");

$usersRemoveLink.on("click", function () {

var newPassword = prompt ("Введите электронную почту: ", user.password);

$.ajax({

url: "/users/" + user.\_id,

type: "PUT",

data: { password: newPassword }

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on edit password!");

});

return false;

});

$usersListItem.append($usersRemoveLink);

return $usersListItem;

}

var liaWithDeleteOnClick = function (appo, callback) {

var $apposListItem = $("<li>");

if (appo.patient != undefined) {

$.ajax ({

url: "/users/" + appo.patient + "/account",

type: "GET"

}).done(function(responde) {

var date = new Date(appo.date);

var options = {

year: 'numeric',

month: 'long',

day: 'numeric',

weekday: 'long',

timezone: 'UTC',

hour: 'numeric',

minute: 'numeric',

};

var str = date.toLocaleString("ru", options);

str = str + ", пациент: " + responde.username;

$apposListItem.text(str);

var $apposRemoveLink = $("<a>").attr("href", "#");

$apposRemoveLink.addClass("linkRemove");

$apposRemoveLink.text("Удалить");

$apposRemoveLink.on("click", function () {

if (confirm("Вы действительно хотите удалить запись на прием на " + date + "?")) {

$.ajax({

url: "appoint/" + appo.\_id,

type: "DELETE"

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on delete 'appo'!");

});

return false;

}

});

$apposListItem.append($apposRemoveLink);

}).fail(function (err) {

console.log("Ошибка: " + err);

});

} else {

var date = new Date(appo.date);

var options = {

year: 'numeric',

month: 'long',

day: 'numeric',

weekday: 'long',

timezone: 'UTC',

hour: 'numeric',

minute: 'numeric',

};

var str = date.toLocaleString("ru", options);

$apposListItem.text(str);

var $apposRemoveLink = $("<a>").attr("href", "#");

$apposRemoveLink.addClass("linkRemove");

$apposRemoveLink.text("Удалить");

$apposRemoveLink.on("click", function () {

if (confirm("Вы действительно хотите удалить запись на прием на " + date + "?")) {

$.ajax({

url: "appoint/" + appo.\_id,

type: "DELETE"

}).done(function (responde) {

callback();

}).fail(function (err) {

console.log("error on delete 'appo'!");

});

return false;

}

});

$apposListItem.append($apposRemoveLink);

}

return $apposListItem;

}

var main = function () {

"use strict";

// создание пустого массива с вкладками

var tabs = [];

tabs.push({

"name": "Обо мне",

// создаем функцию content

// так, что она принимает обратный вызов

"content": function(callback) {

$.get("account/", function(user) {

var $content = $("<ul>");

$content.append(EditUsernameOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditDateOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditPhoneOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditEmailOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

$content.append(EditPasswordOnClick(user, function() {

$(".tabs a:first-child span").trigger("click");

}));

callback(null, $content);

}).fail(function (jqXHR, textStatus, error) {

callback(error, null);

});

}

});

tabs.push({

"name": "Предстоящие приемы",

"content": function(callback) {

var $content = $("<ul>");

$content.append($("<a>").text(""));

$.getJSON("appoint.json", function (appoObjects) {

var i;

for (i = 0; i < appoObjects.length; i++) {

if (new Date(appoObjects[i].date) >= new Date()) {

var $apposListItem = liaWithDeleteOnClick(appoObjects[i], function() {

alert("Вы успешно отказались от приема!");

$(".tabs a:nth-child(2) span").trigger("click");

});

$content.append($apposListItem);

}

}

callback(null, $content);

}).fail(function(jqXHR, textStatus, error) {

callback(error, null);

});

}

});

tabs.push({

"name": "Создать запись на прием",

"content": function(callback) {

var $content = $("<div>").addClass("newAppo"),

$inpDate = $("<input>").attr("type", "datetime-local"),

$linkAdd = $("<a>").text("Создать запись");

$inpDate.attr("placeholder", "Выберите дату и время приема");

$linkAdd.attr("href", "#");

$linkAdd.on("click", function() {

if (confirm("Вы действительно хотите создать запись?")) {

$.ajax({

url: "appoint/",

type: "POST",

data: {date: $inpDate.val()}

}).done(function(responde) {

alert("Вы успешно создали запись на прием!");

$inpDate.val("");

}).fail(function(jqXHR, textStatus, error) {

alert("Произошла ошибка: " + error);

})

}

});

$content.append($inpDate, $linkAdd);

callback(null, $content);

}

});

tabs.forEach(function (tab) {

var $aElement = $("<a>").attr("href",""),

$spanElement = $("<span>").text(tab.name);

$aElement.append($spanElement);

$("main .tabs").append($aElement);

$spanElement.on("click", function () {

var $content;

$(".tabs a span").removeClass("active");

$spanElement.addClass("active");

$("main .content").empty();

tab.content(function (err, $content) {

if (err !== null) {

alert ("Возникла проблема при обработке запроса: " + err);

} else {

$("main .content").append($content);

}

});

return false;

});

});

$(".tabs a:first-child span").trigger("click");

}

$(document).ready(function() {

main();

});