МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт Информационных технологий и управления в технических системах

Кафедра Информационные системы

|  |  |  |  |
| --- | --- | --- | --- |
| № | Дата поступления на кафедру | Подпись отв. за регистрацию | Подпись преподавателя |
|  |  |  |  |
|  |  |  |  |

**ОТЧЕТ**

о производственной (преддипломной) практике

в \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(наименование организации)

Выполнил \_\_\_Долженко Илья\_\_\_\_\_\_

\_Андреевич\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия И.О. обучающегося)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ИС/б-17-2-о\_\_\_\_\_\_\_\_\_\_\_\_\_

(шифр группы)

Направление / специальность 09.03.02

Информационные системы и технологии

(код, наименование)

Руководитель практики от Университета \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(должность)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_Чернега В.С.\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия И.О. руководителя)

Севастополь

2021 г.

**СОДЕРЖАНИЕ**

ВВЕДЕНИЕ 3

1.ОСНОВНАЯ ЧАСТЬ 4

1.1. Разработка компонентов программного модуля 4

1.2. Разработка интерфейса программного модуля 10

ЗАКЛЮЧЕНИЕ 33

СПИСОК ЛИТЕРАТУРЫ И ИНФОРМАЦИОННЫХ РЕСУРСОВ 34

**ВВЕДЕНИЕ**

Целью преддипломной практики является разработка технического приложения для клиентской части системы онлайн-консультаций с врачами.

В процессе достижения поставленной цели решались следующие задачи:

* Разработка компонентов программного модуля;
* Разработка интерфейса программного модуля.

Отчет по производственной практике состоит из содержания, введения, основной части и списка литературных источников.

**1 ОСНОВНАЯ ЧАСТЬ**

**1.1 Разработка компонентов программного модуля**

При разработки клиентской части использовалась библиотека для разработки пользовательских интерфейсов React и библиотека для управления состоянием приложения MobX. Основываясь на лучших практиках разработки реактивных приложений с таким стеком технологий был выбран следующий подход построения структуры приложения:

api/

UserApi.ts

…

components/

Button.tsx

…

fonts/

icons/

pages/

Home/

components/

HomePage.tsx

index.ts

…

stores/

ChatStore.ts

…

styles/

utils/

Классы из директории stores представляют из себя функционально законченные фрагменты программы – модули, которые взаимодействую с серверной частью приложения посредством HTTP-запросов, инстансы которых находятся в директории api.

Таким образом при разработке web-приложения были спроектированы независимые друг от друга компоненты, описание которых представлено ниже.

Модуль HomeStore используется на главной странице для вывода специалистов с наивысшим опытом работы и отправки формы обратной связи. Для данной реализации он содержит в себе массив специалистов и логический индикатор, для отображения анимации загрузки, а также два объекта с полями формы обратной связи и объект для валидации данных полей. Модуль взаимодействует с двумя конечными точками серверной части:

* GET /api/v1/doctor/most-experienced?count={count} – отвечает за получение самых опытных специалистов;
* POST /api/v1/feedback/leave – отвечает за отправку формы обратной связи.

Для выполнения процесса авторизации используется модуль SignInStore, который содержит объекты с данными полей для входа и функцию отправки данного объекта. Данная функция использует метод POST-запроса /api/v1/auth/sign-in. После удачного выполнения данного запроса, внутри функции выполняется установка в localStorage access-токена, который затем используется для отправки в заголовках запросов, чтобы идентифицировать пользователя.

Следующий модуль SignUpStore используется на странице регистрации нового пользователя и содержит метод doSignUp для отправки POST запроса на сервер. В тело запроса передается объект с полями формы, такие как: тип пользователя (пациент или доктор), имя, фамилия, отчество, дата рождения, пол, телефон, пароль и адрес электронной почты, который затем используется для подтверждения регистрации. Модуль взаимодействует с двумя конечными точками сервера:

* POST /api/v1/auth/sign-up – отвечает за отправку формы регистрации;
* POST /api/v1/auth/send-email-with-token – метод для повторной отправки сообщения с подтверждением на почту.

В приложении также реализуется модуль UserStore, предназначенный для хранения текущей информации о пользователе, его статусе авторизации и реализующий функции получения информации о пользователе по access-токену и функцию выхода из аккаунта, в которой происходит очистка закэшированных в других модулях полей. В процессе реализации данного класса применялись следующие API-методы:

* GET /api/v1/user/info – получение информации пользователя по токену в заголовках запроса;
* GET /api/v1/user/fresh-token – метод для обновления токена доступа при каждом входе в приложение.

Для вывода и поиска по фильтрам специалистов используется модуль SearchDoctorStore, который реализует HTTP-методы, представленные ниже, и состоит из массива найденных специалистов, объекта с текущей страницей специалистов, булевых переменных для отображения анимаций загрузки и строкового свойства для реализации поиска по ФИО:

* GET /api/v1/doctor/paginate?page={page}&count={count}&

&specialty={specialty} – отвечает за поиск по странице или специальности;

* GET /api/v1/doctor/paginate?page=1&count=4&fio={fullName} – запрос для поиска специалистов по имени в строке поиска.

Для просмотра выбранного врача существует отдельный модуль DoctorStore. В данном модуле находятся две функции, реализующие запрос на поиск доктора по id из адресной строки и подгрузку комментариев. Методы, используемые в данном модуле:

* GET /api/v1/doctor/info?id={id} – запрос на получение информации о докторе по id;
* GET /api/v1/doctor/review/list?reviewId={reviewId}&doctorId=

={doctorId}&count={count} – метод для получения комментариев по последнему id комментария в списке, id доктора и параметром для получения необходимого количества комментариев.

Модуль AppointmentStore применяется в процессе записи пациентом на консультацию. Данный модуль объединяет в себе всю логику четырех этапов записи. При открытии первого этапа происходит автоматически запрос на получение информации о выбранном специалисте, после чего выполняется получение его расписания на текущее время. В случае если выбранный врач не принимает пациентов в выбранный день, то в соответствующем поле на форме будет установлено “Нет приема”. Также данный класс содержит функцию отправки, заполненной на прошлых этапах формы. Модуль записи на консультацию использует три конечных точки:

* GET /api/v1/consultation/appointment/meta-info?doctorId={doctorId} – получение информации о выбранном докторе по его id;
* GET /api/v1/consultation/appointment/free-doctor-time?doctorId=

={doctorId}&date={date} – запрос на получение свободного времени по id специалиста и дате;

* POST /api/v1/consultation/appointment/create – применяется для отправки итоговой формы записи.

Модуль ModalsStore представляет из себя некий класс-менеджер, который управляет всеми диалоговыми окнами, их открытием и закрытием. Для этого в нем реализуются две соответствующие функции и массив всех существующих в приложении диалогов с их статусом (показывается или скрыто). Данный модуль не реализует запросов к серверу и является компонентом, который может применяться в любой части приложения.

Далее идут модули, применяемые в личном кабинете пациента или специалиста.

Модуль DashboardConsultationsStore применяется на странице консультаций пациента и предназначен для выполнения запросов на получение списков консультаций по типу (активные, предстоящих и завершенных). Таким образом он содержит три специализированных под каждый тип консультации списков и соответствующих функций, реализующих их получение. Также в модуле содержится функция отмены консультации. Данный модуль реализует два запроса на сервер:

* GET /api/v1/consultation/doctors-for-patient?consultationState=

={type} – запрос, в котором зависимости от типа параметра будет возвращаться соответствующий список консультаций;

* POST /api/v1/consultation/cancel – применяется для отмены консультации.

Для управления анализами пациента применяется специализированный для этого модуль DashboardAnalyzesStore. В модуле реализуются функции для получения анализов, добавления файла и его удаление из списка. Таким образом в модуле применяются следующие конечные точки сервера:

* GET /api/v1/patient/analysis/all – запрос для получения всех анализов;
* POST /api/v1/patient/analysis/append – добавление анализа или снимка в список;
* POST /api/v1/patient/analysis/delete – применяется для удаления файла.

Для получения списка назначений от специалиста на соответствующей странице пациента применяется модуль DashboardResultsStore. Он реализует один запрос на сервер, который возвращает список назначений. Данный эндпоинт представляет из себя простой GET запрос без параметров – /api/v1/consultation/appointments.

В личном кабинете пациента существует страница просмотра и редактирования своей медицинской карты. Для данного процесса был реализован модуль DashboardMedicalCardStore. Как уже написано раньше, в нем находится функция редактирования, которая выполняет POST запрос на url /api/v1/patient/profile/change-medical-card, в теле которого отправляется весь объект дополнительной информации о пользователе с уже измененными полями.

Для просмотра своего профиля с аккаунта специалиста в личном кабинете применяется модуль DashboardDoctorProfileStore. По аналогии с модулем для изменения медицинской карты пациента DashboardMedicalCardStore, текущий класс также содержит множества функций для заполнения объекта с дополнительной информацией о себе и функцию отправки этого объекта на сервер. Данный модуль реализует POST запрос /api/v1/doctor/profile/change-info.

Для следующей странице, странице просмотра пациента, был реализован модуль DashboardPatientsStore, в котором на каждое изменения даты в календаре или типа списка (новые или история) выполняется GET запрос /api/v1/consultation/patients-for-doctor?date={date}&state={state}, куда подставляются измененные параметры. После успешного выполнения запроса в массив пациентов заносится ответ с сервера и выполняется перерисовка списка на странице.

Так как доктор имеет возможность просмотра профиля пациента, записанного на консультацию, то целесообразно создать отдельный модуль, в котором хранится вся основная информация пациента и реализуются соответствующие запросы на получения этой информации. Данный модуль называется DashboardPatientInfoStore. В модуле применяется один GET запрос /api/v1/patient/consultation-info?patientId={patientId}&consultationId=

={consultationId} в который передается id пациента и текущей консультации.

Для редактирования расписания специалиста существует модуль DashboardScheduleStore. Данный класс уже содержит реализацию двух запросов:

* GET /api/v1/doctor/profile/schedule – запрос на получение текущего расписания;
* POST /api/v1/doctor/profile/change-schedule – запрос на изменение расписания.

В личном кабинете имеется общий модуль DashboardSettingsStore, который предназначен для редактирования личной информации о пользователе: имя, фамилия, отчество, дата рождения, номер телефона и пол. Для этого происходит взаимодействие с POST методом /api/v1/user/change-user-info и методом для изменения фотографии /api/v1/user/change-photo.

Основная коммуникация пациента и специалиста происходит в модулях ChatStore и SocketsStore. Данные модули взаимосвязаны и реализуют функции отправки сообщений и различных типов файлов, получения диалогов, подгрузки старых сообщений. В модуле SocketsStore реализуется сокет-соединение, а в модуле ChatStore применяются следующие запросы к серверу:

* GET /api/v1/chat/list – получение списка диалогов;
* GET /api/v1/chat/message/list?chatId={chatId}&count={count}&

&lastMessageId={lastMessageId} – запрос на получение старых сообщений по id последнего сообщения в списке и id текущего чата;

* POST /api/v1/chat/message/send-media – запрос для отправки файла.

**1.2 Разработка интерфейса программного модуля**

При запуске web-приложения открывается главная страница, на которой представлена основная информация о нашем сервисе (Рисунок 3) имеется секция с наилучшими специалистами (Рисунок 2) и раздел с отзывами пациентов о сервисе (Рисунок 4).

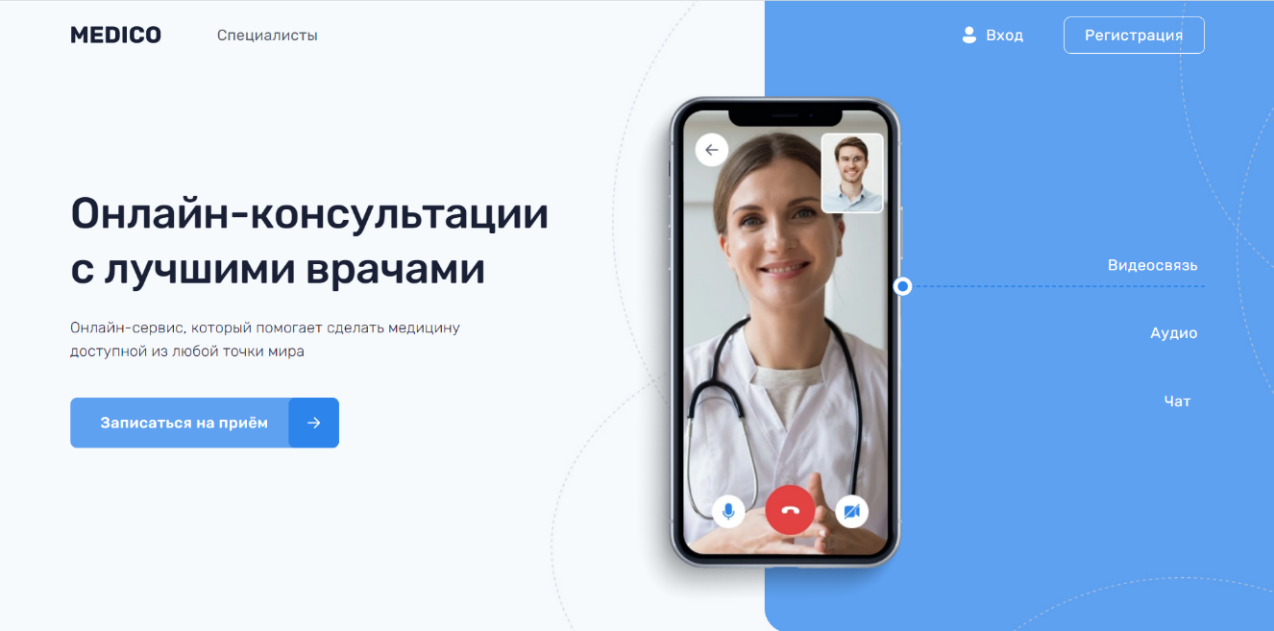


Рисунок 1 – Главный экран

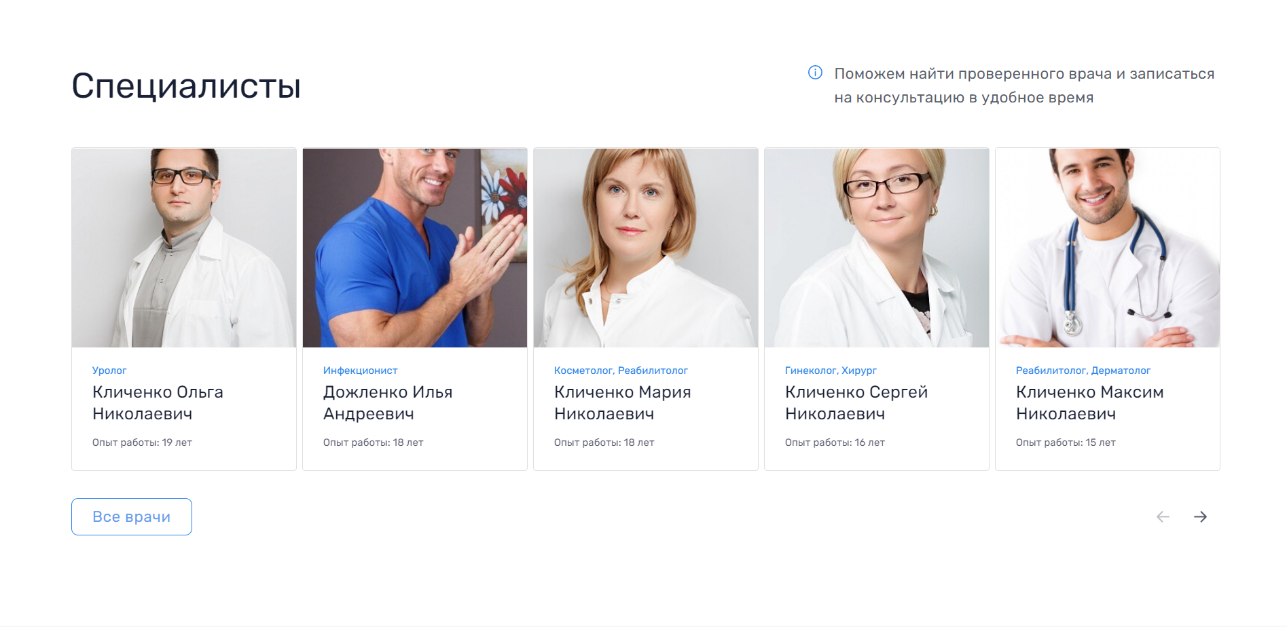


Рисунок 2 – Секция со специалистами



Рисунок 3 – Информация о сервисе

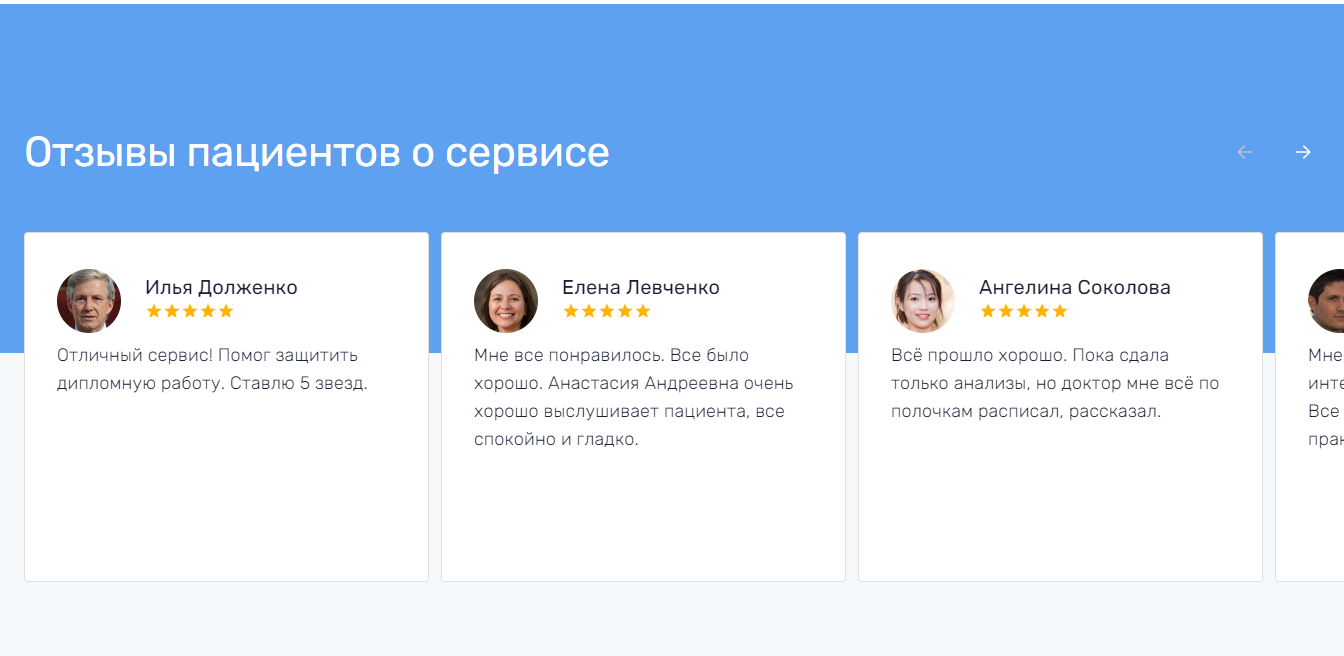


Рисунок 4 – Раздел с отзывами о сервисе

Снизу главной страницы располагается форма обратной связи, представленная на рисунке 5.

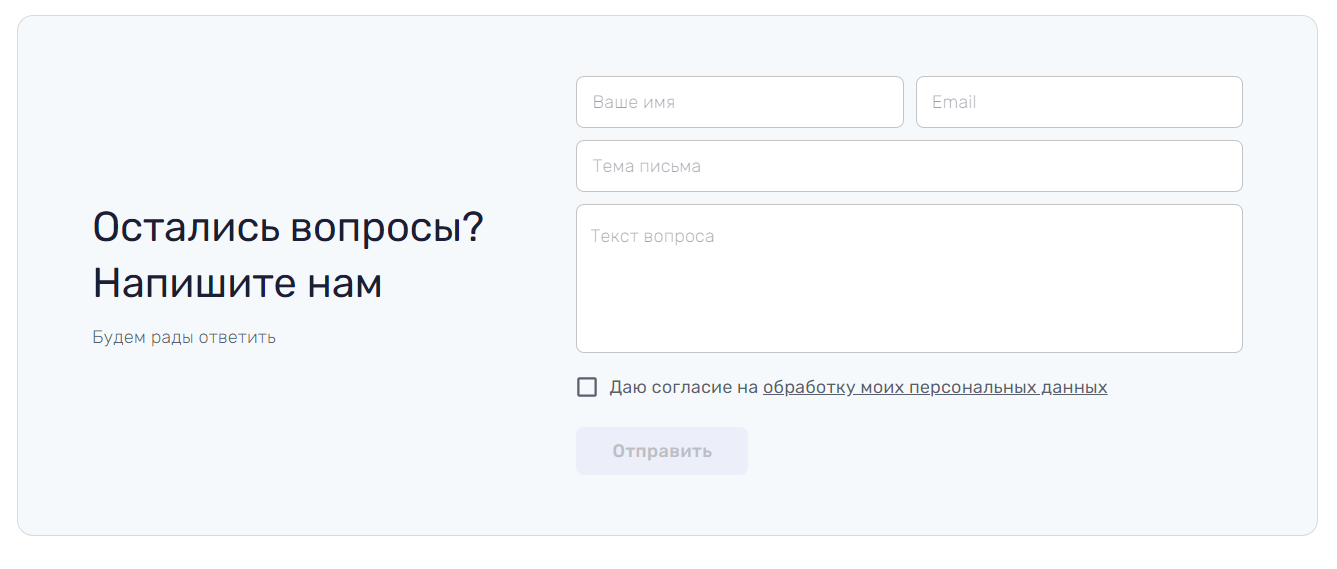


Рисунок 5 – Форма обратной связи

При нажатии на кнопку «Войти» открывается модальное окно с формой входа состоящие из полей «Email» и «Пароль» (Рисунок 5). Так же на этом окне находится кнопка восстановления пароля и регистрации (Рисунок 6)

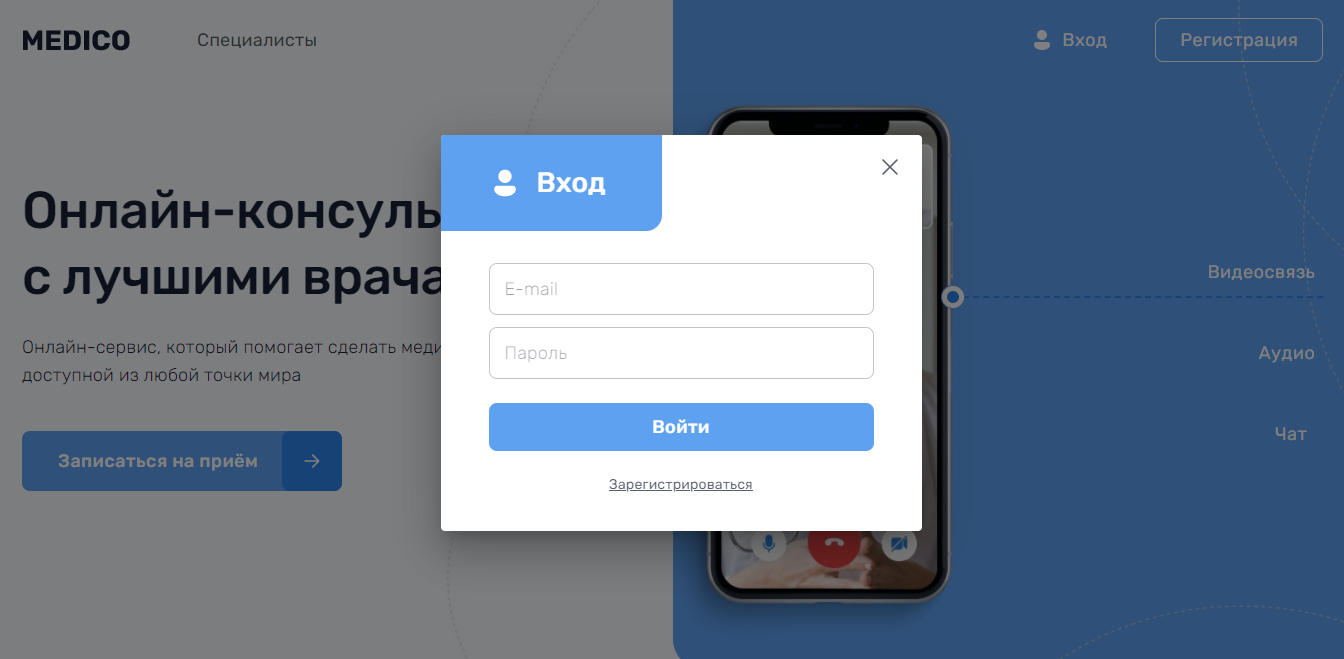


Рисунок 5 – Модальное окно входа

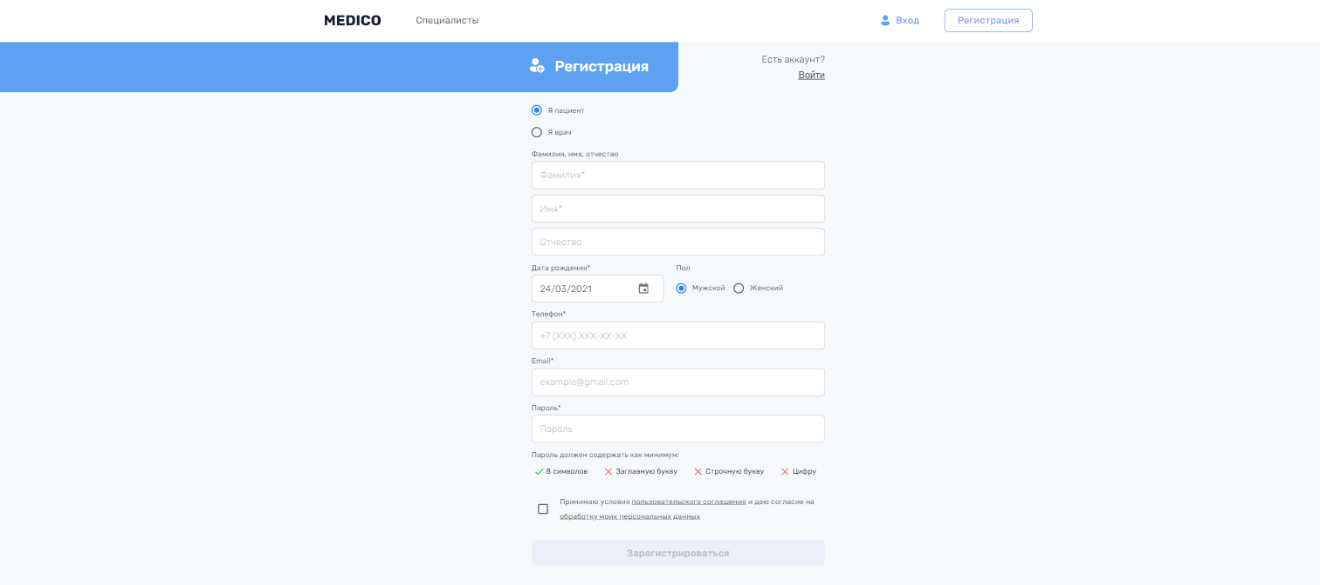


Рисунок 6 – Страница регистрации

После регистрации на почту пользователю приходит письмо с подтверждением аккаунта (Рисунок 7), и появляется модальное окно сообщающее об этом (Рисунок 8).

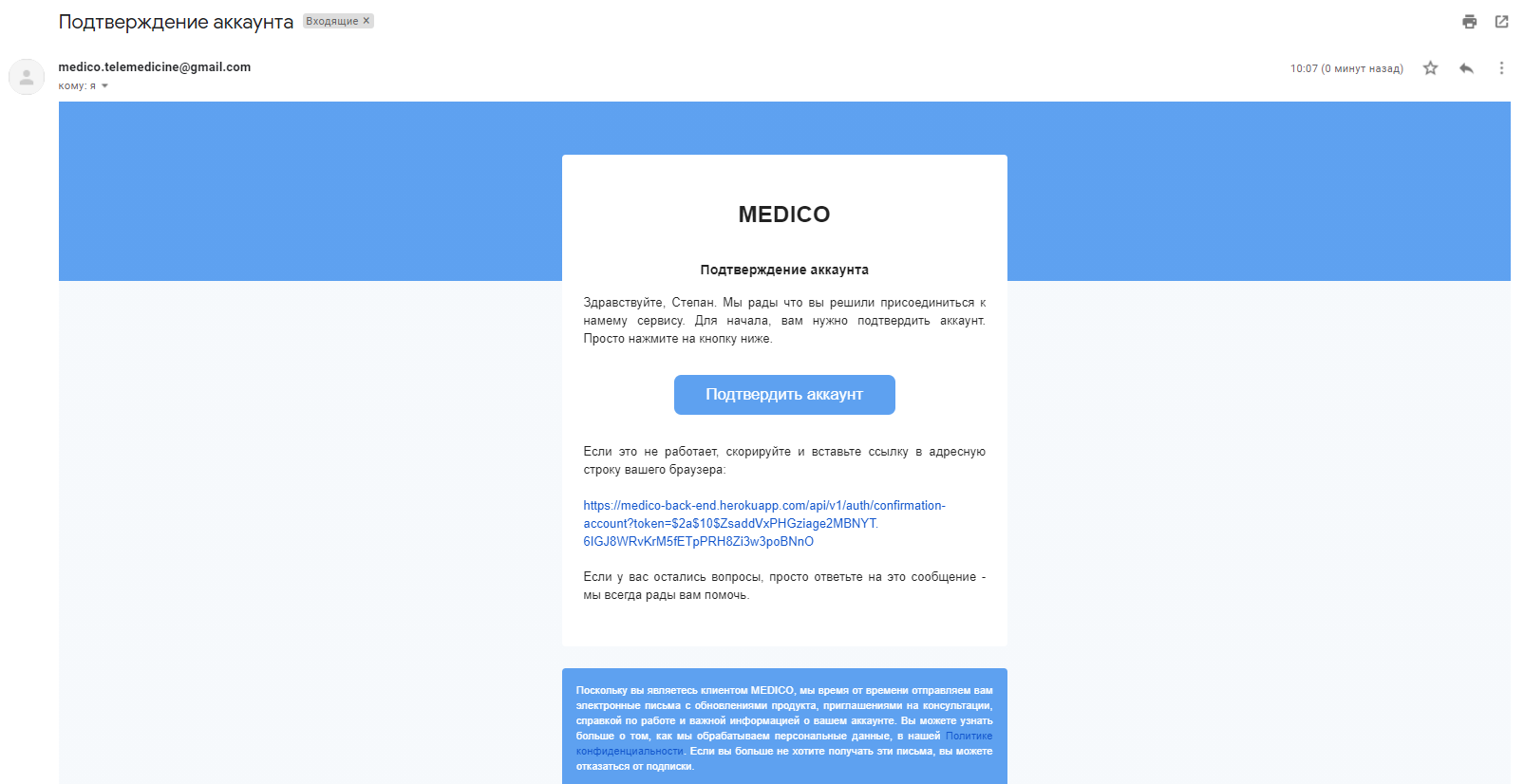


Рисунок 7 – Письмо на почте

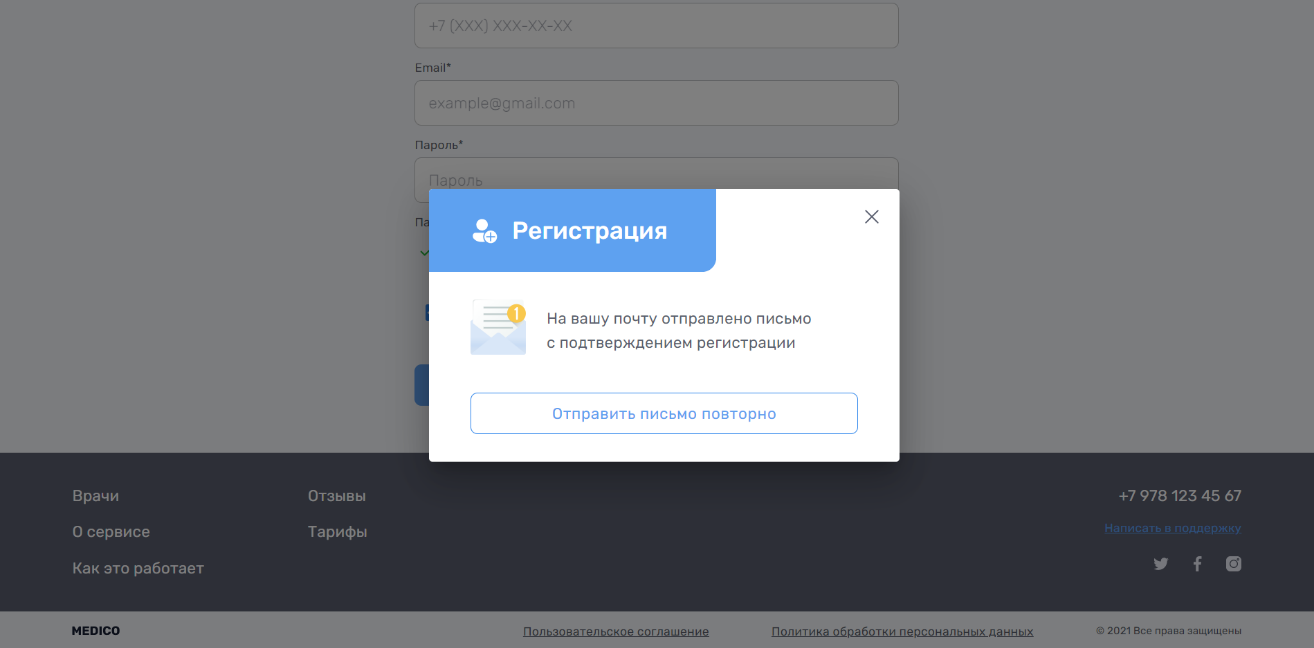


Рисунок 8 – Сообщение об отправки письма

После нажатия на кнопку «Подтвердить аккаунт» в письме пользователь перенаправляется на web-приложение (Рисунок 9).

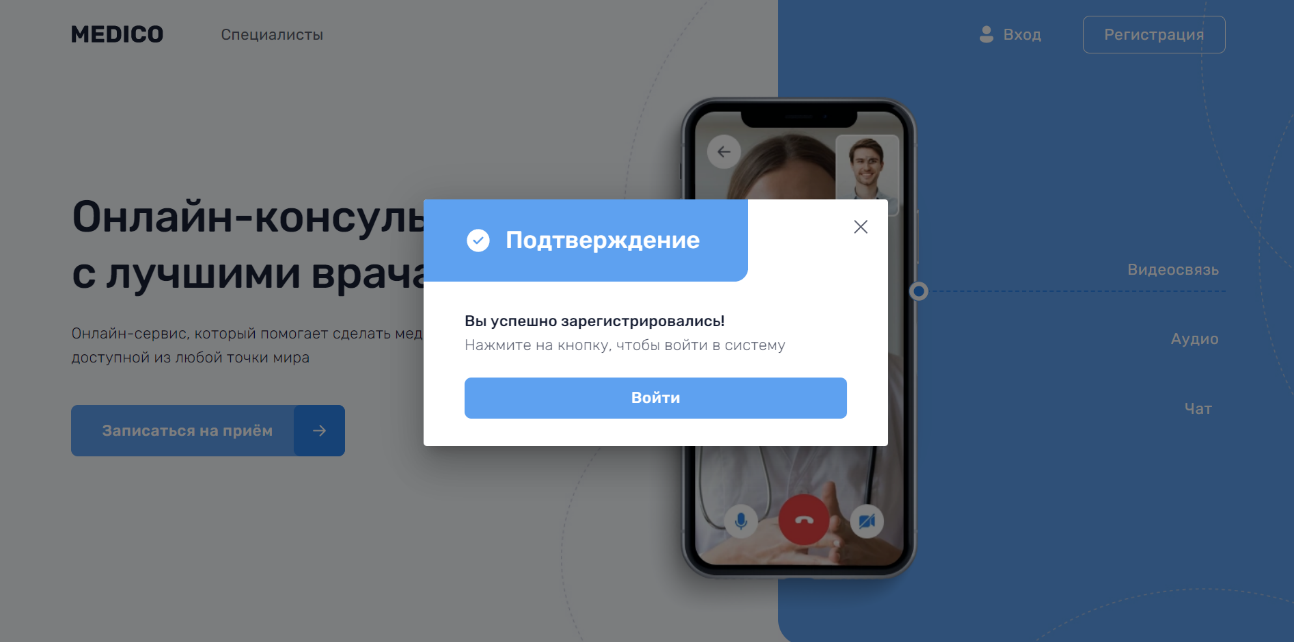


Рисунок 9 – Сообщение об успешном подтверждении аккаунта

После авторизации пациенту нужно заполнить анкету чтобы иметь доступ к личному кабинету (Рисунки 10-11).

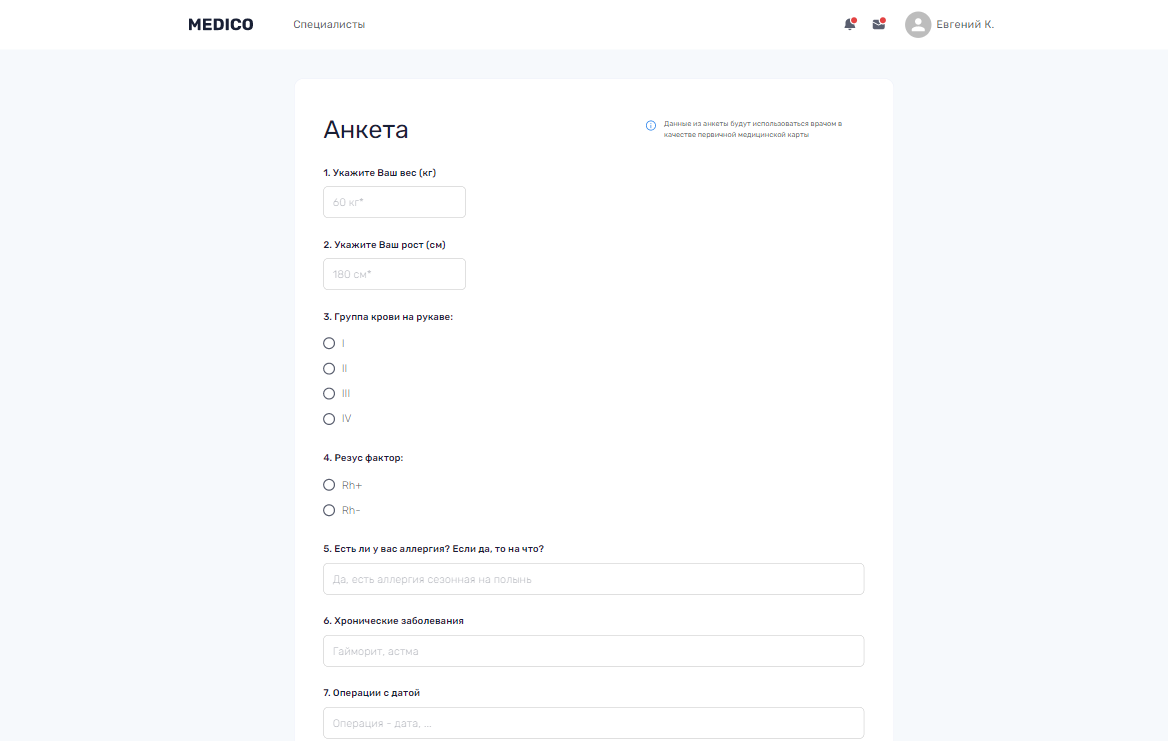


Рисунок 10 – Анкета пациента

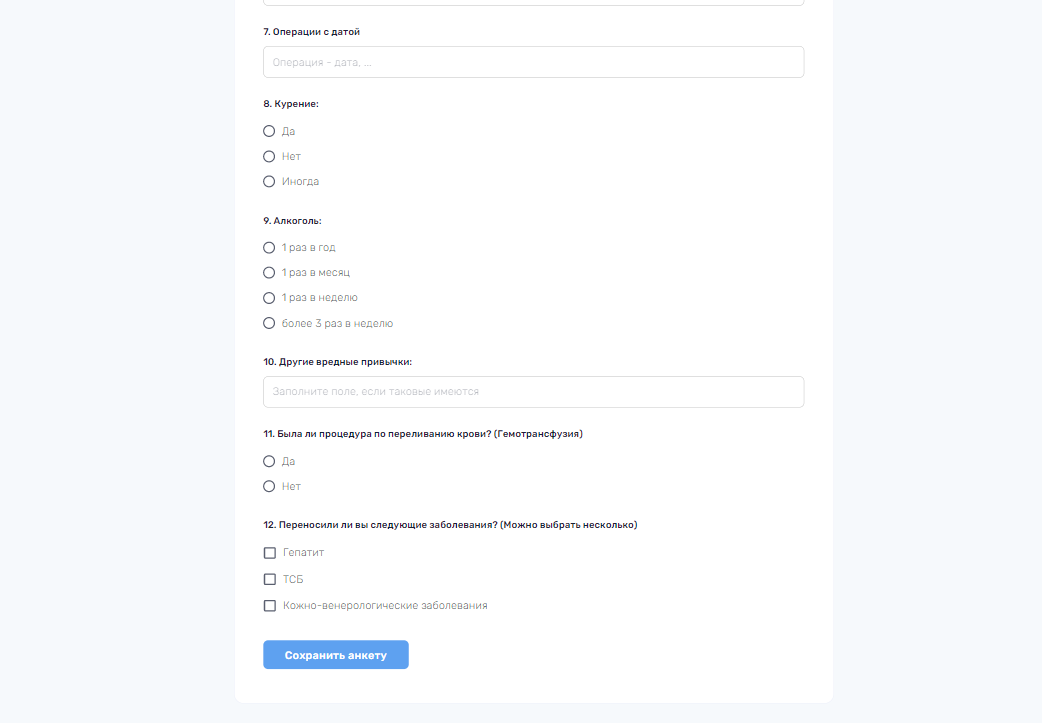


Рисунок 11 – Анкета пациента

Если пользователь является доктором, то ему необходимо заполнить заявку на врача чтобы модератор мог подтвердить его (Рисунки 12-13).

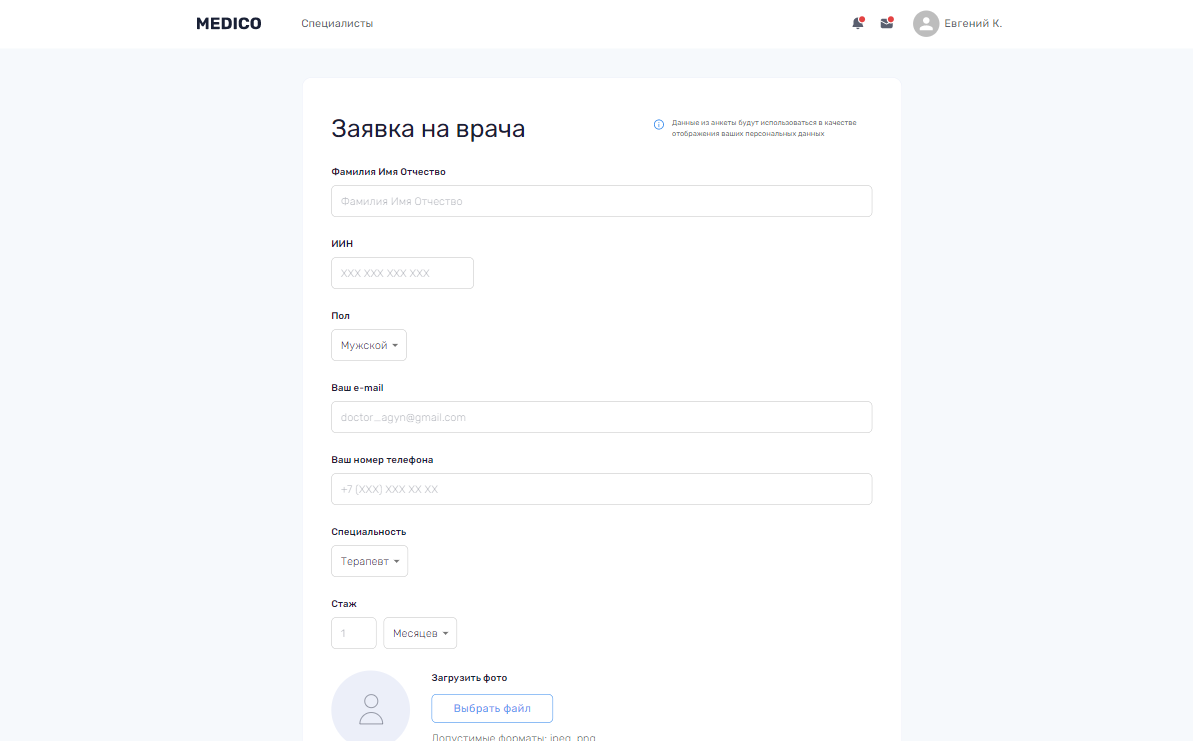


Рисунок 12 – Заявка на врача

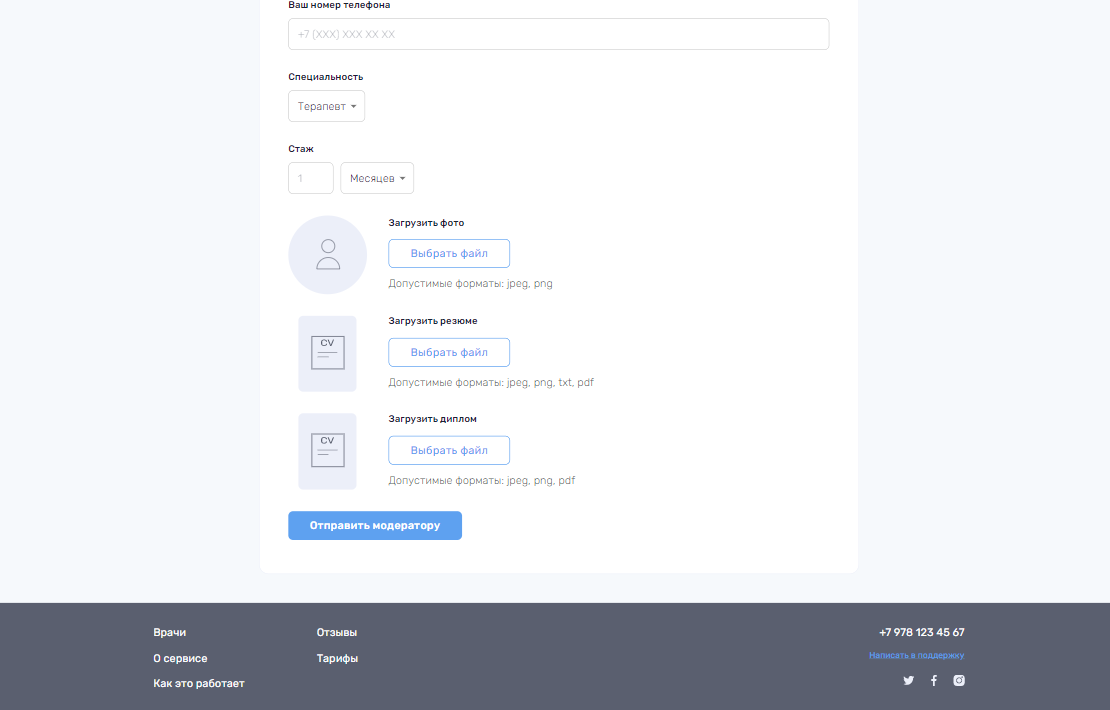


Рисунок 13 – Заявка на врача

После заполнения анкеты пациент имеет возможность найти врача с дальнейшей записью на прием (Рисунки 14-15).

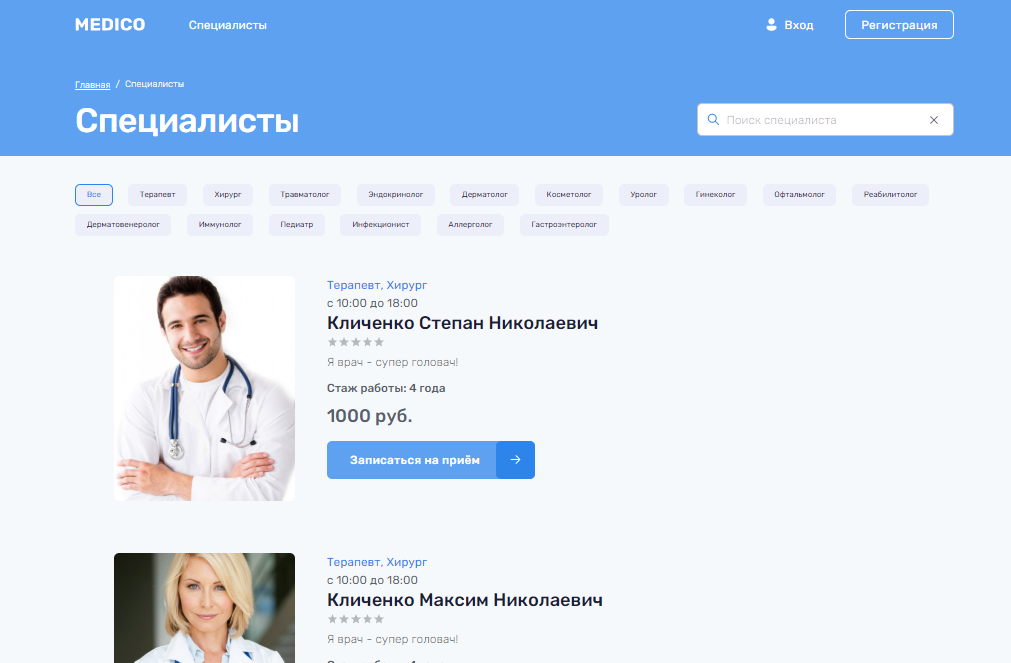


Рисунок 14 – Страница со специалистами (часть 1)

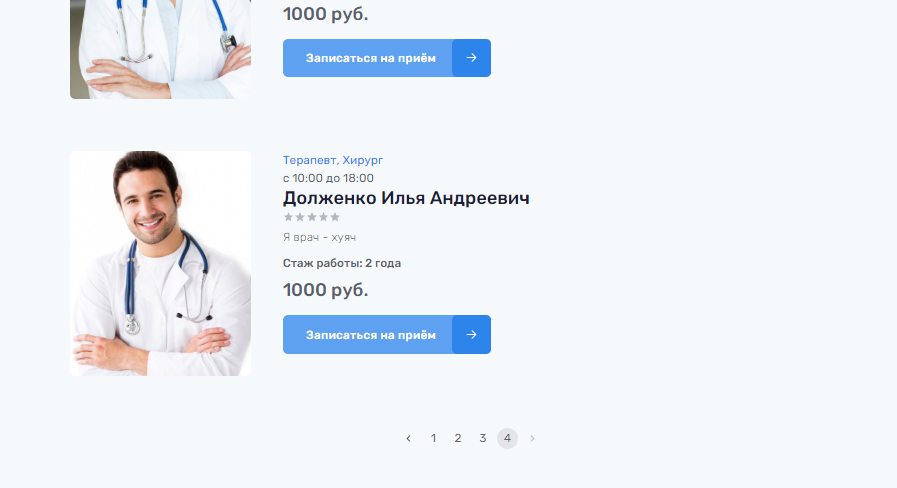


Рисунок 15 – Страница со специалистами (часть 2)

На рисунках 16-17 представлена детальная страница врача, на которой можно посмотреть отзывы пациентов, образование, опыт работы и основные направления по которым работает врач.

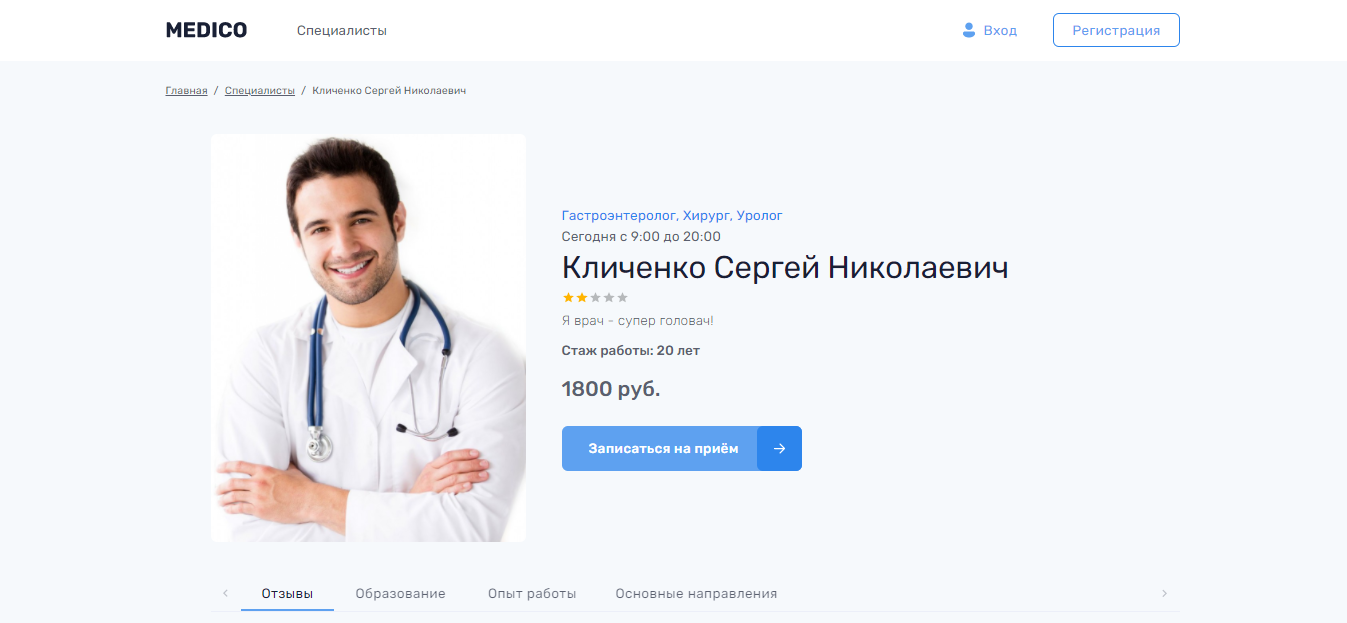


Рисунок 16 – Детальная страница врача

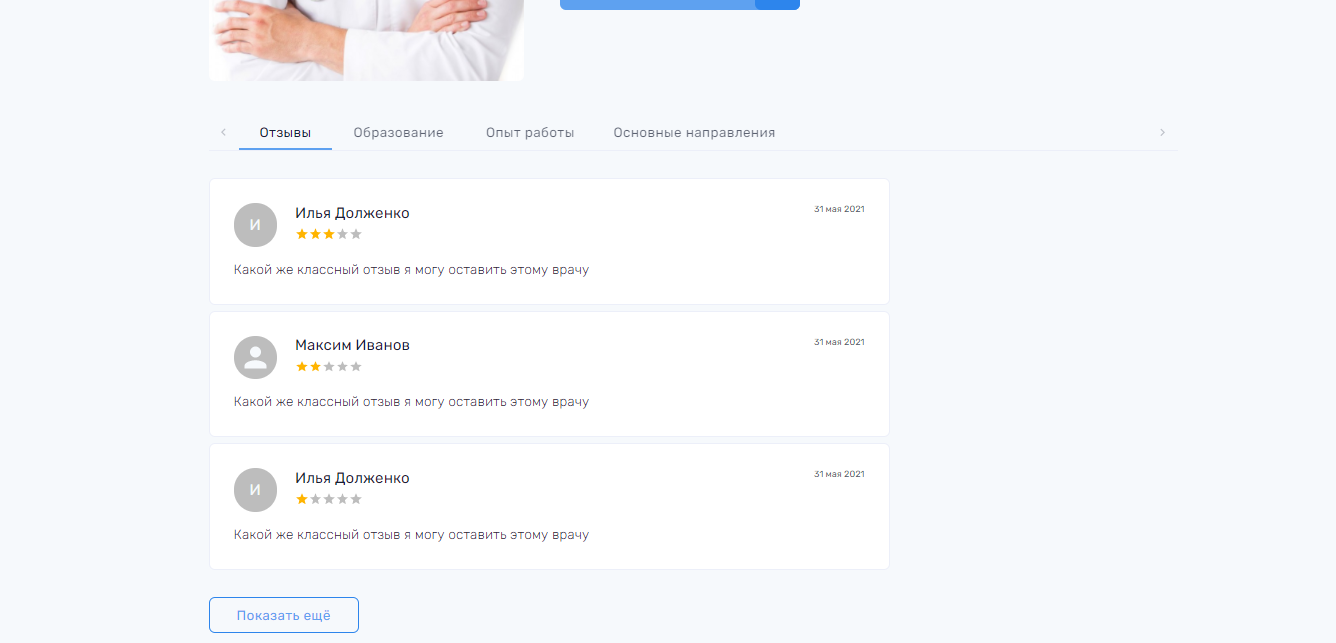


Рисунок 17 – Детальная страница врача

Если пользователь авторизован и является пациентом, то кнопка “Записаться на приём” активна и при клике на нее открывается первый этап записи на консультацию с выбором даты, времени, способом связи и необходимой специальностью. Весь процесс записи состоит из четырех этапов. Первый из них представлен на рисунке 18.

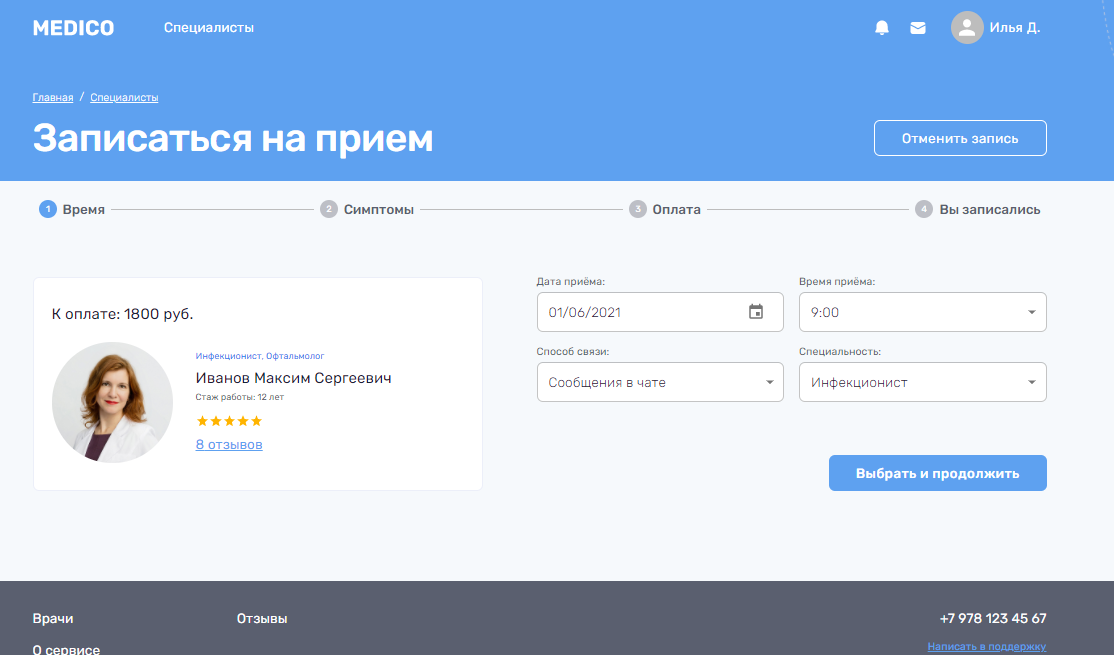


Рисунок 18 – Первый этап записи на консультацию

На втором этапе, представленном на рисунке 19, пациенту представляется возможность заполнить свои симптомы.

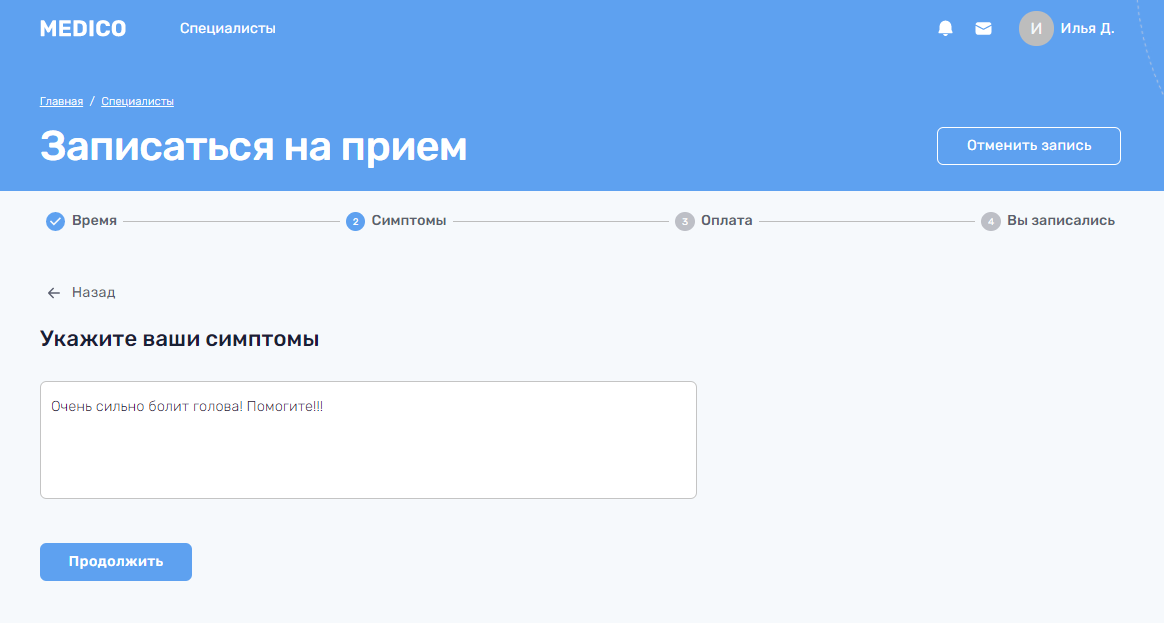


Рисунок 19 – Второй этап записи на консультацию

На третьем этапе пациенту предоставляется возможность удостовериться в введенных данных и оплатить консультацию. Данный этап представлен на рисунке 20.

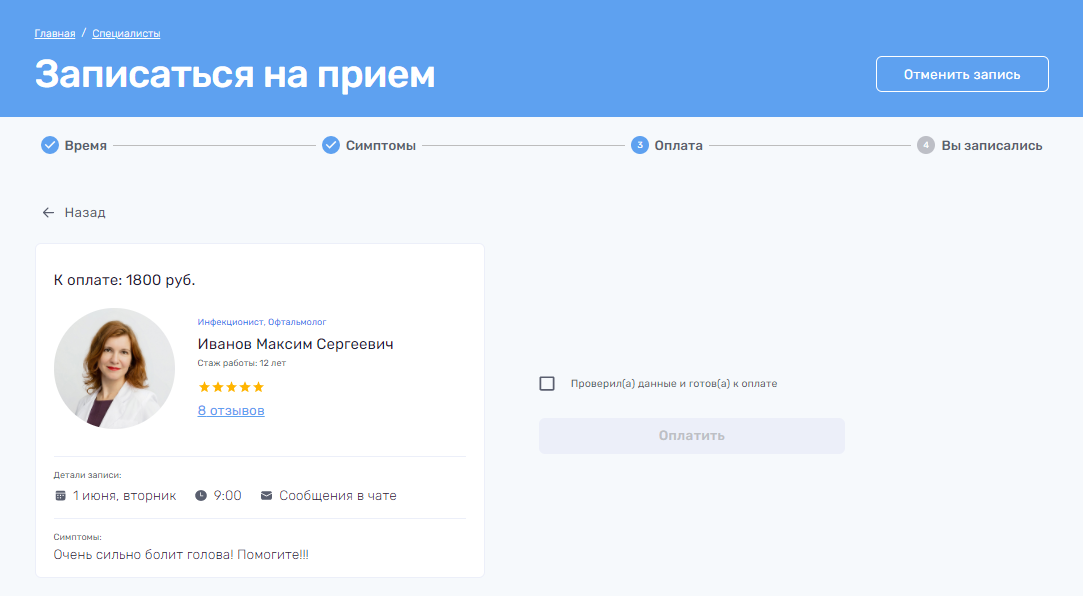


Рисунок 20 – Третий этап записи на консультацию

На завершающем этапе изображен результат записи на консультацию, также продублирована основная информация записи: имя специалиста, дата, время и способ коммуникации. Завершающий этап изображен на рисунке 21.

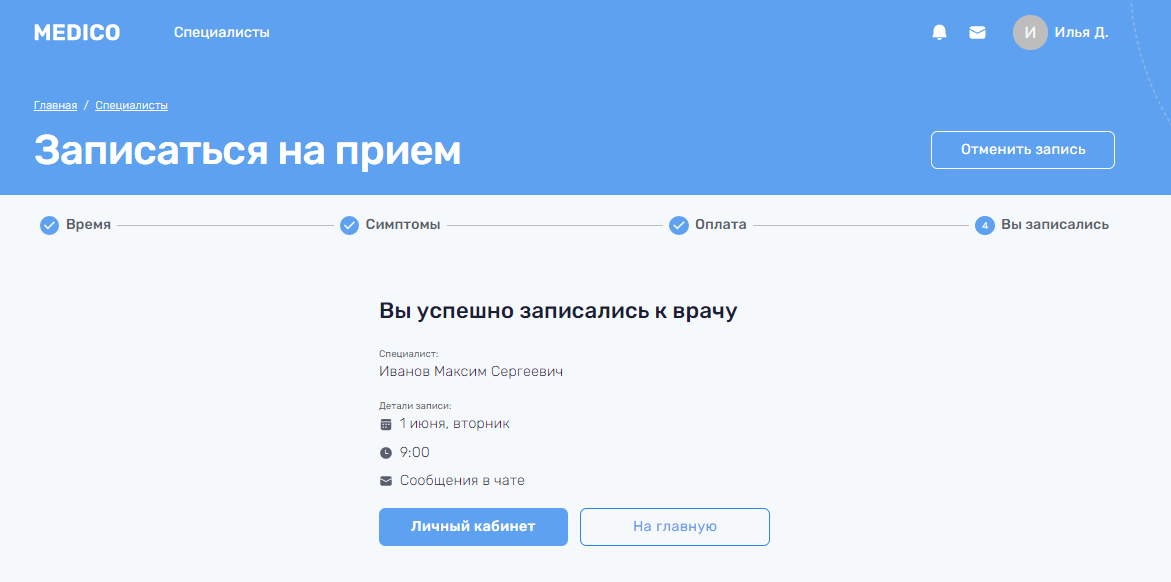


Рисунок 21 – Завершающий этап записи на консультацию

Дальнейший юзерфлоу пациента представляет из себя работу в личном кабинете, который состоит из следующих страниц: мои записи, назначения, мои анализы, мед. карта, сообщения и настройки.

Страница “Мои записи” (рисунок 22) представляет из себя три списка: активные консультации, запланированные и история.

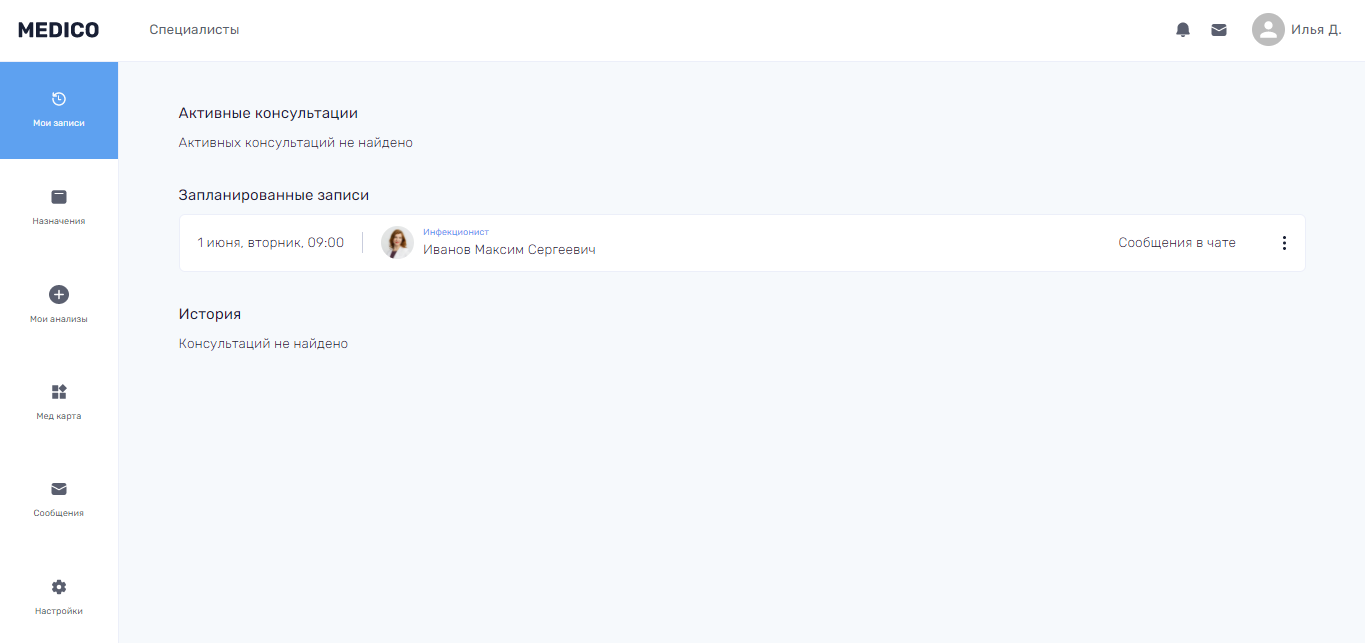


Рисунок 22 – Страница записей у пациента

После консультации пациент имеет возможность оставить отзыв о враче, кликнув на соответствующую кнопку на записи в списке истории. Диалоговое окно отправки отзыва представлено на рисунке 23.

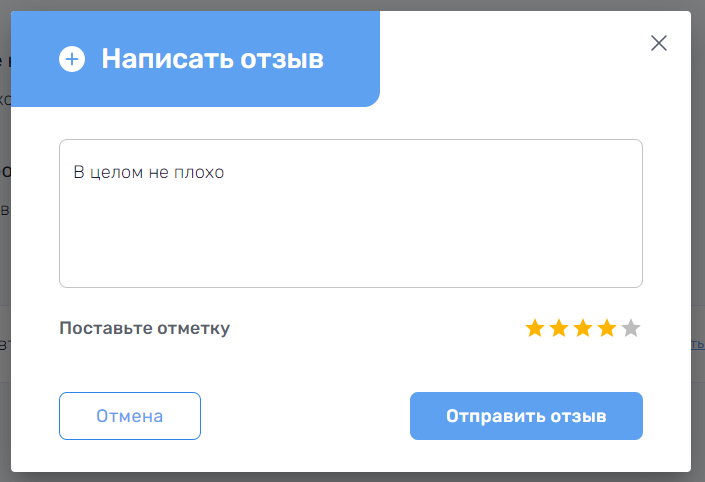


Рисунок 23 – Диалоговое окно отправки отзыва о специалисте

Следующая страница “Назначения” содержит список рекомендаций от врача, заполняемых после процесса консультации с пациентом. Данная страница представлена на рисунке 23.

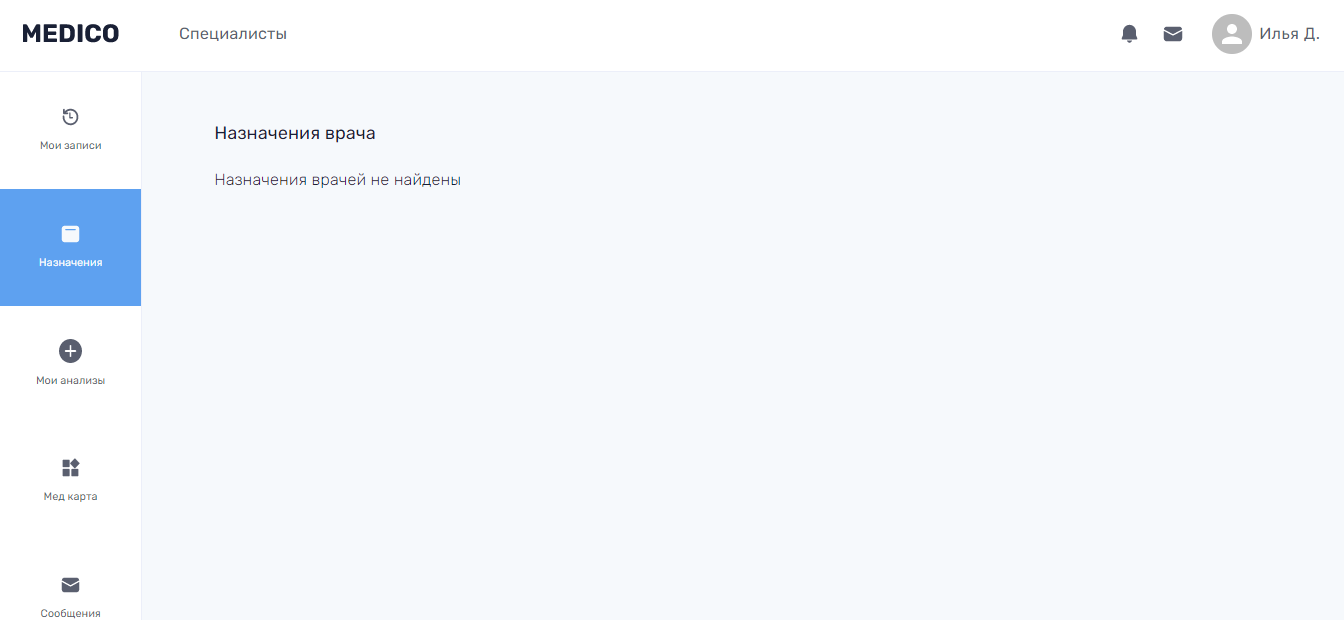


Рисунок 23 – Страница назначений

На странице “Мои анализы” пациент имеет возможность загрузить анализ либо снимок. Диалог загрузки файла, представленный на рисунке 24, вызывается при клике на кнопку, расположенную правее названия страницы.

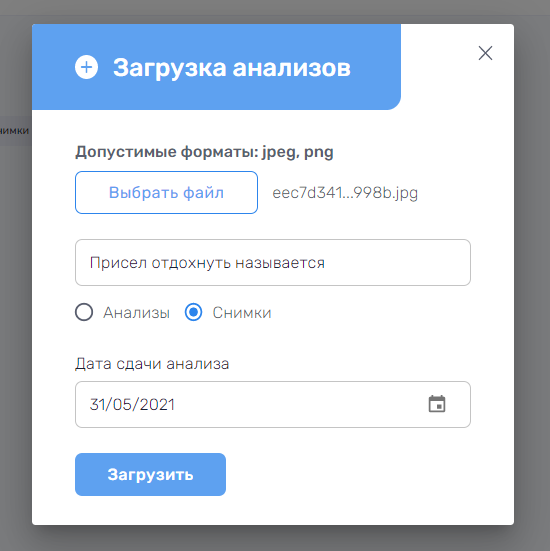


Рисунок 24 – Диалог загрузки анализов

После успешной загрузки файла на странице появится загружаемый анализ/снимок с подписью в виде даты сдачи и названия. Также все загружаемые файлы можно сортировать по их типу. Данная страница представлена на рисунке 25.

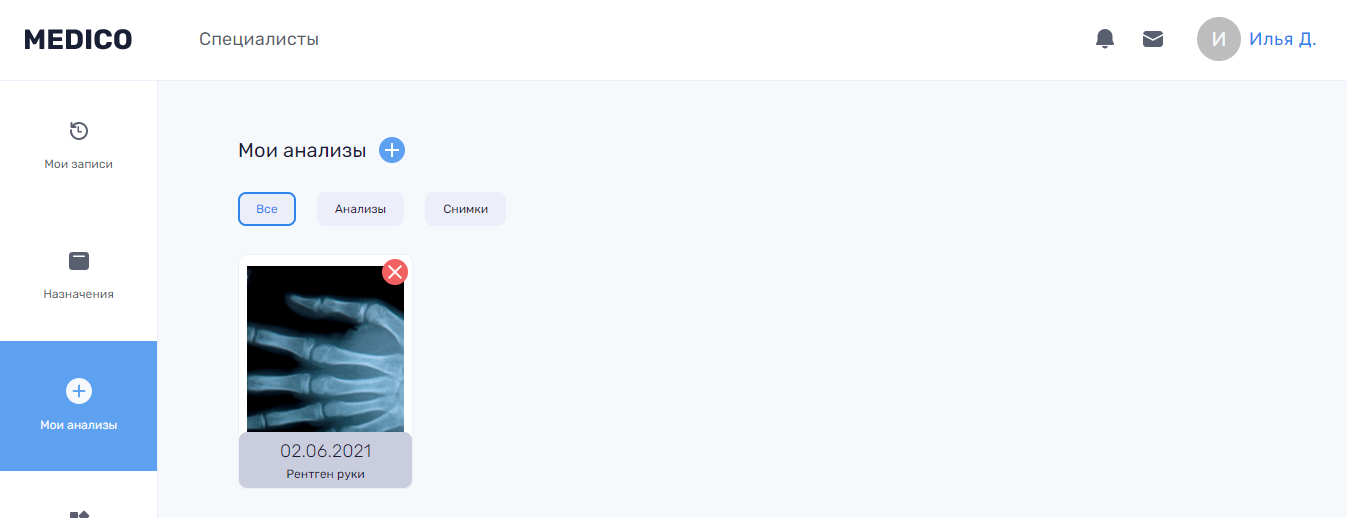


Рисунок 25 – Страница анализов

Также загружаемые файлы можно посмотреть более детально, кликнув по изображению анализа. Данная возможность представлена на рисунке 26.

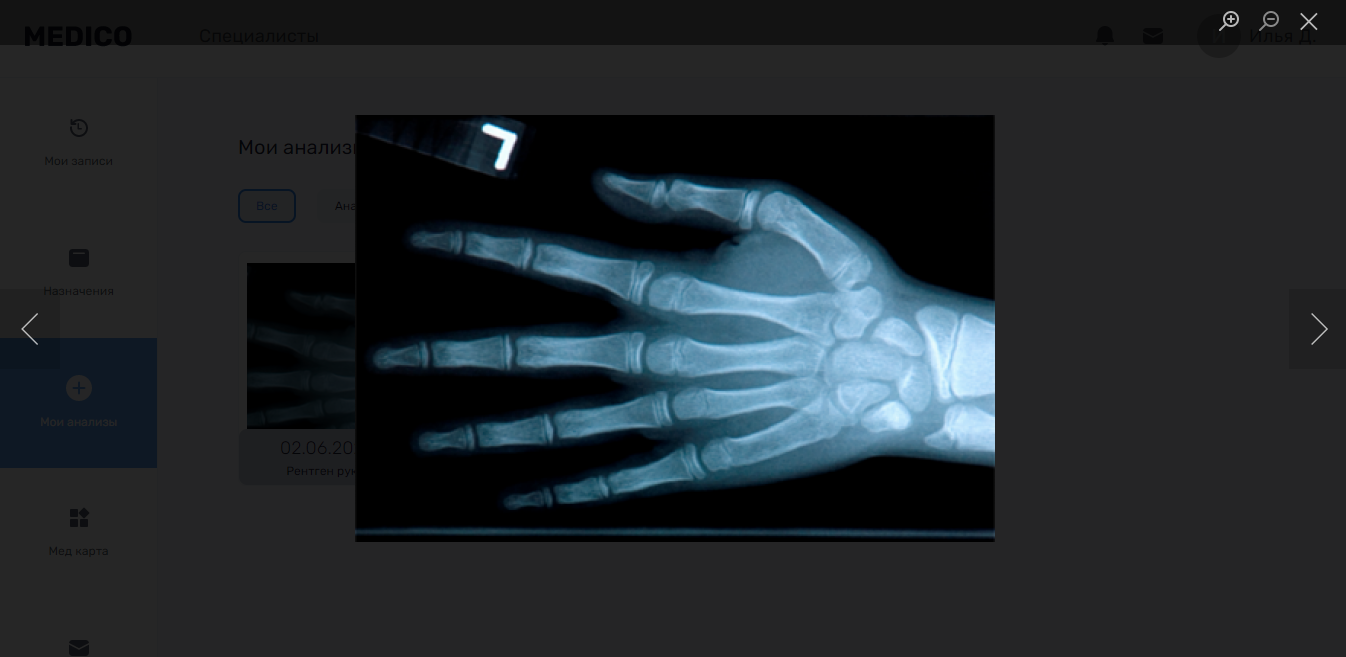


Рисунок 26 – Детальный просмотр анализа

Также при клике на кнопку удаления анализа или снимка появится диалоговое окно, внешний вид которого представлен на рисунке 27.

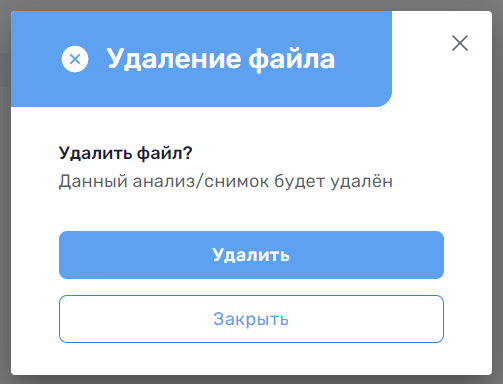


Рисунок 27 – Диалоговое окно удаления файла

Следующая страница “Мед. карта” предназначена для заполнения информации о себе, такой как: рост, вес, группа крови, аллергия, вредные привычки, переливание крови, операции и хронические заболевания. Также на данной странице автоматически считается Индекс Массы Тела с соответствующим пояснением. Данная страница представлена на рисунке 27.

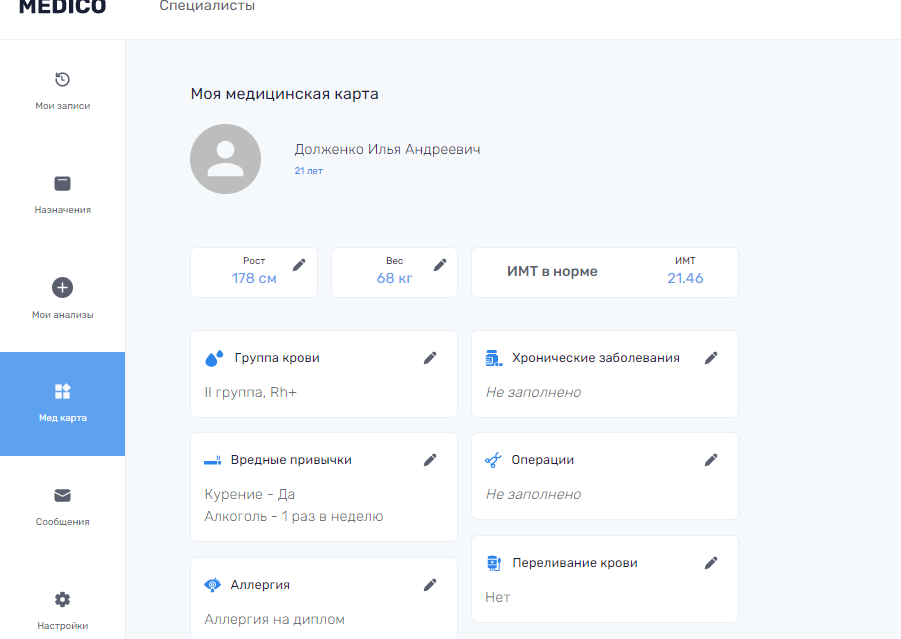


Рисунок 27 – Страница медицинской карты пациента

При клике на иконку редактирования открывается диалоговое окно с выбранными полями. Данный процесс представлен на рисунке 28.

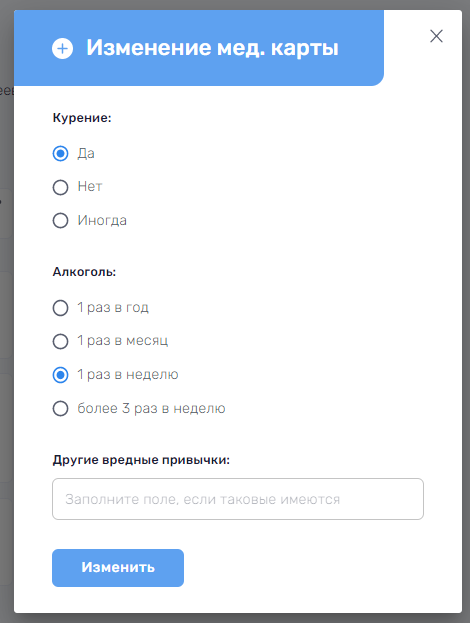


Рисунок 28 – Процесс изменения мед. Карты

Страница настроек является общей и доступна как для пациента, так и для врача. На данной странице, изображенной на рисунке 29, можно изменять как свое имя, фамилию, отчество, так и дату рождения, номер телефона, пол.

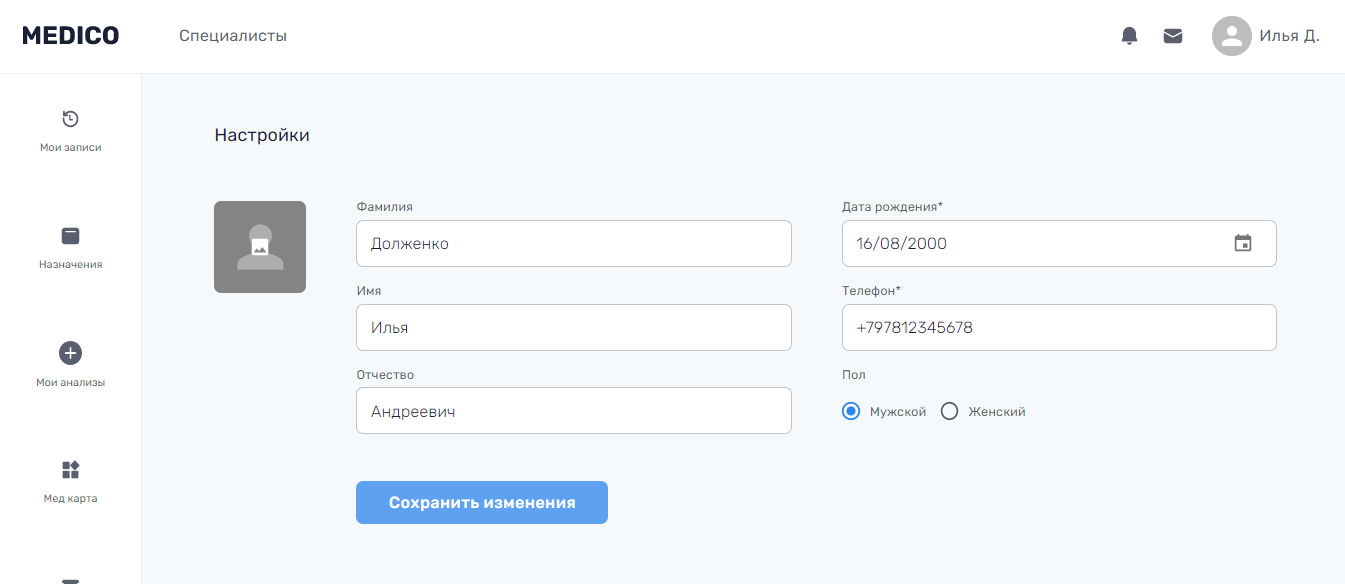


Рисунок 29 – Общая страница настроек

На главной странице личного кабинета врача находится информация о своем профиле. Имеется возможность просмотра отзывов и их подгрузка, просмотр и изменение образования и опыта работы, просмотр основных специальностей, выбранных при заполнении заявки на врача. Данный профиль врача представлен на рисунке 30.

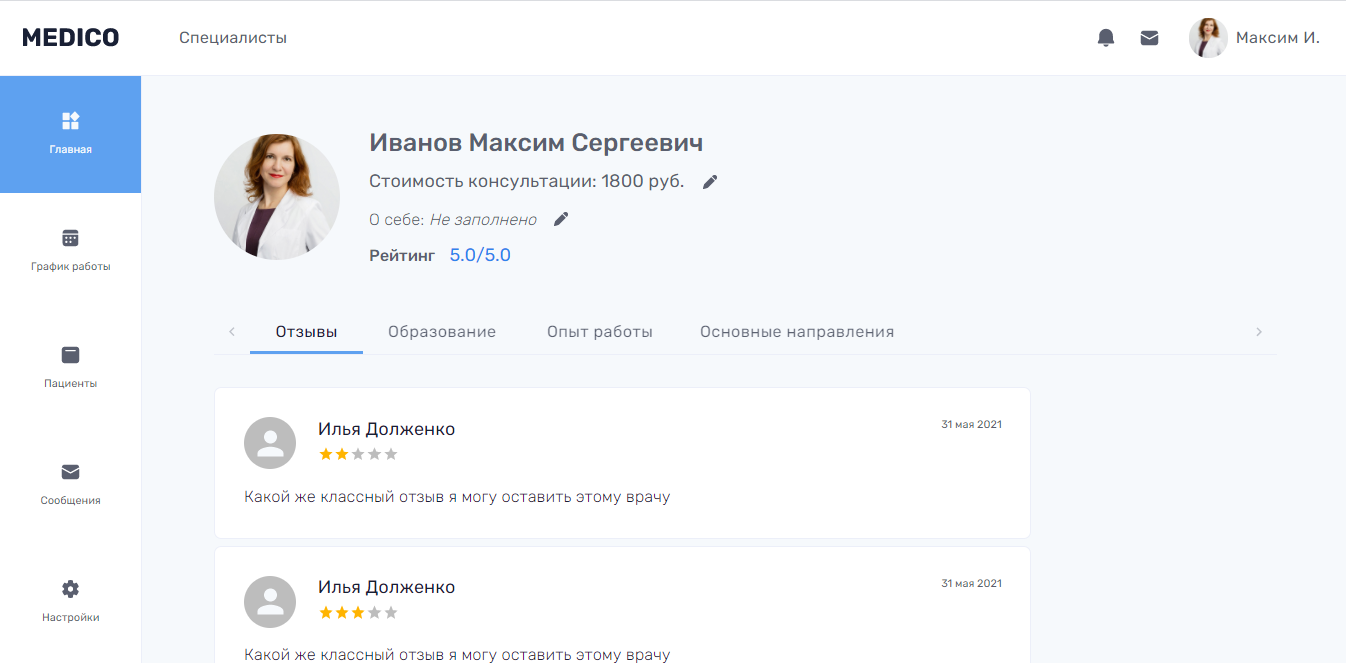


Рисунок 30 – Главная страница врача

При нажатии на кнопку редактирования стоимости консультации или информации о себе появляется диалоговое окно с соответствующей формой. Процесс редактирования информации о себе представлено на рисунке 31.



Рисунок 31 – Форма редактирования “О себе”

Редактирование образования и опыта работы происходит по аналогичному сценарию. Только теперь они содержат динамическое количество записей, т.е. имеется возможность добавить в одной форме сразу несколько записей. Пример редактирования образования приведен на рисунке 32.

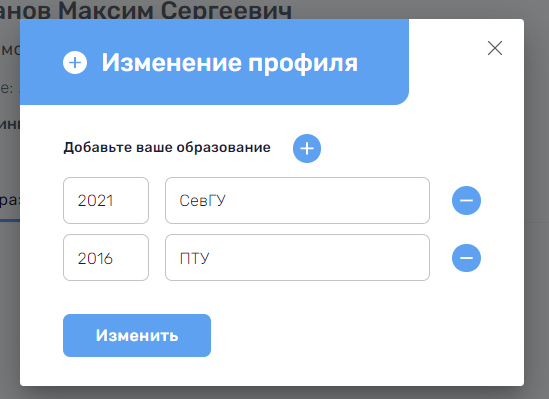


Рисунок 32 – Пример редактирования образования

На следующей странице “График работы” (рисунок 33) врач может изменить время своего приема по дням, либо установить дни без приема.

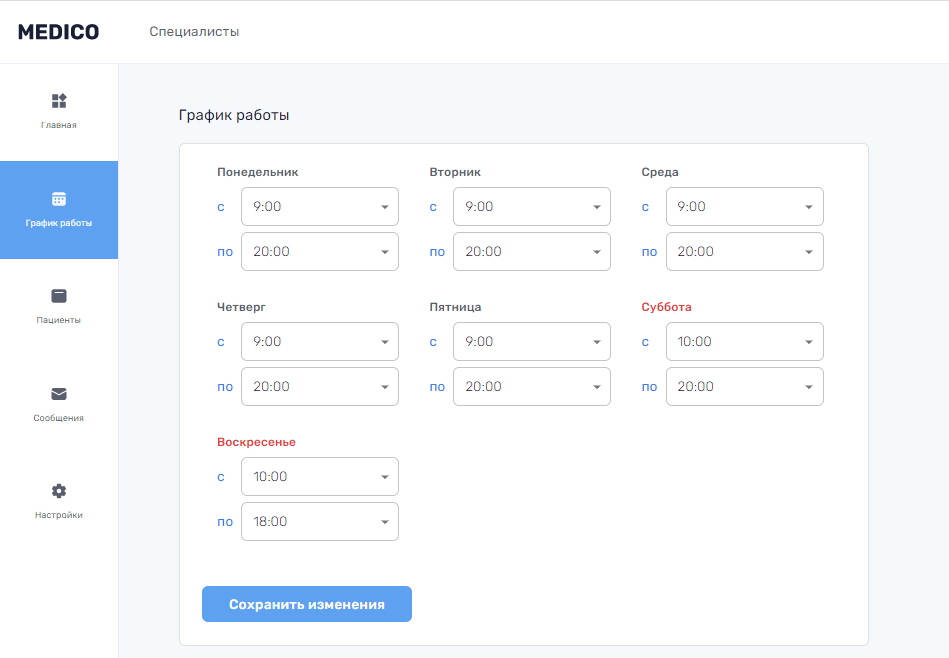


Рисунок 33 – Страница редактирования графика работы

Страница “Пациенты” (рисунок 34) разделена на два блока: список с пациентами по категории (новые или история) и календарь с выбором даты, по которой необходимо отображать пациентов.

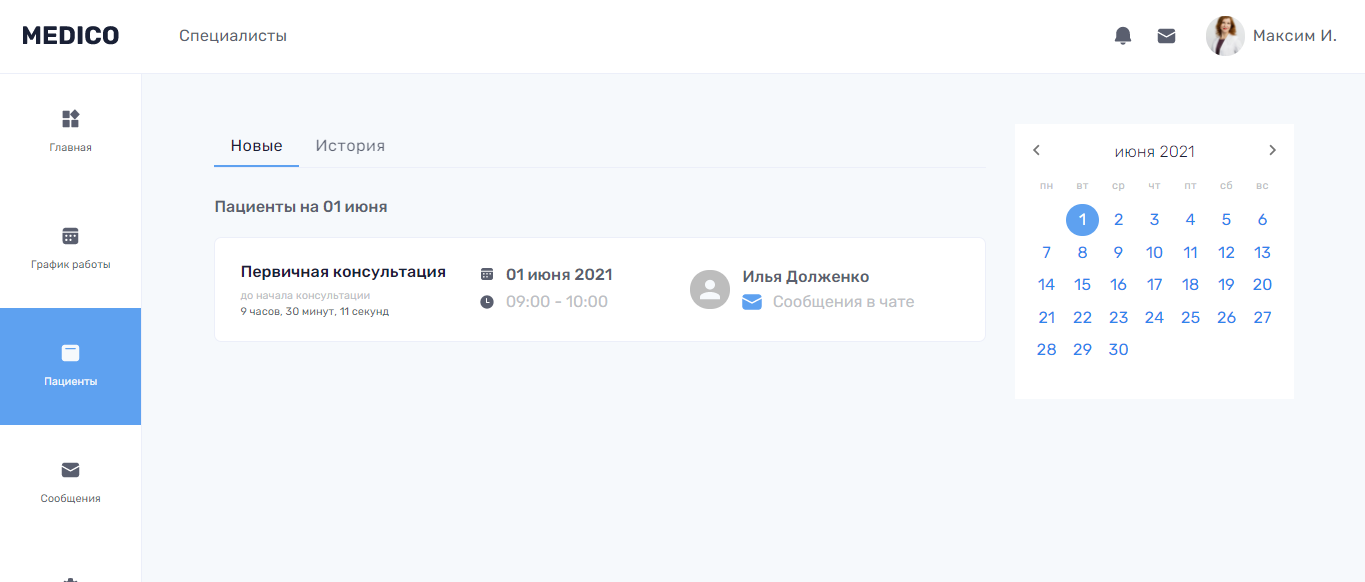


Рисунок 34 – Страница просмотра пациентов

При клике на пациента открывается страница просмотра его профиля, которая содержит его медицинскую карту, симптомы, анализы и прошлые назначения врачей. Страница профиля пациента представлена на рисунке 35.

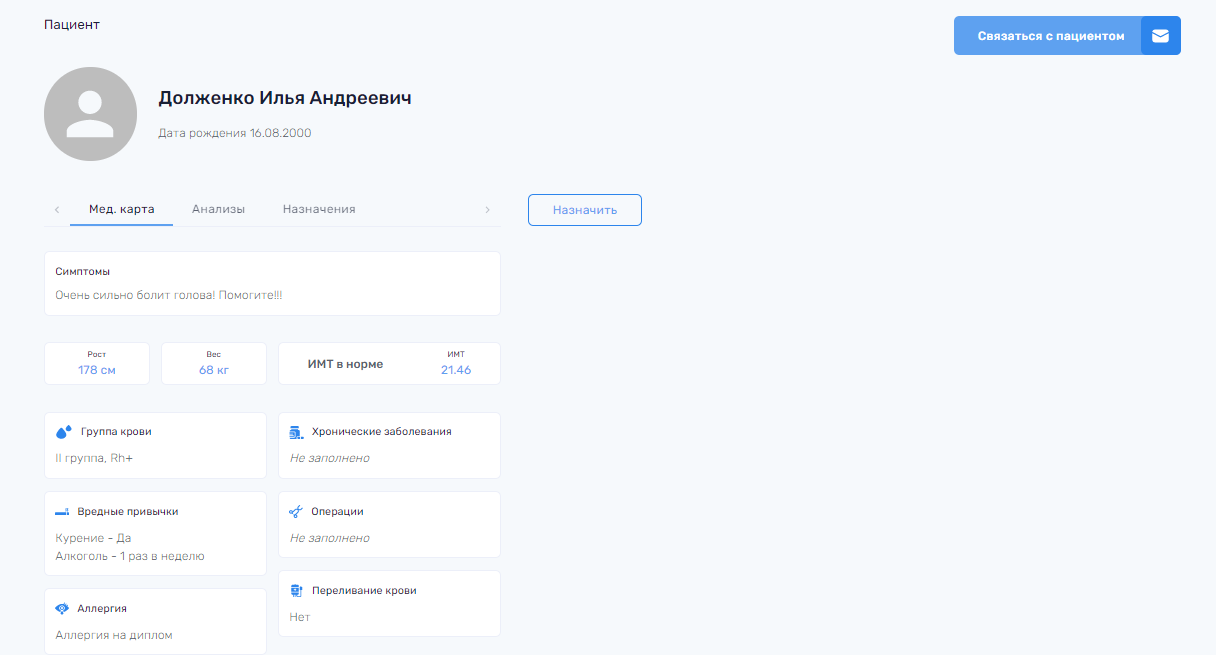


Рисунок 35 – Страница просмотра профиля пациента

На данной странице также находится кнопка назначения рекомендаций, при клике на которую открывается диалоговое окно с текстовым полем (рисунок 36).

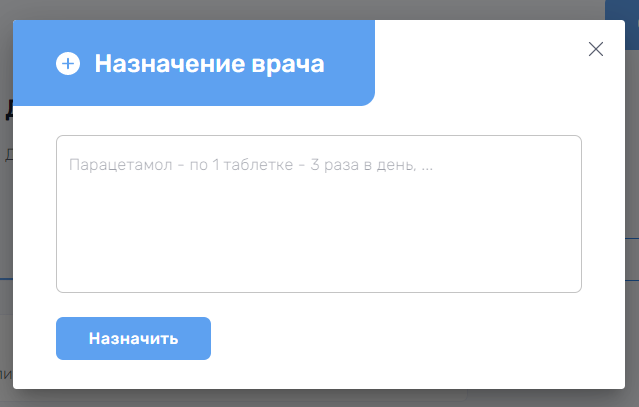


Рисунок 36 – Окно назначения рекомендаций пациенту

Также на странице находится кнопка “связаться с пациентом”, при клике на которую открывается диалог на общей странице чата. Страница “Сообщения” с выбранным диалогом изображена на рисунке 37.

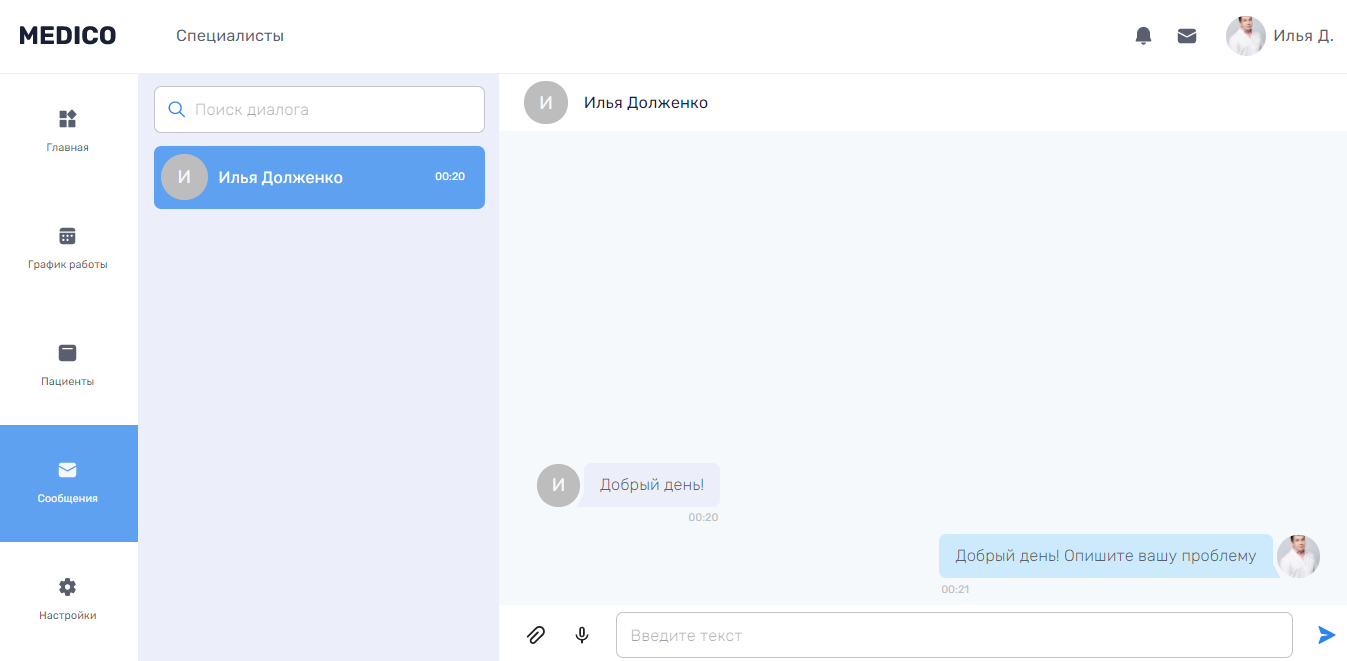


Рисунок 37 – Страница диалога с пациентом

В случае если диалог не выбран, то на странице отображается анимированное изображение с текстом “Выберите диалог”. Данная ситуация представлена на рисунке 38.

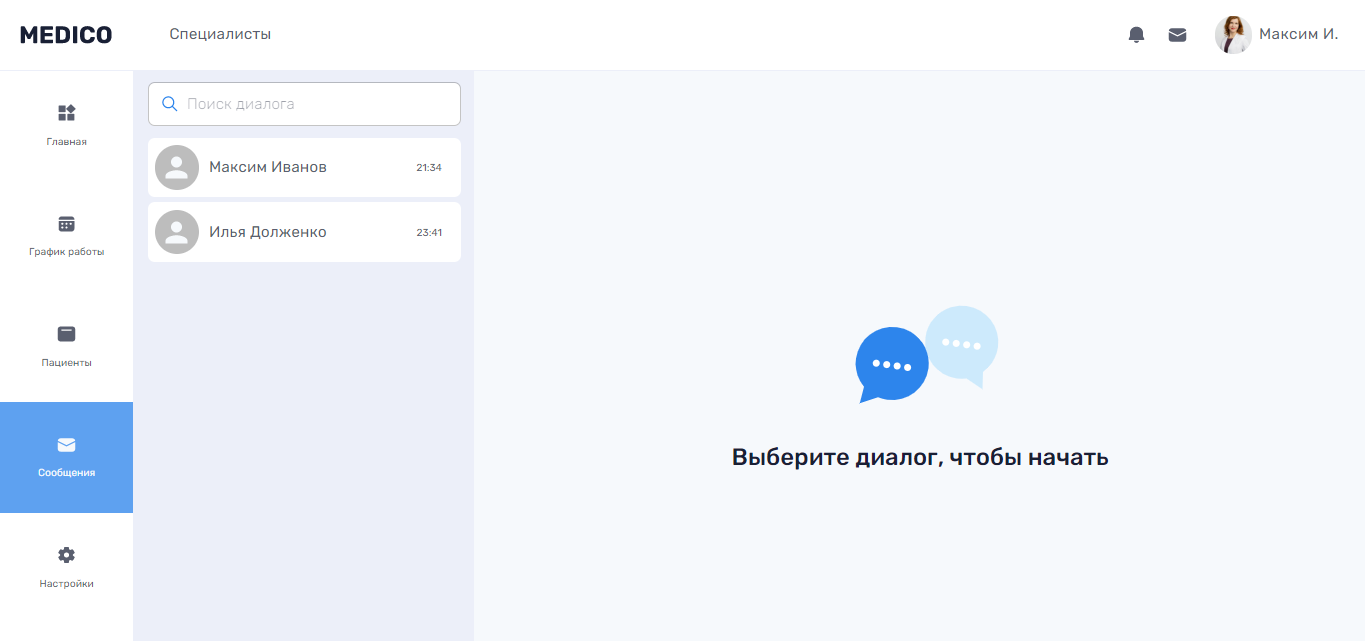


Рисунок 38 – Страница сообщений с невыбранным диалогом

Та же страница, но уже со стороны пациента имеет некоторое отличие – под именем врача в диалогах и верхней части сообщений написаны специальности, которыми обладает врач. Данный вид страницы представлен на рисунке 39.

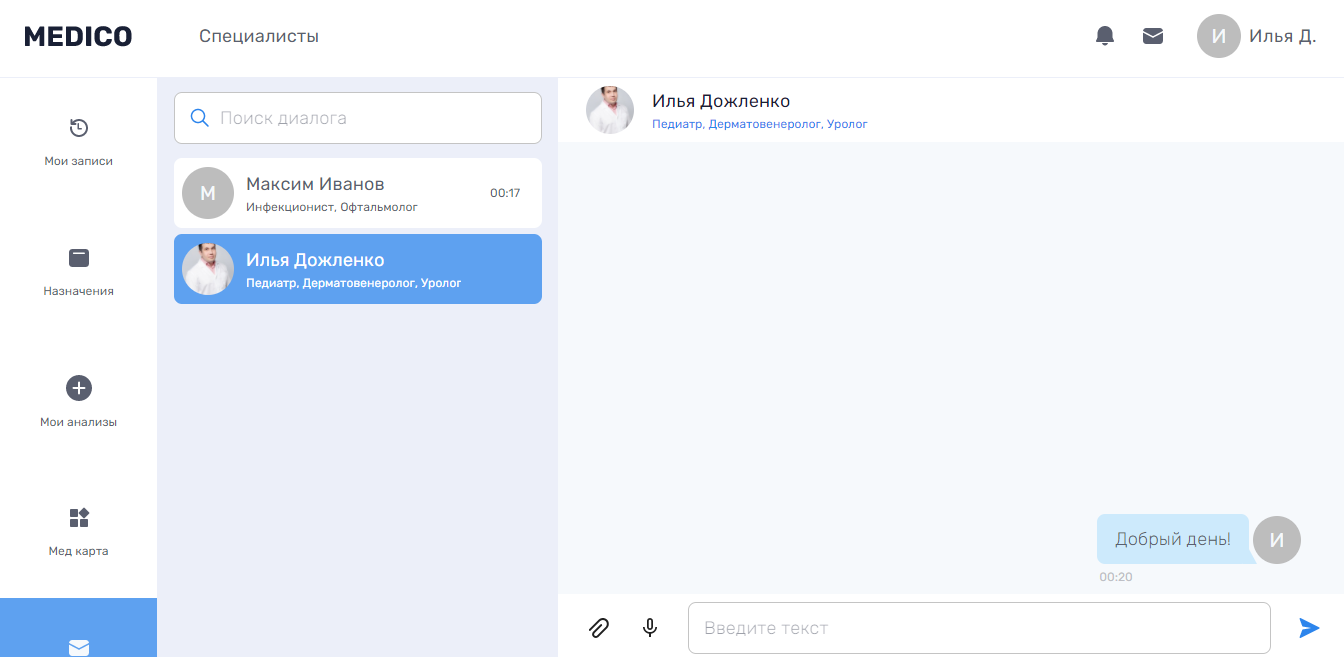


Рисунок 39 – Страница сообщений со стороны пациента

При общении в чате как пациент, так и врач имеют возможность отправить:

* фото (рисунок 40);

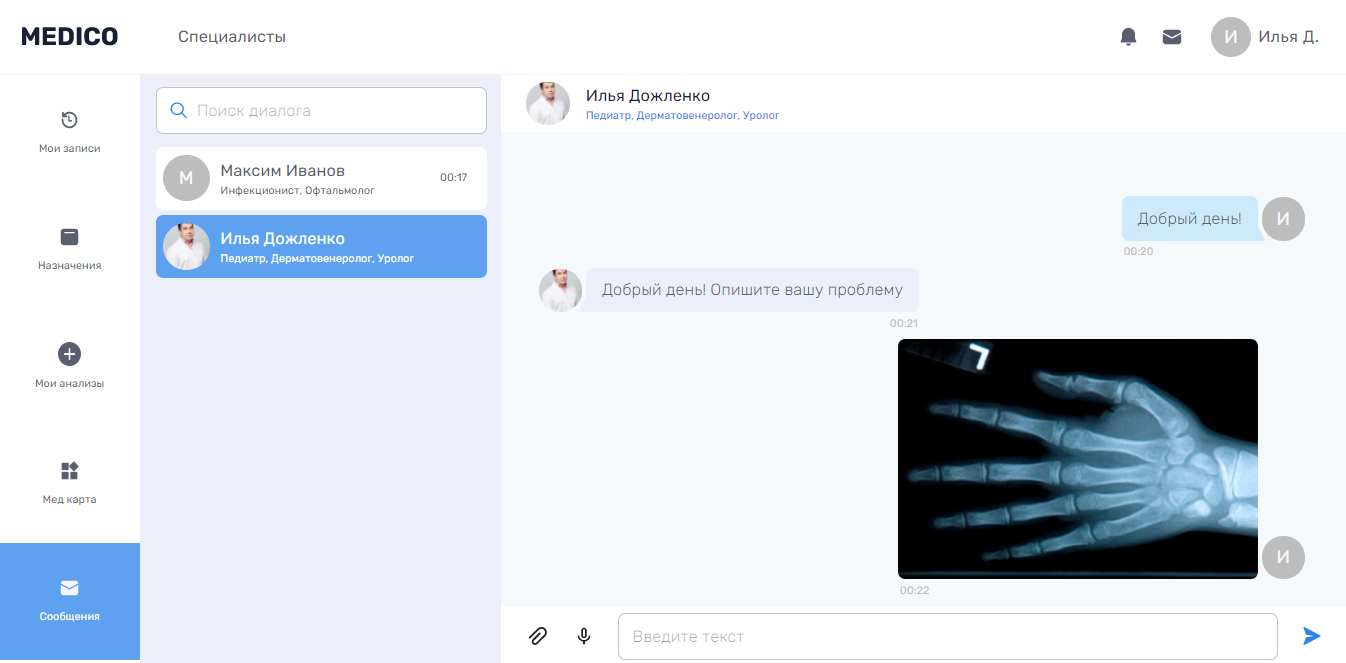


Рисунок 40 – Пример отправки изображения в чат

* аудиосообщение (рисунок 41);



Рисунок 41 – Пример отправки аудиосообщения

* просто текстовый файл (рисунок 42).

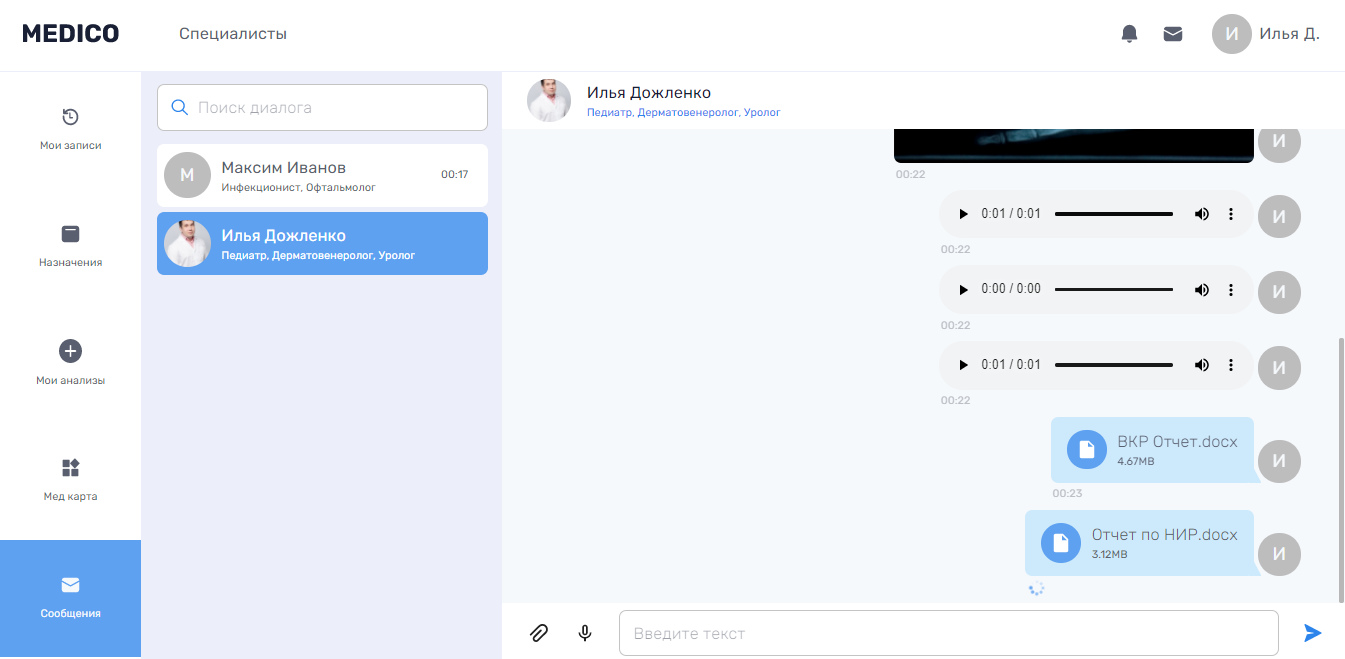


Рисунок 42 – Пример отправки файла

**ЗАКЛЮЧЕНИЕ**

В ходе прохождения преддипломной практики было разработано техническое приложения для клиентской часть системы онлайн-консультаций с врачами. Были разработаны компоненты и интерфейсы программного модуля.

**СПИСОК ЛИТЕРАТУРЫ И ИНФОРМАЦИОННЫХ РЕСУРСОВ**

1. Node.js [Электронный ресурс]. URL: https://nodejs.org/en/ (дата обращения: 02.06.2021).

2. React - JavaScript-библиотека для создания пользовательских интерфейсов [Электронный ресурс]. URL: https://ru.reactjs.org/ (дата обращения: 02.06.2021).