**Министерство науки и высшего образования Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИМТО»**

Факультет СПО

Направление подготовки (специальность) 09.02.07 Информационные системы и программирование

**О Т Ч Е Т**

**о преддипломной практике**

Тема задания: Разработка прототипа клиентской части автоматизированной информационной системы для клиники «Стоматология Доктора Вознесенского» \_

Обучающийся Кунаев Н.Д. Группа Y2432

*(Фамилия И.О.) (номер группы)*

Согласовано:

|  |  |
| --- | --- |
| Руководитель практики от профильной организации: | Костяев Александр Александрович, заместитель генерального директора ООО «АКАДЕМИЯ ЦИФРОВОЙ КУЛЬТУРЫ» |
|  | (Фамилия И.О., должность и место работы) |
| Руководитель практики от Университета ИТМО: | Говорова Марина Михайловна, преподаватель факультета СПО |
|  | (Фамилия И.О., должность и место работы) |

Практика пройдена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_

«19» мая 2021г.

Санкт-Петербург

2021

**Министерство науки и высшего образования Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Факультет СПО**

**И Н Д И В И Д У А Л Ь Н О Е З А Д А Н И Е**

**на преддипломную практику**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Студент** | Кунаев Н.Д. | | | **Группа №** | **Y2432** |
| (Фамилия И. О.) | | | | | |
| **Руководитель от профильной организации** | | | Костяев Александр Александрович, заместитель генерального директора ООО «АКАДЕМИЯ ЦИФРОВОЙ КУЛЬТУРЫ» | | |
| (Фамилия И. О., место работы, должность) | | | | | |
| **Руководитель от Университета ИТМО** | | Говорова Марина Михайловна, преподаватель факультета СПО | | | |
| (Фамилия И. О., место работы, должность) | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Тема задания:** | Разработка клиентской части автоматизированной информационной | | |
| системы для клиники «Стоматология Доктора Вознесенского» | | | |
| **Сроки прохождения практики:** | | 22.04.2021 – 19.05.2021 | |
| **Место прохождения практики:** | | ООО «АКАДЕМИЯ ЦИФРОВОЙ КУЛЬТУРЫ» | |
| **Должность практиканта:** | | - | |
| **1. Виды работ и требования к их выполнению:** | |  | |
| Производственная практика проводится непрерывно на базе «………»***.*** В ходе прохождения практики выполняются следующие виды работ: | | | |
| 1. Вводный инструктаж по технике безопасности и общим целям и задачам практики. | | | |
| 1. Анализ поставленной задачи: уточнение требований технического задания. | | | |
| 1. Детальное проектирование системы/приложения/программы. | | | |
| 1. Реализация, отладка и тестирование системы/приложения/программы. | | | |
| 1. Формирование отчетных материалов и представление программы. | | | |
|  | | | |
| **2. Виды отчетных материалов и требования к их оформлению:** | | |  |
| По результатам прохождения практики составляется отчет, в котором представляется анализ поставленной задачи, описание процесса проектирования системы/приложения/программы, модель системы/приложения/программы и техническое задание. Программный прототип представляется руководителю(ям) практики. Оформление отчета должно соответствовать Рекомендациям по оформлению технических документов факультета СПО Университета ИТМО. | | | |

**1. ПЛАН-ГРАФИК**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №  этапа | Наименование этапа | Срок завершения этапа | Виды работ | Форма отчетности |
| 1 | 2 | 3 | 4 | 5 |
|  | Вводный инструктаж | 22.04.2021 – 25.04.2021 | Ознакомление с инструкцией по технике безопасности. Ознакомление с целями и задачами производственной практики | Журнал по технике безопасности |
|  | Анализ поставленной задачи | 26.04.2021 – 02.05.2021 | Анализ технического задания. Уточнение требований (при необходимости). | Отчет по практике.  Дневник прохождения практики. |
|  | Детальное проектирование системы/приложения/программы. | 03.05.2021 – 16.05.2021 | Детальное проектирование модулей (составных частей программы). | Отчет по практике.  Дневник прохождения практики. |
|  | Реализация, отладка и тестирование системы/приложения/программы. | Реализация, отладка и тестирование системы/приложения/программы (доработка прототипа). | Отчет по практике.  Дневник прохождения практики. |
|  | Подготовка отчетных материалов и представление программы | 17.05 – 19.05.2021 | Формирование отчета о прохождении практики, демонстрация работы системы/приложения/программы. | Отчет по практике.  Дневник прохождения практики. |

**2.  ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ ОТЧЕТНОСТИ**

|  |  |
| --- | --- |
| Наименование формы отчетности | Требования к оформлению |
| Отчет о практике | Отчет о преддипломной практике оформляется в соответствии с Методическими указаниями по преддипломной практике по специальности 09.02.07 Информационные системы и программирование и Рекомендациям по оформлению технических документов факультета СПО Университета ИТМО. Титульный лист отчета оформляется в соответствии с Приложением В Методических указаний. |
| Индивидуальное задание | Индивидуальное задание оформляется в соответствии с Методическими указаниями по преддипломной практике по специальности 09.02.07 Информационные системы и программирование (Приложение А) и включается в отчет о преддипломной практике после титульного листа. |
| Дневник практики | Дневник практикиоформляется в соответствии с Методическими указаниями по преддипломной практике по специальности 09.02.07 Информационные системы и программирование (Приложение Б). |
| Аттестационный лист | Аттестационный лист оформляется в соответствии с Методическими указаниями по преддипломной практике по специальности 09.02.07 Информационные системы и программирование (Приложение Г) |

**Задание выдано:**

**Руководитель практики от Университета ИТМО\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)                                     ФИО

«22» апреля 2021г.

Согласовано:

**Руководитель практики от профильной организации \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)                                   ФИО

«22» апреля 2021г.

**Задание принял к исполнению:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** / \_\_\_\_\_\_\_\_\_

 (подпись обучающегося)                 ФИО

ВВЕДЕНИЕ

Целью преддипломной практики является углубление знаний и практических умений и получение практического опыта по основным видам деятельности «Разработка модулей программного обеспечения для компьютерных систем», «Осуществление интеграции программных модулей», «Сопровождение и обслуживание программного обеспечения компьютерных систем», «Разработка, администрирование и защита баз данных» и овладение соответствующими общими и профессиональными компетенциями: ОК 1, ОК 2, ОК 3, ОК 4, ОК 5, ОК 6, ОК 7, ОК 8, ОК 9, ОК 10, ОК 11, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.1, ПК 2.2, ПК 2.3, ПК 2.4, ПК 2.5, ПК 4.1, ПК 4.2, ПК 4.3, ПК 4.4, ПК.11.1, ПК.11.2, ПК.11.3, ПК.11.4, ПК.11.5, ПК.11. (см. рабочая программа и фонд оценочных средств по преддипломной практике).

Преддипломная практика проводится на базе ООО «Академия цифровой культуры».

Для достижения цели, поставленной в рамках производственной практики, необходимо выполнить следующие задачи:

1. Уточнение функциональных требований (при необходимости).
2. Программная реализация системы (клиентской части).
3. Проведение отладки и тестирования системы.
4. Формирование технического описания разработанной программной системы.
5. Представление реализованной программной системы.

Результатом практики является разработка веб-приложения для учёта пациентов и составление технического описания.

Отчёт по практике содержит следующие разделы:

1. Введение.
2. Разделы Технического описания.
3. Заключение.
4. Список источников.
5. Приложение А Техническое задание.
6. Приложение Б Модель системы.
7. ОБЩИЕ СВЕДЕНИЯ О РАЗРАБОТКЕ
   1. Наименование программы

АИС для медицинской клиники «Стоматология Доктора Вознесенского» (клиентская часть) (URL: <https://doc-stom.ru/>, дата обращения: 04.02.2021).

* 1. Программное обеспечение, необходимое для функционирования программы

Для функционирования программы необходимы следующие веб-браузеры:

* Google Chrome, версии 90.0.4430.212+;
* Mozilla Firefox Developer Edition, версии 89.0 x64+;
* Microsoft Edge, версии 90.0.818.56+;
* Safari, версии 14.1+;
* Opera, версии 76.0.4017.123+.
  1. Программные средства

В приложении использованы парадигмы реактивного, декларативного, объектно(прототипно)-ориентированного и функционального программирования.

Для реализации были использованы:

Во время разработки клиентской части:

NodeJs Server, версии 14.15.4;

Пакетный менеджер Yarn, версии 1.22.10;

Метаязык SASS, версии 1.32.5;

ReactJS, версии 17.0.1;

Bootstrap, версии 4.6.0;

React-bootstrap, версии 1.5.0;

Babel, версии 7.12.3.

В продакшн версии веб-приложения:

HTML, версии 5.3;

CSS, версии 2.1;

JS, версии ES9 (ECMAScript 2018).

1. Функциональное назначение
   1. Описание решаемых задач

Система предоставляет различный функционал для разных групп пользователей (директор/администратор/врач).

Врач может просматривать своё расписание, просматривать и вносить изменения в медицинские карты пациентов.

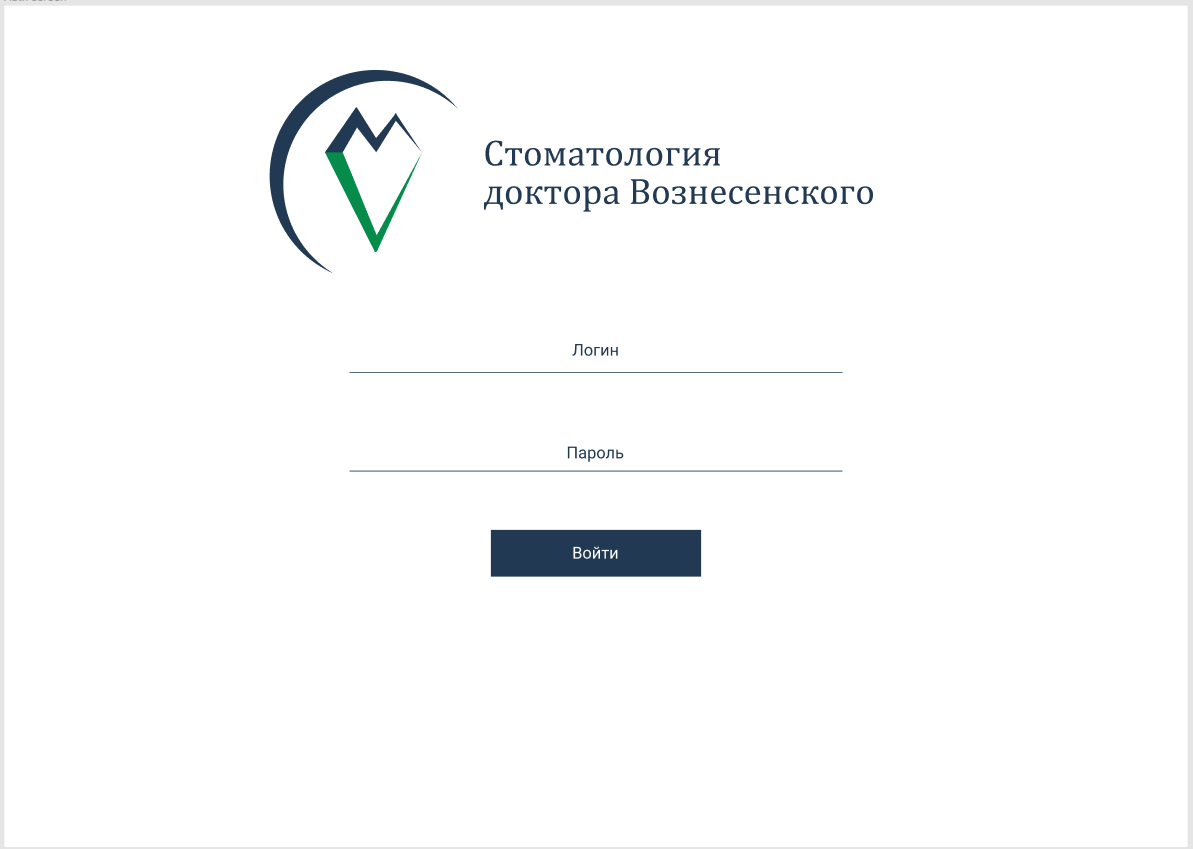
Администратор может делать то, что может делать врач, а также вносить изменения в расписание, добавлять новых пациентов и печатать накладные и прочие документы по выполненным услугам, вписывать пациентов в расписание врача (записывать на приём).

Директору доступен весь функционал других категорий пользователей, а также он может добавлять новые услуги, изменять их цену, добавлять новых пользователей, изменять любые данные пользователей, назначать рабочие дни для администраторов и врачей, добавлять новые товары и изменять их цены.

Просмотр расписания реализован посредством таблицы, отображающей календарь с метками врачей, работающих в определённые дни.

* 1. Описание функций

При входе в веб-приложение открывается форма авторизации (рисунок 1). После ввода корректных данных (логин и пароль) и нажатия на кнопку «Войти», исходя из введенных данных, определяется к какой группе прав относится пользователь, в зависимости от неё пользователь будет видеть разные варианты интерфейса (пока реализован только интерфейс директора).



*Рисунок 1 – Окно аутентификации пользователей*

После удачной аутентификации происходит переадресация на главную страницу, на которой можно увидеть расписание в виде календаря, а так меню навигации, позволяющее перейти в другие разделы веб-приложения.

* 1. Средства контроля правильности выполнения

Поскольку доступ к веб-приложению есть только посредствам локальной сети, то снижается общий уровень уязвимости системы и хранящихся данных.

* 1. Ограничения области применения программы

Программа может быть использована только в медицинской клинике «Стоматология Доктора Вознесенского» (Санкт-Петербург, Фрунзенский район).

1. Описание логической структуры
   1. Алгоритм программы

Функциональная модель представлена в BPMN-диаграмме, на рисунке Б.1, включающей основной бизнес-процесс, который описывает процесс записи пациентов на приём, который включает в себя проведение лечения и оплаты. Описание поведения пользователей представлено на рисунке Б.2 в формате «Пользовательская история» (User story).

* 1. Используемые методы при разработке программы

В процессе разработки системы были использованы:

* git – для контроля версий файлов, версии 2.29.2.windows.3;
* https – протокол для обмена данными между клиентом и сервером;
* json – универсальный формат данных;
* yarn – пакетный менеджер, версии 1.22.10;
* sass – мета-язык для каскадных таблиц стилей (css), версии 1.32.5;
* html – стандартизированный язык разметки веб-документов, версии 5.3;
* ESLint – линтер для JavaScript, статический анализатор кода, версии 0.2.1.
  1. Структура программы

В таблице 1 приведена структура клиентской части веб-приложения до продакшн сборки.

Таблица 1 – Структура клиентской части приложения

|  |  |  |
| --- | --- | --- |
| Файл | Функции | Назначение |
| 1 | 3 | 2 |
| Auth.jsx | Auth | Функция, возвращающая визуальный компонент окна авторизации |
| Auth.setField | Функция, для логирования изменений поля и записи данных в структуру |
| Auth.setAuthorization | Функция, устанавливающая флаг авторизации |
| Auth.module.sass | - | Файл стилей |
| SideBar.jsx | SideBar | Возвращает визуальный компонент бокового меню |
| SideBar.module.sass | - | Файл стилей |

Продолжение таблицы 1.

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| Client.jsx | Client | Функция, возвращающая визуальный интерактивный компонент с информацией о пациенте |
| Client.archiveClient | Функция архивации клиентов |
| Client.dearchiveClient | Функция разархивации клиентов |
| ClientList.jsx | ClientList.componentDidMount | Функция, запускающаяся в момент загрузки компонента Client, которая вызывает функцию getClietns |
| ClientList.getClients | Функция, получающая данные пользователей из Базы Данных по средствам API |
| ClietnList.render | Функция, возвращающая разметку списка клиентов и передающая данные в класс Client |
| ClientList.module.sass | - | Файл стилей |
| ClientAddModalWrapper.jsx | ClientAddModalWrapper | Функция-обёртка для модальных окон, добавляющая возможность обновления и добавления клиентов |
| ClientEditModal.jsx | ClientEditModal | Функция, которая возвращает модальное окно с формой изменения существующих пациентов |
| ClientEditModal.setField | Функция, для логирования изменений поля и записи данных в структуру |
| ClientEditModal.handleSubmit | Функция, собирающая данные и проверяющая на наличие ошибок |
| ClientEditModal.findFromErrors | Функция поиска ошибок во введённых данных |

Продолжение таблицы 1.

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| ClientAddModalClass.jsx | ClientAddModal | Функция, которая возвращает модальное окно с формой добавление пациентов |
| ClientAddModal.setField | Функция для логирования изменений поля и записи данных в структуру |
| ClientAddModal.handleSubmit | Функция, собирающая данные и проверяющая на наличие ошибок |
| ClientAddModal.findFromErrors | Функция поиска ошибок во введённых данных |
| Employee.module.sass | - | Файл стилей |
| EmployeesList.jsx | Employee | Функция, возвращающая интерактивный визуальный компонент со списком сотрудников |
| EmployeesList.module.sass | - | Файл стилей |
| EmployeeAddModal.jsx | EmployeeAddModal | Функция, которая возвращает модальное окно с формой добавление сотрудников |
| EmployeeAddModal.setField | Функция, для логирования изменений поля и записи данных в структуру |
| ClientAddModal.handleSubmit | Функция, собирающая данные и проверяющая на наличие ошибок |
| ClientAddModal.findFromErrors | Функция поиска ошибок во введённых данных |
| EmployeeAddModalWrapper.jsx | EmployeeAddModalWrapper | Функция-обёртка для модальных окон, добавляющая возможность обновления и добавления сотрудников |

Продолжение таблицы 1.

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| EmployeeEditModal.jsx | ClientEditModal | Функция, которая возвращает модальное окно с формой изменения существующих сотрудников |
| ClientEditModal.setField | Функция, для логирования изменений поля и записи данных в структуру |
| ClientEditModal.handleSubmit | Функция, собирающая данные и проверяющая на наличие ошибок |
| ClientEditModal.findFromErrors | Функция поиска ошибок во введённых данных |
| Employee.jsx | Employee | Функция, возвращающая интерактивный визуальный компонент интерфейса, которые выводит таблицу всех сотрудников |
| SideBar.module.sass | - | Файл стилей |
| SideBar.jsx | SideBar | Функция, возвращающая разметку меню боковой навигации |
| Services.module.sass | - | Файл стилей |
| Stats.jsx |  | Функция, возвращающая статистические данные |
| Visits.jsx | Visits | Функция, возвращающая список визитов |
| Visits.module.sass | - | Файл стилей |
| WorkingSpace.jsx | WorkingSpace | Функция-маршрутизатор, определяет, что должно быть выведено, в зависимости от адреса |
| WorkingSpace.module.sass | - | Файл стилей |
| Services.jsx | Service | Функция, возвращающая список всех предоставляемых услуг |
| Stats.module.sass | - | Файл стилей |
| Api.js | axios.create | Обёртка функции из библиотеки axios |
| App.js | App | Функция-маршрутизатор, которая не позволяет неавторизованным пользователям получать доступ к системе |

Продолжение таблицы 1.

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| App.sass | - | Файл стилей |
| index.css | - | Файл стилей |
| index.js | ReactDom.render | Функция, запускающая всё веб-приложение |
| package.json |  | Файл, в котором перечислены зависимости |
| yarn.lock |  | Файл, в котором перечислены зафиксированные версии зависимостей |
| .gitignore |  | Файл, в котором перечислены игнорируемые файлы |
| public/index.html | - | Файл разметки стартового веб-документа |
| public/index.css | - | Файл стилей |

В таблице 2 представлена структура программы после продакшн сборки.

Таблица 2 – Структура клиентской части веб-приложения

|  |  |  |
| --- | --- | --- |
| Файл | Функции | Назначение |
| 1 | 3 | 2 |
| index.html | - | Файл разметки стартового веб-документа |
| index.css | - | Файл стилей |
| static/css/\*.\*.chunk.css | - | Минимизированные файлы стилей |
| static/js/\*.\*.chunk.js | Их слишком много, чтобы это было возможным описать | Минимизированные файлы js, содержащий основной функционал |
| static/js/\*.\*.chunk.js.LICENSE | - | Файлы лицензий |
| static/js/main.\*.chunk.js | Их слишком много, чтобы это было возможным описать | Минимизированный файл webpack-скрипта, содержащий фундаментальные настройки сборки окружения |
| static/js/runtime-main.\*.js | Их слишком много, чтобы это было возможным описать | Минимизированный файл js |

* 1. Форматы обмена данными

Основной формат обмена данными – обращение к Rest API в виде JSON-запросов.

1. Сведения о входных и выходных данных
   1. Входные данные

Входными данными приложения являются следующие данные.

О пациентах:

* паспортные данные;
* ФИО;
* дата рождения;
* адрес регистрации;
* адрес проживания;
* профессия;
* телефон;
* пол;
* место рождения;
* статус архивации;
* откуда узнал;
* адрес электронной почты;
* комментарий;
* баланс;
* постоянная скидка.

О визитах:

* пациент;
* дата;
* статус первичности;
* причина;
* посещённые врачи;
* комментарий;
* статус.

О сотрудниках:

* ФИО;
* Дата рождения;
* должность;
* контактные данные.
* логин;
* пароль;
* права доступа.

Об услугах:

* цена;
* наименование;
* группа.

О предоставленных услугах:

* услуга;
* фактическая цена;
* посещение, в которое услуга оказана.

О товарах:

* цена;
* наименование;
* группа;
* количество.

О проданных товарах:

* товар;
* клиент;
* фактическая цена;
* дата продажи;
* факт использования дисконтной карты;

О дисконтных картах:

* клиент;
* размер скидки.

О транзакциях:

* клиент;
* изменение баланса.
  1. Выходные данные

Данные пациентов выводятся в виде списка, где каждая строка содержит информацию о конкретном пациенте.

Данные о визитах выводятся в календаре с записями.

Данные сотрудников выводятся в виде списка, где каждая строка списка содержит информацию о конкретном сотруднике.

Данные об услугах выводятся в виде списка, содержащего коды услуг, названия и цены.

Данные о предоставленные услугах находятся в конкретных визитах.

Данные о товарах выводятся в виде списка, содержащего коды товаров, названия и цены.

Данные о проданных товарах выводятся в виде списка.

Данные о дисконтных картах выводятся вместе с данными пациентов.

Данные о транзакциях выводятся в виде списка.

1. Используемые технические средства

Для использования разрабатываемого веб-приложения устройство пользователя должно обладать следующими техническими требованиями, соответствующим требованиям использования основных браузеров (Chrome, Firefox, Internet Explorer, Opera, Safari и т.п.) (URL: <https://support.google.com/chrome/a/answer/7100626>, дата обращения: 18.05.21):

* ОС Windows 7 (поддержка завершится не ранее 15.01.2022), Windows 8, Windows 8.1, Windows 10 или более поздней версии или OS X El Capitan 10.11 или более поздней версии.
* процессор Intel Pentium 4 или более поздней версии с поддержкой SSE3.

1. Вызов и загрузка программы
   1. Конфигурирование и загрузка программы

Приложение не нуждается в установке.

* 1. Способ вызова и завершения работы программы

Разрабатываемая программа – веб-приложение, поэтому для начала работы требуется перейти на определённый URL, который будет назначен заказчиком на рабочем месте.

Для прекращения работы необходимо покинуть браузер или вкладку, в которой открыто веб-приложение.

* 1. Режимы работы программы

Пользователи взаимодействуют с веб-приложением в интерактивном режиме.

Заключение

В разделе «Заключение» приводятся выводы по результатам преддипломной практики в соответствии с целью и задачами практики, описывается практический результат проекта и приводятся перспективы его развития (при наличии).

В результате выполнения преддипломной практики было разработано веб-приложение для учёта пациентов в соответствии с техническим заданием и моделями разработки. Сформировано техническое описание разработанной системы.

Для достижения поставленной цели были выполнены следующее задачи:

1. Уточнение функциональных требований (при необходимости).
2. Программная реализация системы (клиентской части).
3. Проведение отладки и тестирования системы.
4. Формирование технического описания разработанной программной системы.
5. Представление реализованной программной системы.

Таким образом, все задачи решены в полном объёме. Цель практики достигнута.

В качестве перспективы дальнейшего развития приложения можно рассмотреть расширение системы для внедрения в другие частные медицинские клиники.

ПРИЛОЖЕНИЕ А  
Техническое задание

ПРИЛОЖЕНИЕ Б

Модель системы

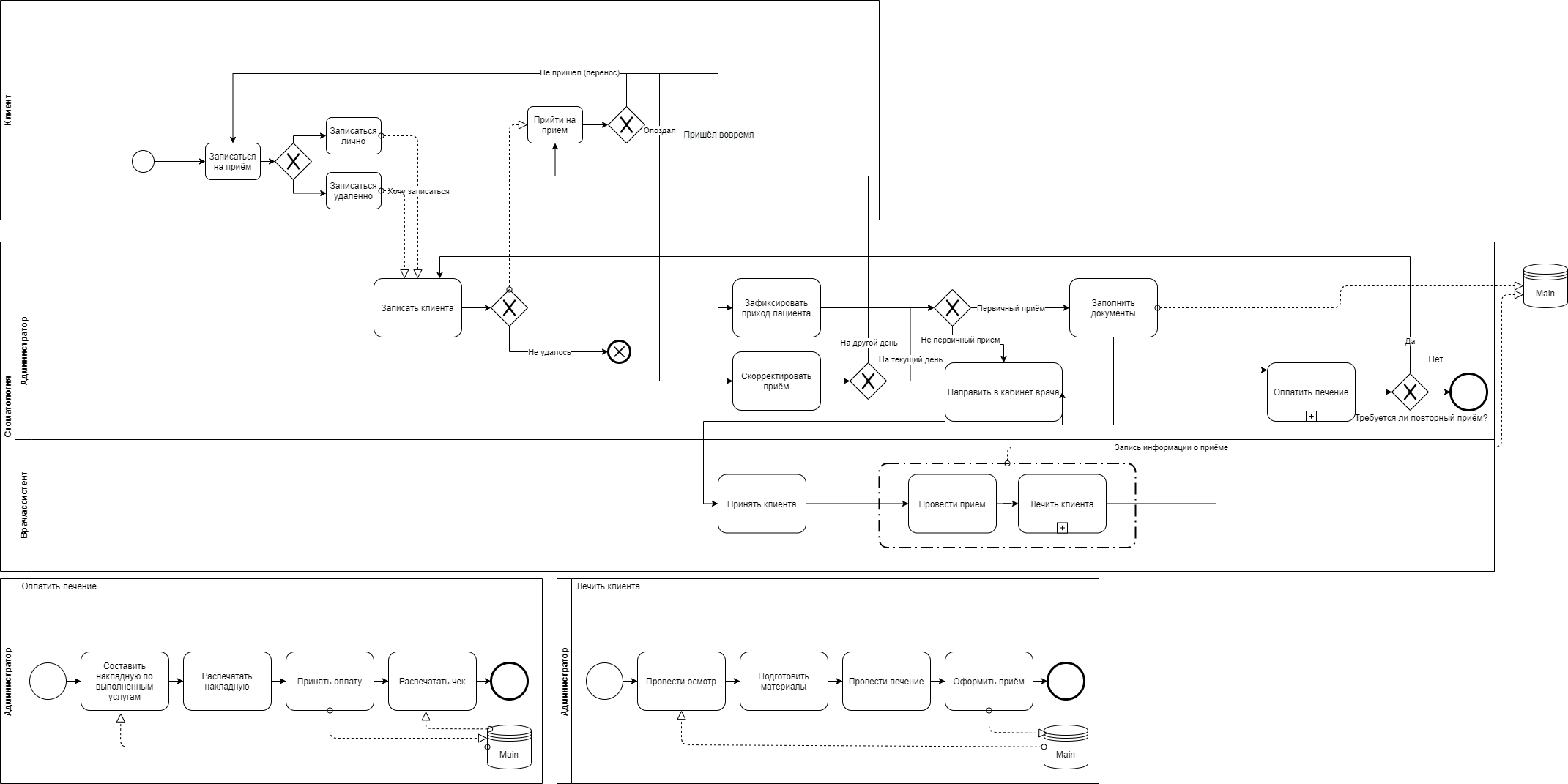


Рисунок Б.1 – Функциональная модель системы в нотации BPMN

Изображение выглядит как текст

Автоматически созданное описание

Рисунок Б.2 – Концептуальная модель: диагрмма User Story