



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИТ)
Кафедра инструментального и прикладного программного обеспечения (ИиППО)

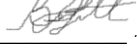
КУРСОВАЯ РАБОТА

по дисциплине: Шаблоны программных платформ языка Джава
по профилю: Разработка программных продуктов и проектирование информационных систем
направления профессиональной подготовки: Программная инженерия (09.03.04)

Тема: Приложение «Стоматологическая клиника»

Студент: Малютина Виталия Андреевна

Группа: ИКБО-13-19

Работа представлена к защите 20.05.2020(дата)  /Малютина В.А. /
(подпись и ф.и.о. студента)

Руководитель: старший преподаватель Зорина Наталья Валентиновна

Работа допущена к защите _____ (дата) _____ /Зорина Н.В. /
(подпись и ф.и.о. рук-ля)

Оценка по итогам защиты: _____

_____/_____
_____/_____

(подписи, дата, ф.и.о., должность, звание, уч. степень двух преподавателей, принявших защиту)

М. РТУ МИРЭА. 2021 г.



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИТ)

Кафедра инструментального и прикладного программного обеспечения (ИиППО)

ЗАДАНИЕ

на выполнение курсовой работы

по дисциплине: Шаблоны программных платформ языка Джава
по профилю: Разработка программных продуктов и проектирование информационных систем направления профессиональной подготовки: Программная инженерия (09.03.04)

Студент: Малютина Виталия Андреевна

Группа: ИКБО-13-19

Срок представления к защите: 20.05.2021

Руководитель: старший преподаватель Зорина Наталья Валентиновна

Тема: Приложение «Стоматологическая клиника»

Исходные данные: индивидуальное задание на разработку; документация по Spring Framework и JEE, документация по языку Java (версия не ниже 8); инструменты и технологии: JDK (не ниже 8), создание Spring MVC web-приложений, Spring ORM, Maven, gitHub, IntelliJIDEA. Нормативный документ: инструкция по организации и проведению курсового проектирования СМКО МИРЭА 7.5.1/04.И.05-18.

Перечень вопросов, подлежащих разработке, и обязательного графического материала:

1. Провести анализ предметной области и формирование основных требований к приложению. 2. Обосновать выбор средств ведения разработки. 3. Разработать приложение с использованием фреймворка Spring и выбранной технологии и инструментария. 4.Провести тестирование приложения. 5. Оформить пояснительную записку по курсовой работе 6. Провести анализ текста на антиплагиат 7. Создать презентацию по выполненной курсовой работе.

Руководителем произведён инструктаж по технике безопасности, противопожарной технике и правилам внутреннего распорядка.

Зав. кафедрой ИиППО: _____/Р. Г. Болбаков/, «_____» _____ 2021 г.

Задание на КР выдал: _____/Н.В. Зорина/, «_____» _____ 2021 г.

Задание на КР получил: _____/В.А. Малютина/, «_____» _____ 2021 г.

Малютина В.А. «Веб-приложение Стоматологическая клиника»/ **Курсовая работа** по дисциплине «Шаблоны программных платформ на языке Джава» профиля «Разработка программных продуктов и проектирование информационных систем» направления профессиональной подготовки бакалавриата 09.03.04 «Программная инженерия» (4-ый семестр) / руководитель старший преподаватель Н.В. Зорина / кафедра ИППО Института ИТ МИРЭА – с. 39, табл. 8, илл.32.

Целью работы является разработка веб-приложения стоматологическая клиника с названием «WhiteApple».

В рамках работы осуществлен краткий анализ аналогов web-приложений. Назначение разработанного веб-приложения – предоставить пользователю возможность просмотреть информацию об услугах, записаться на приём и оставить отзыв.

Malyutina, V. A., "the Web application Dental clinic"/ Course work in discipline "Templates software platforms in the language Java" profile "Development of software products and information systems design" courses undergraduate 09.03.04 "Software engineering" (4th semester) / head of senior lecturer N. V. Zorina / Department of history of the Institute it MIREA – p. 39, tab. 8, ill. 32.

The aim of this work is to develop a web application dental clinic called "WhiteApple".

As part of the work, a brief analysis of analogs of web applications is carried out. The purpose of the developed web application is to provide the user with the opportunity to view information about services, make an appointment and leave a review.

Содержание

СПИСОК ИСПОЛЬЗОВАННЫХ СОКРАЩЕНИЙ	3
ВВЕДЕНИЕ	4
1. СБОР И АНАЛИЗ ТРЕБОВАНИЙ К ВЕБ-ПРИЛОЖЕНИЮ	5
1.1 Обозначение и наименование веб-приложения	5
1.2 Вывод к разделу 1	9
2. ВЫБОР СРЕДСТВ ВЕДЕНИЯ РАЗРАБОТКИ.....	10
2.1 Прикладное программное обеспечение, необходимое для разработки и функционирования интернет-ресурса.....	10
2.2 Функциональное назначение.....	12
2.3 Выбор средств для разработки веб-приложения	13
2.4 Выбор базы данных.	13
2.5 Выбор инструмента для работы с базой данных.....	14
2.6 Spring Security.....	15
2.7 Хранение файлов проекта	16
2.8 Выбор системы сборки.....	18
2.9 Thymeleaf	20
2.10 Вывод к разделу 2.....	21
3. РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ	22
3.1 Создание сущностей	22
3.2 Репозитории.....	25
3.3 Сервисы.....	25
3.4 Контроллеры.....	26
3.5 Конфигурации	28
3.5 Разработка фронтенда.....	28
3.5 Страницы и функции	29
3.6 Вывод к разделу 3	33
4. ТЕСТИРОВАНИЕ ВЕБ-ПРИЛОЖЕНИЯ.....	33
4.1 Вывод к разделу 4	37
5. АНТИПЛАГИАТ	37

6. ЗАКЛЮЧЕНИЕ	38
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ	39

СПИСОК ИСПОЛЬЗОВАННЫХ СОКРАЩЕНИЙ

JDK 11 или Java Development Kit (JDK)— бесплатный пакет для разработчика приложений и апплетов, написанных на языке Java. В него входит компилятор Javac, классические библиотеки классов Java, исполнительная JRE система, документы, некоторые утилиты и образцы кода.

Spring MVC – это фреймворк, позволяющий создавать web-приложения с разделением логики согласно паттерну Model—View—Controller. Spring обеспечивает готовые компоненты, использование которых существенно облегчает и ускоряет разработку веб-приложений.

Restful Web Service — это легкий, обслуживаемый и масштабируемый сервис, построенный на архитектуре REST. Веб-служба Restful, предоставляя вызывающему клиенту API из вашего приложения безопасным, единообразным способом без сохранения состояния.

ORM (Object-Relational Mapping или объектно-реляционное отображение) — это технология для отображения объектов в структуры реляционных баз данных, ну т.е. чтобы представить наш джава-объект в виде строки таблицы.

Apache Maven— Фреймворк для автоматизации сборки проектов на основе описания их структуры в файлах на языке POM, являющемся подмножеством XML.

Gradle — система автоматической сборки, построенная на принципах Apache Ant и Apache Maven, но предоставляющая DSL на языках Groovy и Kotlin вместо традиционной XML-образной формы представления конфигурации проекта.

YouTrack — коммерческая система отслеживания ошибок, программное обеспечение для управления проектами, разработанное компанией JetBrains.

GitHub это система управления проектами и версиями кода, а также платформа социальных сетей, созданная для разработчиков.

IntelliJ IDEA — интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python.

ВВЕДЕНИЕ

Сегодня очень сложно найти человека, который не знаком с Интернетом. Всяческие информационные технологии, к которым относится и Всемирная Паутина, прочно вошли в обыденную жизнь. В наше время прогресс достиг таких высот, что, благодаря высокой скорости Интернета, появилась возможность качать музыку и видео в отличном качестве, общаться в сети и т. д. Сейчас можно даже учиться при помощи Интернета, не выходя из дома. Ежедневно ряды пользователей сети пополняются тысячами новых пользователей. Некоторые люди настолько привязаны к Всемирной Паутине, что современная медицина признала их больными – Интернет-зависимыми. Эти люди в полном смысле этого слова живут в сети. Ведь очень заманчивой выглядит перспектива совершать покупки, искать любую интересующую информацию, общаться с интересными людьми – и всё это сидя дома за компьютером, просто щёлкая мышкой! И интернет невозможно представить без веб-приложений.

Замечательный прогресс в технологиях создания веб-сайтов, веб-приложений и безграничные возможности человеческого воображения определили огромное разнообразие веб-ресурсов, которые мы видим сегодня в Интернете, когда задаем вопросы yandex, google и другим поисковым системам. Исходя из вышеуказанных фактов, можно говорить об актуальности разработки интернет-ресурсов из различных областей.

1. СБОР И АНАЛИЗ ТРЕБОВАНИЙ К ВЕБ-ПРИЛОЖЕНИЮ

Цель работы: разработка веб-приложения «Стоматологическая клиника».

Для достижения цели поставлены задачи, для которых необходимо:

1. Провести анализ предметной области разрабатываемого веб-приложения.
2. Обосновать выбор технологий разработки веб-приложения.
3. Создать веб-приложения с использованием технологий HTML5, CSS3, JavaScript, Java, Spring Framework и системы сборки проектов Gradle.
4. Реализовать систему регистрации и авторизации пользователя.
5. Реализовать систему записи на прием.
6. Реализовать систему отзывов.

1.1 Обозначение и наименование веб-приложения

В ходе выполнения курсовой работы было необходимо разработать веб-приложение, на тему «Стоматологическая клиника». Разработанное приложение в выполненной курсовой работе имеет следующее название: «WhiteApple». Английский язык является самым распространенным языком в мире, поэтому в ходе анализа информационной среды, было принято решение об использовании этого языка в названии разрабатываемого веб-ресурса.

Стоматология - одна из областей медицинской науки, занимающаяся изучением этиологии, патогенеза основных стоматологических заболеваний (кариес зубов, заболевания пародонта и др.), разработкой методов их профилактики, диагностики и лечения. Работа стоматологов направлена на постоянное усовершенствование методов профилактики, ранней диагностики и современных методов лечения стоматологических заболеваний.

Стоматология – наука, которая изучает не только заболевания зубов, но и заболевания полости рта, челюстей и всей лицевой области в целом. Терапевтическая стоматология – часть этой науки, которая занимается

изучением болезней зубов и околозубных тканей, в том числе и тканей полости рта. Это быстро развивающаяся дисциплина, в рамках которой сейчас создаются наиболее передовые методы диагностики, профилактики и лечения данной группы заболеваний. В терапевтической стоматологии различают кариесологию (то есть науку о повреждениях зубной эмали), эндодонтию (науку о болезнях пульпы зуба), пародонтологию (науку о заболеваниях околозубных тканей), болезни слизистой оболочки рта.

Хирургическая стоматология - это клиническая наука о предупреждении, диагностике болезней органов полости рта и челюстно-лицевой области и методах их лечения. Так, к ее помощи обращаются при различного рода воспалениях, механических повреждениях, опухолевых процессах, врожденных и приобретенных дефектах и деформациях.

Современная стоматологическая клиника - это медицинское учреждение, где оказывается диагностическая, терапевтическая, хирургическая, ортодонтическая, ортопедическая и зуботехническая помощь. Стоматологическая клиника включает в свой состав, как лечебные и диагностические кабинеты, так и обязательный набор вспомогательных и служебных помещений: вестибюльная группа, стерилизационная, комната персонала, техническое помещение (компрессорная, северная и др.), санитарная комната, помещение временного хранения отходов, гардеробная, сан.узлы и пр.

Веб-приложение — это любой сайт с элементами интерактива. Это значит, что посетитель может взаимодействовать с материалом, функциями: нажимать кнопки, заполнять формы, запрашивать прайс, совершать покупки. Практически любой интернет-ресурс входит в их число. Это поисковые системы, видео сервисы типа youtube, соцсети, любые веб-сайты с функциями аутентификации пользователя, покупки, заказа, бронирования. Технически это интернет-приложение с архитектурой «клиент-сервер». Клиентом служит браузер, сервером — веб-сервер. Связь происходит посредством сети. Разработанное веб-приложение представляет собой систему ведения

информации и взаимодействия с приложением. Оно позволяет пользователю просматривать информацию об услугах стоматологической клиники и позволяет записаться на приём онлайн.

При разработке проекта для обеспечения его максимальной эффективности очень важно учитывать все особенности информационной среды, в которой будет происходить эксплуатация.

Исходя из заявленных свойств веб-приложения, был проведен поиск и анализ информации. В процессе осуществления поиска, были определены основные направления возможностей: просмотр услуг, запись на прием, личный кабинет и система отзывов.

Также в ходе анализа предметной области были выделены следующие проблемы, с которыми может столкнуться человек, который хочет воспользоваться услугами стоматологической клиники:

- Какую стоматологическую клинику выбрать.
- Какую выбрать услугу.
- Как долго будет проходить лечение.

Наиболее распространённым решением перечисленных проблем является обращение в стоматологии, которые предоставляют консультацию и назначают лечение. Существует множество стоматологий, которые предоставляют услуги разного качества по различной цене.

Исходя из этого можно сделать вывод, что клиент скорее всего столкнётся с проблемой выбора. Был выполнен поиск web-ресурсов, которые бы предоставляли решение обозначенной проблемы. Были обнаружены следующие источники информации:

Центр Современной Стоматологии AppleStom – стоматологическая клиника, которая предлагает клиенту разнообразные услуги. На сайте клиники было найдено много полезной информации об услугах, которую можно было использовать при выполнении курсовой работы, а также отзывы о клинике (рис. 1.1, рис. 1.2).

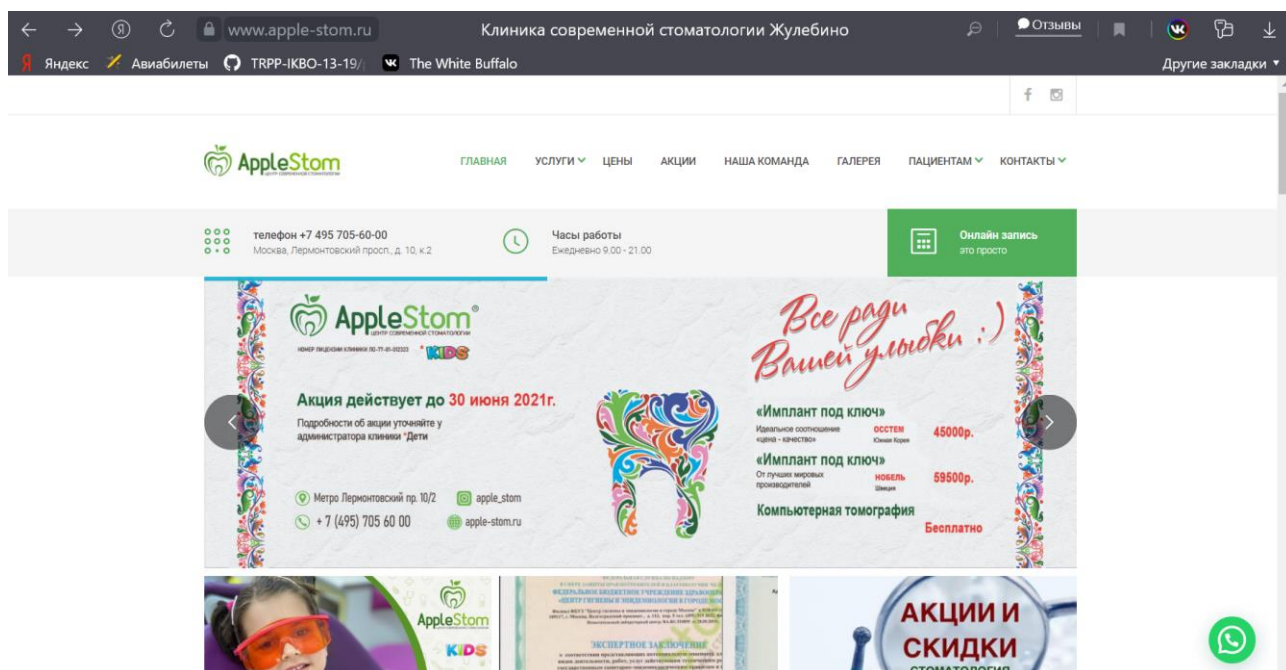


Рис. 1.1 – Сайт клиники AppleStom

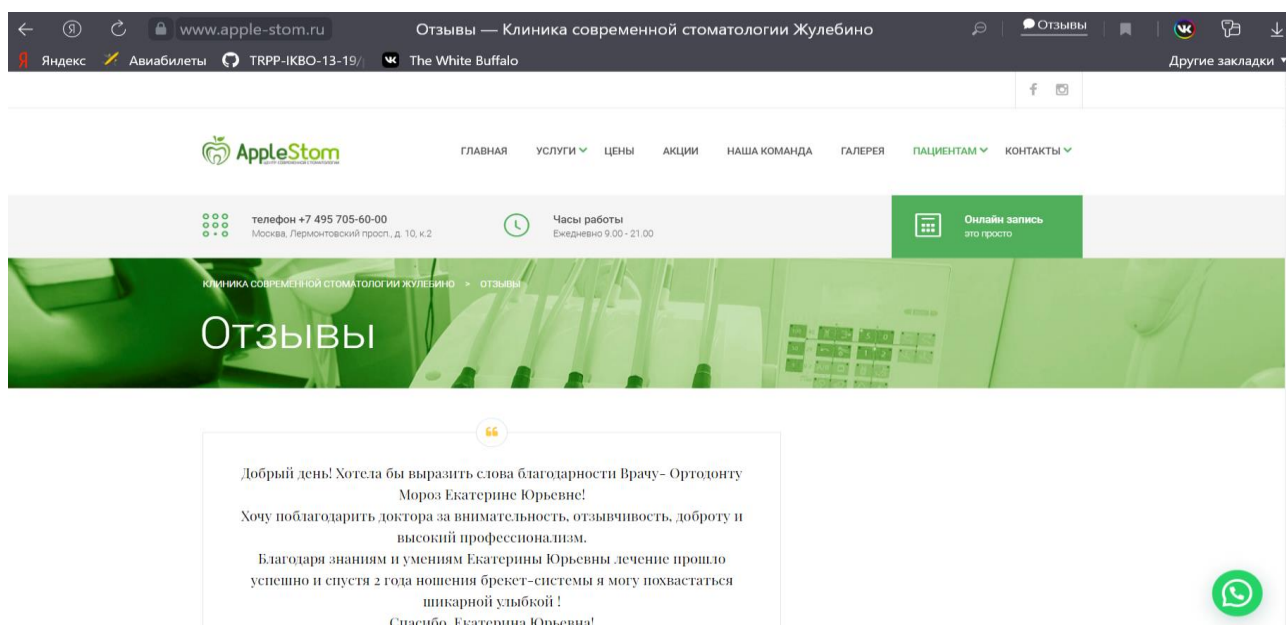


Рис. 1.2 - Отзывы

Zoon.ru – веб-ресурс, который предоставляет информацию о стоматологиях, фильтрует адреса, исходя из геопозиции пользователя, и показывает отзывы о клиниках (рис. 1.3).

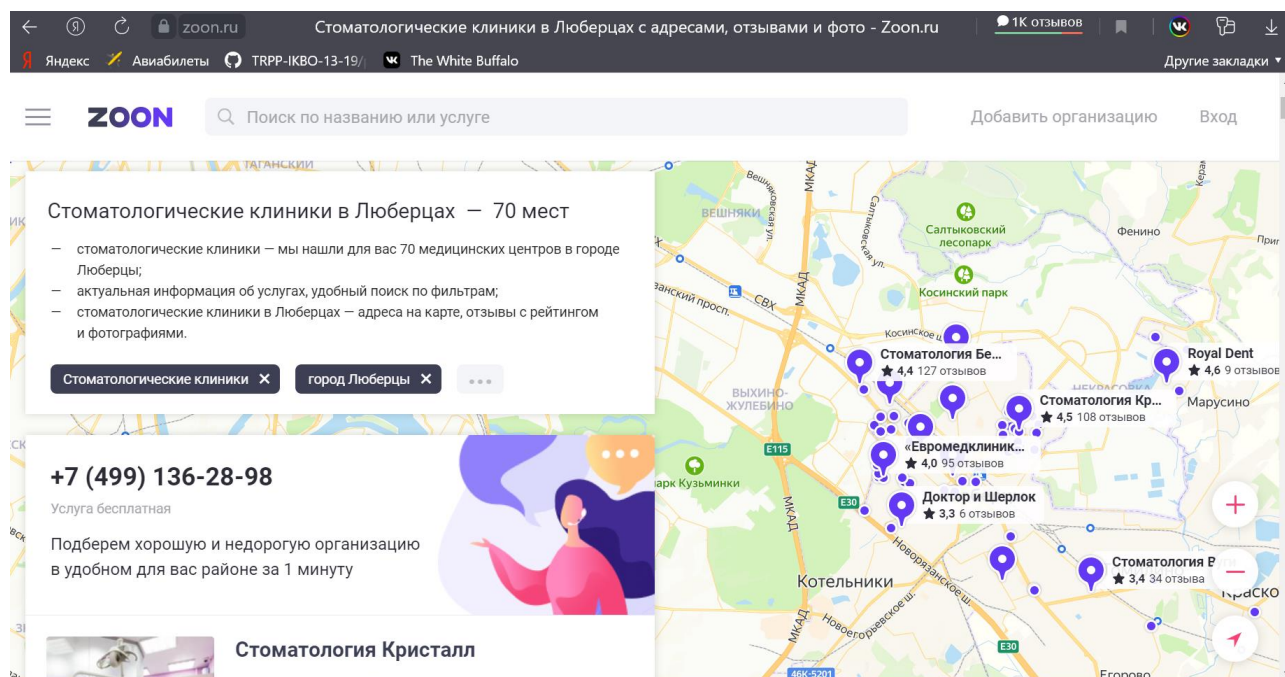


Рис. 1.3 – Zoon.ru

1.2 Вывод к разделу 1

В результате анализа полученной информации было принято решение разработать web-приложение со следующим функционалом:

- Поддержка системы регистрации, авторизации и личного кабинета.
- Возможность просмотреть информацию об услугах.
- Возможность оставить отзыв.
- Просмотр отзывов на сайте.
- Функция записи на прием.
- Функция просмотра информации о себе и истории записей в личном кабинете.

2. ВЫБОР СРЕДСТВ ВЕДЕНИЯ РАЗРАБОТКИ

2.1 Прикладное программное обеспечение, необходимое для разработки и функционирования интернет-ресурса

Следующим шагом в курсовом проектировании было выбрать необходимое ПО для создания веб-приложения.

Клиентская часть:

- операционная система с поддержкой программного обеспечения доступа в Интернет.
- браузер.

Программные средства внешних систем (онлайновые сервисы):

- сервисы Яндекс.Поиск и поиск Google.

Серверная часть:

Для разработки веб-приложения была использована IntelliJ IDEA — интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains. Среди прочих возможностей, среда хорошо совместима со многими популярными свободными инструментами разработчиков, такими как CVS, Subversion, Apache Ant, Maven и JUnit.

Базовые возможности

Возможности IntelliJ IDEA заявлены на официальном сайте.

- Умное автодополнение.
- Автодополнение цепочек вызовов.
- Анализ потоков данных.
- Языковые вставки.

- Рефакторинги для разных языков.
- Поиск дубликатов.
- Инспекции и быстрые исправления.
- Сочетания клавиш для любого действия.
- Эргономичный интерфейс.
- Встроенный отладчик.
- Интеграция с системами контроля версий.
- Инструменты сборки.
- Инструмент запуски тестов и анализ покрытия кода.
- Декомпилятор.
- Инструменты для работы с базами данных.

Для тестирования функциональности веб-приложения, был использован Яндекс.Браузер – браузер, созданный компанией «Яндекс» на основе движка Blink, используемого в открытом браузере Chromium.

Языки и технологии, с помощью которых реализовано веб-приложение (более подробно после п. 2.3)

- HTML (от англ. HyperText Markup Language — «язык гипертекстовой разметки») — стандартизированный язык разметки веб-страниц.
- CSS («Cascading Style Sheets» или «Каскадные таблицы стилей») — это язык, предназначенный для преобразования внешнего вида страницы на сайте в удобную для восприятия пользователем форму.

- JavaScript – мультипарадигменный язык программирования, который обычно применяется в качестве встраиваемого инструмента для программного доступа к различным объектам приложений.
- Сетка Bootstrap версии v4.0.0 для адаптивности веб-страниц.
- Java — multifunctional object-oriented language with strict typing. With multifunctionality it's quite simple: on Java you can develop desktop applications, applications under Android, engage in web-development. (11 version)
- ORM (Object-Relational Mapping или объектно-реляционное отображение) — это технология для отображения объектов в структуры реляционных баз данных.
- Spring Framework, или просто Spring — один из самых популярных фреймворков для создания веб-приложений на Java.
- Spring MVC – это фреймворк, позволяющий создавать web-приложения с разделением логики согласно паттерну Model—View—Controller.

2.2 Функциональное назначение

В век информационных технологий люди в большинстве случаев черпают информацию, делают покупки, записываются на услуги в интернете, а интернет невозможно представить без веб-сайтов. Исходя из этого, было определено назначение разрабатываемого веб-приложения в курсовой работе. «AppleWhite» — это веб-приложение стоматологической клиники, содержащее информацию об услугах, и даёт потенциальному клиенту записаться на прием как с помощью личного кабинета, так и без авторизации.

2.3 Выбор средств для разработки веб-приложения

Исходя из условий поставленной задачи разработка веб-приложения будет осуществлялась на базе фреймворка Spring для типизированного объектно-ориентированного языка программирования общего назначения под названием Java. Разработка выполнялась на языке Java версии 11.

2.4 Выбор базы данных.

В ходе анализа предметной области были выделены следующие задачи:

- Реализация поддержки личного кабинета.
- Реализация функций регистрации и авторизации.
- Возможность оставить отзыв о клинике.
- Реализация системы записи на прием.

Для реализации вышеперечисленных целей необходимо хранить большой объем данных, наиболее эффективный способ достичь результата – это использовать базу данных SQL типа.

Для выполнения вышеперечисленных требований была выбрана база данных SQL типа под название PostgreSQL.

PostgreSQL — свободная объектно-реляционная система управления базами данных. Существует в реализациях для множества UNIX-подобных платформ, включая AIX, различные BSD-системы, HP-UX, IRIX, Linux, macOS, Solaris/OpenSolaris, Tru64, QNX, а также для Microsoft Windows.

Преимущества PostgreSQL:

- поддержка БД неограниченного размера;
- мощные и надёжные механизмы транзакций и репликации;

- расширяемая система встроенных языков программирования и поддержка загрузки С-совместимых модулей;
- наследование;
- легкая расширяемость.

2.5 Выбор инструмента для работы с базой данных

Фреймворк spring предоставляет дополнительный модуль для удобного взаимодействия с базами данных. В ходе анализа был выбран Spring ORM (рис. 2.1).

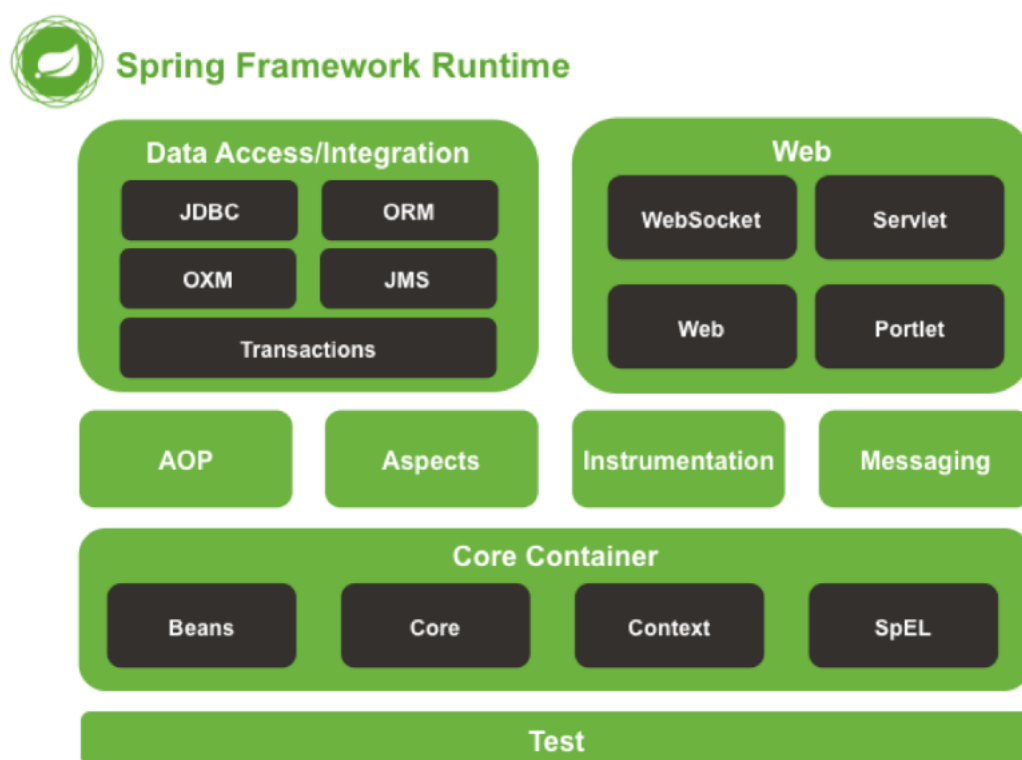


Рис. 2.1 – Spring Framework Runtime

Spring ORM — это зонтичный модуль, который охватывает многие технологии, а именно JPA, JDO, Hibernate. Для каждой из этих технологий Spring предоставляет классы интеграции, так что каждая технология может использоваться в соответствии с принципами конфигурации Spring и плавно интегрируется с управлением транзакциями Spring.

Достоинства использования ORM:

- Возможность работать с сущностями, не тратя время на составление конкретных sql запросов, а значит иметь возможность сосредоточиться на бизнес-логике.
- Возможность легко сохранять, обновлять и получать сущности из базы данных.

Недостатки использования ORM:

- Главным недостатком системы являются высокие затраты по памяти и производительности.

Из вышеперечисленных фактов технология Spring ORM является наиболее перспективной для использования в данном проекте, исходя из этого было принято решение использовать именно Spring ORM.

2.6 Spring Security

Для реализации следующей задачи:

- Поддержка системы регистрации и авторизации.

Был выбран модуль Spring Security (рис. 2.2)

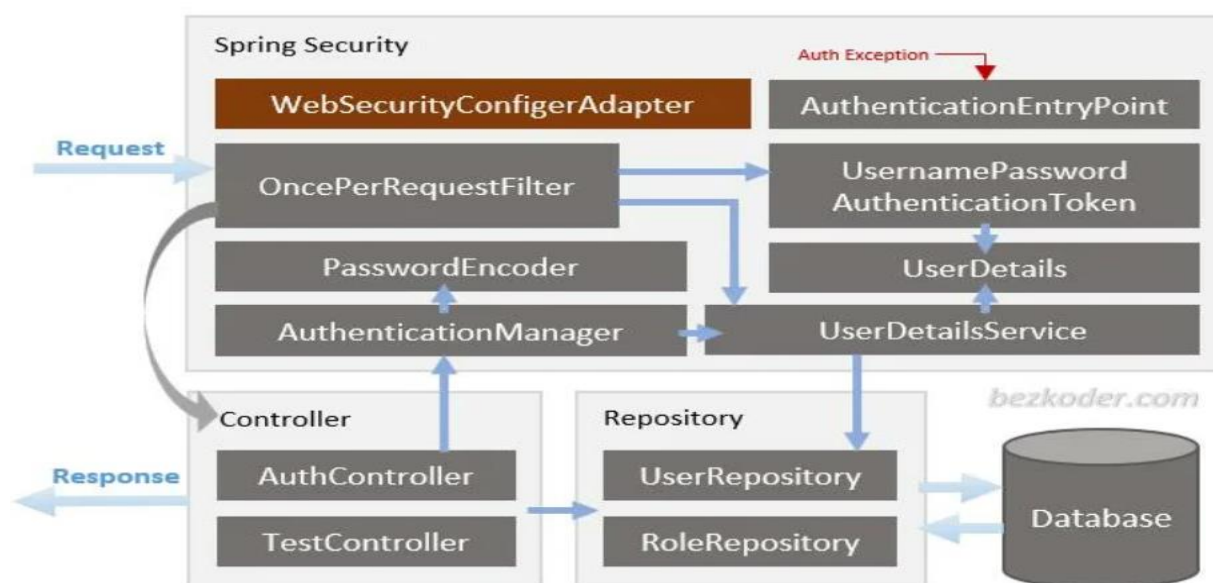


Рис. 2.2 – Spring Security

Spring Security — это Java/Java EE фреймворк, предоставляющий механизмы построения систем аутентификации и авторизации, а также другие возможности обеспечения безопасности для промышленных приложений, созданных с помощью Spring Framework. Для Spring Security 3.0 требуется среда исполнения Java 5.0 или выше. Так как Spring Security стремится работать автономно, то нет никакой необходимости размещать специальные файлы-конфигурации в Java Runtime Environment. В частности, нет необходимости специально настраивать файл политики Java Authentication and Authorization Service (JAAS) или помещать Spring Security в общий CLASSPATH.

Аналогичным образом, если вы используете EJB контейнер или контейнер сервлета, то нет необходимости создавать где-либо специальные конфигурационные файлы или включать Spring Security в загрузчик классов сервера. Все необходимые файлы будут содержаться в вашем приложении.

Такая конструкция обеспечивает максимальную гибкость во время развертывания, так как вы можете просто скопировать целевой артефакт (JAR, WAR или EAR) из одной системы в другую, и он будет работать.

2.7 Хранение файлов проекта

Для начала разработки проекта было необходимо решить проблему с хранением его файлов. Было необходимо обеспечить сохранность не только текущих файлов разрабатываемого продукта, но и возможность сохранять различные его версии. А также обеспечить возможности доступа к проекту с различных устройств, расположенных на значительном расстоянии друг от друга. Реализация всех заявленных требований с наибольшей эффективностью, была возможна с применением системы контроля версий git. Для удаленного хранения git-репозитория был использован сервис Github (рис. 2.3).

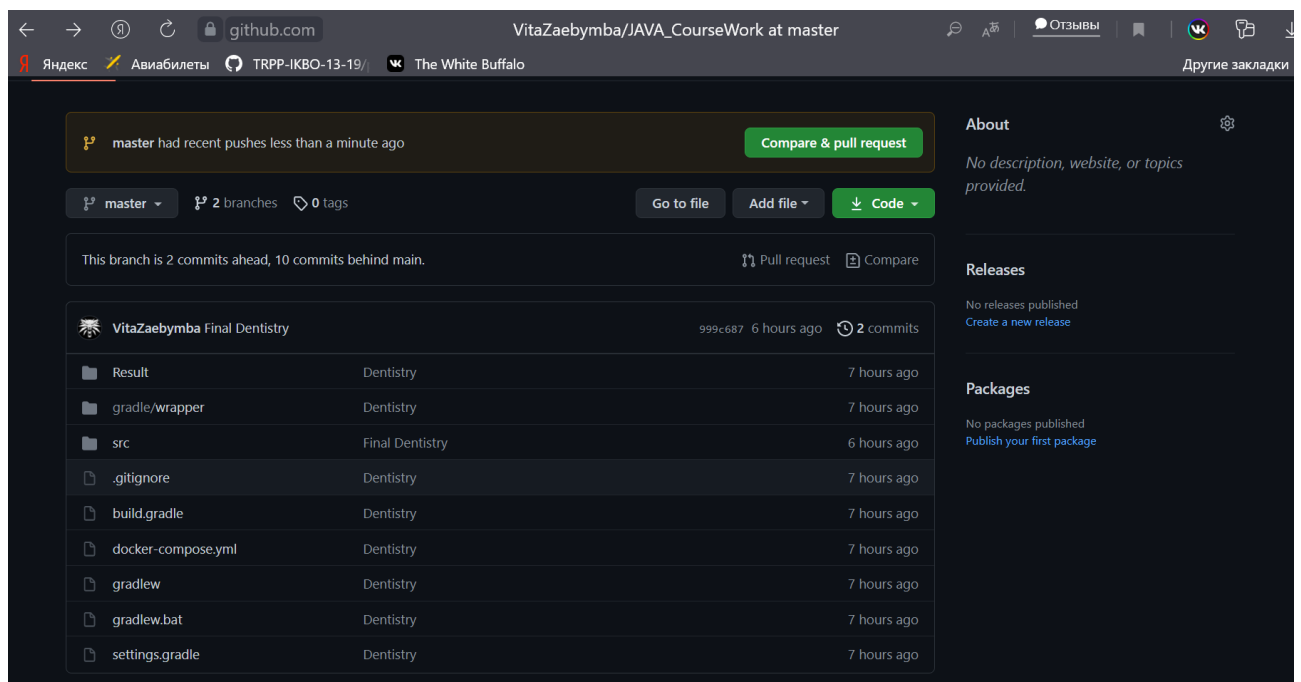


Рис. 2.3 – Git-репозиторий с проектом на Github

GitHub — крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки. Веб-сервис основан на системе контроля версий Git и разработан на Ruby on Rails и Erlang компанией GitHub.

Плюсы использования системы git:

- Быстродействие
- Простота создания нового репозитория
- Весь репозиторий хранится в одной папки
- Удобство взаимодействия с сервисом github

Минусы использования системы git:

- Невозможность извлечь часть целевого репозитория
- Нет отслеживания пустых папок
- При использовании Windows необходимо выполнять дополнительную настройку
- Отсутствие удобного визуального интерфейса управления системой контроля версий

2.8 Выбор системы сборки

Apache Maven — фреймворк для автоматизации сборки проектов на основе описания их структуры в файлах на языке POM, являющемся подмножеством XML. Проект Maven издаётся сообществом Apache Software Foundation, где формально является частью Jakarta Project (рис. 2.4).

Плюсы:

- Независимость от ОС
- Возможность сборки из командной строки
- Хорошая интеграция со средами разработки
- Декларативное описание
- Модульная, расширяемая за счет плагинов архитектура

Минусы:

- Громоздкость описания сборки
- Необходимость доступа в интернет

Maven project

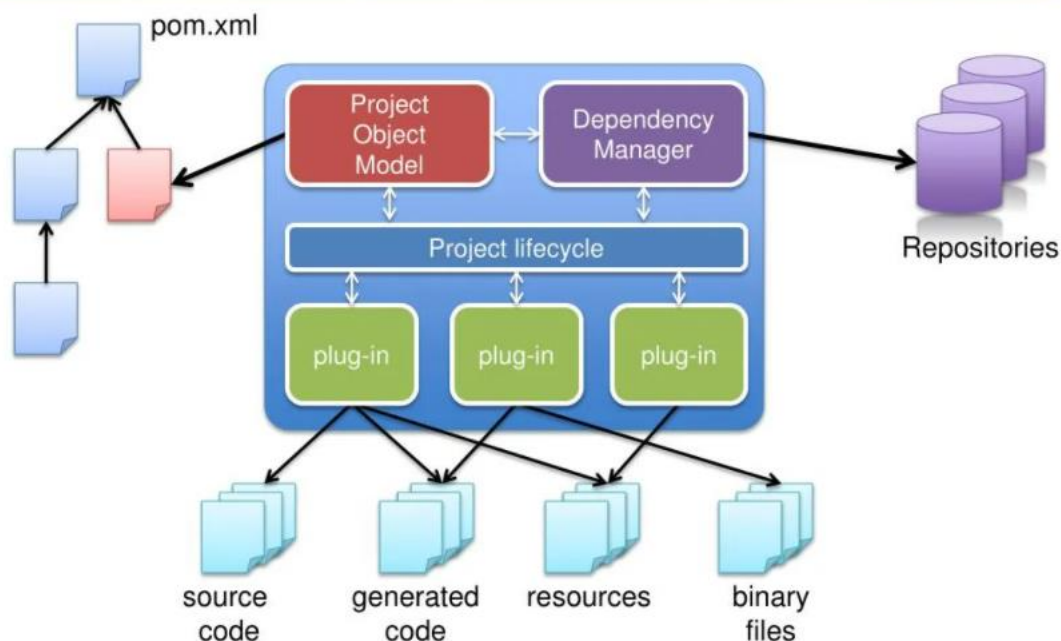


Рис. 2.4 – Maven Project

Gradle — система автоматической сборки, построенная на принципах Apache Ant и Apache Maven, но предоставляющая DSL на языках Groovy и Kotlin вместо традиционной XML-образной формы представления конфигурации проекта (рис. 2.5).

Плюсы:

- Гибкость настройки
- Краткость описания процесса сборки

Минусы:

- Сложность освоения

В качестве системы сборки разрабатываемого проекта была выбрана система сборки Gradle, определяющим фактором стало то, что уже имелся опыт работы с Gradle.

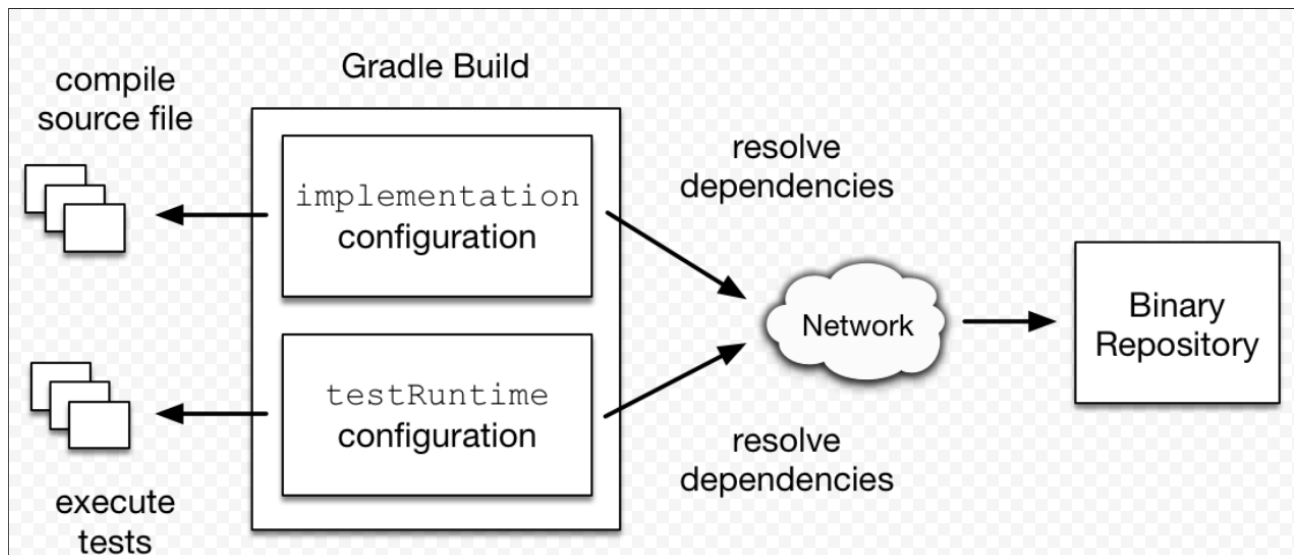


Рис. 2.5 – Gradle

2.9 Thymeleaf

Для реализации поставленных задач было необходимо создавать динамические html-страницы, для достижения указанной цели была выбрана технология Thymeleaf (рис. 2.6).

Thymeleaf — это шаблонизатор Java XML / XHTML / HTML5, который может работать как в веб-среде, так и вне ее. Он лучше подходит для обслуживания XHTML / HTML5 на уровне представления веб-приложений на основе MVC, но он может обрабатывать любой файл XML даже в автономном режиме.

Основаниями для выбора послужила простота внедрения thymeleaf в html-код страницы.

3. РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ

Разработка веб-приложения производится в IntelliJ IDEA - интегрированной среде разработки Java приложений от компании JetBrains. Ее позиционируют как самую умную и удобную среду разработки для Java с поддержкой всех последних технологий и фреймворков.

3.1 Создание сущностей

Были созданы следующие сущности:

- Сущность User, которой будет олицетворять пользователя web-приложения. (Таблица 1, рис. 3.1)

Таблица 1 – Сущность User

Имя	User
Атрибуты	id password confirm username mail
Методы	Стандартный набор методов для установки и получения значений, а также конструктор

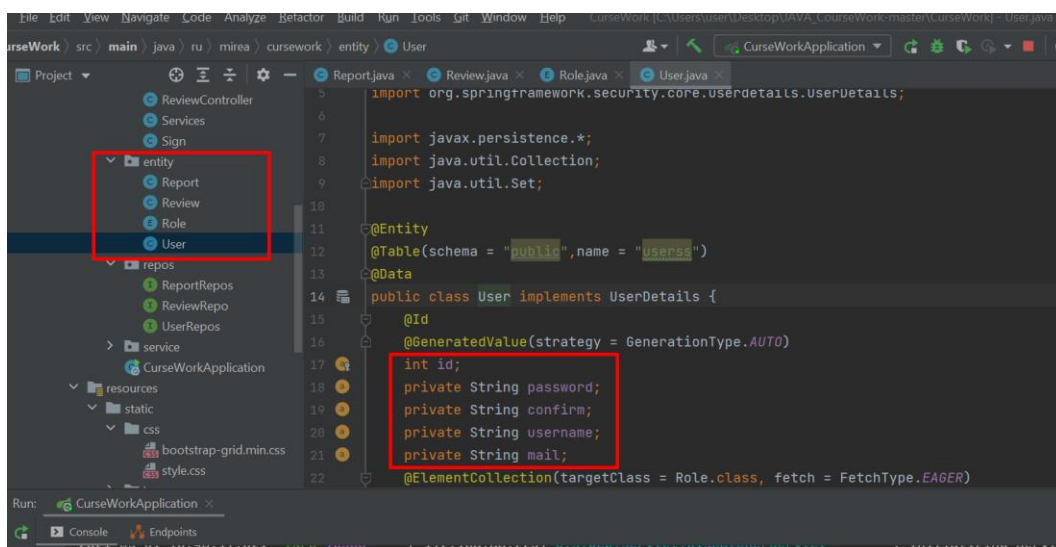


Рис. 3.1 – сущность User

- Сущность Role, класс типа enum с перечислением типов ролей (Таблица 2, рис. 3.2).

Таблица 2 – сущность Role

Имя	Role
Атрибуты	User, Admin
Методы	Стандартный набор методов для установки и получения значений

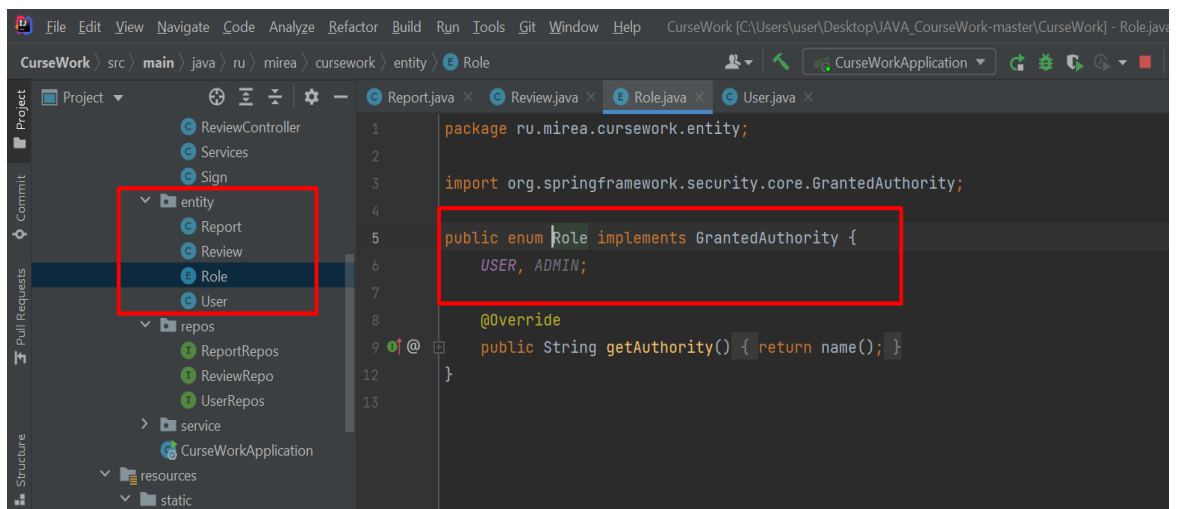


Рис. 3.2 – сущность Role

- Сущность Review, характеризует отзыв, который пользователь может оставить о клинике в разделе отзывы (Таблица 3, рис. 3.3).

Таблица 3 – сущность Review

Имя	Review
Атрибуты	id text
Методы	Стандартный набор методов для установки и получения значений, а также конструктор.

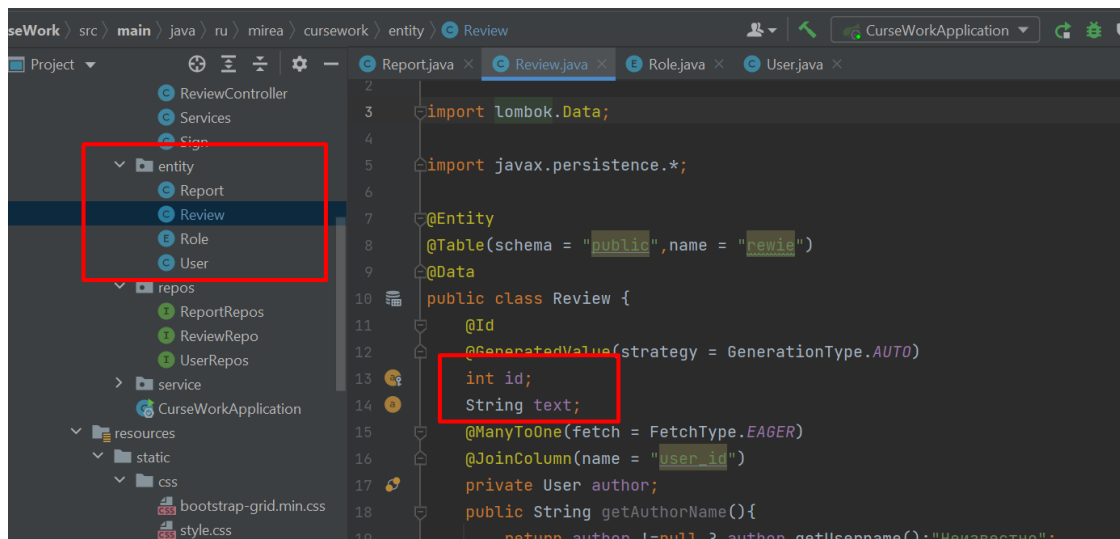


Рис. 3.3 – сущность Review

- Сущность Report, характеризует форму записи на прием.
(Таблица 4, рис. 3.4)

Таблица 1 – сущность Report

Имя	Report
Атрибуты	id serv name mail date note
Методы	Стандартный набор методов для установки и получения значений

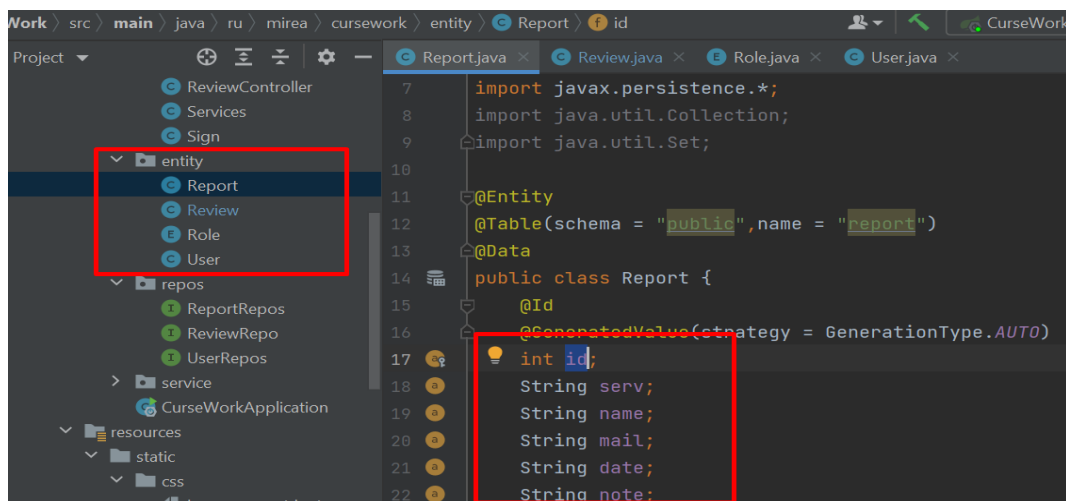


Рис. 3.4 – сущность Report

3.2 Репозитории

Выполнено создание необходимых репозиториях для созданных сущностей (рис. 3.5).

- UserRepos репозиторий для сущности User, расширяет CrudRepository с параметрами User и Long.
- ReportRepos репозиторий для сущности Report, расширяет CrudRepository с параметрами Report и Long.
- ReviewRepos репозиторий для сущности Review, расширяет CrudRepository с параметрами Review и Long.

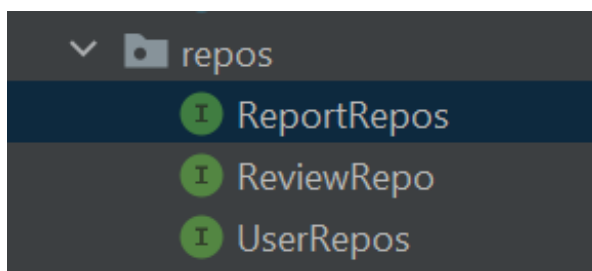


Рис. 3.5 – создание репозиториях

3.3 Сервисы

В рамках разрабатываемого проекта требовался сервис для работы сущности User.

Было выполнено создание UserServ, который имплементировал интерфейс UserDetailsService.

3.4 Контроллеры

Было выполнено создание четырех различных контроллеров (рис. 3.6):

- Services контроллер – осуществляет управление главной страницей, где пользователь может узнать информацию об услугах, а также записаться на прием. (Таблица 5)

Таблица 5 – контроллер Services

Имя	Services
Атрибуты	Report репозиторий
Методы	Обработка get запроса на главную страницу Обработка post запроса на страницу /index Обработка запроса Request для осуществления записи на прием

- ReviewController – осуществляет управление страницей с отзывами. (Таблица 6)

Таблица 6 - ReviewController

Имя	ReviewController
Атрибуты	Review репозиторий
Методы	Обработка get запроса на страницу /reviews Обработка post запроса на страницу /reviews Обработка запроса Request для реализации системы ОТЗЫВОВ

- Log контроллер – осуществляет управление страницей с личным кабинетом. (Таблица 7)

Таблица 7 – контроллер Log

Имя	Log
Атрибуты	User репозиторий Report репозиторий
Методы	Обработка get запроса на страницу /contacts Обработка post запроса на страницу /contacts Обработка запросов Request для реализации входа в личный кабинет и выхода из него

- Sign контроллер – осуществляет управление страницей с регистрацией. (Таблица 8)

Таблица 8 – контроллер Sign

Имя	Sign
Атрибуты	User репозиторий
Методы	Обработка get запроса на страницу /reg Обработка post запроса на страницу /reg Обработка запросов Request для реализации системы регистрации

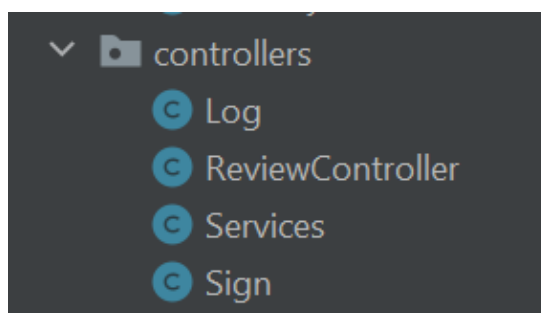


Рис. 3.6 – контроллеры

3.5 Конфигурации

В проекте представлено два класса конфигурации (рис. 3.7).

Security – осуществляет настройку Spring Security расширяет класс WebSecurityConfigurerAdapter, устанавливает страницы /, /reg, /reviews, contacts/** как страницы доступные пользователю без авторизации. Задают страницу /contacts, как страницу авторизации. Задаёт папки /resources/**, /static/**, /js/**, /css/**, /img/** как источники откуда может быть загружен статический контент.

MVCCConfig – аналогично классам контроллерам осуществляет настройку ответов страниц на запрос, но имеет более краткую форму записи. Используется для настройки ответа на запрос к странице по адресу /contacts.

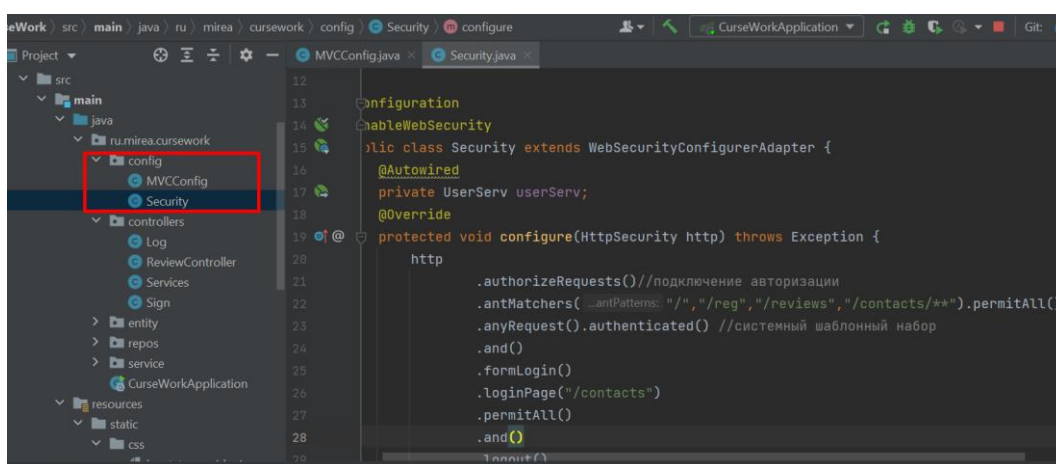


Рис. 3.7 – классы конфигурации

3.5 Разработка фронтенда

Для разработки визуального интерфейса были использованы следующие технологии: html, css, js, thymeleaf. Также был использован фреймворк bootstrap 4.

Bootstrap 4 - свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих

компонентов веб-интерфейса, включая JavaScript-расширения. Использование технологии позволяет существенно сократить временные затраты на разработку макета сайта. Функционал Bootstrap берет на себя решение задач адаптивности и кроссбраузерности. Это значит веб-страницы, созданные с помощью данной технологии, будут корректно отображаться в различных браузерах, а дизайн сайтов будет корректно отображаться на устройствах с разной диагональю.

Исходя из совокупности вышеперечисленных фактов было принято решение об актуальности идеи о использовании Bootstrap 4 при разработке визуального интерфейса веб-приложения.

3.5 Страницы и функции

В ходе разработки интерфейса был разработан ряд страниц для демонстрации функционала веб-приложения. Далее будет рассмотрен функционал каждой страницы отдельно.

Главная страница `index.html` (рис. 3.8). Представляет собой страницу сайта, на которой представлена панель навигации (услуги, отзывы, личный кабинет и регистрация, также показаны название клиники, адрес, часы работы и телефон), слайд-шоу, список услуг с описанием, а также форма записи на приём.

Запись на прием представляет собой форму, где пользователь выбирает услугу, указывает свое имя, почту, выбирает желаемую дату и при необходимости может оставить комментарий (рис. 3.9). Данные из формы передаются как `post` запрос к соответствующему контроллеру.

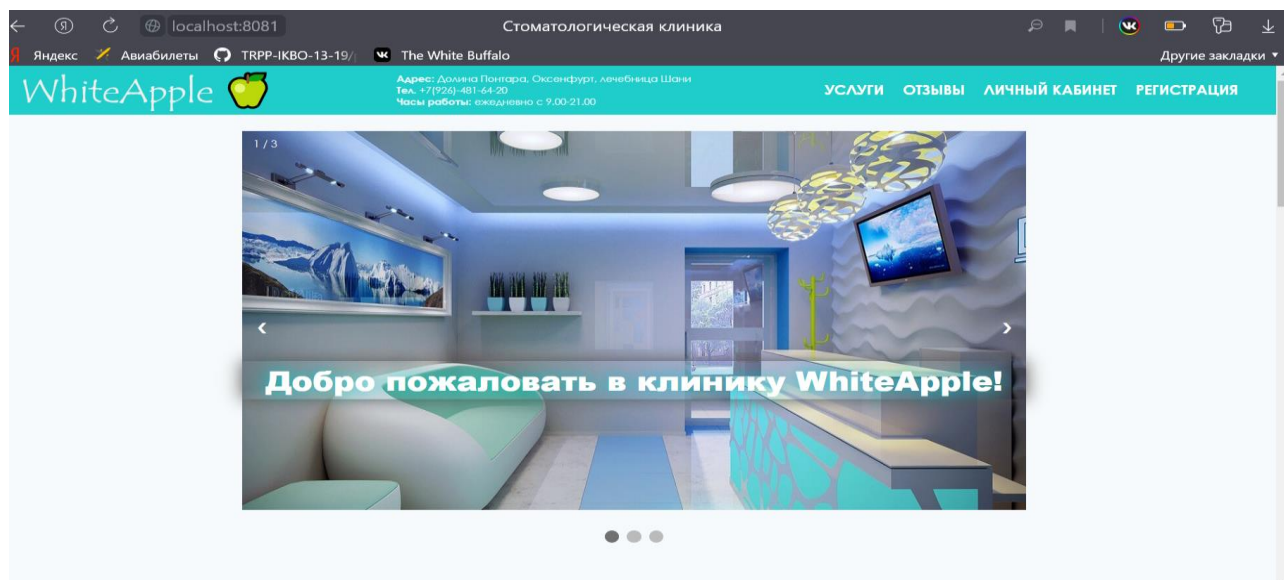


Рис. 3.8 - index.html

Рис. 3.9 – форма записи на прием

Страница регистрации reg.html. Представляет собой форму для ввода данных о новом пользователе (рис. 3.10). Данные из формы передаются как post запрос к соответствующему Sign контроллеру.

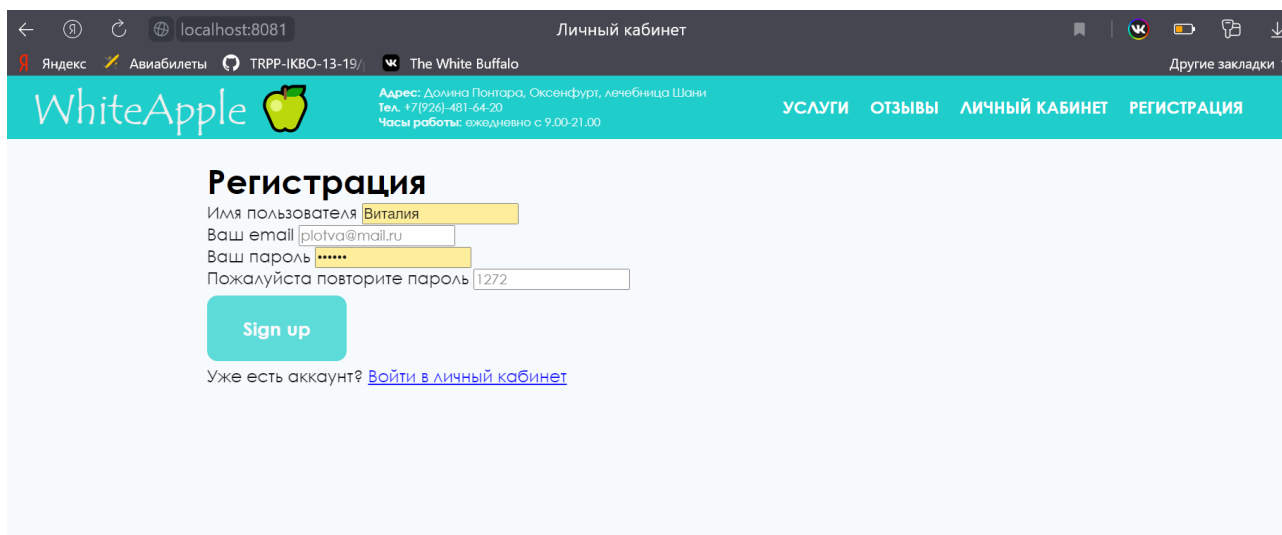


Рис. 3.10 - reg.html

Страница с авторизацией contacts.html. Также как и предыдущая страница представлена формой для ввода данных (рис. 3.11). Полученные данные из формы сверяются с базой данных. В случае успеха пользователь будет перенаправлен на главную страницу, иначе вход выполнен не будет. В личном кабинете отображается информация о пользователе в виде имени и почты, а также показана история записей на приемы.

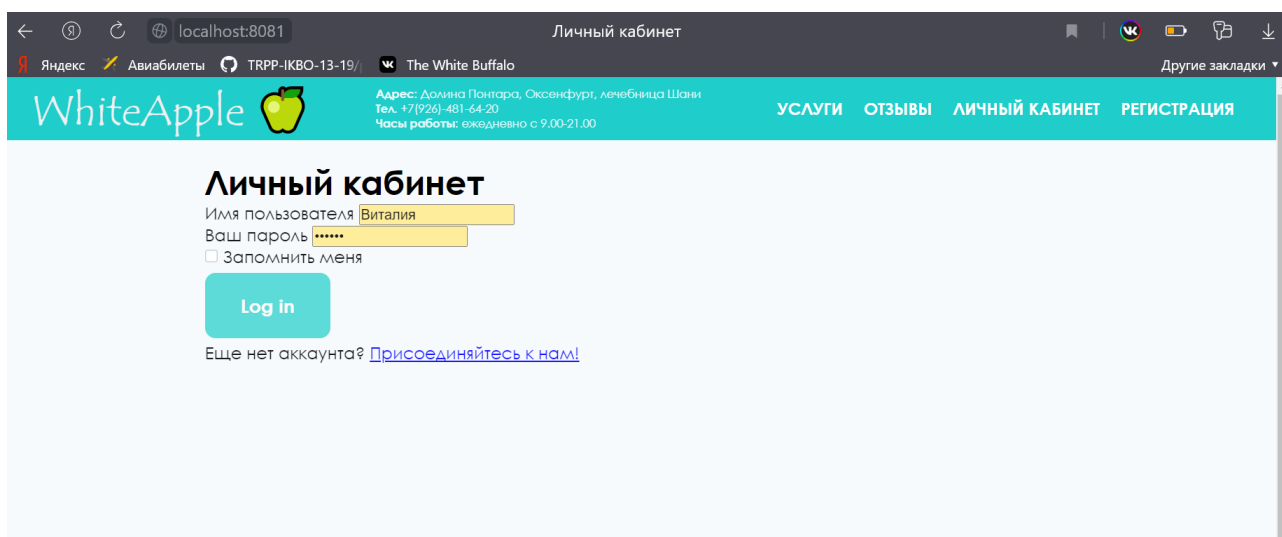


Рис. 3.11 - contacts.html

Отзывы представлены на странице reviews.html. На странице расположен слайд-шоу (рис. 3.12), а также непосредственно сами отзывы клиентов (рис.

3.13). Отзывы могут видеть все пользователи веб-приложения. Отзыв могут оставлять только авторизованные пользователи, у не авторизованных пользователей не будет доступно модальное окно для отзывов. Данные из формы для отзывов передаются как post запрос к соответствующему ReviewController.

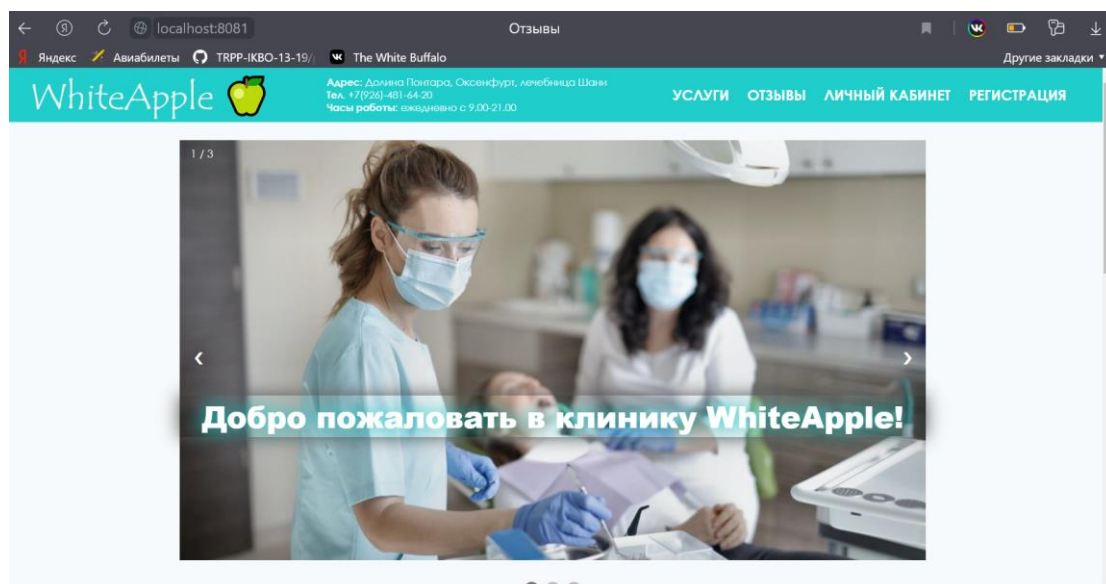


Рис. 3.12 – слайд-шоу на странице с отзывами

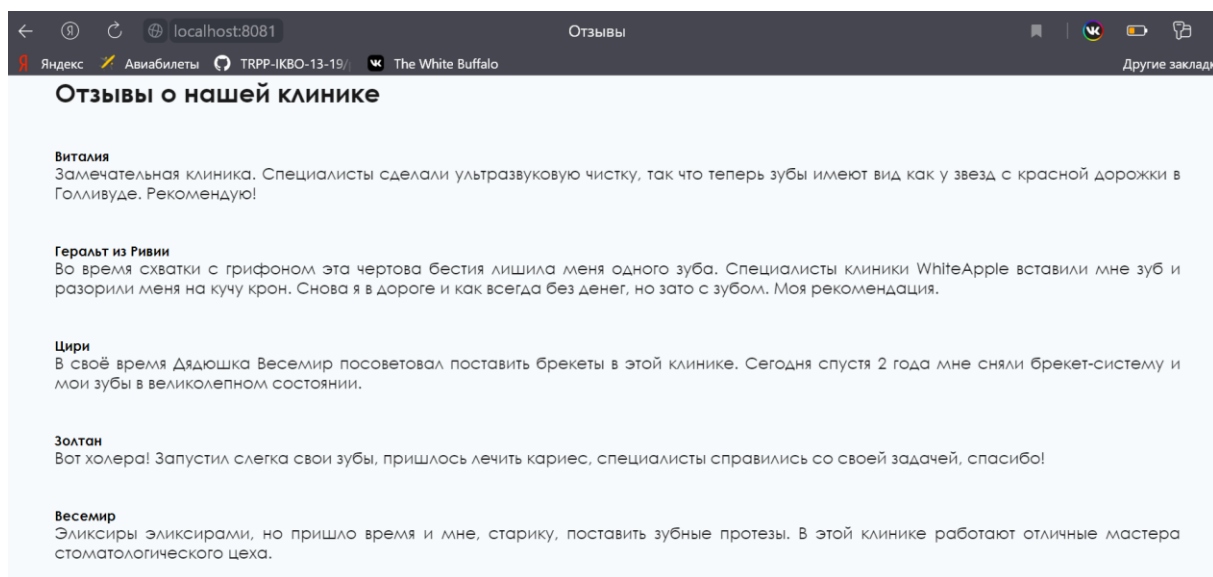


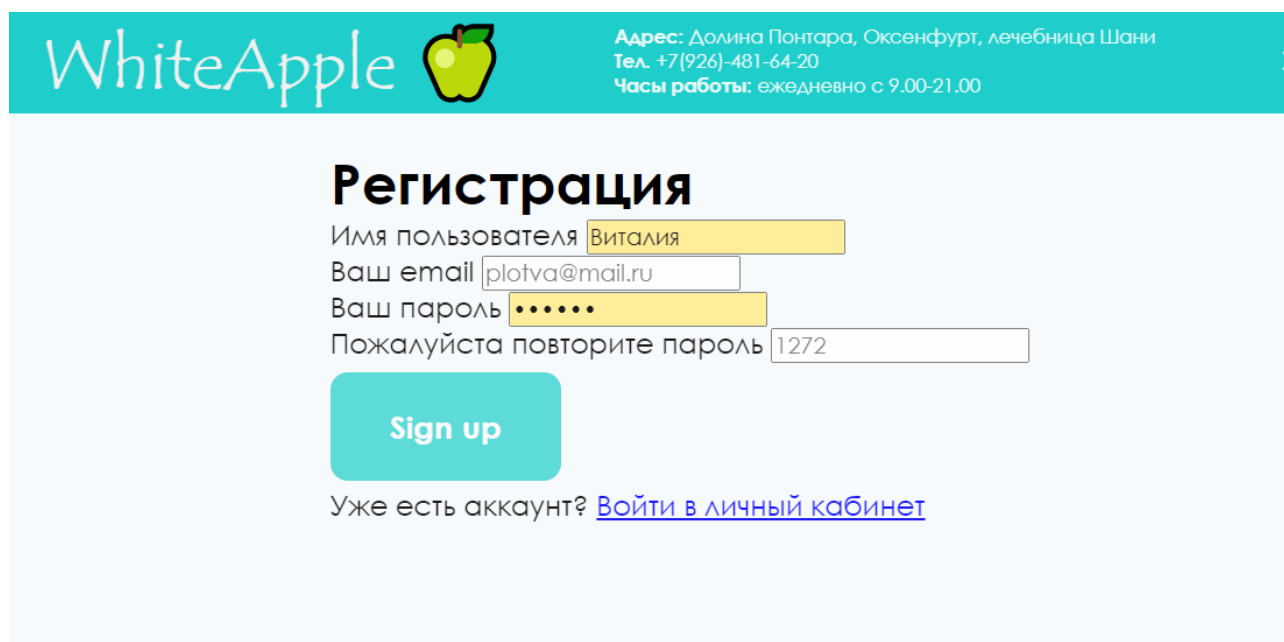
Рис. 3.13 – отзывы клиентов

3.6 Вывод к разделу 3

В ходе выполненной работы было создано 4 сущности, 4 контроллера, 2 класса конфигурации, 3 интерфейса репозитория, 1 сервис. А также создан фронтенд (визуальный интерфейс управления).

4. ТЕСТИРОВАНИЕ ВЕБ-ПРИЛОЖЕНИЯ

Выполнено тестирование функции регистрации. Внесены данные в форму (рис. 4.1). При успешной регистрации Яндекс.Браузер сохраняет пароль для конкретного имени пользователя (рис. 4.2). А сама информация о пользователе после регистрации заносится в базу данных.



The screenshot shows the registration page of the 'WhiteApple' application. The header is teal with the 'WhiteApple' logo and a green apple icon. Contact information is provided: 'Адрес: Долина Понтара, Оксенфурт, лечебница Шани', 'Тел. +7(926)-481-64-20', and 'Часы работы: ежедневно с 9.00-21.00'. The main heading is 'Регистрация'. The form contains four input fields: 'Имя пользователя' with the value 'Виталия', 'Ваш email' with 'plotva@mail.ru', 'Ваш пароль' with masked characters '.....', and 'Пожалуйста повторите пароль' with '1272'. A teal 'Sign up' button is below the fields. At the bottom, there is a link: 'Уже есть аккаунт? [Войти в личный кабинет](#)'.

Рис. 4.1 – регистрация

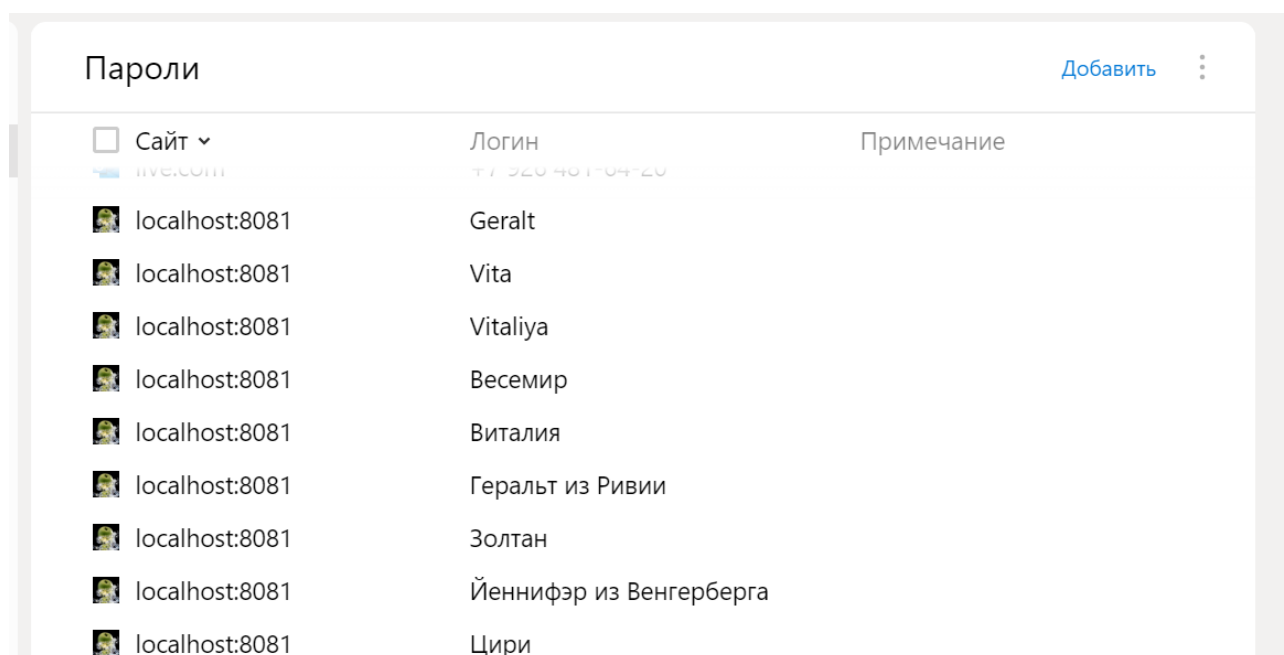



Рис. 4.2 – сохраненные пароли в Яндекс.Браузере

Выполнено тестирование функции входа в личный кабинет (рис. 4.3). После успешной авторизации пользователя перебрасывает на главную страницу, после чего в личном кабинете будет отображаться информация о пользователе и история записей (рис. 4.4). Чтобы избежать вывода истории записей всех пользователей из базы данных, была произведена фильтрация по email. И в личном кабинете будут отображаться записи, оформленные на адрес почты, которая была привязана к личному кабинету. Также в личном кабинете имеется ссылка для выхода.



Адрес: Долина Понтара, Оксенфурт, лечебница Шани
Тел. +7(926)-481-64-20
Часы работы: ежедневно с 9.00-21.00

Личный кабинет

Имя пользователя


Ваш пароль

☐ Запомнить меня

[Log in](#)

Еще нет аккаунта? [Присоединяйтесь к нам!](#)

Рис. 4.3 – авторизация



Адрес: Долина Понтара, Оксенфурт, лечебница Шани
Тел. +7(926)-481-64-20
Часы работы: ежедневно с 9.00-21.00

УСЛУГИ ОТЗЫВЫ ЛИЧНЫЙ КАБИНЕТ РЕГИСТРАЦИЯ

Vitaliya
malavita29@mail.ru

Лечение поверхностного кариеса
Виталия
malavita29@mail.ru
2021-05-31
Связаться по почте

Чистка ультразвуком
Виталия
malavita29@mail.ru
2021-06-05

Удаление зуба мудрости
Виталия
malavita29@mail.ru
2021-06-03

Консультация
Виталия
malavita29@mail.ru
2021-06-02

[Выйти из личного кабинета](#)

Рис. 4.4 – личный кабинет

Выполнено тестирование функции записи на прием (рис. 4.5). Если пользователь не авторизован, то информация о записи переносится в базу данных, а если авторизован, то также еще и заносится в личный кабинет и отображается в виде истории записей на прием (рис. 4.4)

-от 3400 руб.

Записаться

Услуга
 Комплексная чистка+ультразвук ▾

Ваше имя
 Виталия Малютина

Ваш email
 malavita29@mail.ru

Желаемая дата
 31.05.2021 📅

Ваши пожелания
 Просьба ответить на почту 📧

Рис. 4.5 – форма записи на прием

В заключение было выполнено тестирование функции системы отзывов (рис. 4.6). Если пользователь не авторизован, то ему будет видны только отзывы, а если авторизован, то будет высвечено модальное окно, в форме которого клиент может написать отзыв и отправить (рис. 4.7). Все отзывы заносятся в базу данных (рис. 4.8).

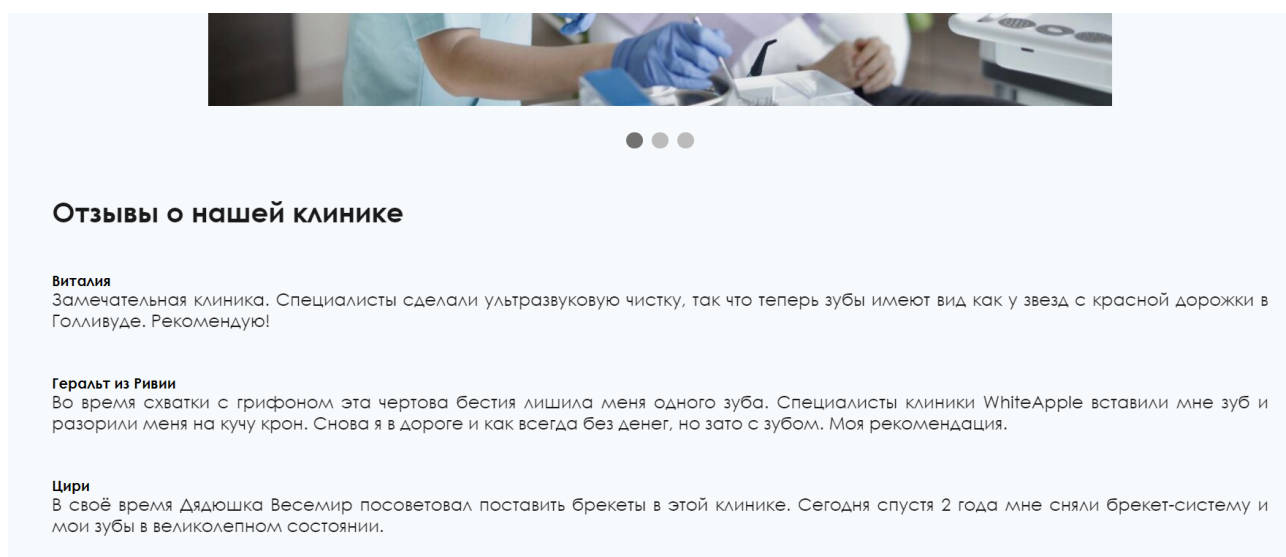


Рис. 4.6 – страница с отзывами без модального окна

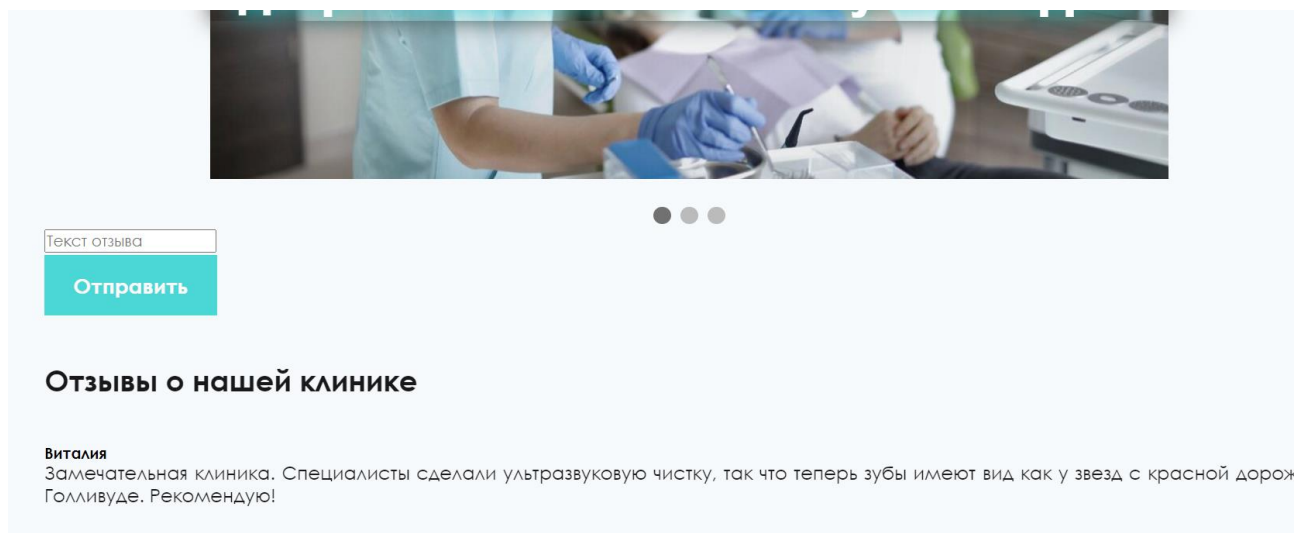


Рис. 4.7 – модальное окно

The image shows a database management tool interface. On the left is a project tree with folders like 'config', 'controllers', and 'entity'. The main area displays a table named 'review' with 6 rows. The table has columns for 'id', 'text', and 'user_id'. The data represents various user reviews of the clinic.

	id	text	user_id
1	102	Замечательная клиника. Специалисты сде...	602
2	252	Во время схватки с грифоном эта чертов...	603
3	253	В своё время Дядюшка Весемир советов...	652
4	254	Вот холера! Запустил слегка свои зубы,...	653
5	302	Эликсиры эликсирами, но пришло время и...	654
6	352	Клиника отличная, только зачем мне кли...	702

Рис. 4.8 – база данных с отзывами

4.1 Вывод к разделу 4

В разделе 4 было описано тестирование основных функций веб-приложения «WhiteApple». А именно системы регистрации, авторизации, личного кабинета, функции записи на прием, а также система отзывов.

5. АНТИПЛАГИАТ

Выполнена проверка на Антиплагиат (рис. 5.1). Использован веб-сервис antiplagiat.ru.

Отчет о проверке на заимствования №1



Автор: malavita29@mail.ru / ID: 7988084
Проверяющий: (malavita29@mail.ru / ID: 7988084)
Отчет предоставлен сервисом «Антиплагиат» - users.antiplagiat.ru

ИНФОРМАЦИЯ О ДОКУМЕНТЕ

№ документа: 4
Начало загрузки: 31.05.2021 18:50:57
Длительность загрузки: 00:00:01
Имя исходного файла: KP_Малютинa.pdf
Название документа: KP_Малютинa
Размер текста: 37 кБ
Символов в тексте: 38353
Слов в тексте: 4365
Число предложений: 355

ИНФОРМАЦИЯ ОБ ОТЧЕТЕ

Начало проверки: 31.05.2021 18:50:59
Длительность проверки: 00:00:16
Комментарии: не указано
Модули поиска: Интернет



ЗАИМСТВОВАНИЯ
6.13%

САМОЦИТИРОВАНИЯ
0%

ЦИТИРОВАНИЯ
0%

ОРИГИНАЛЬНОСТЬ
93.87%

Рис. 5.1 – проверка на антиплагиат

6. ЗАКЛЮЧЕНИЕ

В ходе выполнения курсовой работы было создано веб-приложение на тему «Стоматологическая клиника» с использованием следующих технологий: модуль Spring ORM, база данных PostgreSQL, модуль Spring Security, шаблонизатор Thymeleaf, система контроля версий github, система сборки Gradle. Был создан визуальный интерфейс для управления веб-приложением с использованием фреймворка bootstrap 4 и технологий HTML5, CSS3, JS. Было выполнено тестирование основных функций разработанного продукта. Был пройден тест на антиплагиат. Была создана презентация по выполненной курсовой работе. В ходе выполнения курсовой работы были освоены следующие компетенции:

- ПК-1 - способен выполнять разработку и интеграцию программных модулей и компонент, верификацию выпусков программных продуктов информационных систем
- ПК-1.1 - Знать: методы и средства сборки модулей и компонент программного обеспечения при создании информационных систем

- ПК-1.12 - Уметь: применять методы и средства создания программных интерфейсов информационных систем
- ПК-1.14 - Владеть: разработкой процедур сборки модулей и компонент программного обеспечения при внедрении информационных систем

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Лонг Д., Бастани К. Java в облаке. Spring Boot. — 2019. — 614 с. [Дата обращения: 21.05.21];
2. Ю. Козмина, Р. Харроп, К. Шефер, К. Хо. Spring 5 для профессионалов. — 2019. — 1120 с. [Дата обращения: 13.05.21];
3. Официальный веб-сайт spring.io [Электронный ресурс]: Режим доступа: <https://spring.io/> [Дата обращения: 20.05.21];
4. Официальный веб-сайт [thymeleaf.org](https://www.thymeleaf.org) [Электронный ресурс]: Режим доступа: <https://www.thymeleaf.org/> [Дата обращения: 20.05.21];
5. Spring Boot: делаем простое веб приложение на Java [Электронный ресурс]: Режим доступа: https://www.youtube.com/watch?v=jH17YkBTpI4&list=RDCMUC1g3kT0ZcSXt4_ZyJOshKJQ&index=3 [Дата обращения: 20.05.21];