



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: 09.03.04 «Программная
инженерия» бакалавриата

Тема: «Web-приложение Пейнтбольный клуб»

Студент: Назаркин Владимир Иванович

Группа: ИКБО-13-19

Работа представлена к защите 8.06.2021(дата) _____ /Назаркин В.И./
(подпись и ф.и.о. студента)

Руководитель: ст. преподаватель Зорина Н. В.
(ф.и.о., должность, звание, уч. степень)

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

Оценка по итогам защиты: _____

_____/ _____ /
_____/ _____ /

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

М. МИРЭА. 2021г.



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

Студент: Назаркин Владимир Иванович

Группа: ИКБО-13-19

Срок представления к защите: 25.04.2021

Руководитель: Зорина Наталья Валентиновна, старший преподаватель

Тема «Web-приложение Пейнтбольный клуб»

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 RESTful web-сервисов, 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.

И. о. зав. кафедрой ИиППО: _____ /Р. Г. Болбаков /, «_____» _____ 2021 г.

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

Задание на КР получил: _____ /В.И. Назаркин /, «_____» _____ 2021 г.

Назаркин В.И. «Web-приложение Пейнтбольный клуб»/ **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. 35, табл. 11, ист. 5.

Целью работы является разработка web-приложения для оценки функционирования различных пейнтбольных клубов.

В рамках работы осуществлены краткий анализ аналогов web-приложений их преимущества и недостатки.

Nazarkin V.I «Web-application Paintball club »/ **Coursework** in the discipline «Java software platform templates» of the profile «Software development and information systems design» courses undergraduate 09.03.04. "Software Engineering" (4th semester) / head associate senior lecturer N.V. Zorina / department IPPO of institute IT RTU MIREA – p.35, t.11, s.5.

The aim of the work is to develop a web application for evaluating the functioning of various paintball clubs.

As part of the work, a brief analysis of analogs of web applications, their advantages and disadvantages, has been carried out.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	7
1. Анализ предметной области	7
1.1. Вывод к разделу 1	11
2. Выбор средств ведения разработки.....	11
2.1. Выбор базы данных	12
2.2. Выбор инструмента для работы с базой данных	12
2.2.1. Spring DAO.....	13
2.2.2. Spring JDBC.....	13
2.2.3. Spring ORM.....	14
2.3. Spring Security.....	14
2.4. Хранения файлов проекта	15
2.5. Выбор Системы сборки	16
2.5.1. Maven	16
2.5.2. Gradle	17
2.6. Thymeleaf.....	17
2.7. Вывод к разделу 2	17
3. Разработка приложения	18
3.1. Сущности	18
3.2. Репозитории.....	20
3.3. Сервисы.....	20
3.4. Контроллеры.....	20
3.5. Конфигурация.....	23
3.6. Разработка Визуального интерфейса	24
3.6.1. Bootstrap.....	24
3.6.2. Страницы и функции.....	25
3.7. Вывод к разделу 3	31
4. Тестирование функций Web-приложения	31
4.1. Вывод к разделу 4	33

5. Антиплагиат.....	33
Заключение	34
Список использованной источников.....	35

СПИСОК ИСПОЛЬЗОВАННЫХ СОКРАЩЕНИЙ

JDK 14 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.

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

В современном мире ключевую роль играет информация. С каждым годом её становить всё больше, а передаётся она всё быстрее. Исходя из перечисленных фактов можно сделать вывод, что на передний план выходит проблема получения актуальной и достоверной информации.

Наиболее успешное разрешение данной проблемы можно увидеть у таких всемирно известных web-сервисов, как wikipedia.org или youtube.com. Их подход заключается в возможности свободного обновления информации большим количеством пользователей, что позволяет достичь наибольшей скорости обновления информации, а также наибольшей достоверности так как информация поступает напрямую от пользователей той или иной области. Но также стоит отметить существенный недостаток такого подхода: информация может быть искажена в силу субъективности восприятия отдельных индивидов. Может показаться, что данное утверждение отрицает возможность достижения достоверности, однако, чем больше людей причастны к созданию информационного потока, тем выше достоверность полученной информации.

Исходя из перечисленных фактов следует, что данный подход является наиболее эффективным для одновременного достижения ключевых свойств информации.

Именно на основе данного подхода и планируется проводить разработку web-сервиса по оценке пейнтбольных клубов.

1. Анализ предметной области

Заявленной темой является пейнтбол.

Пейнтбол (англ. Paintball — шар с краской) — командная игра с применением маркеров (пневматическая пушка), стреляющих шариками с краской (желатиновая оболочка с пищевой краской), разбивающимися при ударе о препятствие и окрашивающими его.

В ходе анализа предметной области были выделены следующие проблемы, с которыми может столкнуться человек желающий заниматься озвученным видом активного отдыха:

- Как получить необходимое снаряжение
- Где найти подходящую площадку
- Как обеспечить безопасность свою и окружающих

Наиболее распространённым решением перечисленных проблем является обращение в специализированные клубы, которые предоставляют услуги по проведению мероприятий подобного типа. Существует множество подобных клубов, которые предоставляют услуги разного качества по различной цене.

Ниже приведен краткий перечень наиболее известных клубов Москвы и Московской области (Таблица 1.1).

Таблица 1.1 – Пейнтбольный клубы

№	Название	Местоположение	Тип площадки
1	Пейнтбольный клуб Обитель зла	Перово	крытый
2	Пейнтбольный клуб Paintland	Косино-Ухтомский	открытый
3	Подземный клуб Лабиринт	Дорогомилово	крытый
4	Тактический центр Контакт	Филевский парк	крытый
5	Пейнтбольный клуб Северный Олень	Тимирязевский	открытый

Продолжение таблицы 1.1

№	Название	Местоположение	Тип площадки
6	Пейнтбольный клуб МФП Водный Стадион	Головинский	крытый
7	Клуб Град	Отрадное	открытый
8	Пейнтбольный клуб Пэйнтлэнд Парк	Лосиноостровский	открытый
9	Пейнтбольный клуб Тушино	Покровское-Стрешнево	открытый
10	Пейнтбольный клуб Город	Басманный	открытый
11	Пейнтбольный клуб Арена	Даниловский	открытый и крытый
12	Пейнтбольный клуб МФП Битца	Чертаново Северное	открытый
13	Пейнтбольный клуб МФП Лефортово	Лефортово	крытый
14	МФП Cs Волгоградка	Текстильщики	открытый
15	Пейнтбольный клуб Бункер	Нижегородский	крытый
16	крытый	Южнопортовый	крытый

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:

Moscowplaces.ru – новостной сайт, содержащий информацию о различных интересных местах Москвы. Было обнаружено несколько статей с рейтингом различных пейнтбольных клубов (Рисунок 1.1).

Интересные места Москвы

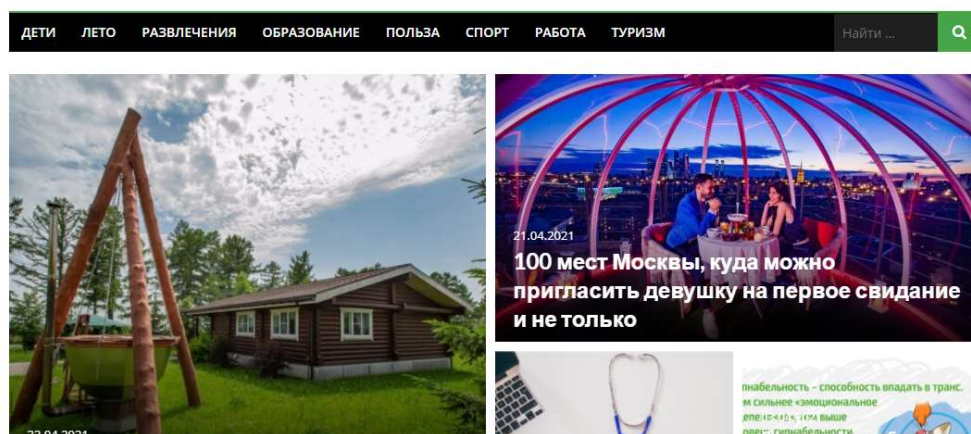


Рисунок 1.1 - Moscowplaces.ru

Freezone.net – ресурс, предоставляющий информация о одноименном Спортивно-развлекательном Парке, также содержит несколько статей с рейтингом клубов (Рисунок 1.2).

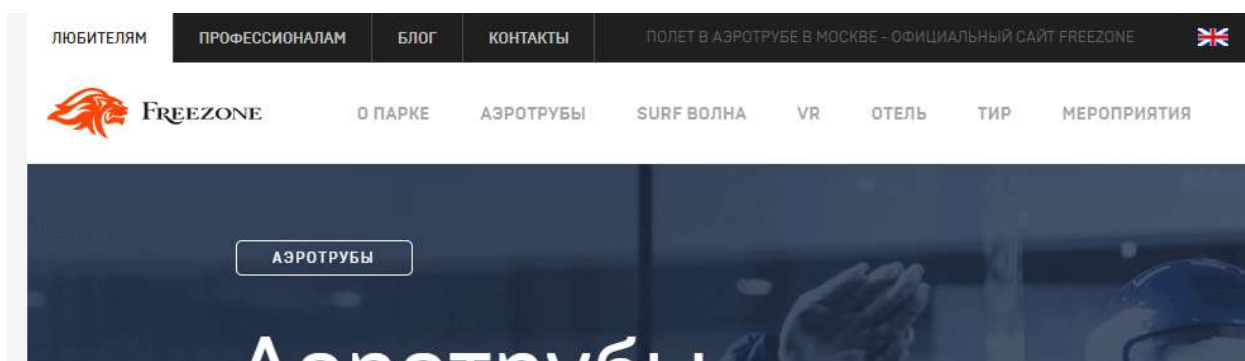


Рисунок 1.2 - Freezone.net

Irecommend.ru – интернет ресурс, предоставляющий возможность отправки различных отзывов, были обнаружены отзывы на некоторые клубы (Рисунок 1.3).

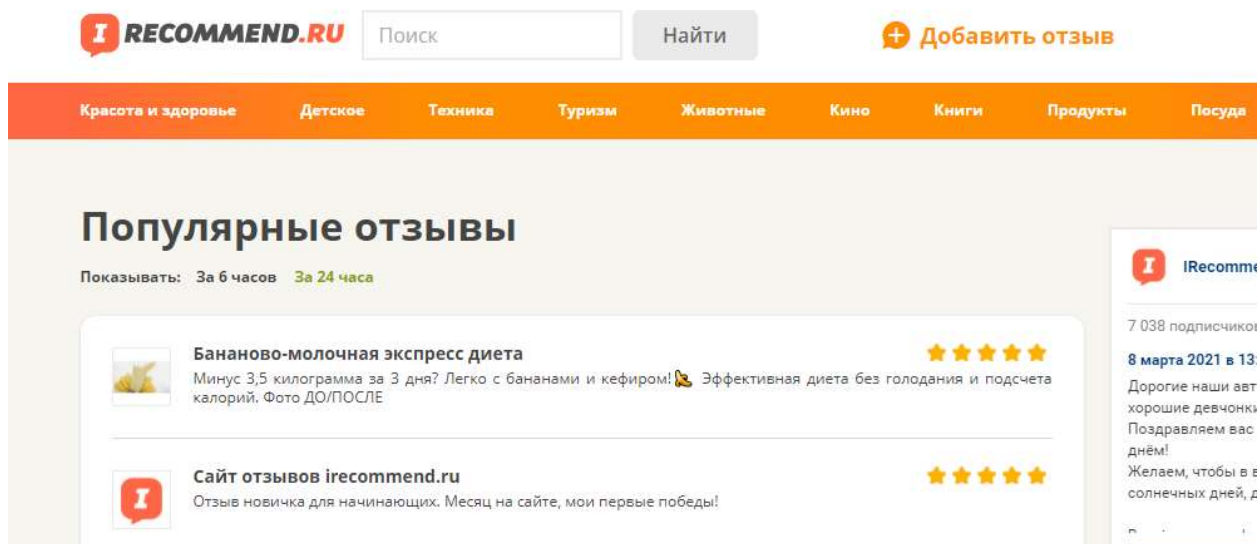


Рисунок 1.3 - Irecommend.ru

1.1. Вывод к разделу 1

В результате анализа полученной информации, был сделан вывод о том, что не один из рассмотренных источников не позволяет в полной мере разрешить проблему неопределенности. Исходя из этого было принято решение разработать web-приложение со следующим функционалом.

- Возможность размещать информацию о различных пейнтбольных клубах.
- Возможность оставлять отзывы об выбранных кандидатов в виде числовой оценки и текстового комментария.
- Составление рейтинга клубов исходя из их оценок.
- Поддержка системы регистрации и авторизации.

2. Выбор средств ведения разработки

Исходя из условий поставленной задачи разработка web-приложения будет осуществляется на базе фреймворка spring для типизированного объектно-ориентированного языка программирования общего назначения под названием java. Разработка выполняется на языке java версии 13.

2.1. Выбор базы данных

В ходе анализа предметной области были выделены следующие задачи:

- Возможность размещать информацию о различных пейнтбольных клубах.
- Возможность оставлять отзывы об выбранных кандидатах в виде числовой оценки и текстового комментария.

Для реализации вышеперечисленных целей необходимо хранить большой объем информации, 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 C-совместимых модулей;
- наследование;
- легкая расширяемость.

2.2. Выбор инструмента для работы с базой данных

Фреймворк spring предоставляет дополнительный модули для удобного взаимодействия с базами данных. В ходе анализа было выявлено два наиболее перспективных кандидата:

- Spring ORM
- Spring DAO

➤ Spring JDBC

Подробнее рассмотрим каждый из пересиленных модулей.

2.2.1. Spring DAO

Spring DAO — это не модуль spring в строгом смысле, а скорее набор условностей. Таким образом, он не предоставляет ни интерфейсов, ни реализаций, ни шаблонов для доступа к вашим данным. При использовании DAO вам необходимо использовать аннотацию `@Repository`, чтобы исключения, связанные с базовой технологией (JDBC, Hibernate, JPA и т. д.), последовательно преобразовывались в соответствующий подкласс `DataAccessException`.

Достоинства использования DAO:

- Шаблон проектирования DAO также обеспечивает низкую связь между различными частями приложения.
- Сила шаблона DAO состоит в том, что они позволяют вам создать хороший уровень абстракции реальной системы хранения. Они обеспечивают более объектно-ориентированное представление уровня сохраняемости и четкое разделение между доменом и кодом, который фактически будет выполнять доступ к данным
- Общие вызовы для получения объектов.

Недостатки использования DAO:

- Добавление дополнительного слоя, что приводит к усложнению функционирования системы.
- Недостаточная гибкость инструмента
- Большое количество шаблонного повторяющегося кода

2.2.2. Spring JDBC

Spring JDBC — предоставляет класс `JdbcTemplate`, который помогает вам сосредоточиться на запросе и параметрах SQL. Вам просто необходимо настроить его с помощью `DataSource`. Spring-JDBC также предоставляет `JdbcDaoSupport`, который можно использовать, чтобы расширить функционал

DAO. Он определяет два свойства, а именно DataSource и JdbcTemplate, которые могут быть использованы для реализации методов DAO. Он также предоставляет преобразователь с SQL исключения на spring DataAccessExceptions.

2.2.3. Spring ORM

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 JDBC:

- 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 остановить поиск на ней.

2.3. Spring Security

Для реализации следующей задачи:

- Поддержка системы регистрации и авторизации.

Был выбран модуль 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.

2.4. Хранения 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.

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

2.5. Выбор Системы сборки

Наиболее удобными системами сборки java приложений являются:

- Maven
- Gradle

Подробнее рассмотрим каждый из вариантов.

2.5.1. Maven

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.

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

2.5.2. Gradle

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.

Плюсы:

- Гибкость настройки
- Краткость описания процесса сборки

Минусы:

- Сложность освоения

В качестве системы сборки разрабатываемого проекта была выбрана система сборки Maven, определяющим фактором выбора послужила относительная простота освоения вышеупомянутой системы сборки.

2.6. Thymeleaf

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

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.

2.7. Вывод к разделу 2

В ходе проведенного анализа средств разработки были выбраны следующие технологии и модули: PostgreSQL, Spring ORM, Github, Maven, Thymeleaf, Spring Security.

3. Разработка приложения

Разработка приложения производится в IntelliJ IDEA. На первой стадии разработки было выполнено создание необходимых сущностей.

3.1. Сущности

Были созданы следующие сущности:

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

Таблица 3.1 - User

Имя	User
Атрибуты	Имя пользователя Пароль Активность Набор Ролей
Методы	Стандартный набор методов для установки и получения значений, а также конструктор

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

Таблица 3.2 - Role

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

- Сущность Post, характеризует раздел с информацией о конкретном пейнтбольном клубе (Таблица 3.3).

Таблица 3.3 - Post

Имя	Post
Атрибуты	Имя Текст Ссылка на изображение Количество просмотров Рейтинг Количество отзывов
Методы	Стандартный набор методов для установки и получения значений Пустой конструктор Конструктор с инициализацией всех атрибутов класса Метод обновления рейтинга Метода добавления сообщения Метод добавления просмотра

- Сущность Message, характеризует сообщение, которое пользователь может оставлять под выбранным постом (Таблица 3.4).

Таблица 3.4 - Message

Имя	Message
Атрибуты	Текст Автор Отзыв
Методы	Стандартный набор методов для установки и получения значений, а также конструктор.

3.2. Репозитории

Выполнено создание необходим репозитория для созданных сущностей.

- UserRepo репозиторий для сущности User, расширяет CrudRepository с параметрами User и Long
- PostRepo репозиторий для сущности Post, расширяет CrudRepository с параметрами Post и Long
- MessageRepo репозиторий для сущности Message, расширяет CrudRepository с параметрами Message и Long

3.3. Сервисы

В рамках разрабатываемого проекта сервис требовался для работы сущности User.

Было выполнено создание UserService, который имплементировал интерфейс UserDetailsService.

3.4. Контроллеры

Было выполнено создание шести различных контроллеров:

- HomeController – осуществляет управление домашней страницей, а также страницами /users и logout (Таблица 3.5).

Таблица 3.5 - HomeController

Имя	HomeController
Атрибуты	User репозиторий Post репозиторий
Методы	Обработка get запроса домашней страницы Обработка get запроса страницы /users, для вывода списка существующих пользователей. Обработка запроса Request для осуществления выхода авторизированного пользователя

- RegController – осуществляет управление страницей регистрации (Таблица 3.6).

Таблица 3.6 - RegController

Имя	RegController
Атрибуты	User репозиторий
Методы	Обработка get запроса на страницу /registration Обработка post запроса на страницу /registration

- ProfileController – осуществляет управление страницей профиля (Таблица 3.7).

Таблица 3.7 - ProfileController

Имя	ProfileController
Атрибуты	
Методы	Обработка get запроса на страницу /profile.

- PostController – осуществляет управление страницами добавления нового поста, удаления, а также просмотра списка доступных постов и вывода подробной информации о каждом конкретном посте (Таблица 3.8).

Таблица 3.8 - PostController

Имя	PostController
Атрибуты	Message репозиторий Post репозиторий
Методы	Обработка get и post запроса на страницу /addpost Обработка get запроса на страницу /posts Обработка get и post запроса на страницу /posts/{id}

- MessageController – создан как инструмент администратора для управления всеми доступными сообщениями (Таблица 3.9).

Таблица 3.9 - MessageController

Имя	MessageController
Атрибуты	Message репозиторий
Методы	Обработка get и post запросов на страницу /messages

- DefaultController – создан как инструмент восстановления в случае потери базы данных для заполнения базы данных стандартным набором параметров (Таблица 3.10).

Таблица 3.10 - DefaultController

Имя	DefaultController
Атрибуты	Post репозиторий User репозиторий
Методы	Обработка get запроса на страницу /setDefault

3.5.Конфигурация

В проекте представлено два класса конфигурации.

WebSecurityConfig – осуществляет настройку Spring Security расширяет класс WebSecurityConfigurerAdapter, устанавливает страницы /, /registration, /messages, /posts, /posts/** как страницы доступные пользователю без авторизации. Задают страницу /login, как страницу авторизации. Задает папки /resources/**, /static/**, /Source/**, /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 /login

3.6. Разработка Визуального интерфейса

Для разработки визуального интерфейса были использованы следующие технологии: html, css, js, thymeleaf. Также был использован фреймворк bootstrap 5.

3.6.1. Bootstrap

Bootstrap — 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 (Рисунок 3.).

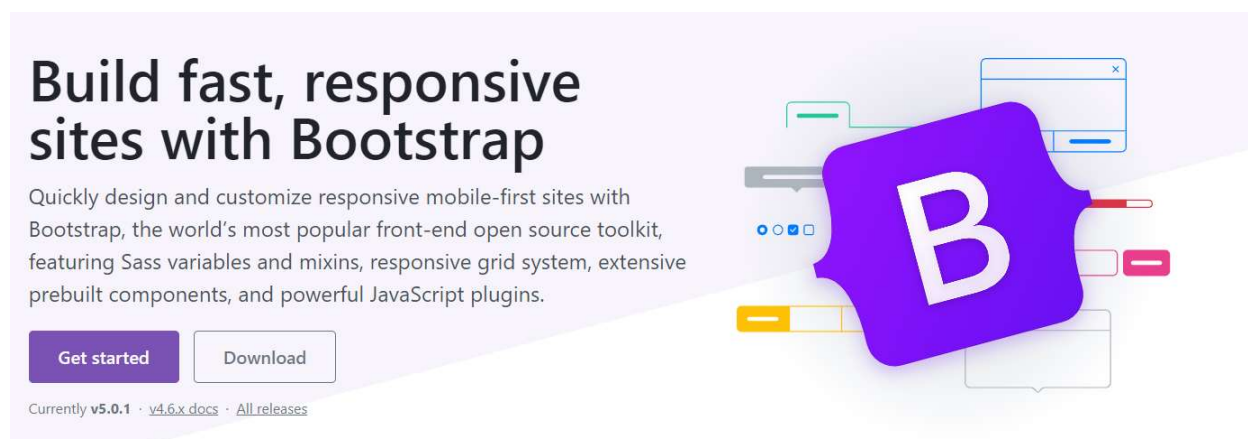


Рисунок 3.1 – Bootstrap

Использование фреймворка значительно облегчает работу разработчика и позволяет разрабатывать проекты быстрее. Bootstrap дает набор готовых решений использование которых позволяет сократить время, затрачиваемое на создание макета сайта. А также позволяет писать качественный и понятный код, который легко поймет другой разработчик.

Однако сайты созданные при помощи Bootstrap похожи друг на друга, что является существенным недостатком. Также сайты созданный при помощи

данного фреймворка могут некорректно отображаться в некоторых старых браузерах.

Исходя из совокупности вышеперечисленных фактов было принято решение об актуальности идеи о использовании Bootstrap 5 при разработке визуального интерфейса web-приложения.

3.6.2. Страницы и функции

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

Домашняя страница home.html. Представляет собой Главную страницу сайта на ней представлена панель навигации, а также топ из трёх постов с наивысшей оценкой пользователей (Рисунок 3.).

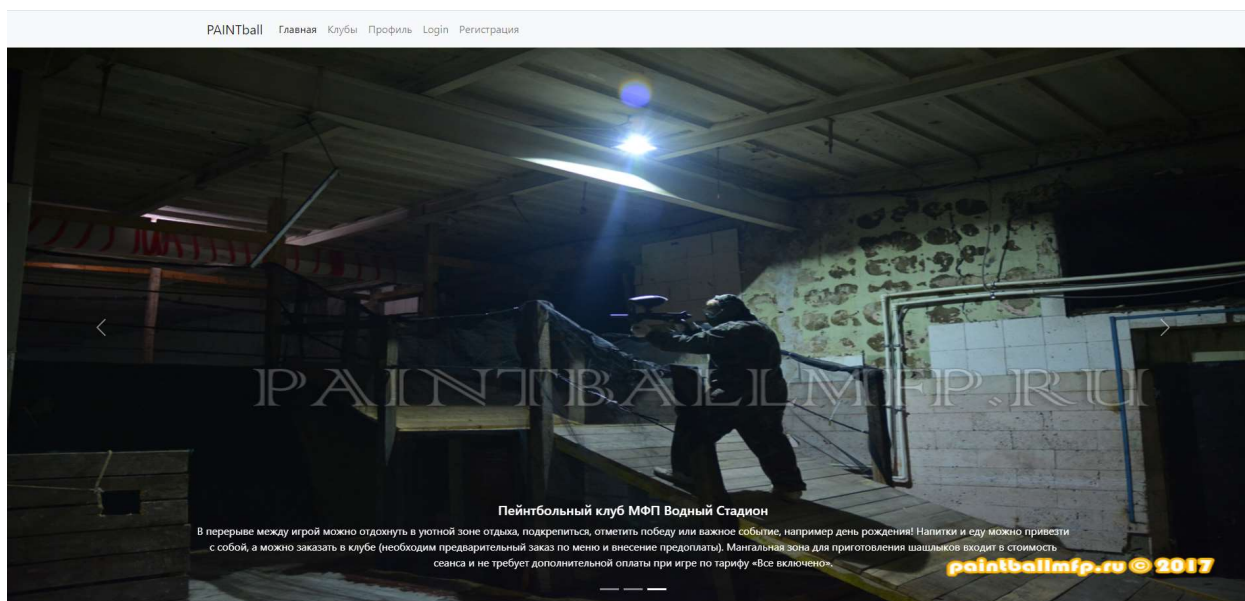


Рисунок 3.2 - home.html

На панели навигации также находятся разделы Login и Регистрация

Страница регистрации registration.html. Представляет собой форму для ввода данных о новом пользователе (Рисунок). Данные из формы передаются как post запрос к соответствующему контроллеру, который проверяет данные

на корректность, в том числе происходит проверка имени пользователя на уникальность.

The registration form is located within a light blue header bar containing the site name 'PAINTball' and navigation links: 'Главная', 'Клубы', 'Профиль', 'Login', and 'Регистрация'. The form itself consists of three white input fields with gray borders, stacked vertically. The first field is labeled 'Имя пользователя', the second 'Email', and the third 'Пароль'. Below these fields is a blue button with the text 'Регистрация' in white.

Рисунок 3.3 - registration.html

Страница Авторизации login.html. Также как и предыдущая страница представлена формой для ввода данных (Рисунок). Полученные данные из формы сверяются с базой данных и выводится результат. В случае успеха пользователь будет перенаправлен на домашнюю страницу, иначе выведется сообщение с соответствующей ошибкой.

The login form is located within the same light blue header bar as the registration form. It contains two white input fields with gray borders, stacked vertically. The first field is labeled 'Имя пользователя' and the second 'Пароль'. Below these fields is a blue button with the text 'Отправить' in white.

Рисунок 3.4 - login.html

Раздел панели навигации клубы представлен страницей posts.html. На странице расположен набор элементов, содержащих изображение, название и краткое описание клуба (Рисунок).

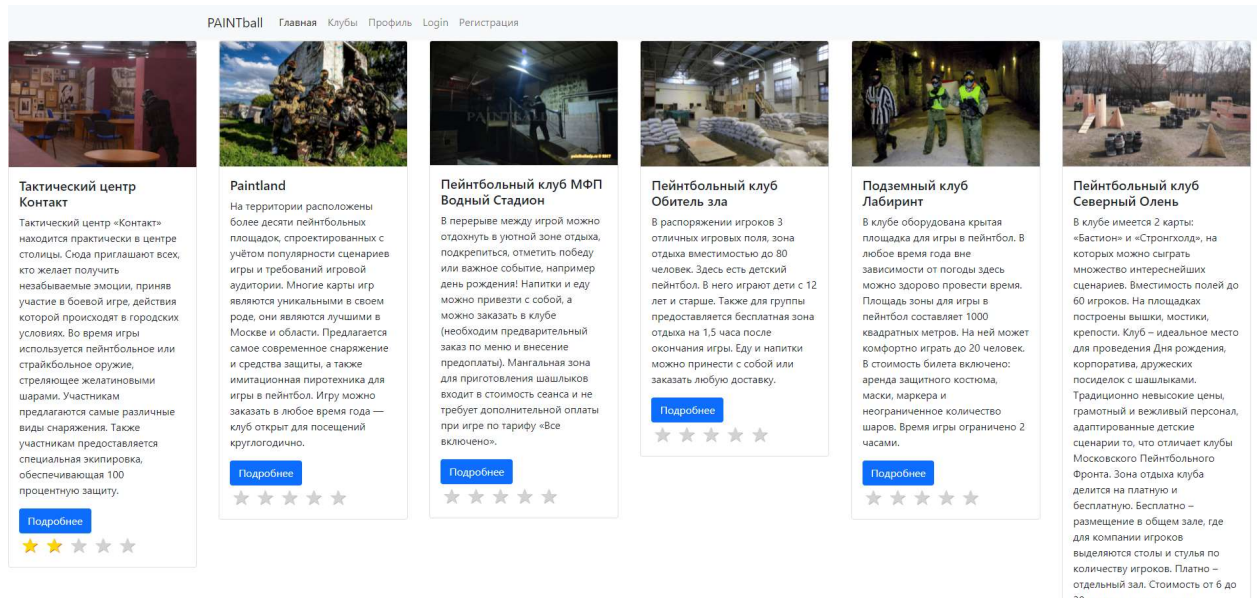


Рисунок 3.5 - posts.html

Нажав на кнопку Подробнее, пользователь переходит по адресу /posts/{id} на страницу представленную post.html. На странице представлено изображение, название клуба, информация о нем, а также средняя оценка и количество просмотров (Рисунок 3.6 - post.html).

Тактический центр Контакт



Тактический центр «Контакт» находится практически в центре столицы. Сюда приглашают всех, кто желает получить незабываемые эмоции, приняв участие в боевой игре, действия которой происходят в городских условиях. Во время игры используется пейнтбольное или страйкбольное оружие, стреляющее желатиновыми шарами. Участникам предлагаются самые различные виды снаряжения. Также участникам предоставляется специальная экипировка, обеспечивающая 100 процентную защиту.



Просмотры
13

Рисунок 3.6 - post.html

В нижней части экрана имеется список сообщений, оставленных пользователями сайта с информацией о пользователе сайта, а также оценкой, которую поставил пользователь (Рисунок).

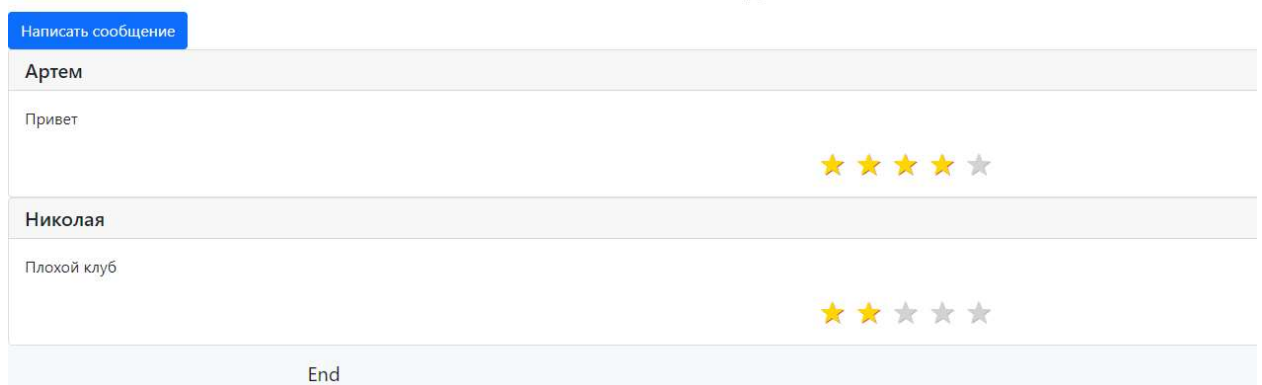


Рисунок 3.7 - Сообщения

Авторизованным пользователям дополнительна доступна функция отправки отзыва, представленная в форме модального окна (Рисунок).

Сообщение

Николая

★★★★☆

Заголовок:

Текст:

Хороший клуб

Очистить Отправить

Рисунок 3.8 - Модальное окно

Пользователю с привилегиями администратора доступны дополнительные функции.

Просмотр списка всех пользователей по адресу /users (Рисунок).

PAINTball Главная Клубы Профиль Login Регистрация
user
user@gmail.com
1802
admin
admin@gmail.com
1804
ad
admin@gmail.com
1803
TestUser
user@gmail.com
1852
sdf
sdf
1902
Арте́м
user@gmail.com
1952

Рисунок 3.9 - users.html

А также возможность добавлять новые клубы в список (Рисунок).

PAINTball	Главная	Клубы	Профиль	Log out
-----------	---------	-------	---------	---------

Имя поста

Текст

Картинка

Выложить

Рисунок 3.10 - addpost.html

3.7. Вывод к разделу 3

В ходе выполненной работы было создано 4 сущности, 6 контроллеров, 2 класса конфигурации, 3 интерфейса репозитория, 1 сервис. А также создан визуальный интерфейс управления.

4. Тестирование функций Web-приложения

Выполнено тестирование функции регистрации. Внесены данные в форму (Рисунок).



The image shows a registration form with three input fields and a button. The first field is labeled 'Имя пользователя' (Username) and contains the text 'Владимирр'. The second field is labeled 'Email' and contains the text 'vladimir@gmail.com'. The third field is labeled 'Пароль' (Password) and contains the text '12345678'. Below the fields is a blue button with the text 'Регистрация' (Registration).

Имя пользователя	Владимирр
Email	vladimir@gmail.com
Пароль	12345678

Регистрация

Рисунок 4.1 - Регистрация. До

Отправлены данные и выполнен переход на панель администратора (Рисунок).

Артём
user@gmail.com
1952
Николая
user@gmail.com
1953
Владимирр
vladimir@gmail.com
2002
End

Рисунок 4.2 - Регистрация. После

Пользователь успешно добавлен.

Выполнена проверку функционирования системы сообщений. Выполнен переход на страницу одного из постов и заполнена форма сообщения (Рисунок).

Сообщение ✕

Владимирр



Заголовок:

Текст:

Привет

Очистить

Отправить

Рисунок 4.3 - Сообщение. До


Форма отправлена сообщение добавлено в список (Рисунок).

Просмотры
4

Написать сообщение

Владимирр

Привет



End

Рисунок 4.4 - Сообщение. После

4.1.Вывод к разделу 4

В данном разделе было выполнено тестирование основных функций web-приложения. А именно системы Авторизации и системы отправки сообщений.

5. Антиплагиат

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

Отчет о проверке на заимствования №1



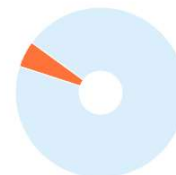
Автор: Назаркин Владимир
Проверяющий: Назаркин Владимир (vohammer2001@gmail.com / ID: 7228339)
Отчет предоставлен сервисом «Антиплагиат» - users.antiplagiat.ru

ИНФОРМАЦИЯ О ДОКУМЕНТЕ

№ документа: 8
Начало загрузки: 23.05.2021 21:08:51
Длительность загрузки: 00:00:01
Имя исходного файла:
Назаркин_Отчет_KP_Spring.pdf
Название документа:
Назаркин_Отчет_KP_Spring
Размер текста: 29 кБ
Символов в тексте: 30174
Слов в тексте: 3454
Число предложений: 233

ИНФОРМАЦИЯ ОБ ОТЧЕТЕ

Начало проверки: 23.05.2021 21:08:53
Длительность проверки: 00:05:24
Комментарии: не указано
Модули поиска: Модуль поиска inet



ЗАИМСТВОВАНИЯ

5,07%

САМОЦИТИРОВАНИЯ

0%

ЦИТИРОВАНИЯ

0%

ОРИГИНАЛЬНОСТЬ

94,93%

Рисунок 4.5 - Антиплагиат

ЗАКЛЮЧЕНИЕ

В ходе выполнения курсовой работы был создано Restful web-сервис с использованием следующих технологий модуль Spring ORM, база данных PostgreSQL, модуль Spring Security, шаблонизатор Thymeleaf, система контроля версий github, система сборки Maven. Было выполнено создание визуального интерфейса для управления web-приложением с использованием фреймворка bootstrap 5. Было выполнено тестирование основных функций разработанного продукта. Была выполнен текст на антиплагиат. Была создана презентации 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 ИСТОЧНИКОВ.

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 getbootstrap.com [Электронный ресурс]: Режим доступа: <https://getbootstrap.com/> [Дата обращения: 19.05.21];
4. Официальный веб-сайт spring.io [Электронный ресурс]: Режим доступа: <https://spring.io/> [Дата обращения: 4.05.21];
5. Официальный веб-сайт [thymeleaf.org](https://www.thymeleaf.org/) [Электронный ресурс]: Режим доступа: <https://www.thymeleaf.org/> [Дата обращения: 2.05.21];