

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Вятский государственный университет»
(ВятГУ)

**ОТЧЕТ
ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

Кудяшев Ярослав Юрьевич

(Ф.И.О. обучающегося)

*09.01.01 Информатика и вычислительная техника. Программное и
аппаратное обеспечение вычислительной техники*

(направление подготовки (специальность), направленность (профиль))

Место прохождения практики *ООО «Синаптика», отдел проектирование
интерфейсов*

(наименование организации, структурного подразделения организации)

Итоговая оценка: _____

Руководитель
практики от университета

(дата)

(подпись)

Крутиков А.К.

(Ф.И.О.)

Киров, 2021 г.

ВВЕДЕНИЕ

В настоящее время представить работу спортивного аналитика без каких-либо сподручных средств практически невозможно. Человек способен наблюдать за ходом спортивного матча и мысленно анализировать действия игроков: выделять слабые и сильные стороны как в защите, так и в нападении.

Конечно, обычный зритель, наблюдающий за игрой, вряд ли сможет запомнить всех игроков. В баскетболе, как правило, люди привыкли в первую очередь акцентировать внимание на количество очков, набираемых игроком, что является большой ошибкой. Они забывают про такие показатели, как эффективность игрока, его защитный рейтинг и остальную статистику, помимо очков. Лишь благодаря тщательной аналитике можно сказать, кто является наиболее ценным игроком команды и выделить тех ролевых игроков, которые оказывают огромное влияние на игру, набирая меньшее количество очков.

Решением этой проблемы может послужить web-сайт, посвящённый NBA, её игрокам, новостям и прогнозированию результатов. Благодаря ему можно проанализировать большинство аспектов игры какого-либо игрока, ознакомиться с его достижениями, проследить за «свежими» новостями вокруг лиги и просто поделиться своим мнением по поводу них. Таким образом, при помощи сайта можно будет проанализировать практически все стороны игры того или иного игрока, следить за актуальными событиями, узнавать мнение других и делиться собственным.

Данный документ представляет собой отчет по производственной практике, проходившей в период с 6 сентября по 28 ноября 2021 года в ООО «Синаптика», г. Киров.

1. СВЕДЕНИЯ О РАБОТЕ, ВЫПОЛНЕННОЙ В ПЕРИОД ПРОХОЖДЕНИЯ УЧЕБНОЙ ПРАКТИКИ

1.1 Сведения о работе, выполненной в период прохождения учебной практики

Таблица 1 – Сведения о работе, выполненной в период практики

Дата	Краткое содержание выполненных работ
06.09.2021	Прохождение инструктажа по ознакомлению с правилами внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологическими правилами и гигиеническими нормативами, а также вводный инструктаж и инструктаж на рабочем месте
07.09.2021- 14.10.2021	Разработка структур, алгоритмов в соответствии с индивидуальным заданием
14.10.2021- 27.11.2021	Программная реализация в соответствии с индивидуальным заданием
28.11.2021	Подготовить и оформить отчет

(дата)

(подпись)

1.2 Организационная структура предприятия

В настоящее время структуру организации можно описать в виде отраслей представленном на рисунке 1.

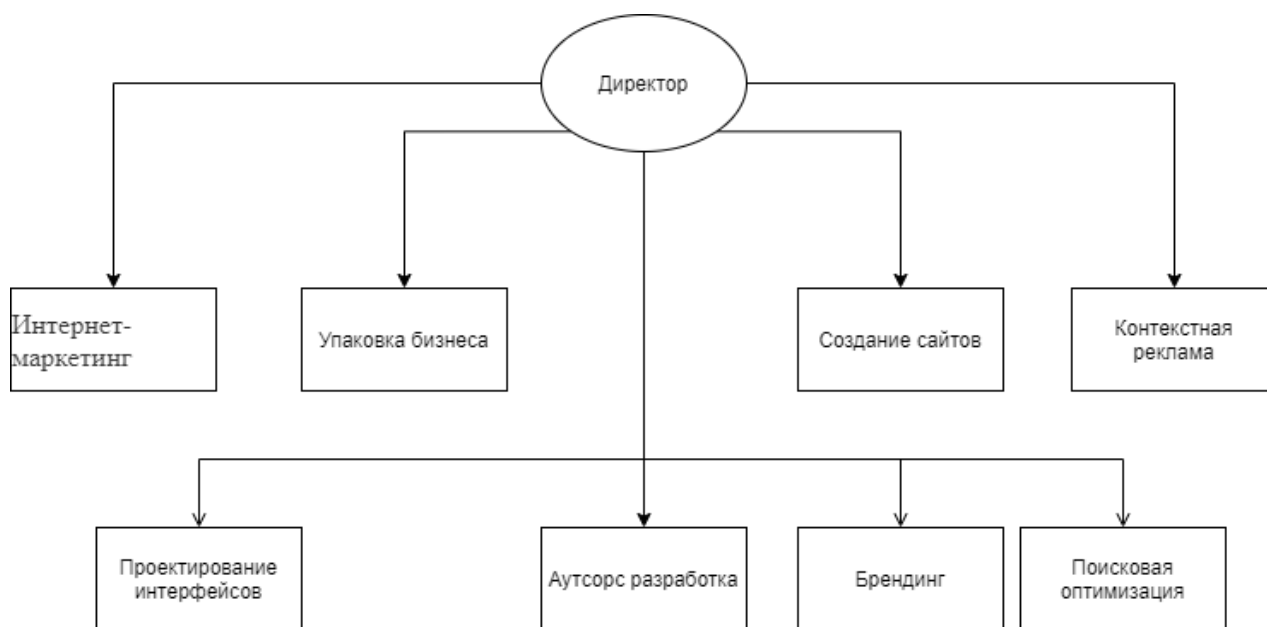


Рисунок 1 – Отрасли организации

2. ФОРМУЛИРОВКА ИНДИВИДУАЛЬНОГО ЗАДАНИЯ

2.1. Назначение системы.

Web-портал «NBA news» предназначен для дескриптивной аналитики спортивных мероприятий и сезонных показателей, включающая в себя выявление слабых и сильных сторон игроков на основе данных, изучения их биографии, просмотра ярких моментов игр и новостей лиги.

2.2. Цели создания системы.

Целью создания системы является:

- Облегчить работу аналитиков при исследовании статистики сезона НБА.
- Сбор наиболее актуальных новостей с наиболее популярных источников;
- Концентрация большого объема информации: новости, медиа, статистика, анализ, биография – в одном ресурсе;
- Русскоязычный аналог.

2.3. Краткие сведения об объекте автоматизации.

Объектом автоматизации является мужская профессиональная баскетбольная лига Северной Америки, в частности, США и Канады. Входит в четвёрку главных профессиональных спортивных лиг Северной Америки, наряду с НХЛ, МЛБ и НФЛ.

2.4. Сведения об условиях эксплуатации объекта автоматизации и характеристика окружающей среды:

- Регулярный сезон в национальной баскетбольной ассоциации начинается в середине октября и заканчивается в середине весны, затем начинается плей-офф
- Статистика данного сезона будет учитываться при выявлении лучших игроков в том или ином аспекте игры и присуждении соответствующих сезонных наград;
- Новостная лента, посвящённая НБА, обновляется каждый день, после ежедневных матчей;
- НБА состоит из 30 команд, каждая из которых имеет свою статистику и игроков;
- Ежедневные матчи влияют на статистику игроков, которая будет меняться после каждого из них.

3. ОПИСАНИЕ ВЫПОЛНЕНИЯ ИНДИВИДУАЛЬНОГО ЗАДАНИЯ

В данном разделе описаны диаграммы, необходимые для реализации проекта

3.1 Разработка диаграмм, необходимых для анализа функциональных и эксплуатационных требований к программному продукту

3.1.1. Функциональные диаграммы



Рисунок 1. Функциональная диаграмма верхнего уровня

- **Функциональная диаграмма верхнего уровня** позволяет сформировать наиболее точное и описание разрабатываемого программного обеспечения. В данном случае в качестве отображения взаимосвязей была выбрана нотация IDEF0. В качестве входных данных выступают параметры, которые пользователь может задать для формирования графика или базы данных, персональные данные, логин/пароль от аккаунта и комментарии, оставленные в разделе «Новости». В качестве субъекта может выступать пользователь и администратор, имеющий более широкий спектр прав. Управление задаётся в соответствии с техническим заданием и некоторыми особенностями НБА, отличающими его от других баскетбольных лиг, таких как длительность четверти и подсчёт рейтинга. В выходных данных будет происходить отображение, выбранных пользователем данных: график, страница об игроке или таблица со статистикой игроков. После ввода логина/пароля происходит авторизация пользователя в системе. Администратор имеет доступ к редактированию комментариев, имеющих нарушения.

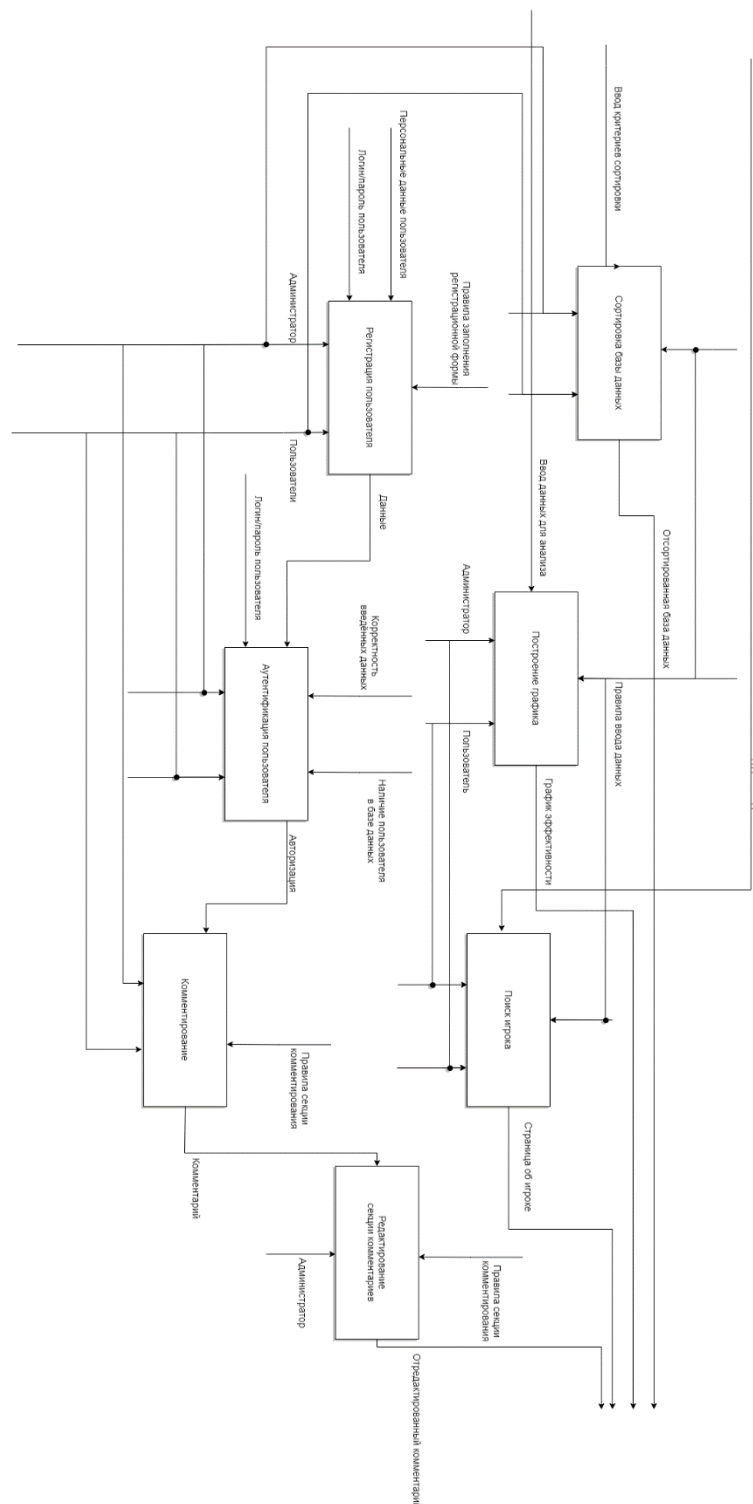


Рисунок 2. Функциональная диаграмма детализированного уровня

- Детализирующая функциональная диаграмма** более подробно раскрывает функциональную диаграмму верхнего уровня: описывает взаимодействия и связи процессов, происходящих внутри системы. На ней можно увидеть, какие процессы взаимосвязаны и что между ними общего. Так практически во всех процедурах в качестве субъекта выступает пользователь и администратор, помимо части редактирования комментариев. Процедура регистрации нового пользователя протекает в 2 этапа: ввод данных и вход в систему. Результатом таких операций, как построение графика, сортировка базы данных и вывод списка игроков на экран являются данные, которые впоследствии анализирует пользователь. После анализа все склоняется к выбору следующего действия.

```

graph TD
    Start([Исходное состояние]) --> MainMenu[Главная страница]
    MainMenu --> News[Новости]
    MainMenu --> Stats[Статистика]
    MainMenu --> Teams[Команды]
    MainMenu --> Events[События]
    MainMenu --> Profile[Профиль пользователя]

    News --> NewsList[Список новостей]
    NewsList --> NewsDetail[Детали новости]
    NewsDetail --> NewsList

    Stats --> StatsList[Список статистики]
    StatsList --> StatsDetail[Детали статистики]
    StatsDetail --> StatsList

    Teams --> TeamList[Список команд]
    TeamList --> TeamDetail[Детали команды]
    TeamDetail --> TeamList

    Events --> EventsList[Список событий]
    EventsList --> EventsDetail[Детали события]
    EventsDetail --> EventsList

    Profile --> ProfileEdit[Редактирование профиля]
    ProfileEdit --> Profile

    MainMenu --> Search[Поиск]
    Search --> SearchResults[Результаты поиска]
    SearchResults --> Search

    MainMenu --> Register[Регистрация]
    Register --> RegisterForm[Форма регистрации]
    RegisterForm --> RegisterSuccess[Успешная регистрация]
    RegisterSuccess --> Profile

    MainMenu --> Login[Вход]
    Login --> LoginForm[Форма входа]
    LoginForm --> LoginSuccess[Успешный вход]
    LoginSuccess --> Profile

    MainMenu --> Logout[Выход]
    Logout --> LogoutSuccess[Выход выполнен]
    LogoutSuccess --> Profile
  
```

Рисунок 3. Диаграмма переходов состояний

- **Диаграмма переходов состояний** используется для описания динамического поведения объекта в ходе его существования. Она наиболее точно выражает как последовательность состояний, испытываемых объектом, событий, вызывающих переходы между состояниями, и действий, сопровождающих переход состояний, происходят в системе. В любом разрабатываемом ПО есть исходное состояние, с которого начинается работа программы, в данном случае – это начальная страница web-сайта «NBA news». Наиболее подробно описаны действия и события, к которым они приводят. После завершения какого-либо события оно переходит в блок «Ожидание», в котором происходит ожидание дальнейших действий пользователя.

3.1.3. Диаграммы потоков данных

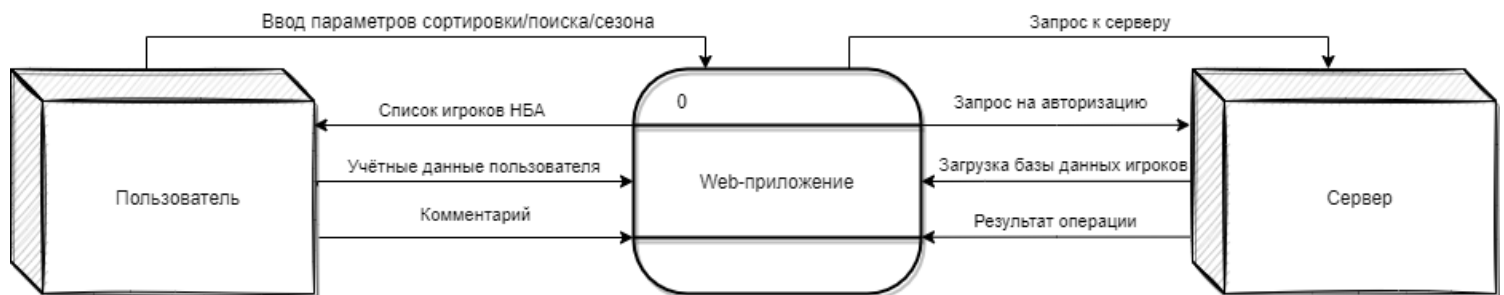


Рисунок 4. Контекстная диаграмма потоков данных

- **Контекстная диаграмма потоков данных** представляет собой иерархию функциональных процессов, связанных потоками данных. Контекстная диаграмма является диаграммой начального уровня и изображает все «заинтересованные» внешние сущности, общающиеся с системой и основные потоки данных между системой и внешним миром. В данном случае в роли сущностей будут выступать пользователь и сервер, которые взаимодействуют друг с другом путём передачи и получения информации с web-приложения. Диаграмма отображает какие запросы приложение передает серверу, а какие получает от него. Помимо этого, можно увидеть, что передает пользователь в качестве входных данных, и что он получает в ответ. Более подробное описание процессов приведена на следующей диаграмме.

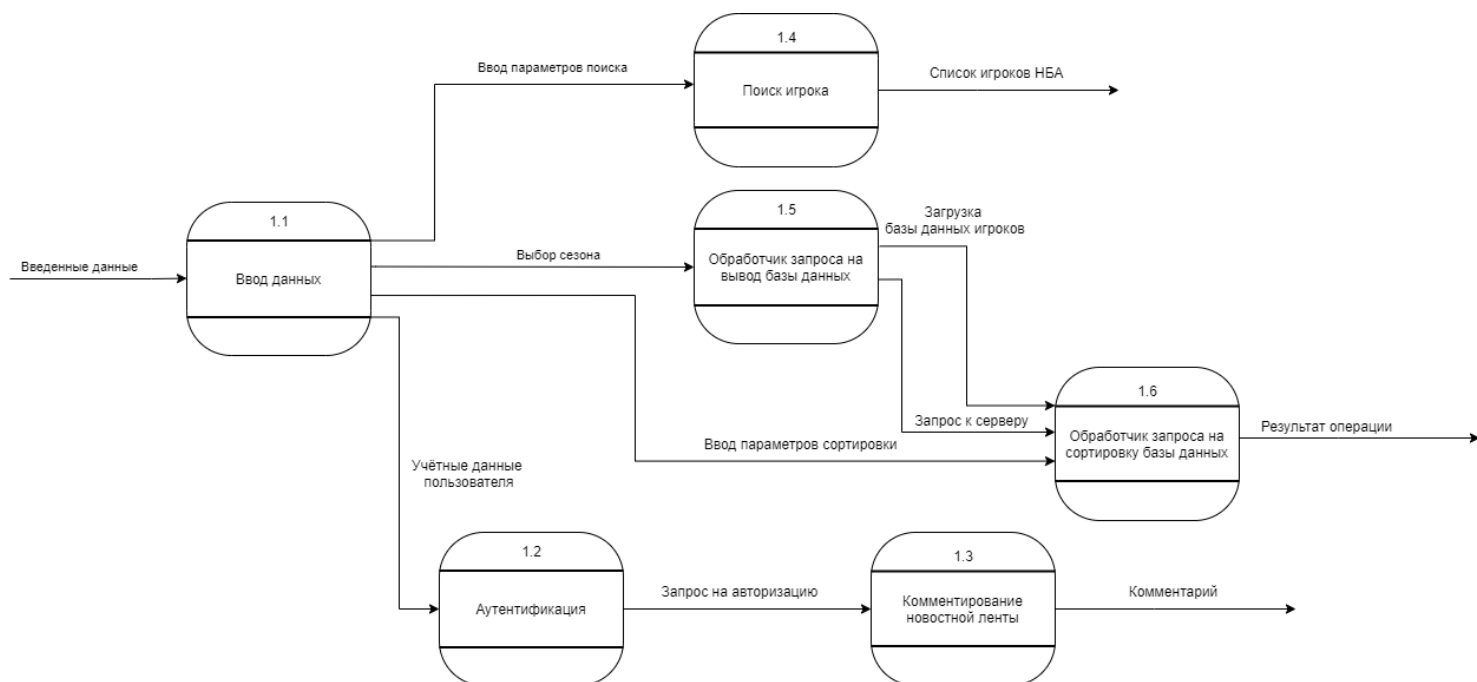


Рисунок 5. Детализированная диаграмма потоков данных

- Детализированная диаграмма потоков данных** демонстрирует, как каждый процесс преобразует свои входные данные в выходные, а также выявляет отношения между этими процессами. В данной системе все процессы связаны вводом данных: выбор сезона, ввод параметров поиска, ввод параметров сортировки, ввод учётных данных пользователя, - что позволяет установить между процессами связь и задать начальную точку иерархии системы. Использование секции комментирования невозможно без аутентификации пользователя, которая позволяет быть точно уверенным в том, кто оставил комментарий. При сортировке базы данных сначала происходит выбор команды для сортировки и загрузка базы данных игроков, и уже только потом становится возможен выбор других критериев, позволяющих наиболее точно отобразить статистику игрока в том или ином аспекте игры. Результатом любой операции в конечном счёте является какое-либо визуальное представление конечных данных.

3.1.4. Укрупнённая схема алгоритма входа на web-сайта

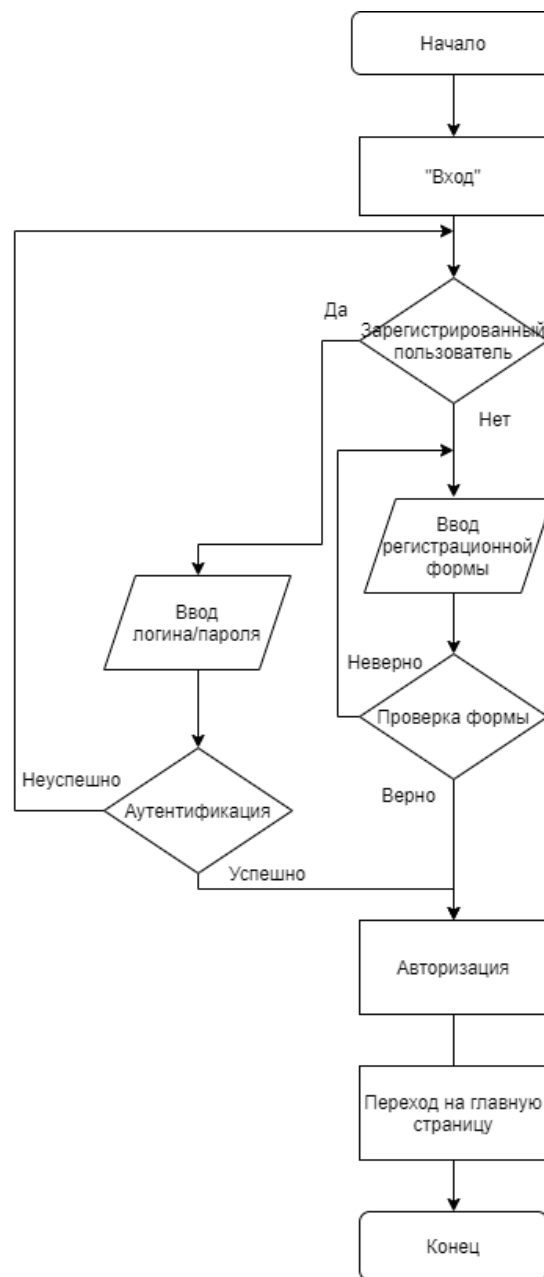


Рисунок 6. Укрупнённая схема алгоритма входа на web-сайта

- **Укрупнённая схема алгоритма** нужна для демонстрации работы алгоритма того или иного процесса. В данном случае изображён алгоритм авторизации в системе. После нажатия вкладки «Вход» пользователь имеет выбор: зарегистрироваться на сайте или войти в уже существующую учётную запись. В обоих случаях присутствует проверка на корректность введённых данных. После успешно введённых данных происходит авторизация пользователя и вход в учётную запись с переходом на главную страницу проекта.

3.2 Разработка диаграмм для объектно-ориентированного подхода к проектированию программного обеспечения

3.2.1. Диаграммы моделей взаимодействия

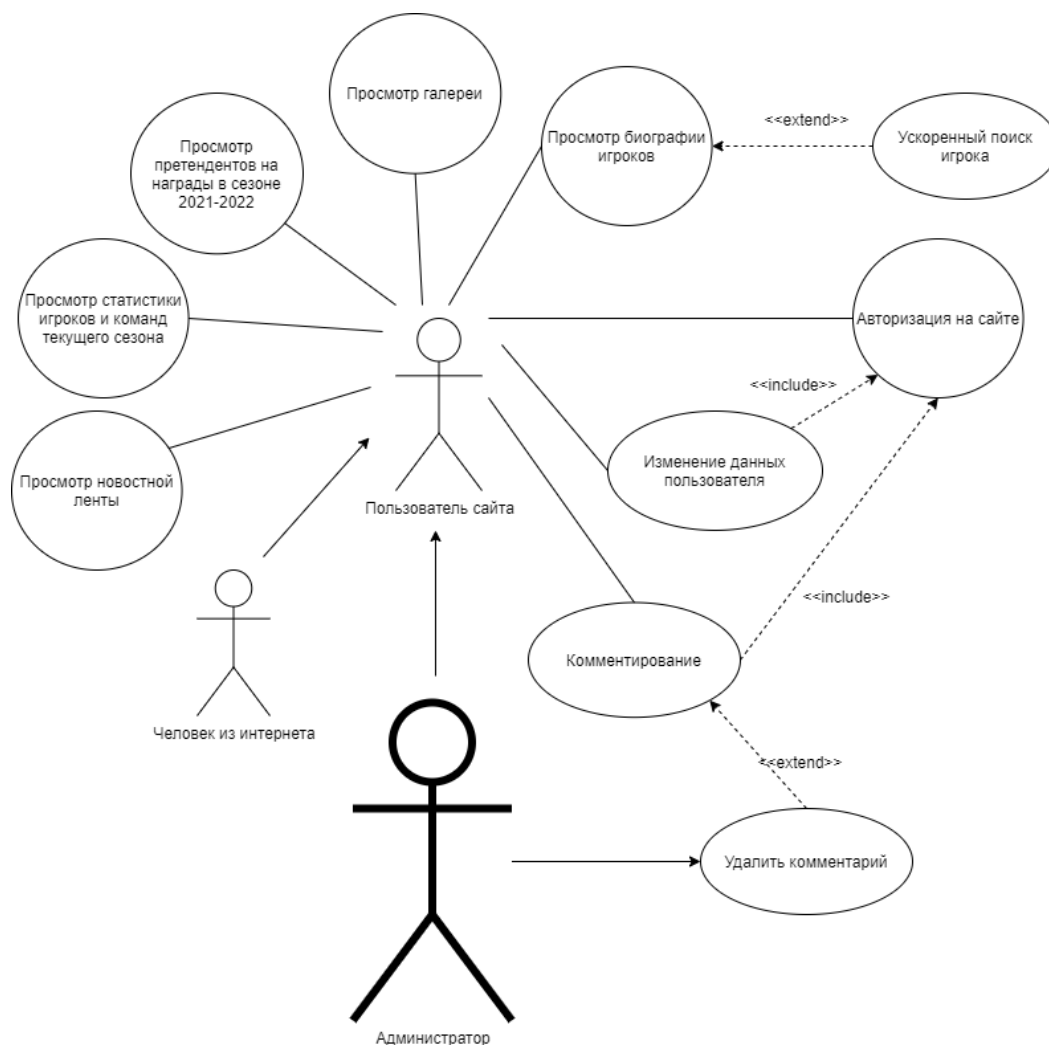


Рисунок 1. Диаграмма «use case»

- **Диаграмма «use case»** отображает отношения между актерами и вариантами использования. В диаграмме для данного проекта отображены возможности, которыми обладает как простой пользователь, так и администратор сайта. Администратор имеет возможность удаления негативных и неприемлемых комментариев. Также показана взаимосвязь таких действий, как комментирование, и его невозможность без авторизации. Благодаря меткам можно увидеть дополнительные возможности определённых страниц сайта и возможность их использования человеком.



Рисунок 2. Диаграмма кооперации

- Диаграмма кооперации** указывает обмен сообщениями между объектами при реализации любого варианта использования. При помощи данной диаграммы можно легко отследить, что происходит при выполнении каких-либо действий на сайте. Отслеживается взаимосвязь между пользователем, сервером и всеми внешними ресурсами, задействованными на сайте. Так взаимосвязь сайта и gettyimages показывает, что будет заимствовано с внешнего ресурса для отображения на сайте. При помощи взаимосвязи сайт-сервер можно увидеть, что сайте передаёт серверу и наоборот.

3.2.2. Диаграммы моделей поведения

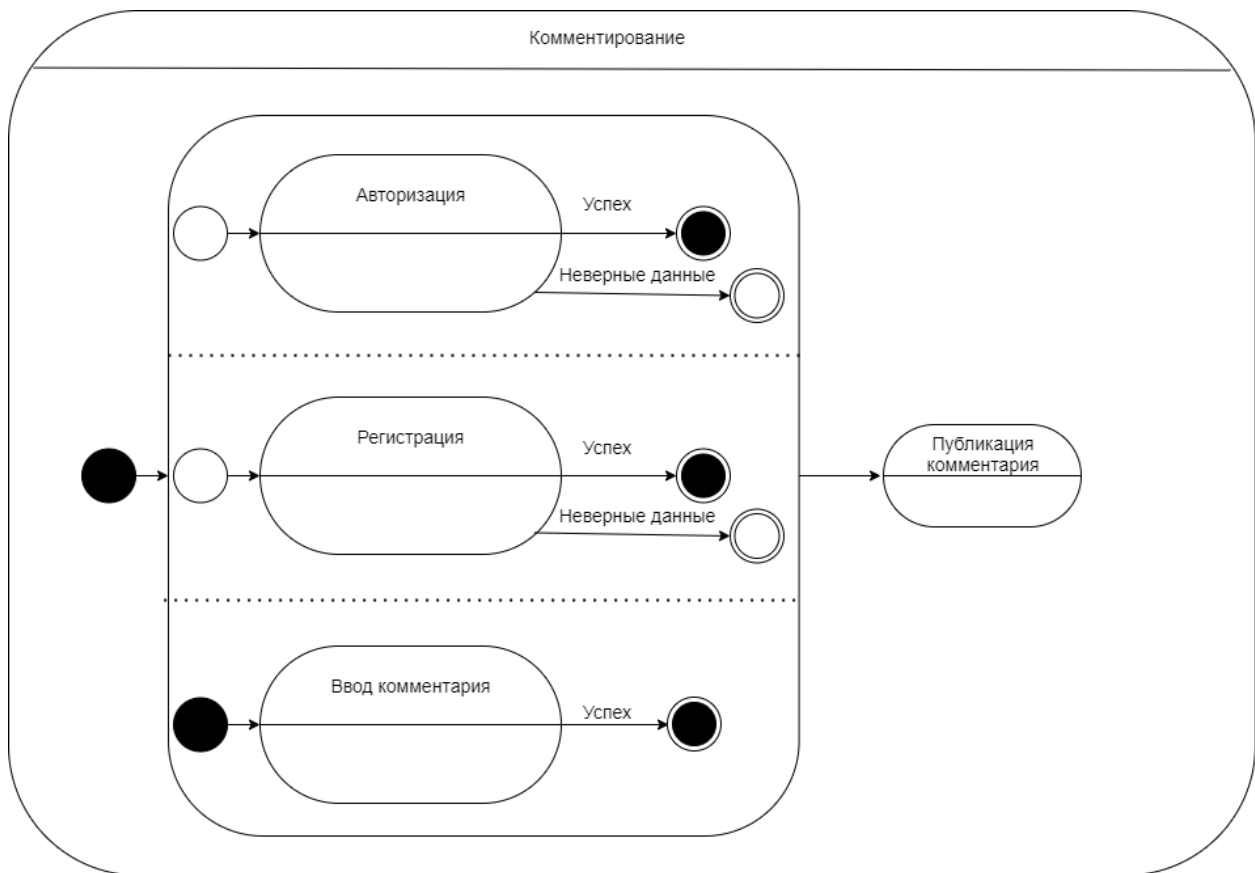


Рисунок 3. Диаграмма состояний входа на сайт

- **Диаграмма состояний** показывает, как объект переходит из одного состояний в другое. Она служит для моделирования сложных динамических аспектов системы. В качестве такого аспекта на диаграмме изображён процесс написания и публикации пользователем комментария. Здесь можно видеть составное состояние, включающее другое состояние, одно из которых содержит также параллельные подсостояния. Так при помощи данной диаграммы можно увидеть, что при неудачной авторизации на сайте у пользователя будет 2 варианты: повторить процесс авторизации или зарегистрироваться по-новому. Комментарий могут оставить только авторизованные на сайте пользователи. Лишь после успешного входа или регистрации у человека появляется возможность оставить комментарий на сайте.

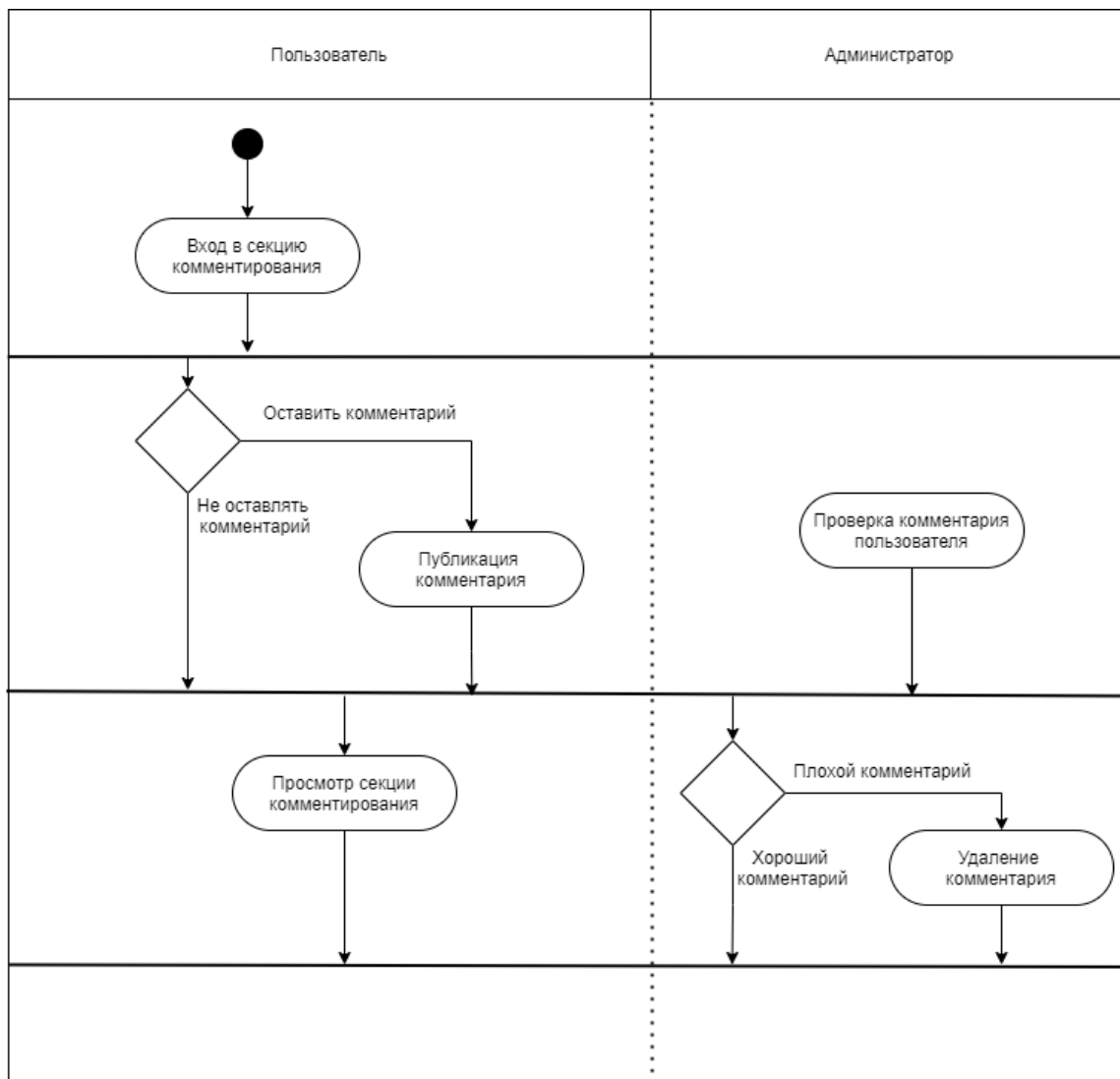


Рисунок 4. Диаграмма деятельности в секции комментирования

- Диаграмма деятельности** представляет из себя алгоритм каких-либо действий, выполняющихся в системе. В качестве действия было выбрано написание и публикация комментария, а также дальнейшие действия с ними. Благодаря диаграмме можно просмотреть как пользователь и администратор могут взаимодействовать с секцией комментирования. Так для написания комментария необходима авторизация на сайте. Администратор имеет больше возможностей в данной секции чем простой пользователь: он может удалять неприемлемые комментарии или оставлять хорошие.

3.2.3. Диаграммы статических моделей

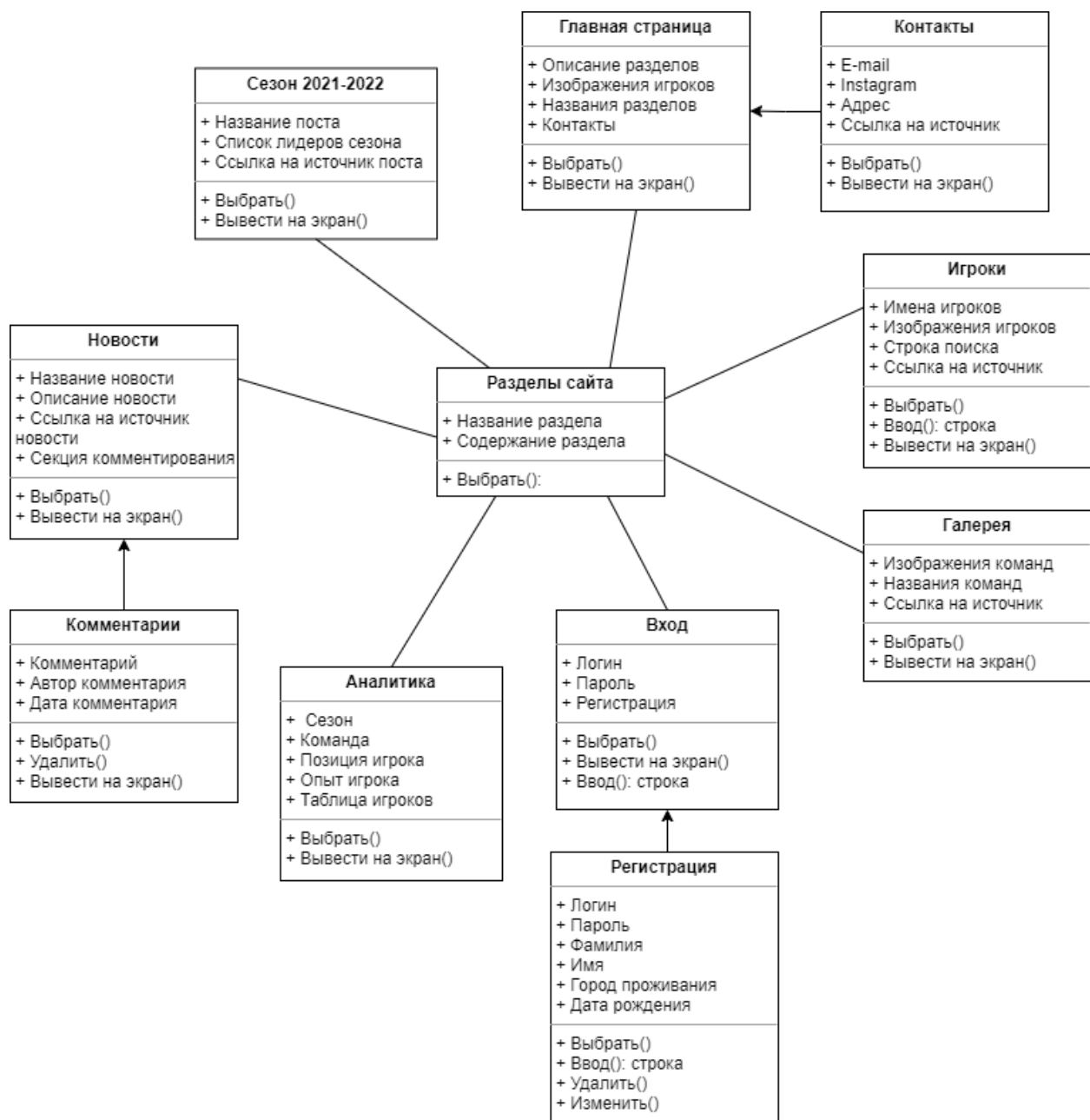


Рисунок 5. Диаграмма классов

- **Диаграмма классов** демонстрирует общую структуру иерархии классов системы, их коопераций, атрибутов, методов, интерфейсов и взаимосвязей между ними. В качестве диаграммы классов было решено описать общую структуру сайта: страницы и их функции. При помощи входных стрелок можно отследить возможности каждой страницы и дополнительные функции, доступные для использования на странице. Так пункт регистрации доступен лишь во вкладке «Вход» или комментирование есть лишь в разделе «Новости».

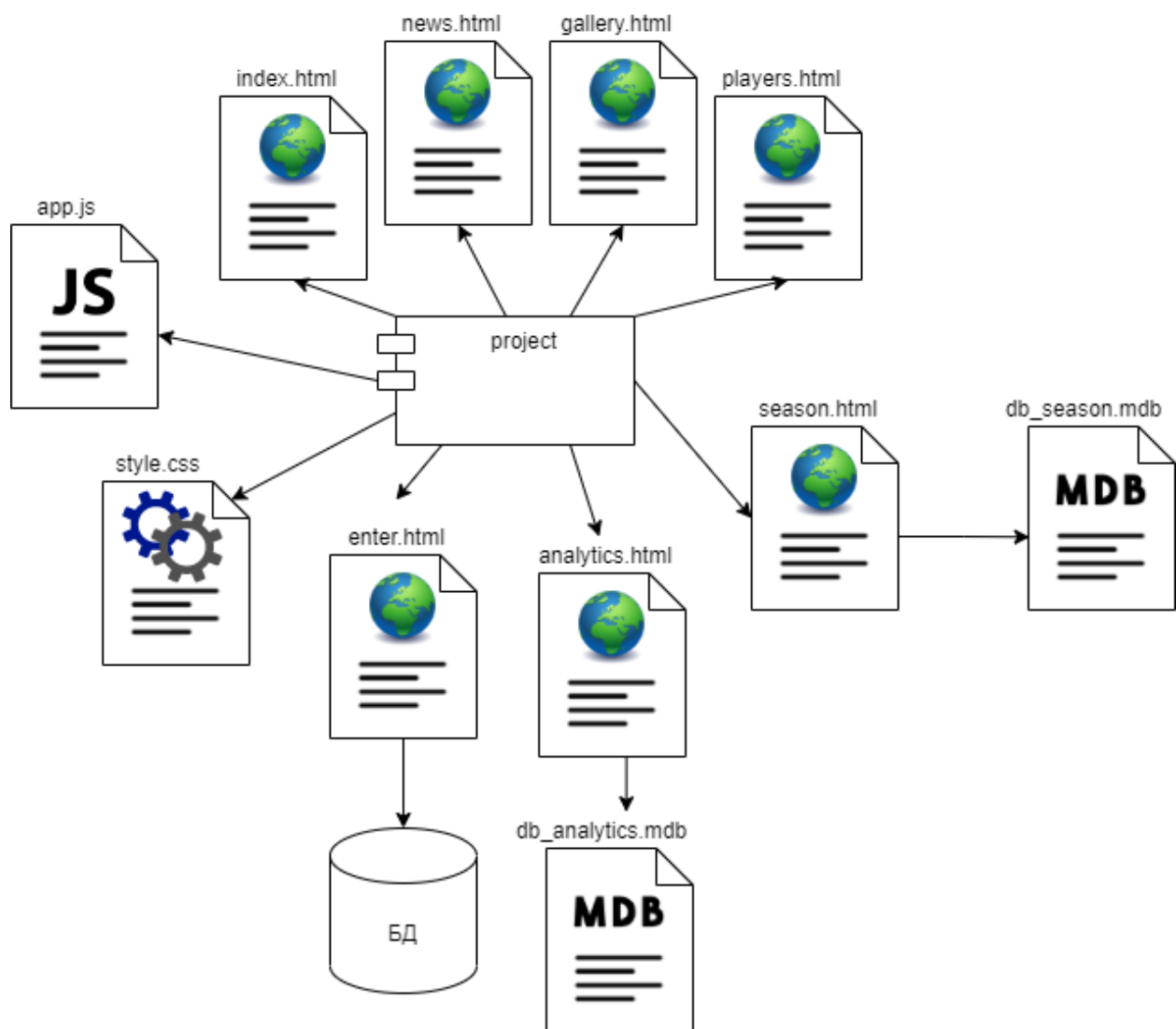


Рисунок 6. Диаграмма реализации

- **Диаграмма реализации** в проекте нужна для визуализации организации компонентов системы и зависимостей между ними. Так в данном проекте project является корневой папкой проекта, в которой располагаются все необходимые файлы. Большую часть сайта занимают html файлы, представляющие страницы главного проекта. База данных пользователей используется при поиске человека на сайте при авторизации. Файлы mdb нужны для отображения и загрузки статических баз данных с игроками и их статистикой. Они доступны во вкладках сезон и аналитика. Также для визуальной составляющей необходимы файлы app.js и style.css, которые задают внешний вид сайта и придают ему динамичность.

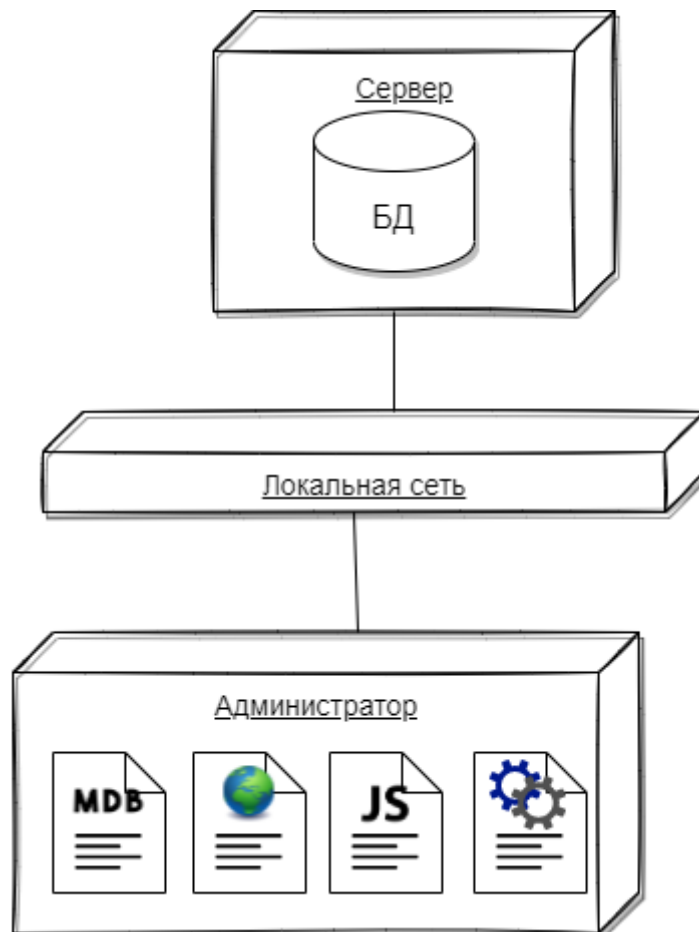


Рисунок 7. Диаграмма размещения

- **Диаграмма размещения** отражает физические взаимосвязи между программными и аппаратными компонентами системы, а также их расположение в системе. В данном проекте в качестве хранилища задействован хост, локальная сеть. В нем хранятся все необходимые для нормального функционирования сайта файлы. В качестве источников данных файлов выступают сервер, который хранит базу данных пользователей, и администратор, который выкладывает в локальную сеть файлы уже самого проекта, описанные в диаграмме компонентов.

3.3. Реализация алгоритмов.

Код модулей предоставлен в Приложении А и является самодокументированным.

ЗАКЛЮЧЕНИЕ

В ходе производственной практики был получен опыт по разработке прототипов пользовательского интерфейса с последующей реализации. Также была освоена работа со структурой web-разработки. При помощи изучения html, css и JavaScript были разработаны структура, дизайн и внутренние составляющие сайта.

Была рассмотрена предметная область, определены основные задачи, которые необходимо было реализовать, также была выбрана система для работы с базами данных, освоены стандартные SQL запросы и проведены работы над ошибками, возникшими в ходе тестирования.

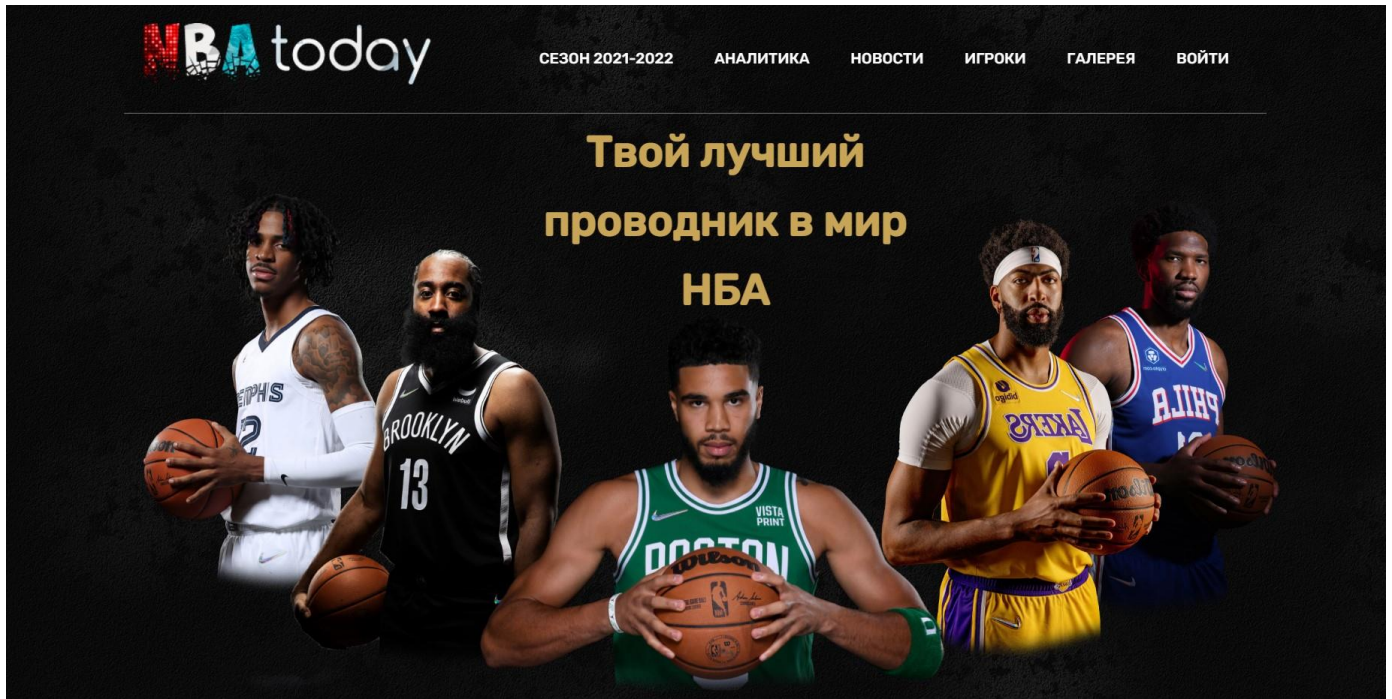
БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Документация JavaScript -[<https://developer.mozilla.org/ru/docs/Web/JavaScript>]
2. Документация HTML - [<https://developer.mozilla.org/ru/docs/Web/HTML>]

ПРИЛОЖЕНИЯ

Приложение А

Экранные формы



Приложение В

Конструктор главной страницы

```
<!DOCTYPE html>
<html></!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>NBA news</title>
    <link href="https://fonts.googleapis.com/css2?family=Rubik:wght@500&display=swap" rel="stylesheet">
    <link rel="shortcut icon" href="assets/images/icon.png">
    <link rel="stylesheet" href="assets/css/style.css">
    <link rel="stylesheet" href="assets/js/app.js">
  </head>
  <body>
    <section class="banner">
      <header id="header" class="header">
        <!-- <div class="container"> -->
        <div class="header__flex">
          <a class="logo" href="#"></a>
          <nav class="nav">
            <a class="nav_link" href="#">Сезон 2021-2022</a>
            <a class="nav_link" href="#">Аналитика</a>
            <a class="nav_link" href="#">Новости</a>
            <a class="nav_link" href="#">Игроки</a>
            <a class="nav_link" href="#">Галерея</a>
            <a class="nav_link" href="#">Войти</a>
          </nav>
        </div>
        <!-- </div> -->
      </header>
```

```
<div class="container">
  <div class="intro">
    
    
    
    
    
    <div class="intro_text">
      <h1>Твой лучший проводник в мир НБА</h1>
      <!-- <p class="second">Не правда ли?</p> -->
    </div>
  </div>
</div>
</section>

<!-- <section class="menu"> -->

<section class="news">
  <div class="container_blocks">
    
    
  </div>
</section>

<section class="season">
  <div class="container_blocks">
    
    
  </div>
</section>

<section class="analitic">
  <div class="container_blocks">
    
    
  </div>
</section>

<section class="players">
  <div class="container_blocks">
    
    
  </div>
</section>

<section class="gallery">
  <div class="container_blocks">
    
    
  </div>
</section>

<section class="end">

</section>
<!-- </section> -->
<script type="text/javascript">

</script>
</body>
</html>>
```