



**МИНОБРАЗОВАНИЯ РОССИИ**  
**федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**«Московский государственный технологический университет «СТАНКИН»**  
**(ФГБОУ ВО «МГТУ «СТАНКИН»)**

---

**Институт информационных  
систем и технологий**

**Кафедра информационных  
технологий и вычислительных  
систем**

**ОТЧЕТ**

**О ПРОХОЖДЕНИИ** Производственной (проектно-технологической) ПРАКТИКИ

**ЗА** 4 **СЕМЕСТР**

**ОБУЧАЮЩЕГОСЯ** 2 **КУРСА** магистратуры **ГРУППЫ** ИДМ-22-08

---

**Филатовой Ольги Владимировны**

---

**КАФЕДРА**

Информационных технологий и вычислительных  
систем (ИТиВС)

---

**НАПРАВЛЕНИЕ ПОДГОТОВКИ**

09.04.04 «Программная инженерия»

---

**МЕСТО ПРОХОЖДЕНИЯ  
ПРАКТИКИ**

Кафедра информационных технологий и  
вычислительных систем ФГБОУ ВО «МГТУ  
«СТАНКИН»

---

**СРОКИ ПРОХОЖДЕНИЯ  
ПРАКТИКИ**

12.02.2024 г. – 11.04.2024 г.

---

**РУКОВОДИТЕЛИ ПРАКТИКИ:**

**ОТ УНИВЕРСИТЕТА**

Тарасов Алексей Геннальевич, к.т.н., доцент

---

**ОТ ОРГАНИЗАЦИИ**

Новоселова Ольга Вячеславовна., к.т.н., доцент,  
зав. кафедрой ИТиВС

---

МОСКВА 2024 г.



**МИНОБРНАУКИ РОССИИ**  
**федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**«Московский государственный технологический университет «СТАНКИН»**  
**(ФГБОУ ВО «МГТУ «СТАНКИН»)**

---

**ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ ДЛЯ ПРОХОЖДЕНИЯ**

**Производственной практики (проектно-технологическая) в 4-ом семестре**

---

**ОБУЧАЮЩЕГОСЯ** 2 **КУРСА**                      **магистратуры**                      **ГРУППЫ** ИДМ-22-08

---

**Филатовой Ольги Владимировны**

---

<b>КАФЕДРА</b>	<b>Информационных технологий и вычислительных систем (ИТиВС)</b>
----------------	--

---

<b>НАПРАВЛЕНИЕ ПОДГОТОВКИ</b>	<b>09.04.04 «Информатика и вычислительная техника»</b>
-------------------------------	--

---

<b>МЕСТО ПРОХОЖДЕНИЯ ПРАКТИКИ</b>	<b>Кафедра информационных технологий и вычислительных систем ФГБОУ ВО «МГТУ «СТАНКИН»</b>
-----------------------------------	---

---

<b>СРОКИ ПРОХОЖДЕНИЯ ПРАКТИКИ</b>	<b>12.02.2024 г. – 11.04.2024 г.</b>
-----------------------------------	--------------------------------------

---

**РУКОВОДИТЕЛИ ПРАКТИКИ:**

<b>ОТ УНИВЕРСИТЕТА</b>	<b>Тарасов Алексей Геннадьевич, к.т.н., доцент</b>
------------------------	--

---

<b>ОТ ОРГАНИЗАЦИИ</b>	<b>Новоселова Ольга Вячеславовна., к.т.н., доцент, зав. кафедрой ИТиВС</b>
-----------------------	--

---

- ### 3. План-график выполнения работ:

10	Сдача отчета	09.04.2024 – 11.04.2024
----	--------------	-------------------------

Филатова Ольга Владимировна

(подпись)

Тарасов Алексей Геннадьевич,

(ФИО полностью, должность)

(подпись)

## Содержание

Введение .....	5
Разработка макета клиентской части веб-приложения.....	6
Анализ и сравнение инструментов для разработки клиентской части веб-приложения .....	7
Angular .....	7
Vue.js .....	7
React JS.....	8
Разработка клиентской части веб-приложения с помощью React JS .....	8
Заключение.....	13
Список литературы.....	14

## Введение

Любое веб-приложение разделяется на 2 части – клиентскую и серверную. Серверная отвечает за логику, но клиентская является не менее важной составляющей. Продуманный UX/UI дизайн значительно повышает лояльность пользователей. Приложения для распознавания заимствований в программной коде также нуждается в современном и удобном дизайне.

Широко используемые инструменты программирования, такие как фреймворки, позволяют оптимизировать создание информационного ресурса, обладающего современным функционалом, высокой скоростью загрузки, адаптированным к мобильным устройствам и т.п [1]. Поэтому важным аспектом в веб-разработке является правильный выбор технологии создания клиентской части веб-приложения.

По результатам исследования ресурса Stack Overflow 2023 и рейтингу крупнейший платформы для хостинга проектов и их совместной разработки GitHub, JavaScript является наиболее часто используемым языком уже 8-й год подряд, и его выбрали 69,9% разработчиков. На сегодняшний день среди JavaScript-фреймворков самыми востребованными являются Angular, React и Vue. Эти инструменты решают задачу синхронизации пользовательского интерфейса и состояния системы, а также позволяют ускорить рабочий процесс, повысить производительность системы без ущерба для качества или функциональности, сохраняя при этом гибкость в разработке.

## Разработка макета клиентской части веб-приложения

Figma - это мощный инструмент для создания интерфейсов, который позволяет дизайнерам и разработчикам работать вместе над проектами в реальном времени. Figma сохраняет макеты в облаке, что означает, что они доступны из любого места и любого устройства с подключением к интернету. Это удобно, когда вам нужно поделиться макетом с коллегами или клиентами или работать удаленно. Figma позволяет создавать компоненты и стили, которые могут быть повторно использованы на разных страницах и в разных проектах. Это сокращает время разработки и обеспечивает консистентность дизайна по всему приложению.

В целом, Figma предлагает мощный и гибкий инструментарий для разработки макетов веб-приложений. Он облегчает совместную работу, повышает эффективность команды и помогает создавать профессиональные и интерактивные макеты, которые могут быть легко переданы разработчикам для реализации

Именно поэтому было решено разрабатывать дизайн-макет клиентской части именно в Figma.

Первым шагом было исследование существующих веб-приложений. Это нужно для того, чтобы проанализировать типовые дизайны для различных аудиторий, подумать, что можно в них улучшить. Макет для приложения антиплагиата должен отвечать запросам преподавателей и студентов, обладать строгим, но современным дизайном. По сравнению с другими подобными приложениями (MOSS, JPlag) новое должно иметь форму для загрузки файлов и форму авторизации.

Важным моментом в процессе создания любого дизайна является брендбук. Брендбук – это общий стиль, концепция дизайна, в нем представлены утвержденные шрифты, цвета, формы и элементы, соответствующие имиджу компании.

Было принято решение придерживаться утвержденного брендбука МГТУ «Станкин», но добавить собственные цветовые (и другие) решения для уникальности повышения удобства.

Из брендбука взято:

- Шрифты – New\_Zeleck и Arial
- Цвета: #D70F3D, #1818B7
- Логотип

Собственные дизайн-решения:

- Окрашивание результатов проверки в цвета: #D9EED1 и # FFEEDB в зависимости от результата
- Цвет для кнопок перехода на следующее окно - #0C1E33
- Использование карточек разных цветов для разграничения групп отчетов

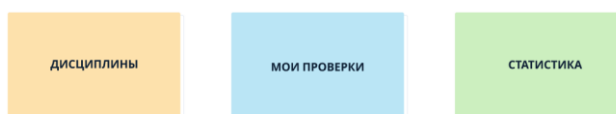


Рис. 1. Карточки отчетов в личном кабинете

## **Анализ и сравнение инструментов для разработки клиентской части веб-приложения**

### **Angular**

В настоящее время общая производительность пользовательских устройств увеличивалась, поэтому стало возможным выполнение логики приложения в браузере. Это привело к подходу создания одностраничных приложений или SPA, для разработки которых назначался AngularJS. Он также предоставлял возможность заменить низкопроизводительный jQuery, предлагая разработчикам такие функции, как двусторонняя привязка данных и возможность организации модулей для импорта внешних скриптов. Затем AngularJS был полностью переписан под Angular, соответственно изменилось множество основных концепций фреймворка. В Angular отсутствуют контроллеры или модели представления, вместо этого есть компоненты, которые состоят из шаблона, классов и метаданных. В то время как AngularJS был сосредоточен на шаблоне MVC, Angular полностью полагается на иерархию компонентов.

Важным нововведением фреймворка является TypeScript, который добавляет в проект строгую типизацию, а также различный новый функционал, такой как декораторы, еnumы и т.д. Впоследствии TypeScript компилируется в JavaScript для работы в браузере.

Angular, также как React и Vue, основан на компонентах, которые отображают информацию и шаблоны, выполняют действия с данными и состоят из трех отдельных файлов: HTML-файл для шаблона, CSS-файл для стилизации и TS-файл для контроля. Следуя этому подходу, реализуется принцип разделения проблем, а также код становится более организованным и структурированным. Внутри компонентов обмен данными между представлением (шаблоном HTML) и моделью (файлом TypeScript) происходит за счет привязки данных и событий. В Angular реализована собственная система маршрутизации для удобства использования SPA. Определив маршруты в отдельном файле или модуле, Angular обрабатывает логику того, какой компонент должен отображаться в зависимости от текущего активного URL-пути.

### **Vue.js**

Vue описывают, как прогрессивный фреймворк, принципы проектирования которого частично взяты из паттерна Model-View-ViewModel. Согласно официальному сайту, фреймворк можно использовать для разработки SPA и небольших проектов в сочетании с другими библиотеками, аналогично React. Все компоненты Vue очень похожи на Custom Elements, которые являются частью спецификации веб-компонентов и позволяют создавать новые теги HTML или изменять существующие, дополняя их функциями с использованием JavaScript, HTML и CSS.

Масштабируемость – одно из главных достоинств данного фреймворка среди других технологий. Vue надежно работает во всех поддерживаемых браузерах. Кроме того, компоненты фреймворка добавляют функциональность, которую не могут обеспечить нативные элементы, например, передачу настраиваемых событий и межкомпонентный поток данных.

Структурирование Vue предполагает, что каждый компонент должен иметь единственный корневой элемент. Функции Vue отображают Virtual DOM. Фреймворк устанавливает минимальный набор компонентов для необходимого количества манипуляций с DOM-деревом.

Фреймворк предоставляет официальный интерфейс командной строки для улучшения разработки, особенностью которого является тот факт, что команда разработчиков предлагает набор предварительно настроенных шаблонов для различных вариантов использования.

## React JS

React – это внешняя библиотека JavaScript для создания пользовательских интерфейсов веб-приложений. Самыми популярными приложениями, использующие React, являются Instagram и WhatsApp. React часто используется в веб-разработке, однако его ядро является отдельной библиотекой и используется в различных применениях, включая нативные приложения iOS и Android.

Поскольку React отвечает только за часть представления приложения, процесс разработки требует таких технологий, как компилятор Babel, модули и соответствующий загрузчик для структуры приложения, процесс сборки, маршрутизация и управление состоянием. С помощью компилятора могут быть использованы новые языковые функции еще не реализованных спецификаций ECMAScript. Что касается модулей и сборки, то Webpack - одно из лучших решений. Для управления состоянием используется Redux либо Mobx. Маршрутизация может выполняться с помощью React Router, основанном на тех же принципах, что и Ember-Router. Основной целью фреймворка является преобразование определенного состояния приложения в представление, которое может отображаться в браузере.

Одной из особенностей React, является возможность использовать расширение JSX, чтобы дополнить код JavaScript фрагментами HTML. Также одной из основных функций фреймворка является построение собственной виртуальной модели DOM-дерева. С помощью элементов ReactElements React повторно визуализирует полный пользовательский интерфейс компонента и генерирует операции, которые выполняются в реальном DOM. Так как операции запроса к реальному DOM-дереву HTML документа очень медленные и ресурсоемкие, React позволяет минимизировать нагрузки на нативную DOM, что является основой его быстродействия.

## Разработка клиентской части веб-приложения с помощью React JS

В ходе реализации макета разработано несколько страниц клиентской части веб-приложения. Например, окно авторизации отвечает за доступ к приложению и выбор роли. Решено, что логины студентов будут соответствовать номеру зачетной книжки.

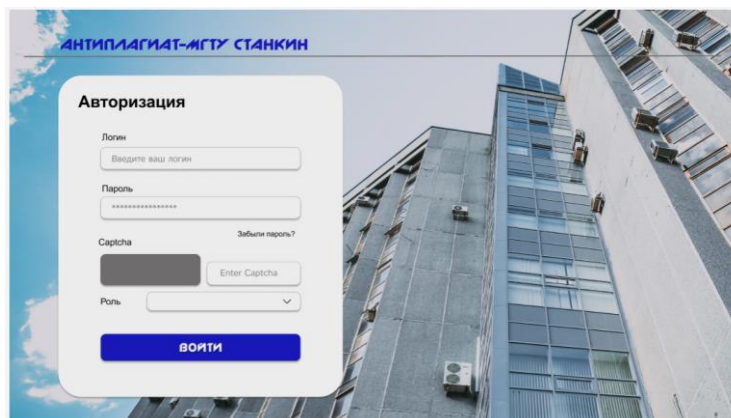


Рис 2. Окно авторизации



Также разработаны личный кабинет студента и преподавателя с разницей в некотором функционале, например:

- Количество доступных отчетов
- Статистические данные
- Возможность создания семинаров

АНТИПЛАГИАТ - МГТУ СТАНКИН

Здравствуйте, Ольга!

Загрузите файл для проверки

Это drag and drop форма, просто положите файл сюда

ОБЗОР

Доступные отчеты

ДИСЦИПЛИНЫ

МОИ ПРОВЕРКИ

СТАТИСТИКА

Последние проверки

Студент	Группа	Дата	Время	Дисциплина	Результат
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	10:00	СМИБ	90%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	02:00	СМИБ	90%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	01:00	АПОиАБС	30%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	09:45	АПОиАБС	30%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	11:15	АПОиАБС	30%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	02:15	АПОиАБС	30%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	2:00	АПОиАБС	30%

Подозрения

Иван Иванов

Системы менеджмента информационной безопасности

Иван Иванов

Системы менеджмента информационной безопасности

Иван Иванов

Системы менеджмента информационной безопасности

Иван Иванов

Системы менеджмента информационной безопасности

Иван Иванов

Системы менеджмента информационной безопасности

Рис. 3. Личный кабинет преподавателя

АНТИПЛАГИАТ - МГТУ СТАНКИН

Здравствуйте, Ольга!

Загрузите файл для проверки

Это drag and drop форма, просто положите файл сюда

ОБЗОР

Доступные отчеты

ДИСЦИПЛИНЫ

СТАТИСТИКА

Последние проверки

Студент	Группа	Дата	Время	Дисциплина	Результат
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	10:00	СМИБ	90%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	02:00	СМИБ	90%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	01:00	АПОиАБС	30%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	09:45	АПОиАБС	30%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	11:15	АПОиАБС	30%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	02:15	АПОиАБС	30%
Петр Петрович Петров	ИДМ-ИИ-44	12.01.2024	2:00	АПОиАБС	30%

Подозрения

Иван Иванов

Системы менеджмента информационной безопасности

Иван Иванов

Системы менеджмента информационной безопасности

Иван Иванов

Системы менеджмента информационной безопасности

Иван Иванов

Системы менеджмента информационной безопасности

Иван Иванов

Системы менеджмента информационной безопасности

Рис. 4. Личный кабинет студента

9

Реализован переход в карточку «Дисциплины» в ЛК преподавателя, откуда доступна статистика по предметам, а также доступен функционал перехода к семинарам по каждому предмету.

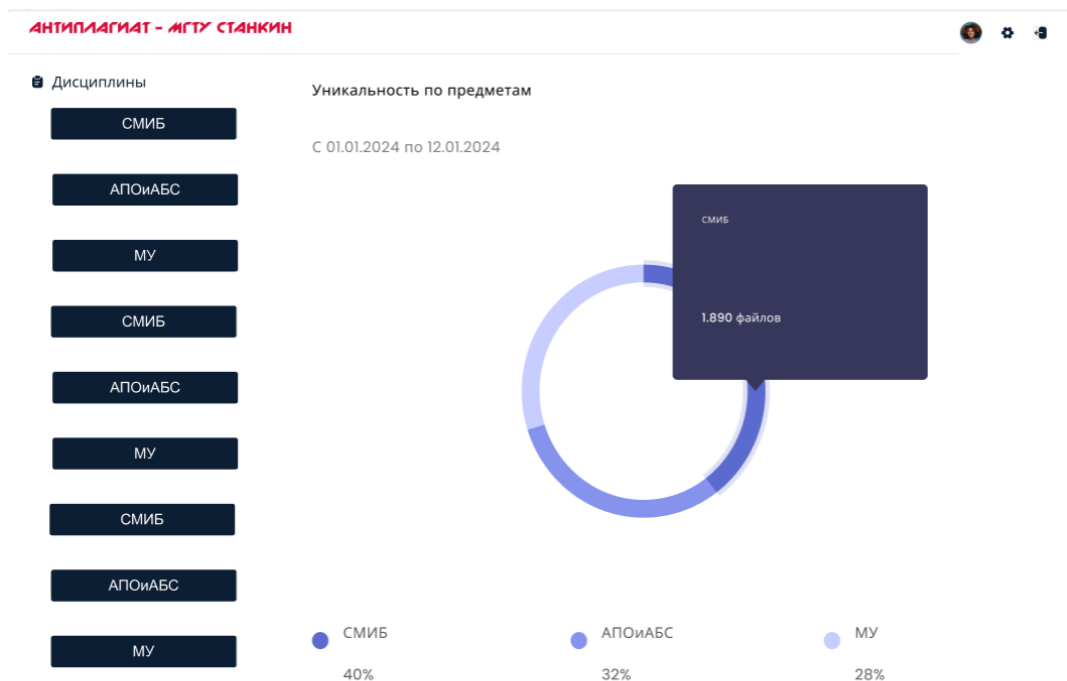


Рис. 5. Окно выбора предмета

Также преподаватель может выбрать нужный семинар или создать новый

АНТИПЛАГИАТ - ИГТУ СТАНКИН

СМИБ

Семинар 1

Семинар 1

Семинар 2

Семинар 3

Семинар 2

Семинар 5

Семинар 7

Семинар 9

Семинар 5

Создать новый семинар

Название семинара

СОЗДАТЬ

Рис. 6. Окно выбора и создания семинара

В конечном итоге преподаватель видит статистику по проверке файлов студентов, которым доступен выбранный семинар выбранного предмета. В таблице также указан референс – файл студента, признанный системой как оригинальный.

АНТИПЛАГИАТ - ИГТУ СТАНКИН

СМИБ > Семинар 1

Студент	Группа	Оригинал	Группа оригинала	Статус	Результат
Петр Петрович Петров	ИДМ-П-44	Иван Иванович Иванов	ИДМ-34-76	Подтверждено	90%
Петр Петрович Петров	ИДМ-П-44	Иван Иванович Иванов	ИДМ-34-76	Не подтверждено	90%
Петр Петрович Петров	ИДМ-П-44	Иван Иванович Иванов	ИДМ-34-76	Не подтверждено	30%
Петр Петрович Петров	ИДМ-П-44	Иван Иванович Иванов	ИДМ-34-76	Подтверждено	30%
Петр Петрович Петров	ИДМ-П-44	Иван Иванович Иванов	ИДМ-34-76	Не подтверждено	30%
Петр Петрович Петров	ИДМ-П-44	Иван Иванович Иванов	ИДМ-34-76	Не подтверждено	30%
Петр Петрович Петров	ИДМ-П-44	Иван Иванович Иванов	ИДМ-34-76	Не подтверждено	30%

Рис. 7. Окно просмотра результатов студентов

Путь студентов к результатам проверок немного отличается – студент не может создавать новые семинары, а также видит статистику в виде пузырьковой диаграммы.

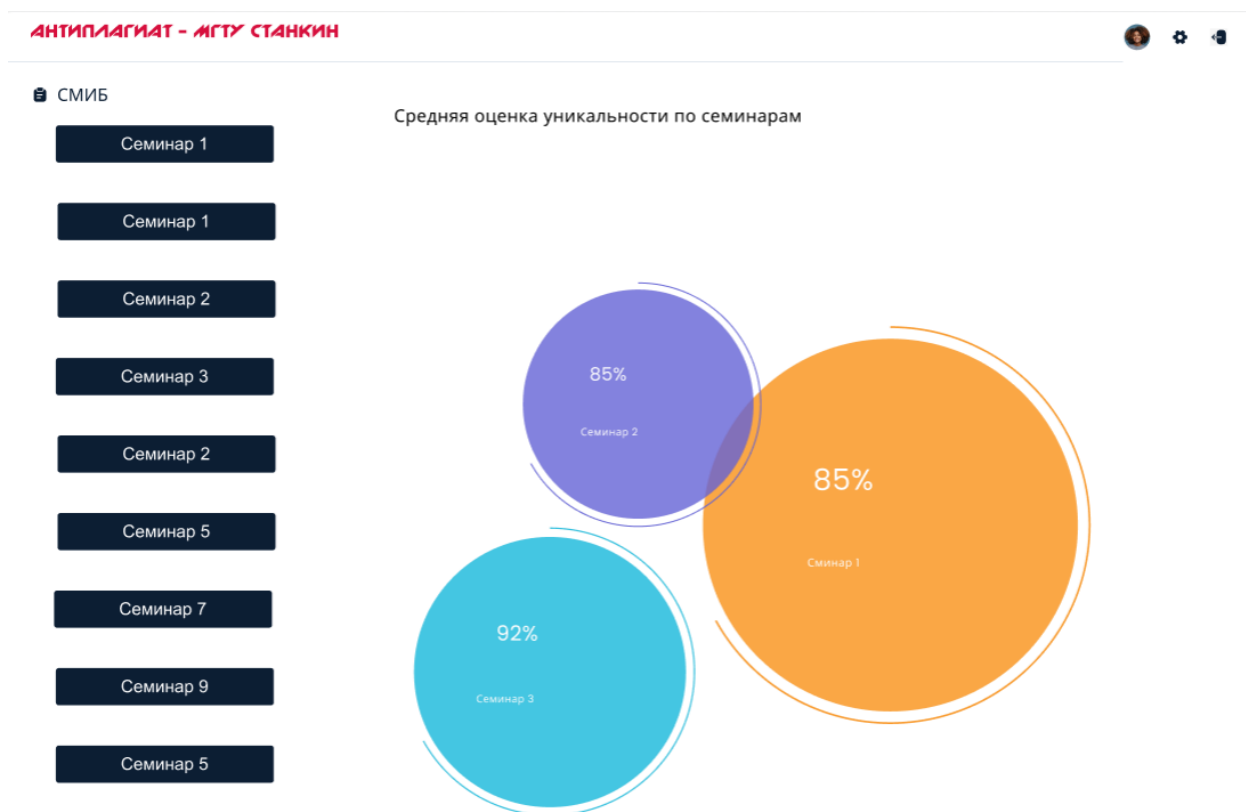





Рис. 8. Окно выбора семинара в личном кабинете студента

Также отличается окно просмотра результатов – студент не видит референс, зато может загрузить новый файл на проверку. Тогда этот файл будет привязан к выбранному семинару. Если же осуществлять загрузку с главной страницы – привязки не будет и документ будет сравниваться с общим пулом программ.


АНТИПЛАГИАТ - МГТУ СТАНКИН




СМИБ > Семинар 1

Загрузите файл для проверки

Это drag and drop форма, просто положите файл сюда




ОБЗОР



Имя файла	Дата	Время	Статус	Результат
Файл 1	12.01.2024	10:00	Подтверждено	90%
Файл 1	12.01.2024	02:00	Не подтверждено	90%
Файл 1	12.01.2024	01:00	Подтверждено	30%
Файл 1	12.01.2024	09:45	Не подтверждено	30%
Файл 1	12.01.2024	11:15	Не подтверждено	30%
Файл 1	12.01.2024	02:15	Не подтверждено	30%
Файл 1	12.01.2024	2:00	Не подтверждено	30%

Рис. 9. Окно просмотра результатов в личном кабинете студента

И роли «Преподаватель», и роли «Студент» доступно редактирование профиля

АНТИПЛАГИАТ - МГТУ СТАНКИН




### Изменить пароль

Логин

st567345

Введите старый пароль

\*\*\*\*\*

Введите новый пароль

\*\*\*\*\*

Забыл пароль

Сохранить

Рис. 10. Окно смены пароля

**АНТИПЛАГИАТ - ИГТУ СТАНКИН**

### Редактирование профиля

Фамилия

Борисов

Имя

Александр


Отчество

Олегович

Группа

ИДБ-67-98

Сохранить



Загрузить фотографию

Рис. 11. Окно редактирования личной информации

## Заключение

В настоящее время JavaScript-фреймворки предлагают полный стек полезных функций и отвечают за архитектурные решения в процессе разработки, включают в себя стратегии для маршрутизации URL-адресов в приложении, управления состоянием, объединения и другое.

Фреймворки React и Vue, использующие Virtual DOM и компонентную структуру, сосредоточены на создании пользовательского интерфейса. Angular используется в крупных коммерческих приложениях с большой функциональной нагрузкой, но в сравнении с React и Vue проявляет меньшую гибкость в разработке, предлагая четко следовать своим правилам. React превосходит Vue в разнообразии доступных средств отрисовки интерфейса пользователя, в то время как Vue в большей мере основывается на классических веб-технологиях.

В результате с помощью библиотеки React JS была разработана клиентская часть приложения для автоматизированного выявления заимствований в программных кодах.

Практикант \_\_\_\_\_ Филатова Ольга Владимировна \_\_\_\_\_

Отчёт о прохождении практики сдан « 13 » апреля 2024г.

Оценка \_\_\_\_\_

Руководитель \_\_\_\_\_ Тарасов Алексей Геннадьевич, к.т.н.,  
практики \_\_\_\_\_ доцент \_\_\_\_\_

### Список литературы

1. Пьюривал С. Основы разработки веб-приложений //СПб.: Питер. – 2015. – 272 с.
2. Елисеева Е. С., Хаханова А. Д., Учанева А. А. Применение Javascript-фреймворков при разработке интерактивных образовательных веб-приложений //Современное образование: традиции и инновации. – 2020.– №. 2. – 240-243 с.
3. Кургасов В. В., Лапшова А. Г. Javascript фреймворки //Центральный научный вестник. – 2018. – Т. 3. – №. 15-16. – 40-41 с.
4. Будаев Е. С., Димова А. В. Обзор и сравнительный анализ библиотек и фреймворков JavaScript //Вестник современных исследований. – 2019. – №. 6.2. – 33-37 с.
5. Акулов А. А. Исследование популярных javascript-фреймворков для разработки клиентской части вебприложения //Цифровизация экономики: направления, методы, инструменты. – 2019. – 358-362 с.
6. Сукиасян В. М., Придиус Е. С. Современные принципы и подходы к frontend архитектуре веб-приложений //Наука, техника и образование. – 2019. – №. 10 (63). – 54-57 с.
7. Бондаренко Ю. В. Почему стоит использовать Vue. JS //наука в современном обществе: закономерности и тенденции развития. – 2018. – 22 с.