

Федеральное агентство связи  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Сибирский государственный университет телекоммуникаций  
и информатики»  
(СибГУТИ)

Кафедра \_\_\_\_\_ БиУТ \_\_\_\_\_

Допустить к защите зав. кафедрой

\_\_\_\_\_ /С.Н. Новиков /

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
СПЕЦИАЛИСТА**

Разработка веб-приложения для защищенного электронного голосования

Пояснительная записка

Студент \_\_\_\_\_ / И.В. Маслов \_\_\_\_\_ /

Факультет \_\_\_\_\_ АЭС \_\_\_\_\_ Группа \_\_\_\_\_ АБ-55 \_\_\_\_\_ /

Руководитель \_\_\_\_\_ / А.А. Буров \_\_\_\_\_ /

Рецензент \_\_\_\_\_ / Е.В. Ткаченко \_\_\_\_\_ /

Консультанты:

— по экономическому обоснованию

\_\_\_\_\_ / И.С. Мухина \_\_\_\_\_ /

— по безопасности жизнедеятельности

\_\_\_\_\_ / Н.Н. Симакова \_\_\_\_\_ /

Новосибирск 2021

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Под. и дата

Федеральное агентство связи  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Сибирский государственный университет телекоммуникаций и информатики»  
(СибГУТИ)

# КАФЕДРА

## Безопасность и управление в телекоммуникациях

## ЗАДАНИЕ

## НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ СПЕЦИАЛИСТА

СТУДЕНТА И.В. Маслова ГРУППЫ АБ-55

«УТВЕРЖДАЮ»

« 28 » ИЮЛЯ 2020 г.Зав. кафедрой БиУТ

/ С.Н. НОВИКОВ /

Новосибирск 2020

1. Тема выпускной квалификационной работы специалиста: \_\_\_\_\_

Разработка веб-приложения для защищенного электронного голосования

утверждена приказом по университету от «28» июля 2020 г. № 4/1011о-20

2. Срок сдачи студентом законченной работы «15» января 2020 г.

3. Исходные данные по проекту (эксплуатационно-технические данные, техническое задание):

Язык программирования PHP 7

Портативная серверная платформа Open Server Panel

Отчет по веб-уязвимостям OWASP

Аналитика веб-уязвимостей от Positive Technologies

4. Содержание расчетно-пояснительной записки (перечень подлежащих разработке вопросов)	Сроки выполнения по разделам
Введение	23.09.2020 г.
1. Анализ предметной области разрабатываемого веб-приложения	29.09.2020 г.
2. Определение технологической составляющей для разработки	13.10.2020 г.
3. Разработка веб-приложения и оценка его защищенности	25.11.2020 г.
4. Безопасность жизнедеятельности	13.12.2020 г.
5. Технико-экономическое обоснование работы	20.12.2020 г.
6. Заключение	25.12.2020 г.
7. Список литературы	26.12.2020 г.
8. Приложения	27.12.2020 г.

Консультанты по ВКР (с указанием относящихся к ним разделов):

1. Раздел по технико-экономическому обоснованию

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_/ И.С. Мухина \_\_\_\_\_/

2. Раздел по безопасности жизнедеятельности

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_/ Н.Н. Симакова \_\_\_\_\_/

Дата выдачи задания

« 01 » сентября 2020 г.

\_\_\_\_\_/ А.А. Буров /

(подпись, Ф.И.О. руководителя)

Задание принял к исполнению

« 01 » сентября 2020 г.

\_\_\_\_\_/ И.В. Маслов /

(подпись, Ф.И.О. студента)

Федеральное агентство связи  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Сибирский государственный университет телекоммуникаций и информатики»  
(СибГУТИ)

---

## РЕЦЕНЗИЯ

на выпускную квалификационную работу студента И.В. Маслова  
по теме «Разработка веб-приложения для защищенного электронного голосо-  
вания»

---

Рассматриваемая в работе И.В. Маслова тема актуальна ввиду активного  
использования веб-приложений на сегодняшний день. Интернет-голосование яв-  
ляется популярным инструментом для проведения различного рода опросов. За-  
дача защиты такого ресурса базируется на предотвращении угроз и уязвимостей,  
о которых автор упоминает в работе.

---

Хочется отметить такие положительные элементы работы как: анализ по-  
тенциальных угроз и уязвимостей, использование подробных UML-диаграмм для  
описания взаимодействия пользователя с ресурсом, адекватный выбор языковых и  
программных средств для разработки, оценка защищенности разработанного веб-  
приложения.

---

В качестве замечаний к работе необходимо отметить следующее:

---

1. Автор не реализовал двухфакторную аутентификацию, которая позволила  
бы снизить риск несанкционированного доступа.

---

2. Отсутствует процедура восстановления пароля от аккаунта пользователя  
в случае его потери.

---

Несмотря на сделанные замечания, считаю, что работа выполнена на высо-  
ком техническом уровне и заслуживает оценки «отлично», а ее автор – Маслов  
Иван Васильевич присвоения квалификации специалиста по защите информации

по специальности 10.05.02 «Информационная безопасность телекоммуникацион-  
ных систем».

---

Старший программист отдела программирования и тестирования

---

ООО «Бэкап ИТ»

Ткаченко Евгений Викторович

« 18 » января 2021 г.

С Рецензией ознакомлен \_\_\_\_\_ /И.В. Маслов/

« 18 » января 2021 г.

Федеральное агентство связи  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Сибирский государственный университет телекоммуникаций и информатики»  
(СибГУТИ)

---

**ОТЗЫВ**

о работе студента И.В. Маслова в период подготовки выпускной квалификационной работы по теме «Разработка веб-приложения для защищенного электронного голосования»

---

---

---

---

---

---

---

---

---

---

Работа имеет практическую ценность  
Работа внедрена  
Рекомендую работу к внедрению  
Рекомендую работу к опубликованию  
Работа выполнена с применением ЭВМ

<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>

Тема предложена предприятием  
Тема предложена студентом  
Тема является фундаментальной  
Рекомендую студента в магистратуру  
Рекомендую студента в аспирантуру

<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>

Руководитель выпускной квалификационной работы специалиста

Доц. каф. БиУТ, к.т.н.

Буров Артем Анатольевич

«19» января 2021 г.

С Отзывом ознакомлен

/И.В. Маслов/

«19» января 2021 г.

## Уровень сформированности компетенций у студента

И.В. Маслова

Компетенции		Уровень сформированности компетенций		
		высокий	средний	низкий
1		2	3	4
Профессиональные	ПК-1 - способностью осуществлять анализ научно-технической информации, нормативных и методических материалов по методам обеспечения информационной безопасности телекоммуникационных систем			
	ПК-5 - способностью проектировать защищенные телекоммуникационные системы и их элементы, проводить анализ проектных решений по обеспечению заданного уровня безопасности и требуемого качества обслуживания, разрабатывать необходимую техническую документацию с учетом действующих нормативных и методических документов			
	ПК-7 - способностью осуществлять рациональный выбор средств обеспечения информационной безопасности телекоммуникационных систем с учетом предъявляемых к ним требований качества обслуживания и качества функционирования			
	ПК-12 - способностью выполнять технико-экономические обоснования, оценивать затраты и результаты деятельности организации в области обеспечения информационной безопасности			



## АННОТАЦИЯ

Выпускной квалификационной работа студента И.В. Маслова  
по теме Разработка веб-приложения для защищенного электронного голосования

---

Объём работы – 90 страниц, на которых размещены 36 рисунков и 12 таблиц. При написании работы использовалось 23 источника.

Ключевые слова: информационная безопасность, электронное голосование, веб-приложение, веб-уязвимости, программирование

---

Работа выполнена на: кафедре БиУТ СибГУТИ

---

Руководитель: доц. каф. БиУТ Буров А.А.

---

Целью работы являлась: разработка веб-приложения для защищенного электронного голосования

---

Решаемые задачи: анализ предметной области разрабатываемого веб-приложения, определение технологической составляющей для разработки, разработка веб-приложения и оценка его защищенности, безопасность жизнедеятельности, технико-экономическое обоснование работы

---

Основные результаты: рабочее и защищенное веб-приложение для электронного голосования

---

## Graduation thesis abstract

of I.V.Maslov on the theme Development of a web application for secure electronic voting

---

The paper consists of 90 pages, with 36 figures and 12 tables/charts/diagrams. While writing the thesis 23 reference sources were used.

Keywords: information security, electronic voting, web application, web vulnerabilities, programming

---

The thesis was written at BIUT department SibSUTIS  
(name of organization or department)

Scientific supervisor associate professor of the BiUT Burov Artyom

---

The goal/subject of the paper is develop a web application for secure electronic voting

---

Tasks: analysis of the subject area of the developed web application, determination of the technological component for development, web application development and security assessment, life safety, technical and economic justification of work

---

Results working and protected web application for electronic voting

---

## ОГЛАВЛЕНИЕ

Введение.....	4
1 Анализ предметной области разрабатываемого веб-приложения .....	5
1.1 Постановка задачи.....	5
1.2 Определение объекта разработки.....	5
1.3 Анализ основных угроз и уязвимостей веб-приложения.....	7
1.4 Разработка модели нарушителя информационной безопасности.....	12
1.5 Выводы по разделу.....	16
2 Определение технологической составляющей для разработки .....	17
2.1 Постановка задачи.....	17
2.2 Планирование архитектуры веб-приложения .....	17
2.3 Выбор языковых и программных средств разработки.....	24
2.4 Выбор практических методов защиты веб-приложения.....	27
2.6 Выводы по разделу.....	32
3 Разработка веб-приложения и оценка его защищенности .....	33
3.1 Постановка задачи.....	33
3.2 Создание веб-интерфейса.....	33
3.3 Разработка серверной логики веб-приложения .....	38
3.4 Настройка протокола SSL .....	44
3.5 Оценка защищенности разработанного веб-приложения.....	48
3.6 Выводы по разделу.....	53
4 Безопасность жизнедеятельности.....	54
4.1 Постановка задачи.....	54
4.2 Характеристика условий труда при работе с ПК.....	54
4.3 Эргономические требования к рабочему месту пользователя .....	57

Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата	3 Разработка веб-приложения и оценка его защищенности..... 33			
					3.1 Постановка задачи..... 33			
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата	3.2 Создание веб-интерфейса..... 33			
					3.3 Разработка серверной логики веб-приложения ..... 38			
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата	3.4 Настройка протокола SSL ..... 44			
					3.5 Оценка защищенности разработанного веб-приложения..... 48			
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата	3.6 Выводы по разделу..... 53			
					4 Безопасность жизнедеятельности..... 54			
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата	4.1 Постановка задачи..... 54			
					4.2 Характеристика условий труда при работе с ПК..... 54			
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата	4.3 Эргономические требования к рабочему месту пользователя..... 57			
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и дата	Инв. № дубл.	Подп. и дата				
Инв. № подл.	Взам. инв. №	Подп. и						

4.4 Требования охраны труда офисных работников .....	59
4.5 Пожарная безопасность .....	62
4.6 Выводы по разделу.....	64
5 Технико-экономическое обоснование работы .....	65
5.1 Постановка задачи.....	65
5.2 Расчет трудоемкости и длительности работ.....	65
5.3 Расчет себестоимости и цены программного продукта .....	69
5.4 Выводы по разделу.....	72
Заключение .....	73
Список литературы .....	74
Приложение А Код разработанного веб-приложения.....	76

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата						
Изм.	Лист	№ докум.	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ					Лист
										3

## Введение

Интернет-голосование, как разновидность электронного голосования, применяется в различных официальных и неофициальных опросах. Оно является доступным и удобным, что делает его популярным и часто используемым, например, для маркетинговых и социологических опросов.

Как и любой веб-ресурс, веб-приложение для электронного голосования подвержено атакам. Компания Positive Technologies ежегодно проводит анализ защищенности веб-приложений. По данным за 2019 год, 50% исследованных веб-приложений содержали уязвимости высокого уровня риска, 39% – критически опасные уязвимости и лишь 11% – уязвимости низкого уровня риска [21]. Поэтому необходимо реализовать систему защиты веб-приложения, учитывая уязвимости, несущие наибольший уровень риска.

Целью выпускной квалификационной работы является разработка веб-приложения для защищенного электронного голосования.

Задачи выпускной квалификационной работы:

- 1) проанализировать предметную область разрабатываемого веб-приложения;
- 2) определить технологическую составляющую для разработки;
- 3) разработать веб-приложение и оценить его защищенность;
- 4) привести технико-экономическое обоснование;
- 5) привести меры по обеспечению безопасности жизнедеятельности.

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата						
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ					Лист
										4

# 1 Анализ предметной области разрабатываемого веб-приложения

## 1.1 Постановка задачи

В данной главе необходимо определить объект разработки и описать его возможности. А также разработать модели потенциальных угроз и нарушителя, на основе которых будет строиться система защиты разрабатываемого веб-приложения.

## 1.2 Определение объекта разработки

Под электронным голосованием понимают такое голосование, в котором используются электронные средства, помогающие или обеспечивающие подачу и подсчет голосов. [11]

Электронное голосование часто рассматривается как инструмент повышения эффективности избирательного процесса и повышения доверия к нему. Правильно реализованные решения для электронного голосования могут повысить безопасность бюллетеня, ускорить обработку результатов и упростить само голосование. [5]

На рисунке 1.1 изображены основные четыре технологии электронного голосования.

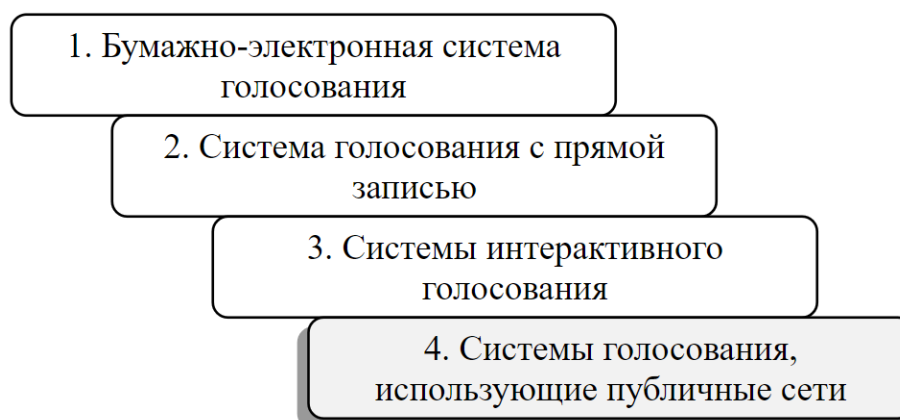


Рисунок 1.1 – Основные технологии электронного голосования

Подпись и дата		Инв. № дубл.		Взам. инв. №		Подпись и дата		Инв. № подл.	
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ				Лист
									5

В системах бумажного-электронного голосования голоса подаются с использованием бумажных бюллетеней, а подсчитываются специальными машинами.

Система голосования с прямой записью осуществляет сбор голосов путём предоставления механических или электрооптических компонентов (как правило, кнопки или сенсорные экраны), которые могут быть использованы избирателем.

Системы интерактивного голосования для сбора результатов используют пульты, которые обычно напоминают пульт от телевизора или калькулятор. [11]

В работе рассматривается система голосования, использующая публичные сети. Такая система включает в себя одновременно и электронные бюллетени, и передачу информации о голосах по открытым компьютерным сетям.

Разрабатываемое веб-приложение для электронного голосования является интерактивным сервисом, дающее возможность посетителям выразить свое мнение по любому кругу вопросов. Сами пользователи имеют возможность создания интересующей их темы для голосования, чтобы провести опрос в той или иной области.

Разработанный сервис можно будет использовать, например, для маркетинговых и социологических опросов.

Веб-приложение для электронного голосования должно предусматривать:

- 1) регистрацию пользователей с подтверждением аккаунта через почту;
- 2) авторизацию пользователей;
- 3) возможность авторизованным пользователям создавать новое голосование, задавать его срок и варианты ответа;
- 5) вывод результатов по истечении срока, отведенного для голосования;
- 4) исключение возможности повторного голосования одним и тем же пользователем.

Электронное голосование в силу своей сущности предполагает использование онлайн-платформы и цифровых устройств для выражения мнения. Этот вариант волеизъявления обладает целым набором преимуществ. Среди них – возмож-

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата						
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ					Лист
										6

ность участия большего количества людей, удобство процедуры для участников, большая надежность и автоматизированный подсчет результатов.

Но, как и в случае с любым процессом, происходящим онлайн, электронное голосование потенциально может быть связано с определенными рисками, присущими веб-приложениям. [22]

Потенциальные угрозы и уязвимости представлены в следующем подразделе.

### 1.3 Анализ основных угроз и уязвимостей веб-приложения

Согласно ГОСТ Р 50922-2006, угрозой безопасности информации является совокупность условий и факторов, создающих потенциальную или реально существующую опасность нарушения безопасности информации. Уязвимость же является свойством информационной системы, обуславливающее возможность реализации угроз безопасности обрабатываемой в ней информации. [8]

Все источники угроз безопасности информации можно разделить на три основные группы:

- 1) обусловленные действиями субъекта (антропогенные источники угроз);
- 2) обусловленные техническими средствами (техногенные источники угрозы);
- 3) обусловленные стихийными источниками. [7]

Наибольший интерес с точки зрения организации защиты представляют антропогенные источники угроз безопасности информации, так как в роли таких источников выступают субъекты, действия которых всегда можно оценить, спрогнозировать и принять адекватные меры.

В качестве антропогенного источника угроз можно рассматривать субъекта, имеющего доступ (санкционированный или несанкционированный) к работе со штатными средствами защищаемого объекта. Субъекты (источники), действия которых могут привести к нарушению безопасности информации, могут быть:

– внешние;

Подпись и дата		<p>Все источники угроз безопасности информации можно разделить на три основные группы:</p> <p>1) обусловленные действиями субъекта (антропогенные источники угроз);</p> <p>2) обусловленные техническими средствами (техногенные источники угрозы);</p> <p>3) обусловленные стихийными источниками. [7]</p> <p>Наибольший интерес с точки зрения организации защиты представляют антропогенные источники угроз безопасности информации, так как в роли таких источников выступают субъекты, действия которых всегда можно оценить, спрогнозировать и принять адекватные меры.</p> <p>В качестве антропогенного источника угроз можно рассматривать субъекта, имеющего доступ (санкционированный или несанкционированный) к работе со штатными средствами защищаемого объекта. Субъекты (источники), действия которых могут привести к нарушению безопасности информации, могут быть:</p> <p>— внешние;</p>																		Лист																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																									
Инв. № дубл.																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													
Взам. инв. №																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													
Подпись и дата																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													
Инв. № подл.																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													



– внутренние. [7]

Внешние источники могут быть случайными или преднамеренными и иметь разный уровень квалификации.

Внутренние субъекты, как правило, представляют собой высококвалифицированных специалистов в области разработки и эксплуатации программного обеспечения и технических средств, знакомы со спецификой решаемых задач, структурой и основными функциями и принципами работы программно-аппаратных средств защиты информации, имеют возможность использования штатного оборудования и технических средств сети.

Уязвимости безопасности информации могут быть:

- объективными;
- субъективными;
- случайными.

Объективные уязвимости зависят от особенностей построения и технических характеристик оборудования, применяемого на защищаемом объекте. Полное устранение этих уязвимостей невозможно, но они могут существенно ослабляться техническими и инженерно-техническими методами парирования угроз безопасности информации.

Субъективные уязвимости зависят от действий сотрудников и, в основном, устраняются организационными и программно-аппаратными методами.

Случайные уязвимости зависят от особенностей окружающей защищаемый объект среды и непредвиденных обстоятельств. Эти факторы, как правило, мало предсказуемы и их устранение возможно только при проведении комплекса организационных и инженерно-технических мероприятий по противодействию, угрозам информационной безопасности. [7]

Следует рассмотреть угрозы и уязвимости, отталкиваясь от особенностей разрабатываемого веб-приложения. В таблице 1.1 приведены особенности будущей системы и прогнозируемые угрозы и уязвимости.

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	<p>ное устранение этих уязвимостей невозможно, но они могут существенно ослабляться техническими и инженерно-техническими методами парирования угроз безопасности информации.</p> <p>Субъективные уязвимости зависят от действий сотрудников и, в основном устраняются организационными и программно-аппаратными методами.</p> <p>Случайные уязвимости зависят от особенностей окружающей защищаемый объект среды и непредвиденных обстоятельств. Эти факторы, как правило, мало предсказуемы и их устранение возможно только при проведении комплекса организационных и инженерно-технических мероприятий по противодействию, угрозам информационной безопасности. [7]</p> <p>Следует рассмотреть угрозы и уязвимости, отталкиваясь от особенностей разрабатываемого веб-приложения. В таблице 1.1 приведены особенности будущей системы и прогнозируемые угрозы и уязвимости.</p>
Изм.	Лист	№ докум	Подпись	Дата	<div>ФАЭС.10.05.02.55.ПЗ</div> <div>Лист</div> <div>8</div>

Таблица 1.1 – Прогнозируемые угрозы и уязвимости веб-приложения

Особенность веб-приложения	Возможные угрозы и уязвимости
Регистрация и авторизация пользователя	Недостатки аутентификации
Подтверждение аккаунта при помощи ссылки с GET-запросом	SQL - инъекция
Создание пользователем нового голосования	Межсайтовое выполнение сценариев (XSS)
Взаимодействие пользователя с HTML-формами	Подделка межсайтового запроса (CSRF)

Благодаря исследованиям Positive Technologies – международной компании, специализирующейся на разработке программного обеспечения в области информационной безопасности, выявлены самые распространенные угрозы и уязвимости из списка OWASP (Open Web Application Security Project – открытый проект обеспечения безопасности веб-приложений) – некоммерческого фонда, который работает над повышением безопасности программного обеспечения [2]. Они представлены на рисунке 1.2 [21].

Ине. № подл.	Подпись и дата	Взам. инв. №	Ине. № дубл.	Подпись и дата	<p>ФАЭС.10.05.02.55.ПЗ</p>					Лист
										9
Изм.	Лист	№ докум	Подпись	Дата						

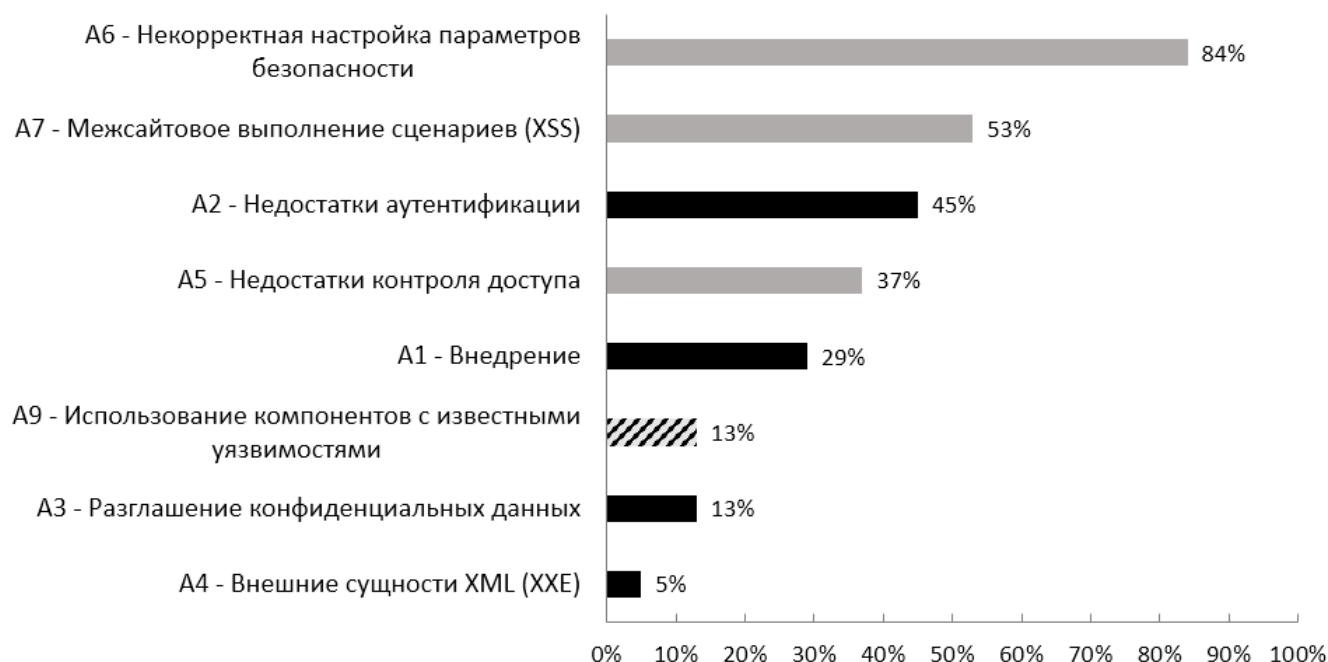


Рисунок 1.2 – Самые распространенные угрозы и уязвимости веб-приложений

Уязвимости А2, А1, А3, и А4 несут в себе высокий риск, А6, А7 и А5 – средний, А9 – низкий.

Угрозы и уязвимости, приведенные в таблице 1.1, входят в топ-5 самых распространенных угроз и уязвимостей веб-приложений за 2019 год. При этом «Недостатки аутентификации» и «Внедрение» несут в себе высокий риск, а «Межсайтовое выполнение сценариев (XSS)» – средний.

Почти треть выявленных уязвимостей из категории «Недостатки аутентификации» – это некорректное ограничение количества неудачных попыток аутентификации. В результате эксплуатации этой уязвимости злоумышленник может подобрать учетные данные пользователя и таким образом получить доступ к веб-приложению.

Треть веб-приложений оказались уязвимы для атаки типа «Подделка межсайтового запроса (CSRF)». В ходе CSRF-атаки злоумышленник с помощью специально сформированных сценариев может выполнять действия от лица пользователя, авторизованного в уязвимом веб-приложении. [21]

Ине. № подл.	Подпись и дата
Взам. инв. №	Ине. № дубл.
Подпись и дата	Подпись и дата

Изм.	Лист	№ докум	Подпись	Дата
------	------	---------	---------	------

ФАЭС.10.05.02.55.ПЗ

Также в каждом третьем веб-приложении присутствуют уязвимости к SQL-инъекциям. Атакующий может выполнить произвольный запрос к базе данных (например, прочитать содержимое любых таблиц, удалить, изменить или добавить данные), получить возможность чтения и/или записи локальных файлов и выполнения произвольных команд на атакуемом сервере. [21]

Таблица 1.2 – Способы предотвращения угроз и уязвимостей веб-приложений

Угроза / уязвимость	Способы предотвращения
SQL-инъекция	1) Для динамических запросов реализовать экранирование спецсимволов. 2) Использовать приведение к типу данных.
Недостатки аутентификации	1) Реализовать проверку надежности паролей, установив их длину и сложность. 2) Предусмотреть защитные меры от неудачных попыток авторизации.
Межсайтовое выполнение сценариев (XSS)	1) Заменять специальные символы HTML-страницы на эквиваленты, не являющиеся символами форматирования. 2) Настраивать параметры безопасности данных в cookie.
Подделка межсайтового запроса (CSRF)	1) К каждой сессии привязывать уникальный сгенерированный csrf-токен.

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата

Согласно ГОСТ Р 53114-2008, нарушителем информационной безопасности является физическое лицо или логический объект, случайно или преднамеренно совершивший действие, следствием которого является нарушение информационной безопасности. [9]

В свою очередь, модель нарушителя информационной безопасности – это набор предположений об одном или нескольких возможных нарушителях информационной безопасности, их квалификации, их технических и материальных средствах и т. д. [7]

Целью разработки модели нарушителя является формирование предположения о типах, видах нарушителей, которые могут реализовать угрозы безопасности информации в информационной системе с заданными структурно-функциональными характеристиками и особенностями функционирования, а также потенциале этих нарушителей и возможных способах реализации угроз безопасности информации. [14]

С учетом наличия прав доступа и возможностей по доступу к информации и/или к компонентам информационной системы нарушители подразделяются на два типа:

1) внешние нарушители (тип I) – лица, не имеющие права доступа к информационной системе, ее отдельным компонентам и реализующие угрозы безопасности информации из-за границ информационной системы;

2) внутренние нарушители (тип II) – лица, имеющие право постоянного или разового доступа к информационной системе, ее отдельным компонентам.

В зависимости от потенциала, требуемого для реализации угроз безопасности информации, нарушители подразделяются на:

– нарушителей, обладающих базовым (низким) потенциалом нападения при реализации угроз безопасности информации в информационной системе;

– нарушителей, обладающих базовым повышенным (средним) потенциалом нападения при реализации угроз безопасности информации в информационной системе;

– нарушителей, обладающих высоким потенциалом нападения при реализации угроз безопасности информации в информационной системе. [14]

Обратимся к банку данных угроз безопасности информации (УБИ), разработанный Федеральной службой по техническому и экспортному контролю России. В банке описаны угрозы и соответствующий им тип нарушителя и его минимально необходимый потенциал. Угрозы, которые можно отнести к разрабатываемому веб-приложению:

– угроза внедрения кода или данных (УБИ. 006);  
– угроза восстановления и/или повторного использования аутентификационной информации (УБИ. 008);

– угроза доступа/перехвата/изменения HTTP cookies (УБИ. 017);  
– угроза использования информации идентификации/аутентификации, заданной по умолчанию (УБИ. 030);

– угроза межсайтового скриптинга (УБИ. 041);  
– угроза несанкционированного доступа к аутентификационной информации (УБИ. 074);

– угроза несанкционированного изменения аутентификационной информации (УБИ. 086);

– угроза обхода некорректно настроенных механизмов аутентификации (УБИ. 100);

– угроза перехвата данных, передаваемых по вычислительной сети (УБИ. 116);

– угроза удаления аутентификационной информации (УБИ. 152). [6]

Приведенным угрозам соответствуют следующие типы нарушителей и их минимально необходимый потенциал:

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата						
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ					Лист
										13

- 1) внешний нарушитель с низким потенциалом;
- 2) внутренний нарушитель с низким потенциалом;
- 3) внешний нарушитель со средним потенциалом. [6]

Внешними нарушителями с низким потенциалом могут быть:

- внешние субъекты (физические лица);
- бывшие работники.

Внутренними нарушителями с низким потенциалом могут быть:

– лица, обеспечивающие функционирование информационных систем или обслуживающих инфраструктуру оператора;

- пользователи;
- лица, привлекаемые для установки, наладки, монтажа, пусконаладочных и иных работ.

Внешними нарушителями со средним потенциалом могут быть:

- преступные группы;
- конкуренты;
- разработчики, производители, поставщики программных, технических и программно-технических средств. [14]

В таблице 1.3 [14] приведена возможная мотивация рассмотренных выше нарушителей.

Инв. № подл.	Подпись и дата				Инв. № дубл.	Взам. инв. №	Подпись и дата	Инв. № подл.	
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ				Лист
									14

Таблица 1.3 – Виды нарушителей и их мотивация

Виды нарушителей	Возможные цели реализации угроз
Преступные группы	<ul style="list-style-type: none"> <li>– причинение имущественного ущерба путем мошенничества или иным преступным путем;</li> <li>– выявление уязвимостей с целью их дальнейшей продажи и получения финансовой выгоды.</li> </ul>
Внешние субъекты (физические лица)	<ul style="list-style-type: none"> <li>– идеологические или политические мотивы;</li> <li>– причинение имущественного ущерба путем мошенничества или иным преступным путем;</li> <li>– любопытство или желание самореализации;</li> <li>– выявление уязвимостей с целью их дальнейшей продажи и получения финансовой выгоды.</li> </ul>
Конкуренты	<ul style="list-style-type: none"> <li>– получение конкурентных преимуществ;</li> <li>– причинение имущественного ущерба путем обмана или злоупотребления доверием.</li> </ul>
Разработчики, производители, поставщики программных, технических и программно-технических средств	<ul style="list-style-type: none"> <li>– внедрение дополнительных функциональных возможностей в программное обеспечение или программно-технические средства на этапе разработки;</li> <li>– причинение имущественного ущерба путем обмана или злоупотребления доверием;</li> <li>– непреднамеренные, неосторожные или неквалифицированные действия.</li> </ul>
Лица, привлекаемые для установки, наладки, монтажа, пусконаладочных и иных видов работ	<ul style="list-style-type: none"> <li>– причинение имущественного ущерба путем обмана или злоупотребления доверием;</li> <li>– непреднамеренные, неосторожные или неквалифицированные действия.</li> </ul>

Инв. № подл.	Подпись и дата
Взам. инв. №	Инв. № дубл.
Подпись и дата	



Продолжение таблицы 1.3

Лица, обеспечивающие функционирование информационных систем или обслуживающие инфраструктур	<ul style="list-style-type: none"> <li>– причинение имущественного ущерба путем обмана или злоупотребления доверием;</li> <li>– непреднамеренные, неосторожные или неквалифицированные действия.</li> </ul>
Пользователи	<ul style="list-style-type: none"> <li>– причинение имущественного ущерба путем мошенничества или иным преступным путем;</li> <li>– любопытство или желание самореализации;</li> <li>– непреднамеренные, неосторожные или неквалифицированные действия.</li> </ul>
Бывшие работники (пользователи)	<ul style="list-style-type: none"> <li>– причинение имущественного ущерба путем мошенничества или иным преступным путем;</li> <li>– месть за ранее совершенные действия.</li> </ul>

При разработке защищенного веб-приложения для электронного голосования необходимо руководствоваться моделями угроз и нарушителя, так как с их помощью удастся построить качественную систему защиты.

### 1.5 Выводы по разделу

В первом разделе был определен объект разработки, спрогнозированы угрозы и уязвимости разрабатываемой системы и рассмотрены способы их предотвращения. Также была разработана модель потенциального нарушителя информационной безопасности веб-приложения для электронного голосования.

Ине. № подл.	Подпись и дата
Взам. инв. №	Инв. № дубл.
Подпись и дата	

Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ	Лист
						16

## 2.1 Постановка задачи

В рамках данной главы необходимо спланировать архитектуру разрабатываемого веб-приложения и отобразить принцип взаимодействия пользователя с системой. Учитывая эти сведения, нужно определить какая техническая база будет использоваться при разработке веб-приложения.

Согласно ГОСТ Р ИСО 9241-151-2014, веб-приложением является приложение, которое предоставляет функциональные возможности пользователю через браузер или другой тип агента пользователя (программное обеспечение конечного пользователя, которое позволяет пользователям взаимодействовать с отдаленной системой через интернет-протоколы), использующего веб-форматы и протоколы.

[11]

Отображением результатов запросов, а также приемом данных от клиента и их передач на сервер обычно занимается специальное приложение – браузер. Как известно, одной из функций браузера является отображение данных, полученных из Интернета, в виде страницы, описанной на языке HTML, следовательно, результат, передаваемый сервером клиенту, должен быть представлен на этом языке.

На стороне сервера веб-приложение выполняется специальным программным обеспечением (веб-сервером), который и принимает запросы клиентов, обрабатывает их, формирует ответ в виде страницы, описанной на языке HTML, и передает его клиенту.

В процессе обработки запроса пользователя веб-приложение компонует ответ на основе исполнения программного кода, работающего на стороне сервера, веб-формы, страницы HTML, другого содержимого, включая графические файлы. В результате, как уже было сказано, формируется HTML-страница, которая и от-

правляется клиенту. Получается, что результат работы веб-приложения идентичен результату запроса к традиционному веб-сайту, однако, в отличие от него, веб-приложение генерирует HTML-код в зависимости от запроса пользователя, а не просто передает его клиенту в том виде, в котором этот код хранится в файле на стороне сервера. То есть веб-приложение динамически формирует ответ с помощью исполняемого кода – так называемой исполняемой части.

Принцип работы веб-приложения представлен на рисунке 2.1.

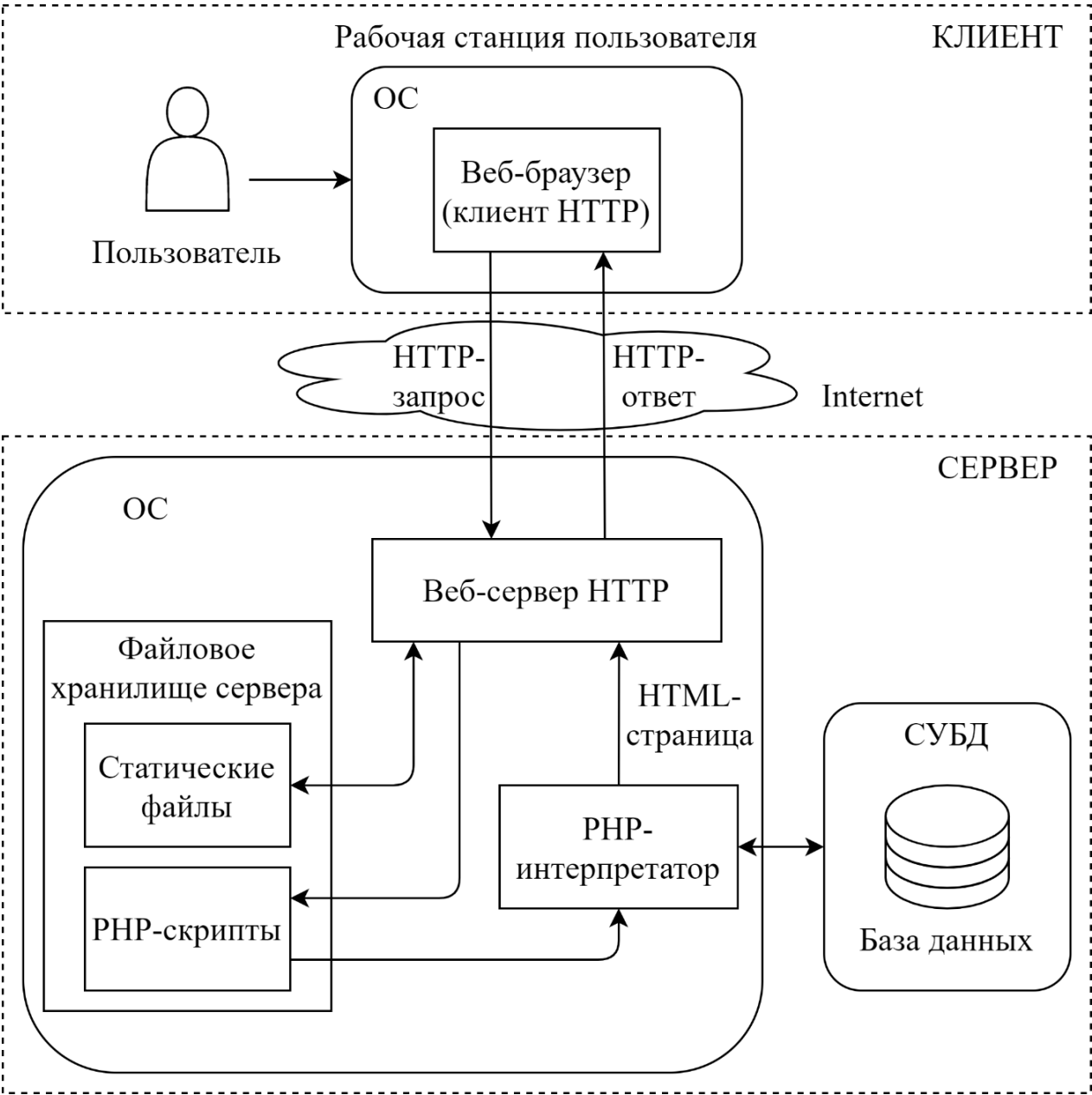


Рисунок 2.1 – Принцип работы веб-приложения

На рисунке 2.2 изображена диаграмма взаимодействия пользователя с веб-приложением при регистрации.

					ФАЭС.10.05.02.55.ПЗ	Лист
Изм.	Лист	№ докум	Подпись	Дата		19





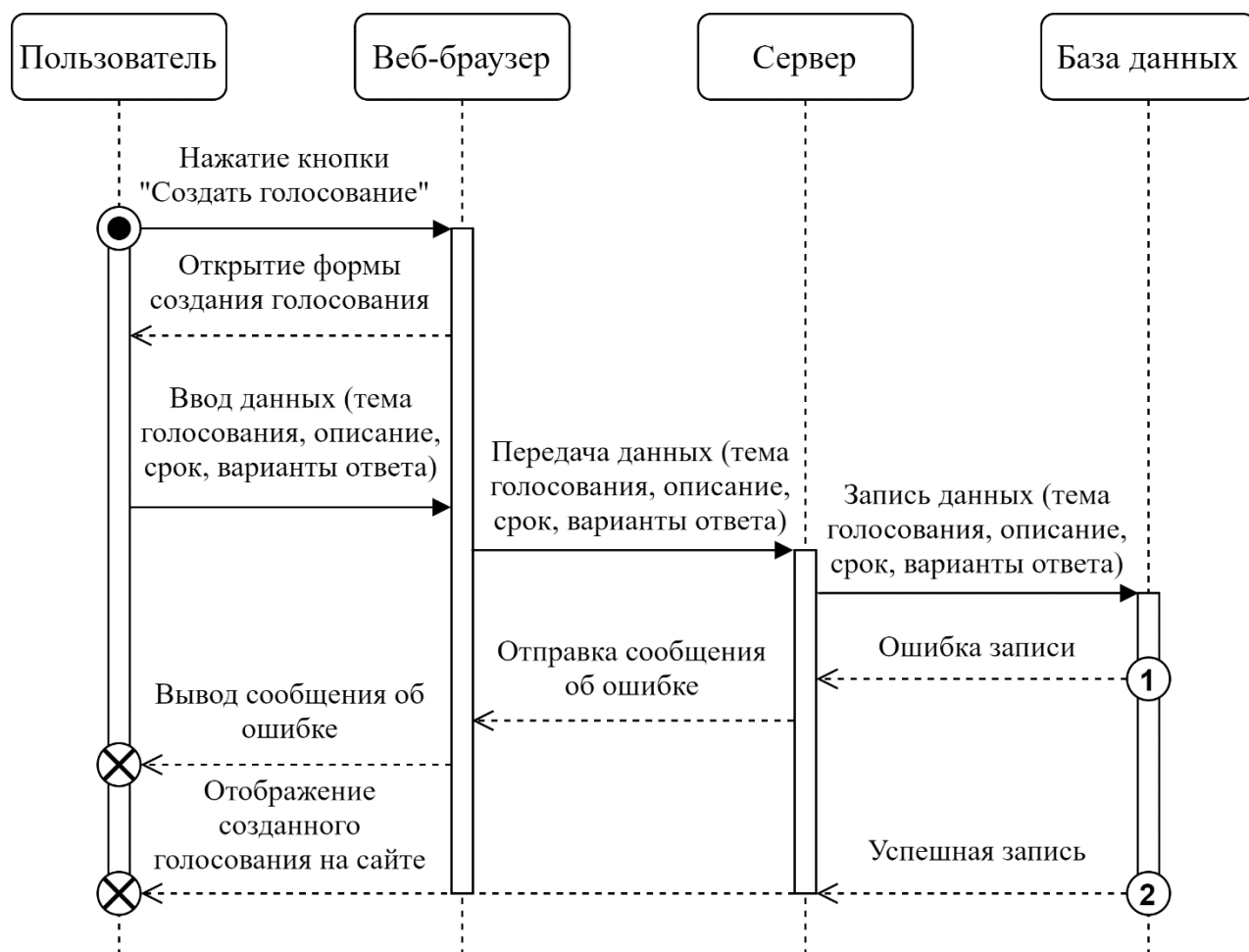


Рисунок 2.4 – Принцип создания нового голосования пользователем

Пользователь открывает форму создания голосования нажатием на кнопку «Создать голосование». Вводит тему, описание, срок и варианты голосования. Данные передаются на сервер, который производит запись в базу данных. Если данные записались успешно, то голосование появляется на странице сайта, в противном случае выводится сообщение об ошибке.

Рассмотрим принцип голосования пользователем (рисунок 2.5).

Инев. № подл.	Подпись и дата
Взам. инв. №	Инев. № дубл.
Подпись и дата	Инев. № дубл.
Инев. № подл.	Инев. № дубл.

Изм.	Лист	№ докум	Подпись	Дата
------	------	---------	---------	------

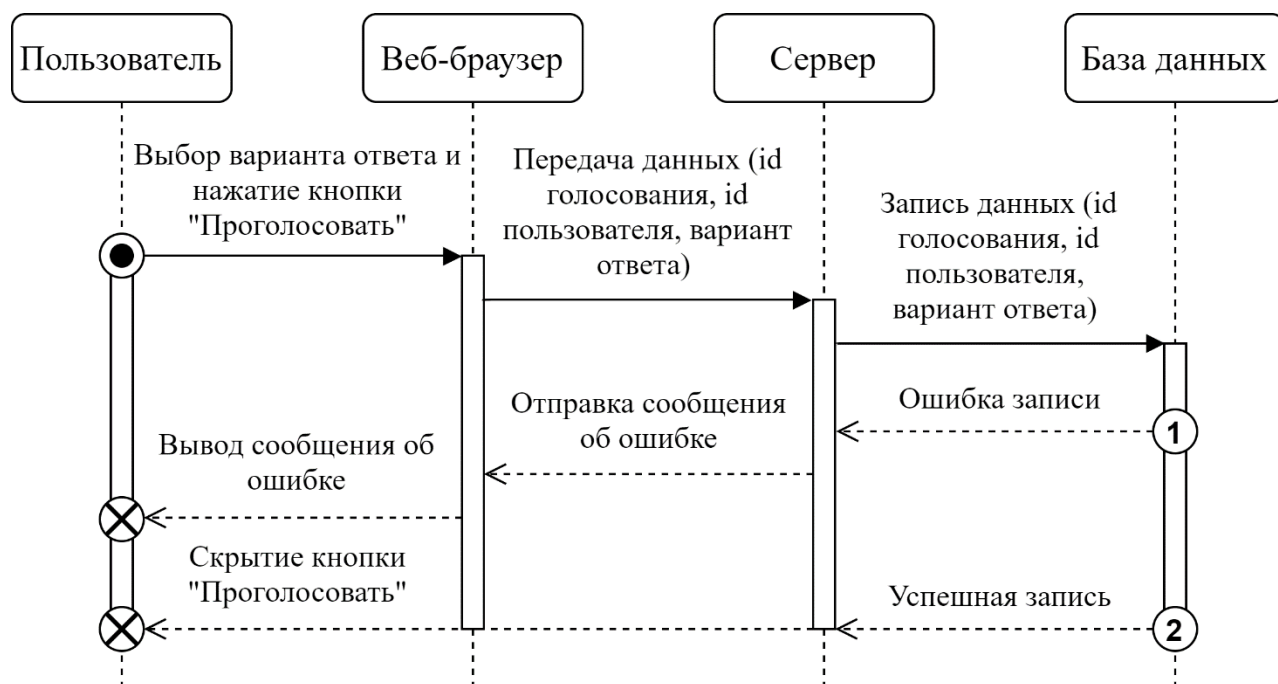


Рисунок 2.5 – Принцип голосования пользователем

Пользователь выбирает вариант ответа и нажимает кнопку «Проголосовать». Серверу передаются данные о пользователе, голосовании и выбранном варианте ответа. Эти данные заносятся в базу данных, после чего пользователю запрещается повторно голосовать в этом голосовании. Если не удастся записать данные в базу данных, то выходит сообщение об ошибке.

На рисунке 2.6 изображен принцип завершения сессии пользователем нажатием на кнопку «Выйти».

Ине. № подл.	Подпись и дата
Взам. инв. №	Инв. № дубл.
Подпись и дата	
Ине. № подл.	

Изм.	Лист	№ докум	Подпись	Дата

ФАЭС.10.05.02.55.ПЗ





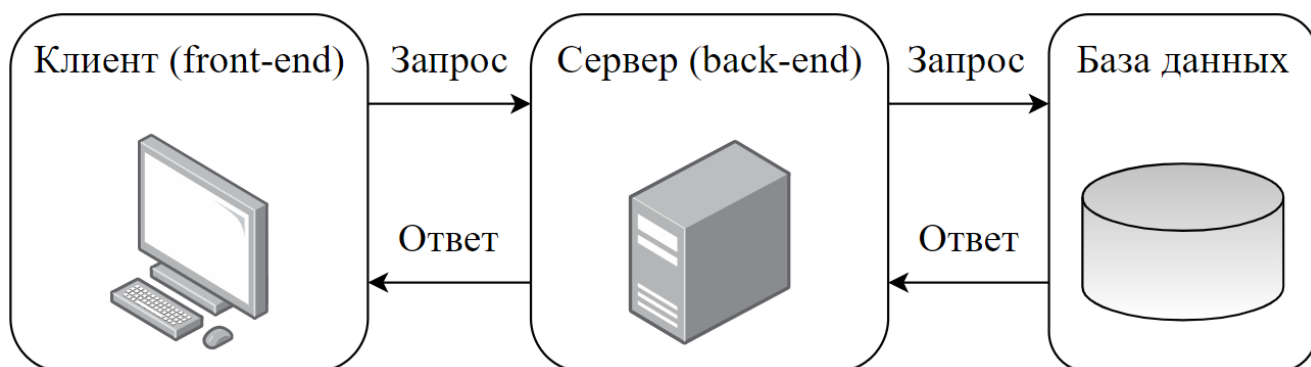


Рисунок 2.7 – Клиент-серверная архитектура

Front-end составляющая содержит в себе 3 основополагающих языковых компонента:

1) HTML (HyperText Markup Language – язык гипертекстовой разметки) – язык разметки документов для создания структуры страницы: заголовки, абзацы, списки и так далее;

2) CSS (Cascading Style Sheets – каскадные таблицы стилей) – язык для описания и стилизации внешнего вида документа;

3) JavaScript – язык программирования, который способствует «оживлению» веб-страниц при взаимодействии пользователя с ними.

Для разработки логики серверной стороны веб-приложения будет использоваться распространенный язык программирования общего назначения PHP (Hypertext Preprocessor – препроцессор гипертекста). Его код может внедряться непосредственно в HTML-разметку. [18]

PHP имеет ряд неоспоримых преимуществ:

1) высокая скорость работы и, соответственно, общая производительность ресурсов;

2) простота освоения, простой синтаксис;

3) отличная совместимость и переносимость – php-коды работают одинаково хорошо с разными платформами;

4) набор текста кода и его редактирование можно осуществлять в любом текстовом или html-редакторе.

Ине. № подл.	Подпись и дата
Взам. инв. №	Ине. № дубл.
Подпись и дата	
Ине. № подл.	

Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ	Лист
						25



Apache – это свободный HTTP-сервер, основными достоинствами которого считаются надёжность и гибкость конфигурации.

В качестве WAMP-платформы будет использоваться Open Server Panel – портативная серверная платформа и программная среда, созданная специально для веб-разработчиков с учётом их рекомендаций и пожеланий. Программный комплекс имеет богатый набор серверного программного обеспечения, удобный, многофункциональный продуманный интерфейс, обладает мощными возможностями по администрированию и настройке компонентов. Платформа широко используется с целью разработки, отладки и тестирования веб-проектов, а также для предоставления веб-сервисов в локальных сетях. [1]

### Подведем итоги:

- для разработки веб-интерфейса будут использоваться HTML, CSS и JavaScript;
- для написания логики на сервере – PHP;
- для управления базой данных выбрана СУБД MySQL, запросы к которой будут описываться с помощью языка SQL;
- в качестве серверной платформы будет использоваться Open Server Panel.

## 2.4 Выбор практических методов защиты веб-приложения

### 2.4.1 Анализ криптографических способов защиты информации

Для защиты информации будем использовать следующие способы:

- 1) сертификат SSL (Secure Sockets Layer – слой защищённых сокетов) – криптографический протокол, обеспечивающий защищённую передачу данных между узлами в сети Интернет;
- 2) хеширование паролей пользователей посредством встроенной функции PHP.

Большую важность имеет проблема защиты информации от несанкционированного доступа (НСД) при передаче и/или хранении. Испытанный метод за-

щиты информации от НСД – шифрование. Шифрованием называют процесс преобразования исходных данных в зашифрованные, нечитаемые без знания специальных параметров преобразования – ключа. Дешифрованием называют обратное преобразование зашифрованных данных в открытые.

Все известные алгоритмы шифрования делятся на два типа:

1) симметричные – с единственным секретным ключом для шифрования и дешифрования (они же single-key). Симметричные алгоритмы также подразделяются на два семейства: потоковые – шифрование данных посимвольно, блочные – шифрование данных кусками (блоками) из нескольких символов конечной и фиксированной длины;

2) асимметричные – с двумя ключами: открытым ключом (или public-key) и закрытым ключом (или private-key); первый ключ служит только для шифрования, второй – для дешифрования.

Каждый из указанных типов криптоалгоритмов имеет свои достоинства и недостатки. Так основным недостатком симметричных методов является необходимость организации закрытого канала для передачи ключа. А основным достоинством – быстрота выполнения криптопреобразования. И наоборот, асимметричные алгоритмы более медленные в выполнении криптопреобразования, но не требуют закрытых каналов обмена ключами, т.к. открытый ключ не позволяет произвести дешифровку, а передавать закрытый ключ не нужно. [13]

Протокол, который используется для передачи данных в сети и получения информации с сайтов, называется HTTP (HyperText Transfer Protocol – протокол передачи гипертекста). У него существует расширение, которое называется HTTPS (HyperText Transfer Protocol Secure – безопасный протокол передачи гипертекста). Его суть в том, что расширение позволяет передавать информацию между клиентом и сервером в зашифрованном виде. То есть информация, которой обмениваются клиент и сервер, доступна только этому клиенту и этому серверу, а не третьим лицам. [3]

Шифрование данных, которые передаются от клиента к серверу, происходит, в свою очередь, в соответствии со своим, криптографическим протоколом.

Инв. № подл.	Подпись и дата				Инв. № рубл.	Взам. инв. №	Подпись и дата	Инв. № подл.	Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ	Лист
															28

Для того чтобы использовать шифрование, у сайта должен быть специальный сертификат, или, как он еще называется, цифровая подпись, который подтверждает, что механизм шифрования действительно надежен и соответствует протоколу. Индикаторами того, что у сайта такой сертификат есть, являются, помимо буквы «S» в HTTPS, зеленый замочек и надпись «Защищено» или название компании в адресной строке браузера. [3]

На рисунке 2.8 представлен принцип внедрения протокола TLS.

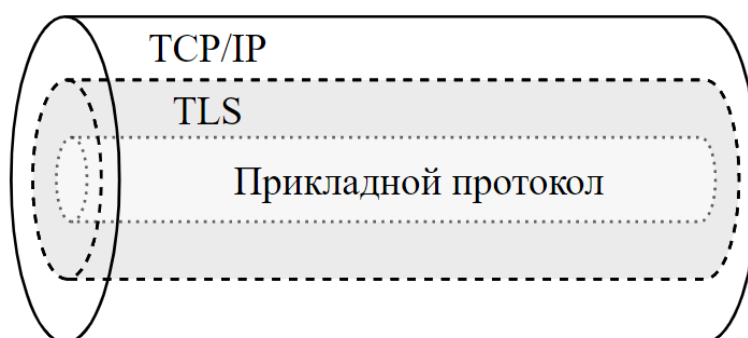


Рисунок 2.8 – Положение протокола TLS при передаче данных

Прикладной протокол (например, HTTP или FTP) «заворачивается» в TLS, а тот в свою очередь в TCP/IP. Получается, что данные по прикладному протоколу передаются по TCP/IP, но уже в зашифрованном виде. [4]

Рассмотрим второй способ защиты данных посредством хеширования паролей. Хеширование паролей является одним из самых основных соображений безопасности, которые необходимо сделать, при разработке приложения, принимающего пароли от пользователей. Без хеширования, пароли, хранящиеся в базе приложения, могут быть украдены, например, если база данных была скомпрометирована, а затем немедленно могут быть применены для компрометации не только приложения, но и аккаунтов пользователей на других сервисах, если они не используют уникальных паролей.

Применяя хеширующий алгоритм к пользовательским паролям перед сохранением их в базе данных, становится невозможным разгадывание оригиналь-

Рисунок 2.8 – Положение протокола TLS при передаче данных				
Подпись и дата				
Инв. № дубл.				
Взам. инв. №				
Подпись и дата				
Инв. № подл.				
<p>Прикладной протокол (например, HTTP или FTP) «заворачивается» в TLS, а тот в свою очередь в TCP/IP. Получается, что данные по прикладному протоколу передаются по TCP/IP, но уже в зашифрованном виде. [4]</p> <p>Рассмотрим второй способ защиты данных посредством хеширования паролей. Хеширование паролей является одним из самых основных соображений безопасности, которые необходимо сделать, при разработке приложения, принимающего пароли от пользователей. Без хеширования, пароли, хранящиеся в базе приложения, могут быть украдены, например, если база данных была скомпрометирована, а затем немедленно могут быть применены для компрометации не только приложения, но и аккаунтов пользователей на других сервисах, если они не используют уникальных паролей.</p> <p>Применяя хеширующий алгоритм к пользовательским паролям перед сохранением их в базе данных, становится невозможным разгадывание оригиналь-</p>				
ФАЭС.10.05.02.55.ПЗ				
Изм.	Лист	№ докум	Подпись	Дата
Лист				
29				

ного пароля для атакующего базу данных, в то же время сохраняя возможность сравнения полученного хеша с оригинальным паролем. [17]

Хеш-функцией, или функцией свёртки называется такая функция, которая осуществляет преобразование массива входных данных произвольной длины в битовую строку установленной длины, выполняемое определённым алгоритмом. Преобразование, производимое хеш-функцией, называется хешированием. [12]

При хешировании паролей существует два важных соображения: это стоимость вычисления и соль. Чем выше стоимость вычисления хеширующего алгоритма, тем больше времени требуется для взлома его вывода методом «грубой силы». Криптографическая соль представляет собой данные, которые применяются в процессе хеширования для предотвращения возможности разгадать оригинальный ввод с помощью поиска результата хеширования в списке заранее вычисленных пар ввод-хеш, известном также как «радужная» таблица. Более простыми словами, соль – это кусочек дополнительных данных, которые делают хеши намного более устойчивыми к взлому.

PHP, начиная с версии 5.5, предоставляет встроенное API хеширования паролей `password_hash()`, которое безопасно работает и с хешированием, и с проверкой паролей.

При хешировании паролей рекомендуется применять алгоритм Blowfish, так как он значительно большей вычислительной сложности, чем MD5 или SHA1. [17]

Итак, для того чтобы сделать хранение паролей пользователей веб-приложения надежным, будет использоваться функция хеширования языка PHP. Для защиты передачи результатов голосования от клиента к серверу будет использоваться сертификат SSL.

#### 2.4.2 Описание защитных мер от SQL-инъекций посредством возможностей языка PHP

Для предотвращения SQL инъекций следует соблюдать основные правила:

Подпись и дата		<p>PHP, начиная с версии 5.5, предоставляет встроенное API хеширования паролей password_hash(), которое безопасно работает и с хешированием, и с проверкой паролей.</p> <p>При хешировании паролей рекомендуется применять алгоритм Blowfish, так как он значительно большей вычислительной сложности, чем MD5 или SHA1.</p> <p>[17]</p> <p>Итак, для того чтобы сделать хранение паролей пользователей веб-приложения надежным, будет использоваться функция хеширования языка PHP. Для защиты передачи результатов голосования от клиента к серверу будет использоваться сертификат SSL.</p> <p>2.4.2 Описание защитных мер от SQL-инъекций посредством возможностей языка PHP</p> <p>Для предотвращения SQL инъекций следует соблюдать основные правила:</p>					
Инв. № дубл.							Лист
Взам. инв. №							
Подпись и дата							
Инв. № подл.							Лист
ФАЭС.10.05.02.55.ПЗ							
Изм.	Лист	№ докум	Подпись	Дата	30		





Функция htmlspecialchars() выполняет фильтрацию переданной строки и заменяет все опасные символы в ней на подходящие HTML-мнемоники. Рассмотрим пример:

```
$text = "<script><script>"; // строка, полученная от пользователя
```

```
$safe_str = htmlspecialchars($text); // безопасная строка.
```

Если вывести эту строку, то увидим следующее: «&lt;script&gt;&lt;/script&gt;». Таким образом, внедрить исполняющий скрипт посредством формы сайта стало невозможным.

Немаловажным способом является установка флага HttpOnly в значение true при установке данных cookie:

```
setcookie ("hash", $hash, time()+60*60*24*30, "/", null, null, true);.
```

Последний параметр функции является булевый флаг HttpOnly, который установлен в true. Это позволит избежать кражи личных данных посредством скриптовых языков, вроде JavaScript.

## 2.6 Выводы по разделу

Во второй главе рассмотрена архитектура будущего веб-приложения и отображен принцип взаимодействия пользователя с системой. Также определена техническая база, которая будет использоваться при разработке веб-приложения, и рассмотрены практические способы предотвращения уязвимостей посредством возможностей языка PHP.

Ине. № подл.	Подпись и дата	Взам. инв. №	Ине. № дубл.	Подпись и дата					
Изм.	Лист	№ докум	Подпись	Дата					
					ФАЭС.10.05.02.55.ПЗ				Лист
									32



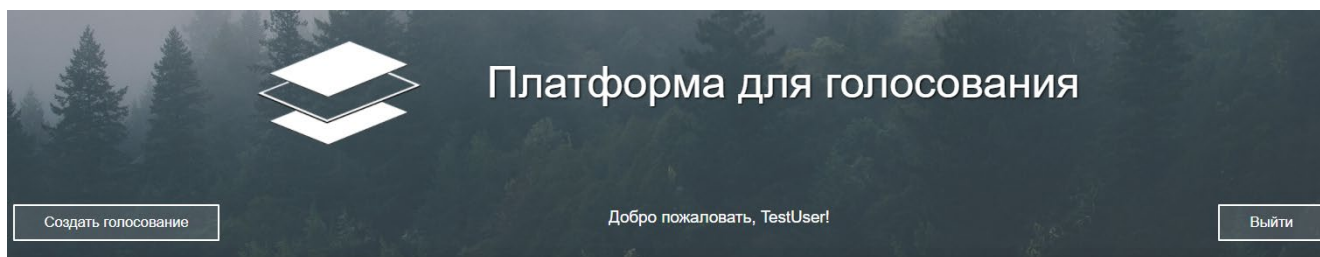


Рисунок 3.2 – Шапка сайта после авторизации пользователя

После успешной авторизации пользователя на шапке сайта появляется кнопка для создания голосования, приветствие и кнопка, чтобы выйти из аккаунта.

При нажатии кнопки «Войти» (рисунок 3.1) появляется окно с формой для авторизации пользователя (рисунок 3.3).

Рисунок 3.3 – Окно с формой для авторизации

При нажатии кнопки «Регистрация» (рисунок 3.1) появляется окно с формой для регистрации нового пользователя (рисунок 3.4).

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	ФАЭС.10.05.02.55.ПЗ					Лист 34
Изм.	Лист	№ докум	Подпись	Дата						

**РЕГИСТРАЦИЯ**

Введите логин

Введите Email

Введите пароль

Повторите пароль

**Зарегистрироваться**

☐ Нажимая кнопку "Зарегистрироваться", Я безоговорочно соглашаюсь с [правилами сайта](#).

Рисунок 3.4 – Окно с формой для регистрации нового пользователя

При нажатии на кнопку «Создать голосование» (рисунок 3.2) появляется окно с формой для создания нового голосования (рисунок 3.5).

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	<p>ФАЭС.10.05.02.55.ПЗ</p>					Лист
										35
Изм.	Лист	№ докум	Подпись	Дата						

×

СОЗДАНИЕ НОВОГО ГОЛОСОВАНИЯ

Тема голосования:

Описание:

Введите текст...

Осталось дней:

Варианты голосования:

1.

2.

+

–

Создать голосование

Рисунок 3.5 – Окно с формой для создания нового голосования

При создании голосования можно задать тему голосования, его описание, количество дней до его завершения и количество вариантов ответа, которые варьируются от 2 до 6 позиций.

На рисунке 3.6 представлен внешний вид созданного голосования, которое размещается на сайте в виде карточки.

Инв. № подл.	Подпись и дата	Инв. № докл.	Взам. инв. №	Подпись и дата							
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ					Лист	36



Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата

### 3.3.1 Описание структуры базы данных

voting_platform_bd results	voting_platform_bd voting
result_id : int	voting_id : int
option_name : varchar(255)	voting_theme : varchar(255)
option_votes_number : int	voting_content : text
voting_id : int	voting_date : datetime
	voting_left_days : float
	user_id : int

voting_platform_bd users	voting_platform_bd voted_users
user_id : int	id : int
user_login : varchar(255)	user_id : int
user_email : varchar(255)	voting_id : int
email_verified : int	
token : varchar(30)	
user_password : varchar(255)	
user_hash : varchar(255)	
user_ip : int	
err_login_number : int	
verified_login_code : varchar(10)	

Созданная база данных voting platform bd имеет четыре таблицы:

– `user id` – ключевой идентификатор пользователя;

*ΦΑЭС.10.05.02.55.ПЗ*

- user\_email – почта пользователя;
- email\_verified – строка, указывающая подтверждена ли почта пользователя;
- token – случайная строка длиной 30 символов, необходимая для подтверждения почты пользователя;
- user\_password – пароль пользователя, который хранится в виде хеша;
- user\_hash – случайно создаваемый хеш пользователя для записи в cookie;
- user\_ip – IP-адрес пользователя;
- err\_login\_number – счетчик неправильных попыток входа;
- verified\_login\_code – случайно сгенерированный код, необходимый для подтверждения пользователем того, что это он пытался зайти в аккаунт;

2) voting – таблица с данными о созданных голосованиях; содержит в себе следующие строки:

- voting\_id – ключевой идентификатор голосования;
- voting\_theme – тема голосования;
- voting\_content – описание голосования;
- voting\_date – дата создания голосования;
- voting\_left\_days – количество дней до закрытия голосования;
- user\_id – идентификатор пользователя из таблицы users, создавшего голосование;

3) voted\_users – таблицы для учёта проголосовавших пользователей, чтобы ограничить возможность повторного голосования в одном и том же голосовании; содержит в себе следующие строки:

- id – ключевой идентификатор таблицы;
- user\_id – идентификатор пользователя из таблицы users, уже проголосовавшего в данном голосовании;
- voting\_id – идентификатор голосования из таблицы voting, в котором уже проголосовал данный пользователь;

Ине. № подл.	Подпись и дата
Взам. инв. №	Ине. № дубл.
Подпись и дата	



4) results – таблица для подсчета голосов; содержит в себе следующие строки:

- result\_id – ключевой идентификатор варианта ответа;
- option\_name – название варианта ответа;
- option\_votes\_number – количество голосов за данный вариант ответа;
- voting\_id – идентификатор голосования из таблицы voting, указывающий к какому голосованию относится данный вариант ответа.

### 3.3.2 Описание разработанных php-скриптов

Общая структура проекта представлена на рисунке 3.10.

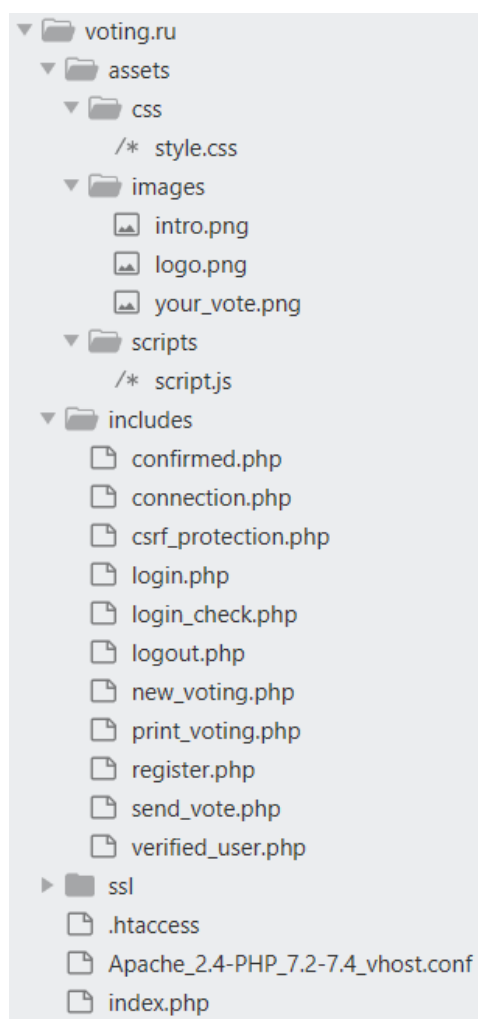
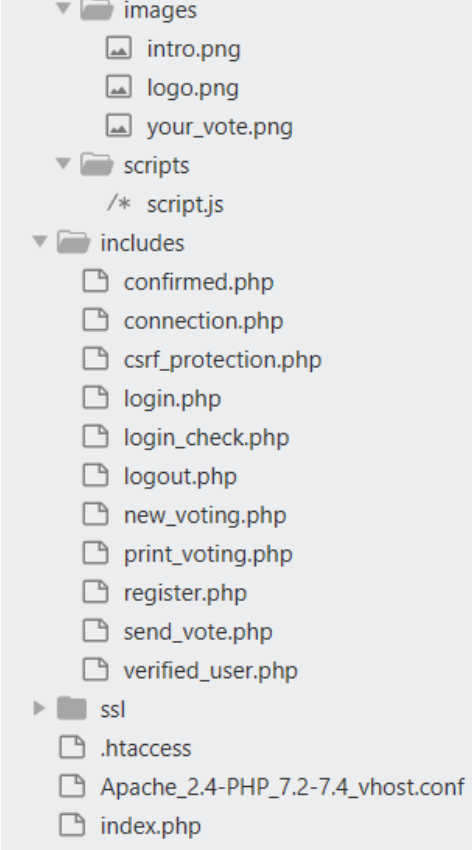


Рисунок 3.10 – Общая структура проекта

Изм.	Лист	№ докум	Подпись	Дата	<p>Рисунок 3.10 – Общая структура проекта</p> 	Лист
						40

Как видно из рисунка 3.10, все php-скрипты расположены в папке includes, помимо index.php, так как он является ядром проекта. Логическая связь скриптов разработанного веб-приложения отображена на рисунке 3.11.

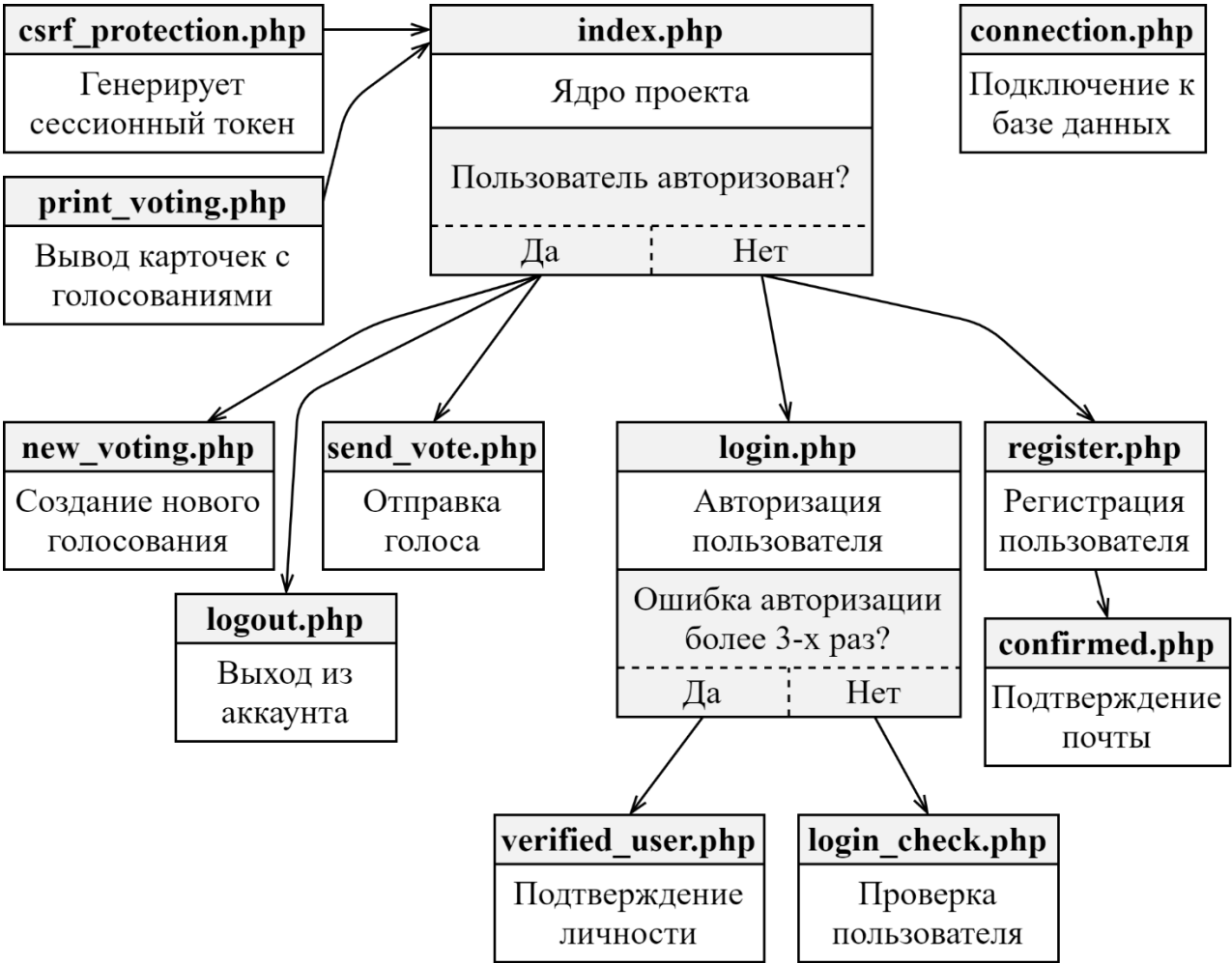


Рисунок 3.11 – Логическая связь скриптов веб-приложения

Скрипт connection.php используется для подключения к базе данных и присутствует во всех скриптах, которые взаимодействуют с ней.

Скрипт index.php является ядром проекта. Он отрисовывает основную разметку страницы сайта. В данном скрипте происходит проверка присутствия данных о пользователе в cookie. Если они имеются, то выводится приветствие и кнопки «Создать голосование» и «Выйти», а если данных нет, то есть пользова-

Если пользователь не авторизировался, то будут доступны кнопки «Войти» и «Зарегистрироваться».

При первом посещении сайта пользователю необходимо на нём зарегистрироваться, за это отвечает скрипт `register.php`.

Регистрация пользователя происходит с подтверждением почты, которую он введёт. Для этого отправляется письмо на указанную почту с ссылкой, содержащей в себе случайно сгенерированный токен. После перехода по этой ссылке аккаунт пользователя активируется.

Пароль пользователя хешируется посредством алгоритма Blowfish и сохраняется в базе данных в виде хеша, так как хранить пароль в открытом виде небезопасно.

Также осуществляется проверка на соответствие введенных паролей, на соответствие необходимой сложности пароля, на корректность введенного логина и на отсутствие в базе данных пользователя с такими же логином и почтой. Если будут обнаружены ошибки, то выведутся сообщения с их описанием, а процесс регистрации остановится.

Так как после регистрации пользователь должен подтвердить свою почту, то рассмотрим теперь скрипт, который отвечает за обработку перехода пользователя по ссылке, которая приходит ему на почту. В скрипте `confirmed.php` происходит сравнение токена из ссылки с тем токеном, который был записан у пользователя в базе данных при регистрации. Если они совпадают, то выходит сообщение о подтверждении почты.

Скрипт `login.php`, отвечающий за авторизацию пользователя, сначала проверяет наличие пользователя в базе данных. Если он присутствует, то проверяется подтверждена ли у него почта. Если нет, то выходит сообщение с ошибкой. В случае, если почта подтверждена, то происходит проверка правильности введенного пароля. Если пароль неверный, то выходит сообщение с ошибкой. Если пароль верный, то формируется уникальный хеш для данной сессии и происходит проверка, стоял ли флаг «Запомнить меня» при авторизации. Это означает привязку данной сессии к IP-адресу пользователя, то есть пока пользователь не разо-

рвет сессию, нажав на кнопку «Выйти», зайти на данный аккаунт с другого IP-адреса не получится. Если данный флаг установлен, то скрипт получает IP-адрес пользователя и переводит его в строку, затем отправляет ее в базу данных вместе с созданным ранее хешем. Если флаг не установлен, то предыдущий этап пропускается. Также в данном скрипте реализована защита от перебора пароля с помощью отправки специального письма пользователю, на аккаунт которого пытались зайти, введя трижды неправильные данные. Пользователю необходимо будет перейти по ссылке, отправленной в письме, чтобы суметь в дальнейшем авторизоваться на сайте.

В конце скрипта устанавливаются сессионные cookie и происходит переадресация на проверочный скрипт login\_check.php, который проверяет, не пытается ли кто-то осуществить авторизацию через другой IP-адрес.

Скрипт verified\_user.php проверяет, перешел ли пользователь по ссылке, которая была отправлена ему на почту после трех неудачных попыток входа в его аккаунт.

В скрипте csrf\_protection.php реализована защита от CSRF-атак путем создания специального токена сессии, который в дальнейшем привязывается к каждой форме отправки данных.

В скрипте new\_voting.php осуществляется отправка данных из формы создания голосования в базу данных.

Скрипт print\_voting.php отвечает за вывод всех карточек с голосованиями на страницу сайта. Для начала происходит получение из базы данных всей коллекции созданных голосований. Затем они поочередно выводятся на страницу сайта. В каждом голосовании происходит скрывание кнопки «Проголосовать» в случаях, если пользователь не авторизовался, кончилось время для голосования или пользователь уже проголосовал.

При истечении времени, отведенного на голосование, строка с датой и временем окончания голосования заменяется на надпись «Голосование завершилось». Также по завершению голосования происходит вывод итогов в виде полос в процентном соотношении напротив каждого варианта ответа. Справа от полос

Ине. № подл.	Подпись и дата	Взам. инв. №	Ине. № дубл.	Подпись и дата						Лист
										43
Изм.	Лист	№ докум	Подпись	Дата						

ФАЭС.10.05.02.55.ПЗ

отображается количество голосов за данный вариант ответа, а ниже приводится статистика, сколько из зарегистрированных пользователей проголосовали в данной теме.

Скрипт `send_vote.php` отвечает за отправку голоса за выбранный вариант. В скрипте осуществляется определение идентификатора голосования и варианта ответа. Затем к данному варианту ответа прибавляется голос, а идентификатор проголосовавшего пользователя заносится в таблицу `voted_users`, чтоб запретить ему повторное голосование в этой теме.

И последний скрипт `logout.php` отвечает за выход пользователя из аккаунта посредством удаления данных cookie.

Полный листинг php-скриптов приведен в приложении А.

### 3.4 Настройка протокола SSL

Протокол SSL использует сочетание открытого сертификата и закрытого ключа. Секретный ключ SSL хранится на сервере. Он используется для шифрования отправляемых на клиентские системы данных. Сертификат SSL находится в открытом доступе для всех, кто запрашивает этот контент. Его можно использовать для расшифровки контента, подписанного соответствующим ключом SSL.

Так как наш проект разрабатывается на локальном сервере, то необходимо создать самоподписанный SSL-сертификат. Для этого создадим два файла: `start.bat` и `config.txt`. Содержимое файла `start.bat`:

```
@echo off
set OPENSSL_CONF=W:\modules\http\Apache_2.4-PHP_7.2-7.4\conf\openssl.cnf
```

```
W:\modules\http\Apache_2.4-PHP_7.2-7.4\bin\openssl req -x509 -
sha256 -newkey rsa:2048 -nodes -days 5475 -keyout rootCA.key -
out rootCA.crt -subj "/CN=OSPanel/"
```

```
W:\modules\http\Apache_2.4-PHP_7.2-7.4\bin\openssl req -newkey  
rsa:2048 -nodes -days 5475 -keyout server.key -out server.csr -  
subj "/CN=VotingPlatform/"
```

Изн. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	ключа. Секретный ключ SSL хранится на сервере. Он используется для шифрования отправляемых на клиентские системы данных. Сертификат SSL находится в открытом доступе для всех, кто запрашивает этот контент. Его можно использовать для расшифровки контента, подписанного соответствующим ключом SSL.					
					Так как наш проект разрабатывается на локальном сервере, то необходимо создать самоподписанный SSL-сертификат. Для этого создадим два файла: start.bat и config.txt. Содержимое файла start.bat:					
					<pre>@echo off set OPENSSL_CONF=W:\modules\http\Apache_2.4-PHP_7.2-7.4\conf\openssl.cnf  W:\modules\http\Apache_2.4-PHP_7.2-7.4\bin\openssl req -x509 -sha256 -newkey rsa:2048 -nodes -days 5475 -keyout rootCA.key -out rootCA.crt -subj "/CN=OSPanel/"  W:\modules\http\Apache_2.4-PHP_7.2-7.4\bin\openssl req -newkey rsa:2048 -nodes -days 5475 -keyout server.key -out server.csr -subj "/CN=VotingPlatform/"</pre>					
Изн. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	ФАЭС.10.05.02.55.ПЗ					Лист
										44
Изн.	Лист	№ докум	Подпись	Дата						

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата

- *keyout*: эта строка указывает OpenSSL, где мы разместим создаваемый закрытый ключ;

Лист
45

– *out*: данный параметр указывает OpenSSL, куда поместить создаваемый сертификат;

– *extfile config.txt*: указывает, что необходимо открыть файл config.txt с конфигурацией;

– *subj*: указывает, кому выдан сертификат.

Файл config.txt содержит в себе:

```
nsComment = "OSPanel Generated Certificate"
basicConstraints = CA:false
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid,issuer
keyUsage = nonRepudiation, digitalSignature, keyEncipherment

subjectAltName = @alt_names
[alt_names]
DNS.1 = voting.ru
DNS.2 = www.voting.ru
```

Рассмотрим назначение строк:

– *nsComment*: строка, содержащая комментарий, который будет отображаться при просмотре сертификата;

– *basicConstraints*: указывает, относится ли сертификат к центру сертификации. Так как у нас самоподписанный сертификат, то стоит значение «false»;

– *subjectKeyIdentifier*: предоставляет средства идентификации сертификатов. Значение «hash» означает работу в автоматическом режиме;

– *authorityKeyIdentifier*: предоставляет средства идентификации открытого ключа, соответствующего закрытому ключу, используемому для подписи сертификата;

– *keyUsage*: определяет цель ключа, содержащегося в сертификате.

В конце файла указаны возможные доменные имена сайта.

Также в корневой папке сайта необходимо создать файл .htaccess со следующим содержанием:

```
RewriteEngine On
RewriteCond %{HTTPS}_%{HTTP_HOST}
^(?|off_(?:www\.)?(.*)|on_www\.(.*)) [NC]
RewriteRule .* https://%1/$0 [R=301,L]
```

Подпись и дата	
Инв. № дубл.	
Взам. инв. №	
Подпись и дата	
Инв. № подл.	

					ФАЭС.10.05.02.55.ПЗ	Лист
						46
Изм.	Лист	№ докум	Подпись	Дата		

Данный файл указывает на использование защищенного протокола HTTPS вместо HTTP.

На рисунке 3.12 изображено состояние подключения до настройки SSL-сертификата.

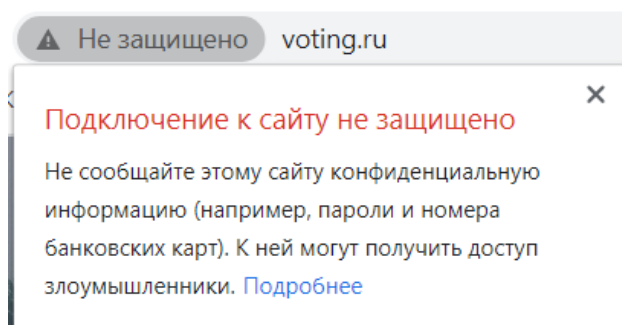


Рисунок 3.12 – Состояние подключения до настройки SSL-сертификата

На рисунке 3.13 изображено состояние подключения после настройки SSL-сертификата.

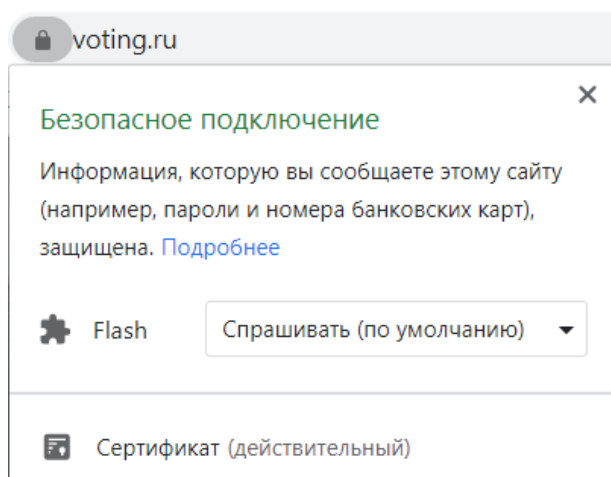


Рисунок 3.13 – Состояние подключения после настройки SSL-сертификата

Теперь подключение между клиентом и сервером защищено, значит SSL-сертификат установлен правильно.

Ине. № подл.	Подпись и дата
Ине. № инв.	Ине. № инв.
Взам. инв. №	Взам. инв. №
Подпись и дата	Подпись и дата
Ине. № подл.	Ине. № подл.

Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ	Лист
						47



### 3.5 Оценка защищенности разработанного веб-приложения

### 3.5.1 Демонстрация защитных мер при регистрации и авторизации пользователя

При регистрации пользователей используется хеширование паролей функцией Blowfish. Таким образом, в базе данных пароль хранится не в открытом виде, а в виде, так называемого, отпечатка (рисунок 3.14).

user_password
\$2y\$10\$IKsNwgqe4MgZv7Ss5EF0meCAY2S5x6NBfSnQ1Lc.oW5...
\$2y\$10\$vbzNIGPbsvoV9S7jWl.iRO5hoh1h1RPRRZlspheRvc...
\$2y\$10\$1ValRi9wY7MFHWerey0tMOKzta1LnX4APK24Da9gfMX...
\$2y\$10\$hdqJ6E5VIVx5R/S7tCmwgOggWOD2/uw9FRZz3jvFrkf...

Рисунок 3.14 – Хранение паролей в базе данных в виде хешей

При авторизации пользователя происходит сравнение хеша введенного пароля с хешем, хранящимся в базе данных.

Также после регистрации пользователю приходит письмо на почту, в котором содержится специальная ссылка, по которой пользователь должен перейти, чтобы подтвердить свой аккаунт. В противном случае не удастся авторизоваться. На рисунке 3.15 изображен пример содержимого письма.

To: new\_user@mail.ru  
Subject: Подтвердите Email на сайте  
X-PHP-Originating-Script: 0:register.php  
From: <mail@voting.ru>

Чтобы подтвердить Email, перейдите сюда:  
<http://voting.ru/includes/confirmed.php?token=ymQZRewai6N7ptLu9swM00YDbAiwji>

Рисунок 3.15 – Пример содержимого письма при регистрации пользователя

Данная ссылка является GET-запросом, который обрабатывается скриптом `confirmed.php`. Значение «*token=ymQZReWai6N7ptLu9swM00YDbAiWhj*» также при регистрации заносится в базу данных и в поле `email_verified` стоит «0», до тех пор пока пользователь не подтвердит почту (рисунок 3.16).

Рисунок 3.16 – Содержимое полей в базе данных до подтверждения почты

Рисунок 3.17 – Содержимое полей в базе данных после подтверждения почты

В разработанном веб-приложении осуществлена защита от перебора пароля. Её принцип заключается в том, что если трижды попытаются неправильно ввести данные при авторизации, то отправится специальное письмо пользователю, а зайти в аккаунт станет невозможно. Пример содержимого такого письма приведен ниже (рисунок 3.18).

To: new\_user@mail.ru  
Subject: Подтвердите Ваши действия на сайте  
X-PHP-Originating-Script: 0:login.php  
From: <mail@voting.ru>

Обнаружена попытка взлома аккаунта, перейдите сюда:  
[http://voting.ru/includes/verified\\_user.php?verified\\_code=g8Sz5RAWfu](http://voting.ru/includes/verified_user.php?verified_code=g8Sz5RAWfu)

Рисунок 3.18 – Пример содержимого письма при попытке взлома аккаунта

Принцип работы аналогичен с подтверждением почты при регистрации. Переменная «*verified\_code=g8Sz5RAWfu*» также заносится в базу данных. После того, как пользователь перейдет по ссылке, то данное поле в базе данных очистится, а счетчик неудачных попыток входа обнулится.

### 3.5.2 Оценка защищенности от CSRF атаки

Подключаемый к странице скрипт `csrf_protection.php` генерирует токен, который записывается в данные сессии. Выведем сгенерированный токен на странице сайта с помощью команды «`echo "<br>".$_SESSION['csrf_token'] = ".$_SESSION['csrf_token']."<br>;`» (рисунок 3.19).

SESSION['csrf token'] = 344a0c3cb86015b3287db4b51c0c2039d42b692dfb00b34df7f5954ceed5c4b

Рисунок 3.19 – Токен сессии, предназначенный для защиты от CSRF атаки

К каждой форме в разметке был добавлен скрытый элемент:

```
<input type="hidden" name="csrf_token" value="<?echo $csrf_token?>">
```

В данном элементе в атрибуте «value» прописывается сгенерированный токен сессии. И затем, при любой отправке формы, система сверяет токен, пропи-

санные в переменной \$\_SESSION, с токеном, прописанным в скрытых элементах форм. То есть проверяется, что пользователь лично в своей сессии совершает это действие, а не кто-то за него.

Таким, образом, если попробовать скомпрометировать токен, записанный в скрытом блоке (рисунок 3.20), то при отправке формы выйдет сообщение об ошибке (рисунок 3.21).

```

▼<form class="auth" action="includes/login.php" method="POST">
  <input id="login" type="text" name="login" placeholder="Логин">
  <input id="password" type="password" name="password" placeholder="Пароль">
  ▶<label for="remember_me">...</label>
  <input type="hidden" name="csrf_token" value="sdfwefwef23r2rf23d32d32d32f">
  <input type="submit" name="submit" value="Войти">
</form>

```

Рисунок 3.20 – Подделка сессионного токена

Произошла ошибка!

Рисунок 3.21 – Сообщение об ошибке

Таким образом, можно сделать вывод, что веб-приложение защищено от CSRF атак.

### 3.5.3 Оценка защищенности от XSS атаки

Рассмотрим вариант, когда защиты от XSS атаки не предусмотрено. Тогда злоумышленник может внедрить в страницу сайта скрипт посредством создания новой карточки для голосования (рисунок 3.22).

Ине. № подл.	Подпись и дата
Взам. инв. №	Инв. № дубл.
Подпись и дата	
Ине. № подл.	

СОЗДАНИЕ НОВОГО ГОЛОСОВАНИЯ

Тема голосования:

Новое голосование

Описание:

```
<script>alert
(document.cookie);</script>
```

Рисунок 3.22 – Внедрение скрипта на страницу сайта

После создания карточки голосования с таким текстом в разметку страницы встраивается исполняющий скрипт (рисунок 3.23), который выводит всплывающее окно с данными из cookie (рисунок 3.24).

```

<div class="voting_structure">
  <p class="voting_theme">Новое голосование</p>
  <p class="voting_content">
    <script>alert (document.cookie);</script> ==
  </p>

```

Рисунок 3.23 – Встроенный в разметку страницы скрипт

Подтвердите действие на странице voting.ru

PHPSESSID=rpluib2q2godsdmamn3t76811tn6r7jb; id=32;

login=NewUser; hash=169b833f62da902ec408ed638bbf6207

ОК

Рисунок 3.24 – Пользовательские данные из cookie

Ине. № подл.	Подпись и дата	Ине. № дубл.	Взам. инв. №	Подпись и дата	<div> <div>ФАЭС.10.05.02.55.ПЗ</div> <div>Лист</div> <div>52</div> </div>			
Изм.	Лист	№ докум	Подпись	Дата				

Добавим защиту пользовательских cookie посредством флага HttpOnly. Теперь посредством скрипта нельзя получать эти значения, во всплывающем окне пропадают пользовательские данные (рисунок 3.25).

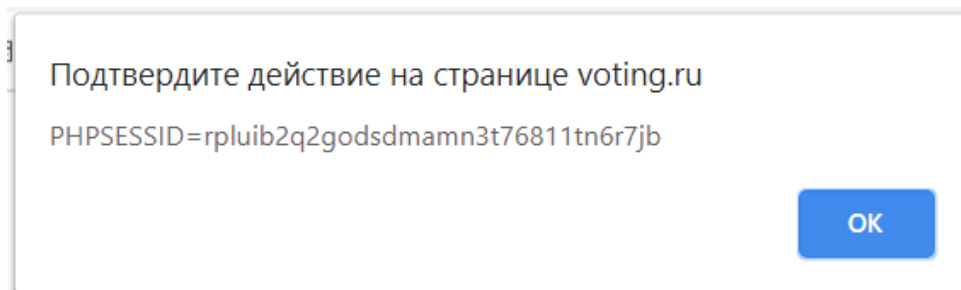


Рисунок 3.25 – Пользовательские cookie теперь не выводятся

Теперь заменим специальные символы HTML мнемониками, тогда скрипт не внедрится в разметку, а будет отображен простым текстом. В базе данных будет строка с текстом, в которой используются мнемоники (рисунок 3.26).



Рисунок 3.26 – Текстовая строка с мнемониками в базе данных

Можно сделать вывод, что веб-приложение защищено от XSS.

### 3.6 Выводы по разделу

В данном разделе было продемонстрировано разработанное веб-приложение для электронного голосования, а также оценен уровень его защищенности.

Инв. № подл.	Подпись и дата	<div><div>voting_content</div><div>&amp;lt;script&amp;gt;alert(document.cookie);&amp;lt;/script&amp;g...</div></div>									
Инв. № дубл.		Рисунок 3.26 – Текстовая строка с мнемониками в базе данных									
Взам. инв. №		Можно сделать вывод, что веб-приложение защищено от XSS.									
Подпись и дата		3.6 Выводы по разделу									
		В данном разделе было продемонстрировано разработанное веб-приложение для электронного голосования, а также оценен уровень его защищенности.									
							ФАЭС.10.05.02.55.ПЗ				Лист
											53
Изм.	Лист	№ докум	Подпись	Дата							



Таблица 4.1 – Требования к условиям труда [18]

№	Требования
Требования к помещениям для работы с ПЭВМ	
1	Естественное и искусственное освещение должно соответствовать требованиям действующей нормативной документации. Окна в помещениях, где эксплуатируется вычислительная техника, преимущественно должны быть ориентированы на север и северо-восток. Оконные проемы должны быть оборудованы регулируемыми устройствами типа: жалюзи, занавесей, внешних козырьков и др.
2	Площадь на одно рабочее место пользователей ПЭВМ с ВДТ на базе электронно-лучевой трубки (ЭЛТ) должна составлять не менее 6 м <sup>2</sup> и с ВДТ на базе плоских дискретных экранов (жидкокристаллические, плазменные) – 4,5 м <sup>2</sup> .
3	Для внутренней отделки интерьера помещений, где расположены ПЭВМ, должны использоваться диффузно отражающие материалы с коэффициентом отражения для потолка - 0,7 - 0,8; для стен - 0,5 - 0,6; для пола - 0,3 - 0,5.
4	Помещения, где размещаются рабочие места с ПЭВМ, должны быть оборудованы защитным заземлением (занулением) в соответствии с техническими требованиями по эксплуатации.
5	Не следует размещать рабочие места с ПЭВМ вблизи силовых кабелей и вводов, высоковольтных трансформаторов, технологического оборудования, создающего помехи в работе ПЭВМ.
Требования к микроклимату, содержанию аэроионов и вредных химических веществ в воздухе на рабочих местах, оборудованных ПЭВМ	
1	В производственных помещениях, в которых работа с использованием ПЭВМ является основной (диспетчерские, операторские, расчетные, кабины и посты управления, залы вычислительной техники и др.) и связана с нервно-эмоциональным напряжением, должны обеспечиваться оптимальные параметры микроклимата для категории работ 1а и 1б в соответствии с действующими санитарно-эпидемиологическими нормативами микроклимата производственных помещений. На других рабочих местах следует поддерживать параметры микроклимата на допустимом уровне, соответствующем требованиям указанных выше нормативов. Подробнее в СанПиН 2.2.4.3359-16 Санитарно-эпидемиологические требования к физическим факторам на рабочих местах.
2	В помещениях, оборудованных ПЭВМ, проводится ежедневная влажная уборка и систематическое проветривание после каждого часа работы на ПЭВМ.
3	Уровни положительных и отрицательных аэроионов в воздухе помещений, где расположены ПЭВМ, должны соответствовать действующим санитарно-эпидемиологическим нормативам.
4	Содержание вредных химических веществ в производственных помещениях, в которых работа с использованием ПЭВМ является основной (диспетчерские, операторские, расчетные, кабины и посты управления, залы вычислительной техники и др.), не должно превышать предельно допустимых концентраций загрязняющих веществ в атмосферном воздухе населенных мест в соответствии с действующими гигиеническими нормативами.

Ине. № подл.	Подпись и дата
Взам. инв. №	Ине. № дубл.
Подпись и дата	Подпись и дата

Изм.	Лист	№ докум	Подпись	Дата
------	------	---------	---------	------

ФАЭС.10.05.02.55.ПЗ

Лист

55



Продолжение таблицы 4.1

№	Требования
Требования к уровням шума и вибрации на рабочих местах, оборудованных ПЭВМ	
1	В производственных помещениях при выполнении основных или вспомогательных работ с использованием ПЭВМ уровни шума на рабочих местах не должны превышать предельно допустимых значений, установленных для данных видов работ в соответствии с действующими санитарно-эпидемиологическими нормативами. Подробнее в СанПиН 2.2.4.3359-16 Санитарно-эпидемиологические требования к физическим факторам на рабочих местах
2	Шумящее оборудование (печатающие устройства, серверы и т.п.), уровни шума которого превышают нормативные, должно размещаться вне помещений с ПЭВМ
Требования к освещению на рабочих местах, оборудованных ПЭВМ	
1	Рабочие столы следует размещать таким образом, чтобы видеодисплейные терминалы были ориентированы боковой стороной к световым проемам, чтобы естественный свет падал преимущественно слева
2	Искусственное освещение в помещениях для эксплуатации ПЭВМ должно осуществляться системой общего равномерного освещения.
3	Освещенность на поверхности стола в зоне размещения рабочего документа должна быть 300 - 500 лк. Освещение не должно создавать бликов на поверхности экрана. Освещенность поверхности экрана не должна быть более 300 лк.
4	Следует ограничивать прямую блескость от источников освещения, при этом яркость светящихся поверхностей (окна, светильники и др.), находящихся в поле зрения, должна быть не более 200 кд/м <sup>2</sup> .
5	Следует ограничивать отраженную блескость на рабочих поверхностях (экран, стол, клавиатура и др.) за счет правильного выбора типов светильников и расположения рабочих мест по отношению к источникам естественного и искусственного освещения, при этом яркость бликов на экране ПЭВМ не должна превышать 40 кд/м <sup>2</sup> и яркость потолка не должна превышать 200 кд/м <sup>2</sup> .
6	В качестве источников света при искусственном освещении следует применять преимущественно люминесцентные лампы типа ЛБ и компактные люминесцентные лампы (КЛЛ). При устройстве отраженного освещения в производственных и административно-общественных помещениях допускается применение металлогалогенных ламп.
7	Применение светильников без рассеивателей и экранирующих решеток не допускается.
8	Коэффициент запаса (Кз) для осветительных установок общего освещения должен приниматься равным 1,4.
9	Коэффициент пульсации не должен превышать 5%.
Требования к уровням электромагнитных полей на рабочих местах, оборудованных ПЭВМ	
1	Временные допустимые уровни ЭМП, создаваемых ПЭВМ на рабочих местах пользователей представлены в таблице 4.2
Требования к визуальным параметрам ВДТ, контролируемым на рабочих местах	
1	Предельно допустимые значения визуальных параметров ВДТ, контролируемые на рабочих местах, представлены в таблице 4.3

Ине. № подл.	Подпись и дата
Взам. инв. №	Ине. № дубл.
Подпись и дата	Подпись и дата

Изм.	Лист	№ докум	Подпись	Дата
------	------	---------	---------	------

ФАЭС.10.05.02.55.ПЗ

Таблица 4.2 - Временные допустимые уровни ЭМП, создаваемых ПЭВМ на рабочих местах [18]

Наименование параметров		ВДУ
Напряженность электрического поля	в диапазоне частот 5 Гц - 2 кГц	25 В/м
	в диапазоне частот 2 кГц - 400 кГц	2,5 В/м
Плотность магнитного потока	в диапазоне частот 5 Гц - 2 кГц	250 нТл
	в диапазоне частот 2 кГц - 400 кГц	25 нТл
Напряженность электростатического поля		15 кВ/м

Таблица 4.3 - Визуальные параметры ВДТ, контролируемые на рабочих местах [18]

N п/п	Параметры	Допустимые значения
1	Яркость белого поля	Не менее 35 кд/кв. м
2	Неравномерность яркости рабочего поля	Не более +/- 20%
3	Контрастность (для монохромного режима)	Не менее 3:1
4	Временная нестабильность изображения (мелькания)	Не должна фиксироваться
5	Пространственная нестабильность изображения (дрожание)	Не более $2 \times 1E(-4L)$ , где L - проектное расстояние наблюдения, мм

#### 4.3 Эргономические требования к рабочему месту пользователя

Требования эргономики – это комплекс мер, направленных на обеспечение эффективности, безопасности и комфортности рабочего места. Продуманный с точки зрения эргономики офис позволяет работодателю увеличить производительность труда сотрудников, обеспечивает здоровье персонала и способствует созданию благоприятного психологического климата в коллективе.

Согласно СанПиН 2.2.2/2.4.1340-03, общие требования к организации рабочих мест пользователей ПЭВМ сведены в таблицу 4.4.

Ине. № подл.	Подпись и дата
Взам. инв. №	Ине. № дубл.
Подпись и дата	

Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ	Лист
						57

Таблица 4.4 – Общие требования к организации рабочих мест пользователей ПЭВМ [18]

№	Требования
1	При размещении рабочих мест с ПЭВМ расстояние между рабочими столами с видеомониторами (в направлении тыла поверхности одного видеомонитора и экрана другого видеомонитора) должно быть не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов - не менее 1,2 м.
2	Рабочие места с ПЭВМ в помещениях с источниками вредных производственных факторов должны размещаться в изолированных кабинах с организованным воздухообменом.
3	Рабочие места с ПЭВМ при выполнении творческой работы, требующей значительного умственного напряжения или высокой концентрации внимания, рекомендуется изолировать друг от друга перегородками высотой 1,5 - 2,0 м.
4	Экран видеомонитора должен находиться от глаз пользователя на расстоянии 600 - 700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов.
5	Конструкция рабочего стола должна обеспечивать оптимальное размещение на рабочей поверхности используемого оборудования с учетом его количества и конструктивных особенностей, характера выполняемой работы. При этом допускается использование рабочих столов различных конструкций, отвечающих современным требованиям эргономики. Поверхность рабочего стола должна иметь коэффициент отражения 0,5 - 0,7.
6	Конструкция рабочего стула (кресла) должна обеспечивать поддержание рациональной рабочей позы при работе на ПЭВМ, позволять изменять позу с целью снижения статического напряжения мышц шейно-плечевой области и спины для предупреждения развития утомления. Тип рабочего стула (кресла) следует выбирать с учетом роста пользователя, характера и продолжительности работы с ПЭВМ.
7	Поверхность сиденья, спинки и других элементов стула (кресла) должна быть полумягкой, с нескользящим, слабо электризующимся и воздухопроницаемым покрытием, обеспечивающим легкую очистку от загрязнений.

Требования к организации и оборудованию рабочих мест с ПЭВМ для взрослых пользователей приведены в таблице 4.5.

Таблица 4.5 – Требования к организации и оборудованию рабочих мест с ПЭВМ для взрослых пользователей [18]

№	Требования
1	Высота рабочей поверхности стола для взрослых пользователей должна регулироваться в пределах 680 - 800 мм; при отсутствии такой возможности высота рабочей поверхности стола должна составлять 725 мм.

Ине. № подл.	Подпись и дата
Взам. инв. №	Ине. № дубл.

Изм.	Лист	№ докум	Подпись	Дата	ФАС.10.05.02.55.ПЗ	Лист
						58

# Продолжение таблицы 4.5

№	Требования
2	Модульными размерами рабочей поверхности стола для ПЭВМ, на основании которых должны рассчитываться конструктивные размеры, следует считать: ширину 800, 1000, 1200 и 1400 мм, глубину 800 и 1000 мм при нерегулируемой его высоте, равной 725 мм.
3	Рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной - не менее 500 мм, глубиной на уровне колен - не менее 450 мм и на уровне вытянутых ног - не менее 650 мм.
4	Конструкция рабочего стула должна обеспечивать: <ul style="list-style-type: none"> <li>- ширину и глубину поверхности сиденья не менее 400 мм;</li> <li>- поверхность сиденья с закругленным передним краем;</li> <li>- регулировку высоты поверхности сиденья в пределах 400 - 550 мм и углам наклона вперед до 15 град, и назад до 5 град.;</li> <li>- высоту опорной поверхности спинки 300 +20 мм, ширину - не менее 380 мм и радиус кривизны горизонтальной плоскости - 400 мм;</li> <li>- угол наклона спинки в вертикальной плоскости в пределах +30 градусов;</li> <li>- регулировку расстояния спинки от переднего края сиденья в пределах 260 - 400 мм;</li> <li>- стационарные или съемные подлокотники длиной не менее 250 мм и шириной - 50 - 70 мм;</li> <li>- регулировку подлокотников по высоте над сиденьем в пределах 230 +30 мм и внутреннего расстояния между подлокотниками в пределах 350 -500 мм.</li> </ul>
5	Рабочее место пользователя ПЭВМ следует оборудовать подставкой для ног, имеющей ширину не менее 300 мм, глубину не менее 400 мм, регулировку по высоте в пределах до 150 мм и по углу наклона опорной поверхности подставки до 20°. Поверхность подставки должна быть рифленой и иметь по переднему краю бортик высотой 10 мм.
6	Клавиатуру следует располагать на поверхности стола на расстоянии 100 - 300 мм от края, обращенного к пользователю или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

## 4.4 Требования охраны труда офисных работников

Согласно ст. 209 ТК РФ, требованиями охраны труда являются государственные нормативные требования охраны труда, в том числе стандарты безопасности труда, а также требования охраны труда, установленные правилами и инструкциями по охране труда. [20]

Требования охраны труда офисных работников сведены в таблицу 4.6.

Ине. № подл.	Подпись и дата
Взам. инв. №	Ине. № дубл.
Подпись и дата	

Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ	Лист
						59

Таблица 4.6 – Требования охраны труда офисных работников

№	Требования
Общие требования безопасности	
1	Офисный работник обязан соблюдать действующие на предприятии правила внутреннего трудового распорядка и графики работы, которыми предусматривается: время начала и окончания работы (смены), перерывы для отдыха и питания, порядок предоставления дней отдыха, чередование смен и другие вопросы использования рабочего времени.
2	Офисный работник обязан: – пользоваться исправными выключателями, розетками, вилками, патронами и другой электроарматурой; – не оставлять без присмотра включенное оборудование и электроприборы, отключать электрическое освещение (кроме аварийного) по окончании работы; – курить только в специально отведенных и оборудованных местах; при использовании в работе горючих и легковоспламеняющихся веществ убирать их в безопасное в пожарном отношении место, не оставлять использованный обтирочный материал в помещении по окончании работы; – соблюдать действующие Правила пожарной безопасности.
3	Офисный работник обязан соблюдать правила личной гигиены: – приходить на работу в чистой одежде и обуви; – постоянно следить за чистотой тела, рук, волос; – мыть руки с мылом после посещения туалета, соприкосновения с загрязненными предметами, по окончании работы.
4	За нарушение (невыполнение) требований нормативных актов об охране труда офисный работник привлекается к дисциплинарной, а в соответствующих случаях – материальной и уголовной ответственности в порядке, установленном законодательством РФ, локальными нормативными актами.
5	На рабочем месте офисный работник получает первичный инструктаж по безопасности труда и проходит: – стажировку; – обучение устройству и правилам эксплуатации используемого оборудования; – проверку знаний по электробезопасности (при использовании оборудования, работающего от электрической сети), теоретических знаний и приобретенных навыков безопасных способов работы.
6	Во время работы офисный работник проходит повторный инструктаж по безопасности труда на рабочем месте – один раз в полгода.
Требования безопасности перед началом работы	
1	Офисный работник обязан подготовить рабочую зону для безопасной работы: – проверить оснащенность рабочего места; – проверить путем внешнего осмотра достаточность освещенности и исправность выключателей и розеток; – осуществить осмотр электрооборудования (проверку комплектности и надежности крепления деталей; проверку путем внешнего осмотра исправности кабеля (шнура); проверку четкости работы выключателя; использовать только штатные приспособления).
2	Офисный работник обязан доложить руководителю при обнаружении дефектов в электрооборудовании и не эксплуатировать неисправное электрооборудование.
3	Включение электрооборудования производить вставкой исправной вилки в исправную розетку для бытовых приборов.
4	Офисный работник во время работы с электрооборудованием обязан поддерживать порядок на рабочем месте.

Ине. № подл.	Подпись и дата
Взам. инв. №	Ине. № дубл.
Подпись и дата	

Изм.	Лист	№ докум	Подпись	Дата
------	------	---------	---------	------

ФАЭС.10.05.02.55.ПЗ

Продолжение таблицы 4.6

№	Требования
5	При работе с электрооборудованием запрещается: – оставлять включенное электрооборудование без надзора; – передавать электрооборудование лицам, не имеющим права работать с ним; – снимать средства защиты; – дергать за подводящий провод для отключения; – держать палец на выключателе при переносе электрооборудования; – натягивать, перекручивать и перегибать подводящий кабель; – ставить на кабель (шнур) посторонние предметы; – допускать касание кабеля (шнура) с горячими или теплыми предметами.
6	Офисный работник обязан выполнять с электрооборудованием только ту работу, для которой предназначено электрооборудование.
7	Если во время работы обнаружится неисправность электрооборудования или работающий с ним почувствует хотя бы слабое действие тока, работа должна быть немедленно прекращена и неисправное электрооборудование должно быть сдано на проверку или в ремонт.
8	Отключение электрооборудования необходимо производить: – при перерыве в работе; – при окончании рабочего процесса.
Требования безопасности во время работы	
1	Офисный работник должен выполнять только ту работу, по которой прошел обучение, инструктаж по охране труда и к которой допущен работником, ответственным за безопасное выполнение работ.
2	Не поручать свою работу посторонним лицам.
3	Во время нахождения на рабочем месте офисный работник не должен совершать действий, которые могут повлечь за собой наступление несчастного случая: – не качаться на стуле; – не касаться оголенных проводов; – не работать на оборудовании мокрыми руками; – не размахивать острыми и режущими предметами.
4	Соблюдать правила перемещения в помещении и на территории организации, пользоваться только установленными проходами. Не загромождать установленные проходы и проезды.
5	Хранить документацию в шкафах в специально оборудованном кабинете.
6	Вследствие того что большая часть времени посвящена работе на компьютере, необходимо каждые два часа делать перерыв на 15 минут для снижения утомляемости общефизического характера.
7	Офисному работнику во время работы запрещается: – допускать захламленность рабочего места бумагой в целях недопущения накопления органической пыли; – производить отключение питания во время выполнения активной задачи; – производить частые переключения питания; – включать сильно охлажденное (принесенное с улицы в зимнее время) оборудование; – производить самостоятельно вскрытие и ремонт оборудования.
Требования безопасности в аварийных ситуациях	
1	В аварийной обстановке следует оповестить об опасности окружающих людей и действовать в соответствии с планом ликвидации аварий.
2	В случае возникновения возгорания или пожара необходимо немедленно сообщить об этом в пожарную часть, окриком предупредить окружающих людей и принять меры для тушения пожара.

Ине. № подл.	Подпись и дата	Взам. инв. №	Ине. № дубл.	Подпись и дата

Изм.	Лист	№ докум	Подпись	Дата

ФАЭС.10.05.02.55.ПЗ

## Продолжение таблицы 4.6

№	Требования
3	При травмировании, отравлении или внезапном заболевании прекратить работу и обратиться за помощью к медицинскому работнику, а в случае его отсутствия оказать себе или другим пострадавшим первую доврачебную помощь и сообщить о случившемся непосредственному руководителю, далее действовать по его указанию.
4	В ситуациях, угрожающих жизни и здоровью, покинуть опасный участок.
Требования безопасности по окончании работы	
1	По окончании работы офисный работник должен произвести уборку рабочего места.
2	Офисный работник должен: – отключить электрооборудование; – проверить противопожарное состояние кабинета; – закрыть окна, выключить свет, закрыть двери.

### 4.5 Пожарная безопасность

Каждый сотрудник независимо от занимаемой должности обязан знать и строго выполнять правила пожарной безопасности, не допускать действий, которые могут привести к пожару. Основные причины пожаров на предприятиях - неосторожное обращение с огнем, оставленные без присмотра электроприборы, проведение с нарушениями требований правил пожарной безопасности огневых, строительных и других пожароопасных работ, курение в не установленных местах, использование легковоспламеняемых веществ, нарушение технологий.

Работодатели обязаны обеспечить полное, своевременное и неукоснительное выполнение правил, норм и условий пожарной безопасности, персональная ответственность за пожарную безопасность возлагается на директора или на его заместителей, а в подразделениях (на участках, в цехах, лабораториях, отделах и т. д.) – на руководителей этих подразделений. [16]

Работодатель или лицо, на которого возложено проведение работ по пожарной безопасности в организации, обязан:

- назначить лиц, ответственных за пожарную безопасность в структурных подразделениях;
- квалифицировать все рабочие места по категориям взрывоопасной и пожарной опасности;

Ине. № подл.	Подпись и дата	Взам. инв. №	Ине. № дубл.	Подпись и дата	<p>ФАЗС.10.05.02.55.ПЗ</p>					Лист
										62
Изм.	Лист	№ докум	Подпись	Дата						





При обнаружении пожара следует немедленно сообщить об этом по номеру 01, 101 или 112 (101, 112 с мобильного телефона) и без паники доложить:

- что горит, чему угрожает;
- адрес объекта;
- есть ли опасность для людей;
- назвать свою фамилию;
- немедленно обесточить всю электротехнику в помещении;
- обеспечить эвакуацию людей.

### Дальнейшие шаги:

- 1) сообщение повторить руководителю, работнику службы безопасности, начальнику отдела и приступить к тушению пожара огнетушителями, подручными средствами;
- 2) подготовить к эвакуации материальные ценности, документацию;
- 3) слушать распоряжения руководителя отдела, организованно покинуть здание;
- 4) рассмотреть вариант эвакуации через запасные выходы, пожарную лестницу, соседние помещения; организовать встречу подразделений пожарной охраны;
- 5) при невозможности покинуть здание (задымление, высокая температура) плотно закрыть дверь помещения, уплотнить тканью щели, вентиляционные отверстия, открыть окно и ждать пожарных. Стоит запомнить, что при задымлении над полом воздух более чист. Это может помочь при эвакуации и ожидании помощи.

#### 4.6 Выводы по разделу

В данной главе были рассмотрены характеристики условий труда при работе с ПК и эргономические требования к рабочему месту пользователя. Определены требования охраны труда офисных работников, а также рассмотрены правила пожарной безопасности.

## 5 Технико-экономическое обоснование работы

### 5.1 Постановка задачи

Целью выпускной квалификационной работы являлась разработка веб-приложения для защищенного электронного голосования. Веб-приложение является программным кодом, который, согласно ст. 1259 ГК РФ, относится к объектам авторских прав, таким образом, является интеллектуальной собственностью.

В данном разделе будут рассмотрены следующие вопросы:

- расчет трудоемкости и длительности работ;
- расчет себестоимости и цены программного продукта.

### 5.2 Расчет трудоемкости и длительности работ

В первую очередь необходимо составить план по разработке программного продукта, который представлен в таблице 5.1. [15]

Таблица 5.1 – План разработки программного продукта

Наименование этапов	Виды работ	Исполнитель (должность, квалификация)	Количество исполнителей
Анализ предметной области разработки	Определение объекта разработки	Студент	1
	Анализ основных угроз и уязвимостей	Студент	1
	Разработка модели нарушителя информационной безопасности	Студент	1
Проектирование	Планирование архитектуры веб-приложения	Студент	1
	Выбор языковых и программных средств разработки	Студент	1

Име. № подл.	Подпись и дата	Взам. инв. №	Име. № дубл.	Подпись и дата

Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ	Лист
						65

Продолжение таблицы 5.1

Проектирование	Выбор практических методов защиты веб-приложения	Студент	1
Разработка	Создание веб-интерфейса	Студент	1
	Разработка серверной логики	Студент	1
	Настройка протокола SSL	Студент	1
Тестирование и отладка	Оценка защищенности веб-приложения	Студент	1
Внедрение	Улучшение, оптимизация и устранение ошибок	Студент	1

Далее требуется рассчитать трудоемкость и длительность работ. Поскольку трудоемкость этапов и видов работ носит вероятностный характер, то предпочтительным будет использование метода экспертных оценок.

В этом методе для каждого этапа требуется экспертным путем определить три оценки трудоемкости, в днях:

- наименее возможная величина затрат,  $a_i$ ;
- наиболее вероятная величина затрат,  $m_i$ ;
- наиболее возможная величина затрат,  $b_i$  [15].

На основании экспертных оценок средняя величина для  $a_i$ ,  $m_i$  и  $b_i$  определяется по формуле (5.1):

$$\bar{T} = \frac{3T_{\text{рук}} + 2T_{\text{авт}}}{5}, \quad (5.1)$$

где  $\bar{T}$  – среднее время, полученное на основании экспертных оценок;

$T_{\text{рук}}$  – оценка затрат времени, данная руководителем;

$T_{\text{авт}}$  – оценка затрат времени, данная автором проекта.

Результаты расчета средней оценки затрат времени на разработку программного продукта приведены в таблице 5.2.

Име. № подл.	Подпись и дата
Взам. инв. №	Име. № дубл.
Подпись и дата	

Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ	Лист
						66



Стандартное отклонение G в целом по программному продукту рассчитывается по следующей формуле:

$$G = \sqrt{\sum G_i^2}, \quad (5.5)$$

где  $G$  – стандартное отклонение;

$G_i$  – стандартное отклонение по i-му этапу.

На основе расчетов математического ожидания (5.4) и стандартного отклонения (5.5) рассчитываем коэффициент вариации – коэффициент согласованности мнения экспертов. Коэффициент вариации рассчитывается по формуле:

$$v_i = \frac{G_i}{MO_i}, \quad (5.6)$$

где  $v_i$  – коэффициент вариации по  $i$ -му этапу [15].

Все произведенные расчеты сведены в таблицу 5.3.

Таблица 5.3 – Затраты на разработку программного продукта

Этапы разработки программного продукта	Средняя величина затрат по этапам, дни			Матем. ожидание (МО <sub>i</sub> , дни)	Станд. отклонение (G <sub>i</sub> , дни)	Коэффициент вариации (v <sub>i</sub> )
	Наименее возможная величина затрат (a <sub>i</sub> , дни)	Наиболее вероятная величина затрат (m <sub>i</sub> , дни)	Наиболее возможная величина затрат (b <sub>i</sub> , дни)			
1. Анализ предметной области разработки	5,8	8,8	14,8	9,30	1,5	0,161
2. Проектирование	12,4	15,8	19,4	15,83	1,17	0,074
3. Разработка	20,2	24,2	27,2	24,03	1,17	0,049
4. Тестирование и отладка	5,2	7,2	9,2	7,20	0,67	0,093
5. Внедрение	2,4	3,4	4,4	3,40	0,33	0,098
Итого	46	59,4	75	59,77	2,35	0,039

В итоге коэффициент вариации равен 0,039 и не превосходит 0,33. Поэтому мнения экспертов считаются согласованными.

### 5.3 Расчет себестоимости и цены программного продукта

Себестоимость программного продукта – это все виды затрат, понесенные при разработке продукта. Чтобы определить себестоимость разработки применяется метод экспертных оценок.

Себестоимость программного продукта определяется по формуле (5.7):

$$C = \frac{3}{m} \cdot k \cdot k_{\text{TEP}} \cdot k_{\text{ПП}} \cdot (t_1 + t_2) \cdot (1 + k_H) + 8 \cdot t_3 \cdot C_M + 8 \cdot t_4 \cdot C_H, \quad (5.7)$$

где  $З$  – среднемесячная заработная плата php-разработчика,  $З = 30000$ ;

$k_{\text{ТЕР}}$  – территориальный коэффициент,  $k_{\text{ТЕР}} = 1,2$  (для НСО);

$k_{\text{пр}}$  – коэффициент премии,  $k_{\text{пр}} = 1$ ;

$k$  – коэффициент, учитывающий страховые взносы (фонды пенсионного, социального и медицинского страхования),  $k = 1,3$ ;

$m$  – количество рабочих дней в месяце,  $m = 22$ ;

$k_H$  – коэффициент, учитывающий накладные расходы (отопление, освещение, уборка и т. д.),  $k_H = 0,4$ ;

$t_1$  – время, затраченное разработчиком на разработку требований к программе, т.е. подготовительное время, которое необходимо потратить, чтобы приступить к написанию программы и отладки программы, чел./дни;

$t_2$  – сборка устройства, составление алгоритма в программе, время, затраченное на написание и отладку программы, чел./дни;

$t_3$  – время, затраченное на разработку программы с использованием машинного времени, чел./дни;

$t_4$  – время работы в сети интернет, дни;

$C_{\text{И}}$  – стоимость 1 часа работы в сети интернет, руб. (оценивается через абонентскую плату);

$C_M$  – стоимость одного часа машинного времени.

Для расчета стоимости одного часа машинного времени, необходимо определить затраты на эксплуатацию ПК за год по следующей формуле:

$$C_m = \frac{З_{эл} + З_a + З_{компл} + З_{пр}}{T_{общ}}. \quad (5.8)$$

Общее время работы компьютера за год составляет:

$$T_{общ} = 22 * 12 * 8 = 2112 \text{ (часов)}$$

Затраты на электроэнергию за год работы (на данный момент тариф  $C_{эл}$  составляет 2,68 руб. за кВт/ч):

$$З_{эл} = T_{общ} * C_{эл} * P, \quad (5.9)$$

где  $P$  – потребляемая мощность ноутбука по паспортным данным в час,  $P = 135$  Вт/ч.

По (5.9) затраты на электроэнергию за год работы составляют:

$$З_{эл} = 2112 * 2,68 * 0,135 = 764,1 \text{ (руб.)}$$

Амортизационные отчисления в год определяются как процент отчисления на амортизацию от первоначальной стоимости основных производственных фондов. Процент отчисления на амортизацию, согласно ст. 258 НК РФ, составляет 34-50% от первоначальной стоимости ПК (компьютер относится ко второй группе имущества со сроком полезного использования свыше 2 лет до 3 лет включительно). Затраты на ПК определяются по формуле:

$$З_a = C * P_p, \quad (5.10)$$

где  $C$  – стоимость ноутбука, руб.;

$P_p$  – процент отчисления на амортизацию,  $P_p = 40\%$ .

Получим:

$$З_a = 55000 * 0,4 = 22000 \text{ (руб.)}$$

Затраты на комплектующие материалы составляют:

$$З_{компл} = 5000 \text{ (руб.)}$$

Прочие расходы составляют 5% от общей суммы затрат:

$$З_{пр} = \frac{0,05 * (З_{эл} + З_a + З_{компл})}{0,95}. \quad (5.11)$$

Ине. № подл.	Подпись и дата	Взам. инв. №	Ине. № дубл.	Подпись и дата						
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ					Лист
										70

По (5.11) прочие расходы равны:

$$З_{пр} = \frac{0,05 * (764,1 + 22000 + 5000)}{0,95} = 1461,27 \text{ (руб.)}$$

По формуле 5.8 стоимость одного часа машинного времени равна:

$$C_m = \frac{764,1 + 22000 + 5000 + 1461,27}{2112} = 13,84 \text{ (руб.)}$$

Тариф на услугу интернет составляет 635 руб. в месяц, следовательно, стоимость 1 часа работы в сети интернет равен:

$$C_{и} = \frac{635}{30} = 21,2 \text{ (руб.)}$$

Заключительным этапом расчета является распределение ранее рассчитанной трудоемкости (таблица 5.3) по 4 направлениям:

–  $t_1$  включает первые два этапа: анализ предметной области разработки и проектирование:

$$t_1 = 9,3 + 15,83 = 25,1 \text{ (дней)}$$

–  $t_2$  включает этапы: разработка, тестирование и отладка и внедрение:

$$t_2 = 24,03 + 7,2 + 3,4 = 34,6 \text{ (дней)}$$

–  $t_3$  включает время работы ПК для разработки программы:

$$t_3 = 50 \text{ (дней)}$$

–  $t_4$  включает время использования интернета для разработки программы:

$$t_4 = 45,5 \text{ (дней)}$$

Наконец, итоговая себестоимость программного продукта составляет:

$$C = \frac{30000}{22} \cdot 1,3 \cdot 1,2 \cdot 1 \cdot (25,1 + 34,6) \cdot (1 + 0,4) + 8 \cdot 50 \cdot 13,84 + 8 \cdot 45,5 \cdot 21,2 = 191050,25 \text{ (руб.)}$$

В случае, если программный продукт будет доработан и реализован на рынке, следует рассчитать цену по следующей формуле:

$$Ц = C * (1 + \frac{P}{100}), \quad (5.12)$$

где  $C$  – себестоимость разработки программы, руб;

$P$  – рентабельность, руб.

Инв. № подл.	Подпись и дата				Лист
	Инв. № дубл.				
	Взам. инв. №				
	Подпись и дата				
<p>– <math>t_2</math> включает этапы: разработка, тестирование и отладка и внедрение:</p> $t_2 = 24,03 + 7,2 + 3,4 = 34,6 \text{ (дней)}$ <p>– <math>t_3</math> включает время работы ПК для разработки программы:</p> $t_3 = 50 \text{ (дней)}$ <p>– <math>t_4</math> включает время использования интернета для разработки программы:</p> $t_4 = 45,5 \text{ (дней)}$ <p>Наконец, итоговая себестоимость программного продукта составляет:</p> $C = \frac{30000}{22} \cdot 1,3 \cdot 1,2 \cdot 1 \cdot (25,1 + 34,6) \cdot (1 + 0,4) + 8 \cdot 50 \cdot 13,84 + 8 \cdot 45,5 \cdot 21,2$ $= 191050,25 \text{ (руб.)}$ <p>В случае, если программный продукт будет доработан и реализован на рынке, следует рассчитать цену по следующей формуле:</p> $Ц = C * (1 + \frac{P}{100}), \tag{5.12}$ <p>где <math>C</math> – себестоимость разработки программы, руб;</p> <p><math>P</math> – рентабельность, руб.</p>					
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ



Определим цену программного продукта, при условии, что значение рентабельности равно 20%:

$$Ц = 191050,25 \cdot \left(1 + \frac{20}{100}\right) = 229260,3 \text{ (руб.)}$$

Цена с учетом налога на добавленную стоимость находится по формуле:

$$Ц_{\text{НДС}} = Ц * K_{\text{НДС}}, \quad (5.13)$$

где Ц – цена программного продукта;

$K_{\text{НДС}}$  – коэффициент, учитывающий ставку налога на добавленную стоимость (НДС),  $K_{\text{НДС}} = 1,20$ . [15]

Цена с учетом налога на добавленную стоимость составит:

$$Ц_{\text{НДС}} = 229260,3 * 1,20 = 275112,36 \text{ (руб.)}$$

#### 5.4 Выводы по разделу

В данном разделе была определены и рассчитаны трудоемкость и длительность работ, а также рассчитаны себестоимость и цена программного продукта.

Ине. № подл.	Подпись и дата	Взам. инв. №	Ине. № дубл.	Подпись и дата					
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ				
									Лист
									72

## Заключение

В результате выполнения выпускной квалификационной работы была достигнута поставленная цель и ее задачи.

В первой главе было определено назначение разрабатываемого веб-приложения. Также выявлены актуальные риски и угрозы: недостатки аутентификации, SQL-инъекции, XSS и CSRF. В качестве нарушителей ИБ определены: внешние и внутренние нарушители с низким и средним потенциалом.

Во второй главе рассмотрен принцип взаимодействия пользователя с веб-приложением. В качестве языковых средств разработки выбраны: HTML, CSS, JavaScript, PHP. В качестве программных средств выбраны: локальная серверная платформа Open Server Panel с HTTP-сервером Apache и СУБД MySQL. Также были выбраны практические методы защиты веб-приложения с использованием возможностей языка PHP.

В третьей главе продемонстрирован интерфейс разработанного веб-приложения, рассмотрена логика взаимодействия PHP-скриптов. Также был настроен SSL-сертификат и продемонстрирована его работа. В конце главы оценена защищенность разработанного веб-приложения, в следствие чего можно сказать, что веб-приложение защищено от рассмотренных в первой главе угроз и уязвимостей.

В четвертой и пятой главах были рассмотрены вопросы по безопасности жизнедеятельности и выполнено технико-экономическое обоснование.

Подпись и дата					<p>приложения, рассмотрена логика взаимодействия РНР-скриптов. Также был настроен SSL-сертификат и продемонстрирована его работа. В конце главы оценена защищенность разработанного веб-приложения, в следствие чего можно сказать, что веб-приложение защищено от рассмотренных в первой главе угроз и уязвимостей.</p> <p>В четвертой и пятой главах были рассмотрены вопросы по безопасности жизнедеятельности и выполнено технико-экономическое обоснование.</p>	
Инв. № дубл.						
Взам. инв. №						
Подпись и дата						
Инв. № подл.						
Изм.	Лист	№ докум	Подпись	Дата	<p><i>ФАЭС.10.05.02.55.ПЗ</i></p>	Лист
						73

## Список литературы

1 Open Server Panel // Open Server. Лучшая панель управления сервером для Windows. – URL: <https://ospanel.io/> (дата обращения: 05.11.20).

2 OWASP Top Ten 2017 // OWASP – 2017. – URL: <https://owasp.org/www-project-top-ten/2017/> (дата обращения: 19.10.20).

3 SSL-сертификаты бывают разные // Kaspersky daily. – URL: <https://www.kaspersky.ru/blog/certificates-are-different/20227/> (дата обращения: 01.11.20).

4 TLS и SSL: Необходимый минимум знаний // MNorin.com. – URL: <https://mnorin.com/tls-ssl-neobhodimy-j-minimum-znaniy.html/> (дата обращения: 01.11.20).

5 Wolf, P. Introducing Electronic Voting: Essential Considerations / P. Wolf, R. Nackerdien, D. Tuccinardi. – Stockholm.: Bulls Graphics, 2011. – 36 p.

6 Банк данных угроз безопасности информации // ФСТЭК России. – URL: <https://bdu.fstec.ru/threat/> (дата обращения: 25.10.20).

7 Буя, П.М. Защита информации в телекоммуникационных системах / П.М. Буя // Модель нарушителя информационной безопасности. – 2016. – №4. – С. 7-15.

8 ГОСТ Р 50922-2006. Защита информации. Основные термины и определения. – М.: Стандартинформ, 2008. – 8 с.

9 ГОСТ Р 53114-2008. Защита информации. Обеспечение информационной безопасности в организации. Основные термины и определения. – М.: Стандартинформ, 2009. – 16 с.

10 ГОСТ Р ИСО 9241-151-2014. Эргономика взаимодействия человек-система. Часть 151. Руководство по проектированию пользовательских интерфейсов сети Интернет. – М.: Стандартинформ, 2019. – 46 с.

11 Ерохина, О.В. Технологии электронного голосования в России / О.В. Ерохина // Вестник университета. – 2019. – № 11. – С. 5-11.

12 Иванов, М.А. Хеш-функции. Теория, применение и новые стандарты (часть 1) / М.А. Иванов, А.В. Стариковский. – 2017. – 31 с.

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	<p>https://bdu.fstec.ru/threat/ (дата обращения: 25.10.20).</p> <p>7 Буя, П.М. Защита информации в телекоммуникационных системах / П.М. Буя // Модель нарушителя информационной безопасности. – 2016. – №4. – С. 7-15.</p> <p>8 ГОСТ Р 50922-2006. Защита информации. Основные термины и определения. – М.: Стандартинформ, 2008. – 8 с.</p> <p>9 ГОСТ Р 53114-2008. Защита информации. Обеспечение информационной безопасности в организации. Основные термины и определения. – М.: Стандартинформ, 2009. – 16 с.</p> <p>10 ГОСТ Р ИСО 9241-151-2014. Эргономика взаимодействия человек-система. Часть 151. Руководство по проектированию пользовательских интерфейсов сети Интернет. – М.: Стандартинформ, 2019. – 46 с.</p> <p>11 Ерохина, О.В. Технологии электронного голосования в России / О.В. Ерохина // Вестник университета. – 2019. – № 11. – С. 5-11.</p> <p>12 Иванов, М.А. Хеш-функции. Теория, применение и новые стандарты (часть 1) / М.А. Иванов, А.В. Стариковский. – 2017. – 31 с.</p>					
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ					Лист
										74

23 Язык программирования PHP // Depix. – URL: [https://depix.ru/articles/yazyk\\_programmirovaniya\\_php/](https://depix.ru/articles/yazyk_programmirovaniya_php/) (дата обращения: 30.10.20).

Подпись и дата		18 Руководство по PHP. Что такое PHP? // PHP. – URL: <a href="https://www.php.net/manual/ru/intro-what-is.php/">https://www.php.net/manual/ru/intro-what-is.php/</a> (дата обращения: 28.10.20).			
Инв. № дубл.		19 СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы.			
Взам. инв. №		20 Трудовой кодекс Российской Федерации от 30.12.2001 № 197-ФЗ (ред. от 09.11.2020) // Собрание законодательства РФ. – 07.01.2002.			
Подпись и дата		21 Уязвимости и угрозы веб-приложений в 2019 году // Positive Technologies – 2018. – URL: <a href="https://www.ptsecurity.com/ru-ru/research/analytics/web-vulnerabilities-2020/">https://www.ptsecurity.com/ru-ru/research/analytics/web-vulnerabilities-2020/</a> (дата обращения: 20.10.20).			
Инв. № подл.		22 ЭММ, Д. Главные риски в онлайн-голосовании: мнение охотника за киберугрозами / Д. ЭММ // Information Security/ Информационная безопасность – 2018. – № 1. – URL: <a href="http://lib.itsec.ru/articles2/Oborandteh/glavnye-riski-v-onlayn-golosovanii-mnenie-ohotnika-za-kiberugrozami/">http://lib.itsec.ru/articles2/Oborandteh/glavnye-riski-v-onlayn-golosovanii-mnenie-ohotnika-za-kiberugrozami/</a> (дата обращения: 15.10.20).			
		23 Язык программирования PHP // Depix. – URL: <a href="https://depix.ru/articles/yazyk_programmirovaniya_php/">https://depix.ru/articles/yazyk_programmirovaniya_php/</a> (дата обращения: 30.10.20).			
		ФАЭС.10.05.02.55.ПЗ	Лист		
Изм.	Лист		№ докум	Подпись	Дата

## Приложение А

## Код разработанного веб-приложения

Ниже представлен код всех скриптов разработанного веб-приложения.

*index.php:*

```
<?php
include("includes/csrf_protection.php");
?>

<!DOCTYPE html>
<html lang="ru">
<head>
    <meta charset="UTF-8">
    <link rel="stylesheet" href="assets/css/style.css">
    <title>Платформа для голосования</title>
</head>
<body>
    <header>
        <div class="container">
            <a class="intro" href="index.php">
                <div class="logo">
                    
                </div>
                <div class="brand">
                    Платформа для голосования
                </div>
            </a>
            <div class="panel" id="panel">
                <? if (!empty($_COOKIE["id"])): ?>
                <div class="user_panel">
                    <button class="create_voting"
id="create_button">Создать голосование</button>
                    <?
                        echo "Добро пожаловать,
".$_COOKIE["login"]."!";
                    <?>
                    <form action="includes/logout.php" method="POST">
                        <input type="submit" value="Выйти">
                    </form>
                </div>
                <? else: ?>
                <div class="auth_reg">
                    <button class="auth_button"
id="auth_button">Вход</button>
                    <button class="registration"
id="reg_button">Регистрация</button>
                </div>
                <? endif ?>
            </div>
        </div>
    </header>

```

Изм.	Лист	№ докум	Подпись	Дата	Изм. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	<pre> <div class="logo">  </div> <div class="brand"> Платформа для голосования </div> &lt;/a&gt; <div class="panel" id="panel"> &lt;? if (!empty(\$_COOKIE["id"])): ?&gt; <div class="user_panel"> &lt;button class="create_voting" id="create_button"&gt;Создать голосование&lt;/button&gt; &lt;? echo "Добро пожаловать, ".\$_COOKIE["login"]."!"; ?&gt; &lt;form action="includes/logout.php" method="POST"&gt; &lt;input type="submit" value="Выйти"&gt; &lt;/form&gt; &lt;/div&gt; &lt;? else: ?&gt; <div &gt;="" &gt;вход&lt;="" &gt;регистрация&lt;="" &lt;="" &lt;?="" &lt;button="" <="" ?&gt;="" button&gt;="" class="registration" div&gt;="" endif="" id="reg_button" pre=""> </div></div></div></pre>
										<p style="text-align: center; font-size: 24px; font-weight: bold;">ФАС.10.05.02.55.ПЗ</p>

```

        </div>
    </header>
    <div class="container">
        <main>
            <div class="container_main">
                <h2 class="title">Последние темы голосований</h2>
                <div id="created_votings">
                    <?php
                        include("includes/print_voting.php");
                    ?>
                </div>
            </div>
        </main>
        <div class="modal" id="modal_auth">
            <div class="modal_container">
                <span class="modal_close"
id="modal_close_auth">&times;</span>
                <h3>Авторизация</h3>
                <form class="auth" action="includes/login.php" meth-
od="POST">
                    <input id="login" type="text" name="login" place-
holder="Логин">
                    <input id="password" type="password"
name="password" placeholder="Пароль">
                    <label for="remember_me">
                        <input type="checkbox" id="remember_me"
name="remember_me">
                            Запомнить меня
                        </label>
                    <input type="hidden" name="csrf_token" value="<?
echo $csrf_token?>">
                    <input type="submit" name="submit" value="Войти">
                </form>
            </div>
        </div>
        <div class="modal" id="modal_reg">
            <div class="modal_container">
                <span class="modal_close"
id="modal_close_reg">&times;</span>
                <h3>Регистрация</h3>
                <form action="includes/register.php" method="POST"
class="modal_reg_form">
                    <input type="text" id="new_login" name="new_login"
placeholder="Введите логин" required>
                    <input type="email" id="new_email"
name="new_email" placeholder="Введите Email" required>
                    <input type="password" id="new_password"
name="new_password" placeholder="Введите пароль" required>
                    <input type="password" id="new_password_repeat"
name="new_password_repeat" placeholder="Повторите пароль" required>
                    <input type="hidden" name="csrf_token" value="<?
echo $csrf_token?>">
                    <input type="submit" name="submit"
value="Зарегистрироваться">
                    <div class="checkbox_label">
                        <input type="checkbox" id="agreement"
name="agreement" required>
                            <label for="agreement">

```

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата						
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ					Лист
										77



```

        </form>
    </div>
</div>
</div>
<footer>
    <div class="container">
        <p class="rights">&copy; 2020 Платформа для голосования.
Разработано специально для выпускной квалификационной работы.</p>
    </div>
</footer>
<script src="assets/scripts/script.js"></script>
</body>
</html>

```

### connection.php:

```

<?php

// Подключение к базе данных
$connection = mysqli_connect('127.0.0.1', 'root', 'root', 'vot-
ing_platform_bd');

// Если подключение не произошло, то выйдет сообщение об ошибке
if($connection == false) {
    echo 'Ошибка подключения к базе данных!<br>';
    echo mysqli_connect_error();
    exit();
}
?>

```

### register.php:

```

<?php

session_start();

// Сравниваем токен, присвоенный форме, с токеном, привязанным к сес-
сии
if (!hash_equals($_SESSION['csrf_token'], $_POST['csrf_token'])) {
    echo "Произошла ошибка!";
} else {

    include("connection.php");

    // Функция для генерации случайной строки
    function generateCode($length) {
        $chars =
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
        $code = "";
        $chars_len = strlen($chars) - 1;
        while (strlen($code) < $length) {
            $code .= $chars[random_int(0, $chars_len)];
        }
        return $code;
    }
}

```

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	<pre>exit(); } ?&gt;  register.php:  &lt;?php  session_start();  // Сравниваем токен, присвоенный форме, с токеном, привязанным к сес- сии if (!hash_equals(\$_SESSION['csrf_token'], \$_POST['csrf_token'])) {     echo "Произошла ошибка!"; } else {      include("connection.php");      // Функция для генерации случайной строки     function generateCode(\$length) {         \$chars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";         \$code = "";         \$chars_len = strlen(\$chars) - 1;         while (strlen(\$code) &lt; \$length) {             \$code .= \$chars[random_int(0, \$chars_len)];         }         return \$code;     } }</pre>	
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ	Лист
						79



подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата

```

$token_slash = mysqli_real_escape_string($connection, $token);

// Переменная $headers для Email заголовка
$headers = "From: <mail@voting.ru>\r\n";

// Сообщение для Email
$message = "
    Чтобы подтвердить Email, перейдите сюда:
    http://voting.ru/includes/confirmed.php?token=".$token."
";

// Добавление данных о пользователе в БД
mysqli_query($connection, "INSERT INTO `users` SET `user_login` = '". $new_login_slash.'" , `user_email` =
'". $new_email_slash.'" , `token` = '". $token_slash.'" , `user_password`
= '". $new_password_hash_slash.'"");

// Проверка, отправилось ли письмо
if (mail($new_email, "Подтвердите Email на сайте", $message,
$headers)) {
    // Если да, то выводит сообщение
    echo "Проверьте свою почту для подтверждения аккаунта";
}
exit();
}
else {
    print_r("<b>При регистрации произошли следующие ошибки:</b><br>");

    foreach($errors as $err) {
        print_r($err . "<br>");
    }
}
?>

```

### *confirmed.php:*

```

<?php

include("connection.php");

// Проверка есть ли токен
if ($_GET['token']) {

    $token = $_GET['token'];

    // Запрашиваем данные о пользователе, с которым совпадает токен
    $query_user = mysqli_query($connection, "SELECT `user_id`, `token` FROM `users` WHERE `token` =
'".mysqli_real_escape_string($connection, $token)."'");
    $query_user_data = mysqli_fetch_assoc($query_user);

    // Подтверждаем в БД то, что пользователь активировал аккаунт
    mysqli_query($connection, "UPDATE `users` SET `email_verified` =
'1', `token` = 'NULL' WHERE `user_id` =

```

Подпись и дата		<pre>        foreach(\$errors as \$err) {             print_r(\$err . "&lt;br&gt;");         }     }     ?&gt;</pre>													
Инв. № дубл.		<p><i>confirmed.php:</i></p> <pre>&lt;?php  include("connection.php");  // Проверка есть ли токен if (\$_GET['token']) {      \$token = \$_GET['token'];      // Запрашиваем данные о пользователе, с которым совпадает токен     \$query_user = mysqli_query(\$connection, "SELECT `user_id`, `to- ken` FROM `users` WHERE `token` = '".\$_mysqli_real_escape_string(\$connection, \$token)."'");     \$query_user_data = mysqli_fetch_assoc(\$query_user);      // Подтверждаем в БД то, что пользователь активировал аккаунт     mysqli_query(\$connection, "UPDATE `users` SET `email_verified` = '1', `token` = 'NULL' WHERE `user_id` =</pre>													
Взам. инв. №															
Подпись и дата															
Инв. № подл.															
							ФАЭС.10.05.02.55.ПЗ				Лист				
											81				
Изм.	Лист	№ докум		Подпись	Дата										

```
".mysqli_real_escape_string($connection, $query_user_data['user_id'])."");
```

```
    if (mysqli_num_rows($query_user) != 0){
        echo "Email подтверждён!";
        exit();
    } else {
        echo "Произошла ошибка!";
    }

} else {
    echo "Что-то пошло не так";
}
?>
```

*login.php:*

```
<?php
```

```
session_start();
```

```
// Сравниваем токен, присвоенный форме, с токеном, привязанным к сессии
if (!hash_equals($_SESSION['csrf_token'], $_POST['csrf_token'])) {
    echo "Произошла ошибка!";
} else {
```

```
    include("connection.php");
```

```
    // Функция для генерации случайной строки
```

```
    function generateCode($length) {
```

```
        $chars =
```

```
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
```

```
        $code = "";
```

```
        $chars_len = strlen($chars) - 1;
```

```
        while (strlen($code) < $length) {
```

```
            $code .= $chars[random_int(0, $chars_len)];
```

```
        }
```

```
        return $code;
```

```
    }
```

```
$login = $_POST['login'];
```

```
$login_slash = mysqli_real_escape_string($connection, $login);
```

```
// Запрашиваем в БД запись, у которой логин совпадает с введенным
```

```
$query_login = mysqli_query($connection, "SELECT `user_id`, `user_email`, `user_password`, `email_verified`, `err_login_number`, `verified_login_code` FROM `users` WHERE `user_login` = '". $login_slash. "' LIMIT 1");
```

```
$query_login_data = mysqli_fetch_assoc($query_login);
```

```
// Смотрим наличие пользователя в БД
```

```
if (mysqli_num_rows($query_login) == 0) {
```

```
    print_r("Вы ввели неправильный логин/пароль");
```

```
}
```

```
else {
```

Подпись и дата

Инв. № докл.

Взам. инв. №

Подпись и дата

Инв. № подл.

Лист

ФАЭС.10.05.02.55.ПЗ

82

Изм. Лист № докум Подпись Дата

```

// Смотрим, подтвержден ли email
if ($query_login_data['email_verified'] == 0) {
    echo "Ваш Email не подтверждён! Пожалуйста, подтвердите
его.";
} else {
    // Проверка на то, что пользователь подтвердил, что это
он пытался зайти на свой аккаунт
    if (!empty($query_login_data['verified_login_code'])) {
        echo "Была попытка взлома аккаунта! Проверьте свою
почту.";
    } else {
        // Сравниваем пароли
        if (password_verify ($_POST['password'], $query_login_data['user_password'])) {

            // Генерируем случайную строку и хешируем её
            $hash = md5(generateCode(10));

            $ip_to_string = ", `user_ip` = 0";

            // Если пользователь выбрал "Запомнить меня"
            if(!empty($_POST['remember_me'])) {

                // Переводим IP в строку
                $ip_to_string = ", `user_ip` =
INET_ATON('".$_SERVER['REMOTE_ADDR']."'");
            }

            $hash_slash =
mysqli_real_escape_string($connection, $hash);
            $ip_to_string_slash =
mysqli_real_escape_string($connection, $ip_to_string);
            $user_id_slash =
mysqli_real_escape_string($connection, $query_login_data['user_id']);

            // Записываем в БД новый хеш авторизации и IP
            mysqli_query($connection, "UPDATE `users` SET
`user_hash` = '".$_$hash_slash.'" ".$ip_to_string_slash." WHERE `user_id` = '".$_$user_id_slash.'"");

            // Устанавливаем cookie
            setcookie("id", $query_login_data['user_id'],
time()+60*60*24*30, "/");
            setcookie("login", $login, time()+60*60*24*30,
"/");
            setcookie("hash", $hash, time()+60*60*24*30, "/",
null, null, true);

            // Переадресовываем браузер на страницу проверки
header("Location: login_check.php");
exit();
        }
    } else {
        if ($query_login_data['err_login_number'] >= 3) {

            $verified_code = generateCode(10);
            $verified_code_slash =
mysqli_real_escape_string($connection, $verified_code);

```

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата	Подпись и дата	
Изм.	Лист	№ докум	Подпись	Дата	<div style="text-align: center; font-size: 1.2em; font-weight: bold;">ФАЭС.10.05.02.55.ПЗ</div>	
						Лист
						83



```

        setcookie("login", "", time() - 3600*24*30*12, "/");
        setcookie("hash", "", time() - 3600*24*30*12, "/", null,
null, true);

```

```

        print_r("Хм, что-то не получилось");
    }
    else {
        header("Location: /");
    }
}
else {
    print_r("Включите cookie");
}
?>

```

### *verified\_user.php:*

```

<?php

include("connection.php");

// Проверка есть ли код подтверждения
if ($_GET['verified_code']) {

    $verified_code = $_GET['verified_code'];

    // Запрашиваем данные о пользователе, с которым совпадает код
    $query_user = mysqli_query($connection, "SELECT `user_id` FROM
`users` WHERE `verified_login_code` =
'".mysqli_real_escape_string($connection, $verified_code)."'");
    $query_user_data = mysqli_fetch_assoc($query_user);

    // Подтверждаем в БД то, что пользователь перешел по ссылке, по-
сле этого происходит сброс счётчика неправильных входов на сайт
    mysqli_query($connection, "UPDATE `users` SET `err_login_number`
= '0', `verified_login_code` = '' WHERE `user_id` =
'".mysqli_real_escape_string($connection, $que-
ry_user_data['user_id'])."'");

    echo "Теперь попробуйте снова зайти в аккаунт!";
    exit();

} else {
    echo "Что-то пошло не так";
}
?>

```

### *logout.php:*

```

<?php

// Удаляем cookie
setcookie("id", "", time() - 3600*24*30*12, "/");
setcookie("login", "", time() - 3600*24*30*12, "/");
setcookie("hash", "", time() - 3600*24*30*12, "/", null, null, true);

```

Подпись и дата		Инв. № дубл.		Взам. инв. №		Подпись и дата		Инв. № подл.		
<pre>// Запрашиваем данные о пользователе, с которым совпадает код \$query_user = mysqli_query(\$connection, "SELECT `user_id` FROM `users` WHERE `verified_login_code` = '".mysqli_real_escape_string(\$connection, \$verified_code)."'"); \$query_user_data = mysqli_fetch_assoc(\$query_user);  // Подтверждаем в БД то, что пользователь перешел по ссылке, по- сле этого происходит сброс счётчика неправильных входов на сайт mysqli_query(\$connection, "UPDATE `users` SET `err_login_number` = '0', `verified_login_code` = '' WHERE `user_id` = '".mysqli_real_escape_string(\$connection, \$que- ry_user_data['user_id'])."'");  echo "Теперь попробуйте снова зайти в аккаунт!"; exit();  } else {     echo "Что-то пошло не так"; } ?&gt;</pre> <p><i>logout.php:</i></p> <pre>&lt;?php  // Удаляем cookie setcookie("id", "", time() - 3600*24*30*12, "/"); setcookie("login", "", time() - 3600*24*30*12, "/"); setcookie("hash", "", time() - 3600*24*30*12, "/",null,null,true);</pre>										
					ФАЭС.10.05.02.55.ПЗ					Лист
										85
Изм.	Лист	№ докум	Подпись	Дата						

```
// Переадресовываем браузер главную страницу
header("Location: /");
exit();
```

?>

### *csrf\_protection.php:*

```
<?php
```

```
// Начинаем сессию
session_start();
```

```
// Генерируем случайный ключ для данной сессии, чтобы сгенерировать
токен
if (empty($_SESSION['key'])) {
    $_SESSION['key'] = bin2hex(random_bytes(32));}
// Создаем токен для защиты от CSRF и привязываем его к сессии
$csrf_token = hash_hmac('sha256', 'this is anyone string: index.php',
$_SESSION['key']);
$_SESSION['csrf_token'] = $csrf_token;
?>
```

### *new\_voting.php:*

```
<?php
```

```
session_start();
```

```
// Сравниваем токен, присвоенный форме, с токеном, привязанным к сес-
сии
```

```
if (!hash_equals($_SESSION['csrf_token'], $_POST['csrf_token'])) {
    echo "Произошла ошибка!";
} else {
    include("connection.php");
```

```
/*----- Добавление данных о созданном голосовании в таблицу
voting -----*/
```

```
$voting_theme_new = htmlspecialchars($_POST['voting_theme_new']);
$voting_theme_new_slash = mysqli_real_escape_string($connection,
$voting_theme_new);
```

```
$voting_content_new = htmlspecialchars(
chars($_POST['voting_content_new']);
$voting_content_new_slash =
mysqli_real_escape_string($connection, $voting_content_new);
```

```
$voting_left_time = htmlspecialchars($_POST['voting_left_time']);
$voting_left_time_slash = mysqli_real_escape_string($connection,
$voting_left_time);
```

```
$query_table_voting = "INSERT INTO `voting` SET `voting_theme` =
'".$voting_theme_new_slash."', `voting_content` =
'".$voting_content_new_slash."', `voting_date` =
'".$mysqli_real_escape_string($connection, date("Y-m-d H:i:s"))."',
```

Инв. № подл.	Подпись и дата				Лист
	Инв. № дубл.				
	Взам. инв. №				
	Подпись и дата				
<pre>session_start();  // Сравниваем токен, присвоенный форме, с токеном, привязанным к сес- сии if (!hash_equals(\$_SESSION['csrf_token'], \$_POST['csrf_token'])) {     echo "Произошла ошибка!"; } else {     include("connection.php");      /*----- Добавление данных о созданном голосовании в таблицу voting -----*/      \$voting_theme_new = htmlspecialchars(\$_POST['voting_theme_new']);     \$voting_theme_new_slash = mysqli_real_escape_string(\$connection, \$voting_theme_new);      \$voting_content_new = htmlspecialchars- chars(\$_POST['voting_content_new']);     \$voting_content_new_slash = mysqli_real_escape_string(\$connection, \$voting_content_new);      \$voting_left_time = htmlspecialchars(\$_POST['voting_left_time']);     \$voting_left_time_slash = mysqli_real_escape_string(\$connection, \$voting_left_time);      \$query_table_voting = "INSERT INTO `voting` SET `voting_theme` = '".\$voting_theme_new_slash."', `voting_content` = '".\$voting_content_new_slash."', `voting_date` = '".mysqli_real_escape_string(\$connection, date("Y-m-d H:i:s"))."',</pre>					
ФАЭС.10.05.02.55.ПЗ					86
Изм.	Лист	№ докум	Подпись	Дата	

```

`voting_left_days` = ' ".$voting_left_time_slash."', `user_id` =
' ".mysqli_real_escape_string($connection, $_COOKIE['id'])."'";

mysqli_query($connection, $query_table_voting);
$last_voting_id = mysqli_insert_id($connection);
$last_voting_id_slash = mysqli_real_escape_string($connection,
$last_voting_id);

/*----- Добавление вариантов созданного голосования в таблицу results
-----*/
foreach (array_keys($_POST) as $key) {
    if (strpos($key, 'option') !== false) {
        mysqli_query($connection, "INSERT INTO `results` SET `op-
tion_name` = ' ".mysqli_real_escape_string($connection, htmlspecialchars(
$_POST[$key]))."', `voting_id` = ' ".$last_voting_id_slash."'";
    }
}

header("Location: /");
exit();
}
?>

```

*print\_voting.php:*

```

<?php

include("connection.php");

// Получаем данные о всех созданных голосованиях из БД
$query_voting = mysqli_query($connection, "SELECT * FROM `voting` OR-
DER BY `voting_id` DESC");

$voting_number = mysqli_num_rows($query_voting) + 1;

foreach($query_voting as $voting):
    $voting_number--; ?>
    <div class="voting">
        <div class="number_and_img">
            <div class="voting_number">#<?echo $voting_number?></div>
            
        </div>
        <div class="voting_structure">
            <p class="voting_theme"><?echo $voting['voting_theme']?></p>
            <p class="voting_content"><?echo $vot-
ing['voting_content']?></p>

            <form class="voting_form" action="includes/send_vote.php"
method="POST">
                <div class="options">
                    <?
                        $voting_id_slash =
mysqli_real_escape_string($connection, $voting['voting_id']);

```

Инв. № подл.	Подпись и дата				Лист
Инв. № докл.	Подпись и дата				87
Взам. инв. №					ФАЭС.10.05.02.55.ПЗ
Инв. № докл.					Лист
Изм. Лист № докум Подпись Дата					

<pre>include("connection.php");  // Получаем данные о всех созданных голосованиях из БД \$query_voting = mysqli_query(\$connection, "SELECT * FROM `voting` OR- DER BY `voting_id` DESC");  \$voting_number = mysqli_num_rows(\$query_voting) + 1;  foreach(\$query_voting as \$voting):     \$voting_number--; ?&gt;     &lt;div class="voting"&gt;         &lt;div class="number_and_img"&gt;             &lt;div class="voting_number"&gt;#&lt;?echo \$voting_number?&gt;&lt;/div&gt;             &lt;img src="assets/images/your_vote.png" alt="Ваш голос важен!" width="200" height="200"&gt;         &lt;/div&gt;         &lt;div class="voting_structure"&gt;             &lt;p class="voting_theme"&gt;&lt;?echo \$voting['voting_theme']?&gt;&lt;/p&gt;             &lt;p class="voting_content"&gt;&lt;?echo \$vot- ing['voting_content']?&gt;&lt;/p&gt;              &lt;form class="voting_form" action="includes/send_vote.php" method="POST"&gt;                 &lt;div class="options"&gt;                     &lt;?                         \$voting_id_slash = mysqli_real_escape_string(\$connection, \$voting['voting_id']);</pre>				
--	--	--	--	--



```

        // Получаем из БД варианты ответа для данного голосо-
вания
        $query_options = mysqli_query($connection, "SELECT *
FROM `results` WHERE `voting_id` = '". $voting_id_slash. "'");

        // Перевод дней, выделенных на голосование, и даты
начала голосования в секунды
        $duration = $voting['voting_left_days'] * 24*60*60;
        $voting_date_sec = strtotime($voting['voting_date']);
        $voting_end_datetime_sec = $voting_date_sec + $dura-
tion;

        // Запрос в БД в таблицу voted_users
        $query_voted_users = mysqli_query($connection, "SE-
LECT `id` FROM `voted_users` WHERE `user_id` =
'".mysqli_real_escape_string($connection, $_COOKIE['id']).'" AND
`voting_id` = '". $voting_id_slash. "'");

        // Подсчёт общей суммы голосов на определенном голо-
совании
        $sum_votes = 0;
        foreach($query_options as $option) {
            $sum_votes += $option['option_votes_number'];
        }

        // Вывод результатов голосования, если вышло время
для голосования
        if(strtotime(date("d.m.Y H:i:s")) >= $vot-
ing_end_datetime_sec): ?>
            <table class="option_result">
            <?
            $option_number = 0;
            foreach($query_options as $option):
            $option_number++;
            ?>
                <tr>
                    <td class="option_name_result">
                        <?echo $option['option_name']?>&nbsp;
                    </td>
                    <td class="progress">
                        <progress max="100" value="<?echo
round(($option['option_votes_number']/$sum_votes)*100)?>"></progress>
                    </td>
                    &nbsp;&nbsp;&nbsp;<? echo
"('.$option['option_votes_number'].')"; ?>
                    </td>
                </tr>
            <? endforeach ?>
            <tr>
                <td class="number_users_voting" col-
span="3">
                    Пользователей проголосовало: <?echo
$sum_votes?> из <?echo mysqli_num_rows(mysqli_query($connection, "SE-
LECT `user_id` FROM `users`"))?>
                </td>
            </tr>
            </table>

```

Ине. № подл.	Подпись и дата
Взам. инв. №	Инв. № дубл.
Подпись и дата	



```
// Сравниваем токен, присвоенный форме, с токеном, привязанным к сес-
сии
if (!hash_equals($_SESSION['csrf_token'], $_POST['csrf_token'])) {
    echo "Произошла ошибка!";
} else {
    include("connection.php");

    // Определение ID голосования по атрибуту name
    $voting_id = intval(explode("_", key($_POST))[1]);

    $voting_id_slash = mysqli_real_escape_string($connection, $vot-
ing_id);

    // Определение порядкового номера варианта голосования по атрибуту
value
    $option_count = intval(explode("_", array_values($_POST)[0])[1]);

    $query_options = mysqli_query($connection, "SELECT * FROM `re-
sults` WHERE `voting_id` = '". $voting_id_slash. "'");

    // Добавление голоса к варианту, за который проголосовали
    $i = 0;
    foreach($query_options as $option) {
        if ($i === $option_count-1) {
            mysqli_query($connection, "UPDATE `results` SET `op-
tion_votes_number` = `option_votes_number` + 1 WHERE `result_id` =
'".mysqli_real_escape_string($connection, $option['result_id']).'");
        }
        $i++;
    }
    // Добавление id пользователя в таблицу проголосовавших, чтобы он
не смог проголосовать более 1 раза
    mysqli_query($connection, "INSERT INTO `voted_users` SET `us-
er_id` = '".mysqli_real_escape_string($connection,
$_COOKIE['id'])."', `voting_id` = '". $voting_id_slash. "'");
    header("Location: /");
}
?>
```

Инв. № подл.	Подпись и дата	Взам. инв. №	Инв. № дубл.	Подпись и дата						
Изм.	Лист	№ докум	Подпись	Дата	ФАЭС.10.05.02.55.ПЗ					Лист
										90