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

По д. и дата

Пояснительная записка

Студент / И.В. Маслов / Факультет АЭС Группа АБ-55 / Руководитель / А.А. Буров / Рецензент / Е.В. Ткаченко / Консультанты:

Взам. инв . №

Инв . № ду бл.

* по экономическому обоснованию

/ И.С. Мухина /

* по безопасности жизнедеятельности

По дп. и дата

/ Н.Н. Симакова /

Новосибирск 2021

Инв . № по дп.

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, техниче- ское 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 выпол- нения по раз-  делам |
| Введение | 23.09.2020 г. |
| 1. Анализ предметной области разрабатываемого веб-  приложения | 29.09.2020 г. |
| 2. Определение технологической составляющей для разработ-  ки | 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 /

1. Раздел по безопасности жизнедеятельности

/ Н.Н. Симакова /

|  |  |
| --- | --- |
| Дата выдачи задания  « 01 » сентября 2020 г. | Задание принял к исполнению  « 01 » сентября 2020 г. |
| / А.А. Буров /  (подпись, Ф.И.О. руководителя) | / 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 голосова- ния»

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 защищенно- го электронного голосования»

|  |  |  |
| --- | --- | --- |
| Работа имеет практическую ценность |  | Тема предложена предприятием |
| Работа внедрена |  | Тема предложена студентом |
| Рекомендую работу к внедрению |  | Тема является фундаментальной |
| Рекомендую работу к опубликованию |  | Рекомендую студента в магистратуру |
| Работа выполнена с применением ЭВМ |  | Рекомендую студента в аспирантуру |

|  |
| --- |
|  |
|  |
|  |
|  |
|  |

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.

Приложение к Отзыву

## 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 теле- коммуникационные системы и их элементы, проводить анализ проектных решений по обеспечению заданного уровня безопасности и требуемого качества обслужива- ния, разрабатывать необходимую техническую докумен- тацию с учетом действующих нормативных и методиче-  ских документов |  |  |  |
| ПК-7 - способностью осуществлять рациональный выбор средств обеспечения информационной безопасности теле- коммуникационных систем с учетом предъявляемых к ним требований качества обслуживания и качества функ-  ционирования |  |  |  |
| ПК-12 - способностью выполнять технико-экономические обоснования, оценивать затраты и результаты деятельно- сти 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 электрон- ного голосования

Решаемые задачи: анализ предметной области разрабатываемого веб-приложения, определение технологической составляющей для разработки, разработка веб- приложения и оценка его защищенности, безопасность жизнедеятельности, тех- нико-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 vot- ing

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 securi- ty assessment, life safety, technical and economic justification of work

Results working and protected web application for electronic voting

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | ОГЛАВЛЕНИЕ [Введение 4](#_bookmark0)   1. [Анализ предметной области разрабатываемого веб-приложения 5](#_bookmark1)    1. [Постановка задачи 5](#_bookmark2)    2. [Определение объекта разработки 5](#_bookmark3)    3. [Анализ основных угроз и уязвимостей веб-приложения 7](#_bookmark4)    4. [Разработка модели нарушителя информационной безопасности 12](#_bookmark5)    5. [Выводы по разделу 16](#_bookmark6) 2. [Определение технологической составляющей для разработки 17](#_bookmark7)    1. [Постановка задачи 17](#_bookmark8)    2. [Планирование архитектуры веб-приложения 17](#_bookmark9)    3. [Выбор языковых и программных средств разработки 24](#_bookmark10)    4. [Выбор практических методов защиты веб-приложения 27](#_bookmark11)   [2.6 Выводы по разделу 32](#_bookmark12)   1. [Разработка веб-приложения и оценка его защищенности 33](#_bookmark13)    1. [Постановка задачи 33](#_bookmark14)    2. [Создание веб-интерфейса 33](#_bookmark15)    3. [Разработка серверной логики веб-приложения 38](#_bookmark16)    4. [Настройка протокола SSL 44](#_bookmark17)    5. [Оценка защищенности разработанного веб-приложения 48](#_bookmark18)    6. [Выводы по разделу 53](#_bookmark19) 2. [Безопасность жизнедеятельности 54](#_bookmark20)    1. [Постановка задачи 54](#_bookmark21)    2. [Характеристика условий труда при работе с ПК 54](#_bookmark22)    3. [Эргономические требования к рабочему месту пользователя 57](#_bookmark23) | | | | | | | | | | |
| [Подп. и дата](#_bookmark7) |  |
| [Инв. № дубл.](#_bookmark7) |  |
| [Взам. инв. №](#_bookmark7) |  |
| [Подп. и дата](#_bookmark7) |  |
|  |  |  |  |  | *ФАЭС.10.05.02.055 ПЗ* | | | | | |
|  |  |  |  |  |
| Из | Лист | № докум. | Подп. | Дата |
| Инв. № подп |  | Разраб. | | *И.В. Маслов* |  |  | Разработка веб-приложения для за-  щищенного электронного голосова- ния  Содержание | Лит | | | Лист | Листов |
| Пров. | | *А.А. Буров* |  |  |  |  |  | 2 | 90 |
| Н/контр | |  |  |  |  | | | | |
| Рецензент | | *Е.В. Ткаченко* |  |  |
| Утвердил | | *С.Н. Новиков* |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | * 1. [Требования охраны труда офисных работников 59](#_bookmark24)   2. [Пожарная безопасность 62](#_bookmark25)   3. [Выводы по разделу 64](#_bookmark26)  1. [Технико-экономическое обоснование работы 65](#_bookmark27)    1. [Постановка задачи 65](#_bookmark28)    2. [Расчет трудоемкости и длительности работ 65](#_bookmark29)    3. [Расчет себестоимости и цены программного продукта 69](#_bookmark30)    4. [Выводы по разделу 72](#_bookmark31)   [Заключение 73](#_bookmark32)  [Список литературы 74](#_bookmark33)  [Приложение А Код разработанного веб-приложения 76](#_bookmark55) | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 3 |
| *Изм.* | *Лист* | *№ докум.* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Введение  Интернет-голосование, как разновидность электронного голосования, при- меняется в различных официальных и неофициальных опросах. Оно является до- ступным и удобным, что делает его популярным и часто используемым, напри- мер, 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](#_bookmark52)]. Поэто- му необходимо реализовать систему защиты веб-приложения, учитывая уязвимо- сти, 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. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 4 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1. Анализ предметной области разрабатываемого веб-приложения    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. Определение объекта разработки   Под электронным голосованием понимают такое голосование, в котором используются электронные средства, помогающие или обеспечивающие подачу и подсчет голосов. [[11](#_bookmark43)]  Электронное голосование часто рассматривается как инструмент повыше- ния эффективности избирательного процесса и повышения доверия к нему. Пра- вильно реализованные решения для электронного голосования могут повысить безопасность бюллетеня, ускорить обработку результатов и упростить само голо- сование. [[5](#_bookmark38)]  На рисунке 1.1 изображены основные четыре технологии электронного го- лосования.  P324#yIS1  Рисунок 1.1 – Основные технологии электронного голосования | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 5 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | В системах бумажного-электронного голосования голоса подаются с ис- пользованием бумажных бюллетеней, а подсчитываются специальными машина- ми.  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](#_bookmark43)]  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 мне- ние 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 маркетин- говых и социологических опросов.  Веб-приложение для электронного голосования должно предусматривать:   1. регистрацию пользователей с подтверждением аккаунта через почту; 2. авторизацию пользователей; 3. возможность авторизированным пользователям создавать новое голосо- вание, задавать его срок и варианты ответа;   5) вывод результатов по истечении срока, отведенного для голосования;  4) исключение возможности повторного голосования одним и тем же поль- зователем.  Электронное голосование в силу своей сущности предполагает использова- ние онлайн-платформы и цифровых устройств для выражения мнения. Этот вари- ант волеизъявления обладает целым набором преимуществ. Среди них – возмож- | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 6 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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, при- сущими веб-приложениям. [[22](#_bookmark53)]  Потенциальные угрозы и уязвимости представлены в следующем подразде-  ле.  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 суще- ствующую опасность нарушения безопасности информации. Уязвимость же явля- ется свойством информационной системы, обусловливающее возможность реали- зации угроз безопасности обрабатываемой в ней информации. [[8](#_bookmark41)]  Все источники угроз безопасности информации можно разделить на три ос- новные группы:   1. обусловленные действиями субъекта (антропогенные источники угроз); 2. обусловленные техническими средствами (техногенные источники угро-   зы);   1. обусловленные стихийными источниками. [[7](#_bookmark40)]   Наибольший интерес с точки зрения организации защиты представляют ан- тропогенные источники угроз безопасности информации, так как в роли таких ис- точников выступают субъекты, действия которых всегда можно оценить, спро- гнозировать 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 ко- торых могут привести к нарушению безопасности информации, могут быть:  – внешние; | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 7 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | * внутренние. [[7](#_bookmark40)]   Внешние источники могут быть случайными или преднамеренными и иметь разный уровень квалификации.  Внутренние субъекты, как правило, представляют собой высококвалифици- рованных 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 техниче- ских характеристик оборудования, применяемого на защищаемом объекте. Пол- ное устранение этих уязвимостей невозможно, но они могут существенно ослаб- ляться 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 орга- низационных и инженерно-технических мероприятий по противодействию, угро- зам информационной безопасности. [[7](#_bookmark40)]  Следует рассмотреть угрозы и уязвимости, отталкиваясь от особенностей разрабатываемого веб-приложения. В таблице 1.1 приведены особенности буду- щей системы и прогнозируемые угрозы и уязвимости. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 8 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Таблица 1.1 – Прогнозируемые угрозы и уязвимости веб-приложения  Благодаря исследованиям Positive Technologies – международной компании, специализирующейся на разработке программного обеспечения в области инфор- мационной безопасности, выявлены самые распространенные угрозы и уязвимо- сти из списка OWASP (Open Web Application Security Project – открытый проект обеспечения безопасности веб-приложений) – некоммерческого фонда, который работает над повышением безопасности программного обеспечения [[2](#_bookmark35)]. Они представлены на рисунке 1.2 [[21](#_bookmark52)]. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 9 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |
| --- | --- |
| 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) |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P366#yIS1  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 аутенти- фикации» – это некорректное ограничение количества неудачных попыток аутен- тификации. 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 меж- сайтового запроса (CSRF)». В ходе CSRF-атаки злоумышленник с помощью спе- циально сформированных сценариев может выполнять действия от лица пользо- вателя, авторизованного в уязвимом веб-приложении. [[21](#_bookmark52)] | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 10 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Для 9 из 10 веб-приложений актуальна угроза атак на клиентов. Как и преж- де, существенную роль при этом играет «Межсайтовое выполнение сценариев (XSS)». В результате эксплуатации уязвимостей злоумышленник может заражать компьютеры пользователей вредоносным ПО, проводить фишинговые атаки, например для получения учетных данных, а также выполнять действия от имени пользователя.  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 выпол- нения произвольных команд на атакуемом сервере. [[21](#_bookmark52)]  В таблицу 1.2 [[2](#_bookmark35)] сведем рекомендации и способы предотвращения рас- смотренных угроз и уязвимостей.  Таблица 1.2 – Способы предотвращения угроз и уязвимостей веб-приложений | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 11 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

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

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | При разработке веб-приложения необходимо реализовать все защитные ме- ры, рассмотренные в данном подразделе.  1.4 Разработка модели нарушителя информационной безопасности  Согласно ГОСТ Р 53114-2008, нарушителем информационной безопасности является физическое лицо или логический объект, случайно или преднамеренно совершивший действие, следствием которого является нарушение информацион- ной безопасности. [[9](#_bookmark42)]  В свою очередь, модель нарушителя информационной безопасности – это набор предположений об одном или нескольких возможных нарушителях инфор- мационной безопасности, их квалификации, их технических и материальных средствах и т. д. [[7](#_bookmark40)]  Целью разработки модели нарушителя является формирование предполо- жения о типах, видах нарушителей, которые могут реализовать угрозы безопасно- сти информации в информационной системе с заданными структурно- функциональными характеристиками и особенностями функционирования, а так- же потенциале этих нарушителей и возможных способах реализации угроз без- опасности информации. [[14](#_bookmark46)]  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 инфор- мационной системе, ее отдельным компонентам и реализующие угрозы безопас- ности 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 безопасно- сти информации, нарушители подразделяются на: | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 12 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | * 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 реализа- ции угроз безопасности информации в информационной системе. [[14](#_bookmark46)]   Обратимся к банку данных угроз безопасности информации (УБИ), разра- ботанный Федеральной службой по техническому и экспортному контролю Рос- сии. В банке описаны угрозы и соответствующий им тип нарушителя и его мини- мально необходимый потенциал. Угрозы, которые можно отнести к разрабатыва- емому веб-приложению:   * угроза внедрения кода или данных (УБИ. 006); * угроза восстановления и/или повторного использования аутентификаци- онной информации (УБИ. 008); * угроза доступа/перехвата/изменения HTTP cookies (УБИ. 017); * угроза использования информации идентификации/аутентификации, за- данной по умолчанию (УБИ. 030); * угроза межсайтового скриптинга (УБИ. 041); * угроза несанкционированного доступа к аутентификационной информа- ции (УБИ. 074); * угроза несанкционированного изменения аутентификационной информа- ции (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](#_bookmark39)]   Приведенным угрозам соответствуют следующие типы нарушителей и их минимально необходимый потенциал: | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 13 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1. внешний нарушитель с низким потенциалом; 2. внутренний нарушитель с низким потенциалом; 3. внешний нарушитель со средним потенциалом. [[6](#_bookmark39)]   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](#_bookmark46)]   В таблице 1.3 [[14](#_bookmark46)] приведена возможная мотивация рассмотренных выше нарушителей. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 14 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

Таблица 1.3 – Виды нарушителей и их мотивация

|  |  |
| --- | --- |
| Виды нарушителей | Возможные цели реализации угроз |
| Преступные группы | * причинение имущественного ущерба путем мо- шенничества или иным преступным путем; * выявление уязвимостей с целью их дальнейшей   продажи и получения финансовой выгоды. |
| Внешние субъекты (физи- ческие лица) | * идеологические или политические мотивы; * причинение имущественного ущерба путем мо- шенничества или иным преступным путем; * любопытство или желание самореализации; * выявление уязвимостей с целью их дальнейшей продажи и получения финансовой выгоды. |
| Конкуренты | * получение конкурентных преимуществ; * причинение имущественного ущерба путем об- мана или злоупотребления доверием. |
| Разработчики, производи- тели, поставщики про- граммных, технических и программно-технических средств | * внедрение дополнительных функциональных воз- можностей в программное обеспечение или про- граммно-технические средства на этапе разработки; * причинение имущественного ущерба путем об- мана или злоупотребления доверием; * непреднамеренные, неосторожные или неквали-   фицированные действия. |
| Лица, привлекаемые для установки, наладки, мон- тажа, пусконаладочных и  иных видов работ | * причинение имущественного ущерба путем об- мана или злоупотребления доверием; * непреднамеренные, неосторожные или неквали-   фицированные действия. |

*Взам. инв. № Инв. № дубл.*

*Подпись и дата*

*Изм.*

*Инв. № подл.*

*Лист*

*№ докум*

*Подпись Дата*

# *ФАЭС.10.05.02.55.ПЗ*

*Лист*

15

*Подпись и дата*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Продолжение таблицы 1.3  При разработке защищенного веб-приложения для электронного голосова- ния 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 угро- зы и уязвимости разрабатываемой системы и рассмотрены способы их предот- вращения. Также была разработана модель потенциального нарушителя инфор- мационной безопасности веб-приложения для электронного голосования. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 16 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |
| --- | --- |
| Лица, обеспечивающие функционирование инфор- мационных систем или об- служивающие инфра-  структур | * причинение имущественного ущерба путем об- мана или злоупотребления доверием; * непреднамеренные, неосторожные или неквали- фицированные действия. |
| Пользователи | * причинение имущественного ущерба путем мо- шенничества или иным преступным путем; * любопытство или желание самореализации; * непреднамеренные, неосторожные или неквали- фицированные действия. |
| Бывшие работники (поль- зователи) | * причинение имущественного ущерба путем мо- шенничества или иным преступным путем; * месть за ранее совершенные действия. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1. Определение технологической составляющей для разработки    1. Постановка задачи   В рамках данной главы необходимо спланировать архитектуру разрабаты- ваемого веб-приложения и отобразить принцип взаимодействия пользователя с системой. Учитывая эти сведения, нужно определить какая техническая база бу- дет использоваться при разработке веб-приложения.   * 1. Планирование архитектуры веб-приложения   Согласно ГОСТ Р ИСО 9241-151-2014, веб-приложением является прило- жение, 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 передачей на сервер обычно занимается специальное приложение – браузер. Как известно, одной из функций браузера является отображение данных, полу- ченных из Интернета, в виде страницы, описанной на языке HTML, следователь- но, результат, передаваемый сервером клиенту, должен быть представлен на этом языке.  На стороне сервера веб-приложение выполняется специальным программ- ным обеспечением (веб-сервером), который и принимает запросы клиентов, обра- батывает их, формирует ответ в виде страницы, описанной на языке HTML, и пе- редает его клиенту.  В процессе обработки запроса пользователя веб-приложение компонует от- вет 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- | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 17 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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 помо- щью исполняемого кода – так называемой исполняемой части.  Принцип работы веб-приложения представлен на рисунке 2.1.  P436#yIS1  Рисунок 2.1 – Принцип работы веб-приложения | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 18 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Общий принцип работы веб-приложения можно описать несколькими ша-  гами:   1. пользователь посредством веб-браузера переходит по адресу веб- приложения. Отправляется HTTP-запрос на веб-сервер HTTP; 2. веб-сервер обращается к файловому хранилищу для запуска необходимо- го PHP-скрипта; 3. PHP-скрипт обрабатывается PHP-интерпретатором, который в свою оче- редь обменивается данными с базой данных при необходимости; 4. затем генерируется необходимая HTML-страница и подгружаются необ- ходимые статические файлы; 5. полученная страница передается в HTTP-ответе к клиенту HTTP, то есть веб-браузеру пользователя.   Предполагаемая архитектура базы данных разрабатываемого веб- приложения будет состоять из четырех таблиц:   * таблица пользователей, содержащая данные об аккаунтах; * таблица с данными о созданных голосованиях; * таблица учета голосов; * таблица учета проголосовавших пользователей.   Опираясь на техническое задание, рассмотрим принцип взаимодействия пользователя с веб-приложением. Для этого рассмотрим веб-приложение как со- вокупность из четырех частей: авторизация, регистрация, создание голосования пользователем, отправка голоса пользователя.  На рисунке 2.2 изображена диаграмма взаимодействия пользователя с веб- приложением при регистрации. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 19 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P451#yIS1  Рисунок 2.2 – Взаимодействие пользователя с веб-приложением при регистрации  Сначала пользователь открывает форму регистрации нажатием кнопки «Ре- гистрация». Затем вводит логин, почту и пароль. Если такого пользователя в базе данных не существует, то тогда создается новый пользователь и отправляется письмо ему на почту для подтверждения аккаунта. В случае наличия такого логи- на или почты в базе данных, выводится сообщение об ошибке.  На рисунке 2.3 изображен принцип взаимодействия пользователя с веб- приложением при авторизации. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 20 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P455#yIS1  Рисунок 2.3 – Взаимодействие пользователя с веб-приложением при авторизации Сначала пользователь открывает форму авторизации нажатием на кнопку  «Войти». Затем вводит логин и пароль. Данные отправляются на сервер, который проверяет наличие такого пользователя в базе данных. Если пользователя нет, то выводится сообщение об ошибке. Если пользователь есть, то сервер проверяет введенный пароль. Если пароль неверный, то выводится сообщение об ошибке, в противном случае, выводится приветствие пользователя и появляются кнопки  «Создать голосование» и «Выйти».  Рассмотрим принцип создания нового голосования пользователем (рисунок  2.4). | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 21 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P459#yIS1  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 про- тивном случае выводится сообщение об ошибке.  Рассмотрим принцип голосования пользователем (рисунок 2.5). | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 22 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P464#yIS1  Рисунок 2.5 – Принцип голосования пользователем  Пользователь выбирает вариант ответа и нажимает кнопку «Проголосо- вать». Серверу передаются данные о пользователе, голосовании и выбранном ва- рианте ответа. Эти данные заносятся в базу данных, после чего пользователю за- прещается повторно голосовать в этом голосовании. Если не удается записать данные в базу данных, то выходит сообщение об ошибке.  На рисунке 2.6 изображен принцип завершения сессии пользователем нажа- тием на кнопку «Выйти». | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 23 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P469#yIS1  Рисунок 2.6 – Принцип завершения сессии пользователем  Пользователь нажимает на кнопку «Выйти» и происходит удаление данных cookie. Таким образом, пользователь перестает быть авторизированным и у него снова появляются кнопки «Войти» и «Зарегистрироваться».  2.3 Выбор языковых и программных средств разработки  Любое веб-приложение содержит в себе три основные составляющие:   1. front-end – пользовательский интерфейс и функции, которые работают на клиентской стороне веб-сайта или приложения; 2. back-end – набор аппаратно-программных средств, при помощи которых реализована логика работы сайта. 3. база данных – организованная структура, предназначенная для хранения, изменения и обработки взаимосвязанной информации, преимущественно больших объемов.   На рисунке 2.7 представлена клиент-серверная архитектура. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 24 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P480#yIS1  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 опи- сания и стилизации внешнего вида документа; 3. JavaScript – язык программирования, который способствует «оживле- нию» веб-страниц при взаимодействии пользователя с ними.   Для разработки логики серверной стороны веб-приложения будет использо- ваться распространенный язык программирования общего назначения PHP (Hy- pertext Preprocessor – препроцессор гипертекста). Его код может внедряться непо- средственно в HTML-разметку. [[18](#_bookmark50)]  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 одинако- во хорошо с разными платформами; 4. набор текста кода и его редактирование можно осуществлять в любом   текстовом или html-редакторе. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 25 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1. высокая гибкость, емкость и функциональность; 2. многозадачность и широкие возможности. [[23](#_bookmark54)]   Помимо front-end’а и back-end’а неотъемлемой частью любого веб- приложения являются базы данных. Они бывают реляционными и нереляцион- ными. Реляционные базы данных хранят структурированные данные, которые обычно представляют объекты реального мира. Скажем, это могут быть сведения о человеке, или о содержимом корзины для товаров в магазине, сгруппированные в таблицах, формат которых задан на этапе проектирования хранилища. Нереля- ционные базы данных устроены иначе. Например, документо-ориентированные базы хранят информацию в виде иерархических структур данных. Речь может ид- ти об объектах с произвольным набором атрибутов. То, что в реляционной базе данных будет разбито на несколько взаимосвязанных таблиц, в нереляционной может храниться в виде целостной сущности. Для надежной сохранности данных лучше использовать реляционную базу данных, поэтому в работе будет приме- няться именно этот вариант.  Чтобы взаимодействовать с реляционной базой данных, используется SQL (Structured Query Language – язык структурированных запросов) – декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных (СУБД).  При разработке веб-приложения будет использовать реляционную СУБД MySQL. Она является быстрой, надежной и универсальной, а также самой попу- лярной в мире: по итогам прошлого года ее использовали 39% разработчиков [20]. Она отлично подходит для веб-разработки.  Таким образом, для разработки данного веб-приложения можно использо- вать технологический стек WAMP – акроним от «Windows, Apache, MySQL и PHP».  Windows является распространенной операционной системой, ориентиро- ванной на управление с помощью графического интерфейса. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 26 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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 возможно- стями по администрированию и настройке компонентов. Платформа широко ис- пользуется с целью разработки, отладки и тестирования веб-проектов, а также для предоставления веб-сервисов в локальных сетях. [[1](#_bookmark34)]  Подведем итоги:   * для разработки веб-интерфейса будут использоваться HTML, CSS и JavaS-   cript;   * 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.   1. Выбор практических методов защиты веб-приложения      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 несанкциони- рованного доступа (НСД) при передаче и/или хранении. Испытанный метод за- | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 27 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | щиты информации от НСД – шифрование. Шифрованием называют процесс пре- образования исходных данных в шифрованные, нечитаемые без знания специаль- ных параметров преобразования – ключа. Дешифрованием называют обратное преобразование шифрованных данных в открытые.  Все известные алгоритмы шифрования делятся на два типа:   1. cимметричные – с единственным секретным ключом для шифрования и дешифрования (они же single-key). Симметричные алгоритмы также подразделя- ются на два семейства: потоковые – шифрование данных посимвольно, блочные – шифрование данных кусками (блоками) из нескольких символов конечной и фик- сированной 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 необхо- димость организации закрытого канала для передачи ключа. А основным досто- инством – быстрота выполнения криптопреобразования. И наоборот, асиммет- ричные 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](#_bookmark45)]  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 ги- пертекста). 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](#_bookmark36)]  Шифрование данных, которые передаются от клиента к серверу, происхо-  дит, в свою очередь, в соответствии со своим, криптографическим протоколом. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 28 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Для того чтобы использовать шифрование, у сайта должен быть специаль- ный сертификат, или, как он еще называется, цифровая подпись, который под- тверждает, что механизм шифрования действительно надежен и соответствует протоколу. Индикаторами того, что у сайта такой сертификат есть, являются, по- мимо буквы «S» в HTTPS, зеленый замочек и надпись «Защищено» или название компании в адресной строке браузера. [[3](#_bookmark36)]  На рисунке 2.8 представлен принцип внедрения протокола TLS.  P523#yIS1  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](#_bookmark37)]  Рассмотрим второй способ защиты данных посредством хеширования паро- лей. Хеширование паролей является одним из самых основных соображений без- опасности, которые необходимо сделать, при разработке приложения, принима- ющего пароли от пользователей. Без хеширования, пароли, хранящиеся в базе приложения, могут быть украдены, например, если база данных была скомпроме- тирована, а затем немедленно могут быть применены для компрометации не толь- ко приложения, но и аккаунтов пользователей на других сервисах, если они не используют уникальных паролей.  Применяя хеширующий алгоритм к пользовательским паролям перед со- хранением их в базе данных, становится невозможным разгадывание оригиналь- | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 29 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | ного пароля для атакующего базу данных, в то же время сохраняя возможность сравнения полученного хеша с оригинальным паролем. [[17](#_bookmark49)]  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](#_bookmark44)]  При хешировании паролей существует два важных соображения: это стои- мость вычисления и соль. Чем выше стоимость вычисления хеширующего алго- ритма, тем больше времени требуется для взлома его вывода методом «грубой си- лы». Криптографическая соль представляет собой данные, которые применяются в процессе хеширования для предотвращения возможности разгадать оригиналь- ный ввод с помощью поиска результата хеширования в списке заранее вычислен- ных 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 па- ролей password\_hash(), которое безопасно работает и с хешированием, и с провер- кой паролей.  При хешировании паролей рекомендуется применять алгоритм Blowfish, так как он значительно большей вычислительной сложности, чем MD5 или SHA1. [[17](#_bookmark49)]  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 ис- пользоваться 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* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 30 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | * все параметры в запросах должны быть экранированы; * необходимо использовать обратные одинарные кавычки при написании названий таблиц и полей в запросе.   Экранирование добавляет в строке перед кавычками (и другими спецсимво- лами) знак обратного слэша. Такая обработка лишает кавычки их статуса – они больше не определяют конец значения и не могут повлиять на логику SQL- выражения.  За экранирование значений отвечает функция mysqli\_real\_escape\_string().  Данный запрос является безопасным:  *$login = mysqli\_real\_escape\_string($link, $\_POST['login']);*  *$query\_login = mysqli\_query($link, "SELECT \* FROM users WHERE login =*  *'".$login."");*.  Так как имя поля или таблицы могут совпадать с ключевыми словами mysql, то необходимо использовать обратные кавычки, чтобы MySQL понял всё правильно. Теперь предыдущий запрос стал еще и надёжным в исполнении:  *$login = mysqli\_real\_escape\_string($link, $\_POST['login']);*  *$query\_login = mysqli\_query($link, "SELECT \* FROM `users` WHERE `login`*  *= '".$login."");*.  2.4.3 Описание защитных мер от XSS посредством возможностей языка PHP Основными методами защиты от XSS являются:   * фильтрация данных от пользователей, поступающих через форму сайта; * использование данных cookie только для HTTP.   Принцип фильтрации поступающих данных заключается в том, чтобы спе- циальные символы HTML заменять мнемониками – кодовыми представлениями символов в HTML, которые начинаются со знака амперсанда «&» и завершается точкой с запятой «;». Это нужно для того, чтобы злоумышленник не мог внедрить на страницу сайта JS-код, который навредил бы всем пользователем. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 31 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Функция htmlspecialchars() выполняет фильтрацию переданной строки и за- меняет все опасные символы в ней на подходящие HTML-мнемоники. Рассмот- рим 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 по- средством 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. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 32 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1. Разработка веб-приложения и оценка его защищенности    1. Постановка задачи   В данном разделе необходимо рассмотреть создание веб-интерфейса разра- батываемого веб-приложения, реализацию его серверной логики и настройку про- токола SSL. Также требуется оценить защищенность разработанного веб- приложения.   * 1. Создание веб-интерфейса   Рассмотрим интерфейс разработанного веб-приложения, который был реа- лизован с помощью языков HTML и CSS.  На рисунке 3.1 представлена шапка сайта до момента авторизации пользо- вателя.  P572#yIS1  Рисунок 3.1 – Шапка сайта до авторизации пользователя  На рисунке 3.1 можно видеть логотип сайта, его название, кнопку для авто- ризации пользователя и кнопку для регистрации пользователя.  На рисунке 3.2 изображена шапка сайта после авторизации пользователя. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 33 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P577#yIS1  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 аккаун- та.  При нажатии кнопки «Войти» (рисунок 3.1) появляется окно с формой для авторизации пользователя (рисунок 3.3).  P582#yIS1  Рисунок 3.3 – Окно с формой для авторизации  При нажатии кнопки «Регистрация» (рисунок 3.1) появляется окно с фор- мой для регистрации нового пользователя (рисунок 3.4). | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 34 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P586#yIS1  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). | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 35 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P590#yIS1  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 варь- ируются от 2 до 6 позиций.  На рисунке 3.6 представлен внешний вид созданного голосования, которое размещается на сайте в виде карточки. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 36 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P595#yIS1  Рисунок 3.6 – Карточка созданного голосования  На рисунке 3.7 представлен внешний вид голосования, у которого заверши- лось отведенное время.  P599#yIS1  Рисунок 3.7 – Карточка голосования, у которого истекло время На рисунке 3.8 представлен внешний вид подвала сайта.  P603#yIS1  Рисунок 3.8 – Подвал сайта | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 37 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | В следующем подразделе рассмотрим устройство всей серверной логики разработанного веб-приложения.   * 1. Разработка серверной логики веб-приложения      1. Описание структуры базы данных   Визуальная структура таблиц созданной базы данных представлена на ри- сунке 3.9.  P611#yIS1  Рисунок 3.9 – Структура созданной базы данных Созданная база данных voting\_platform\_bd имеет четыре таблицы:   1. users – таблица с данными о пользователях платформы; содержит в себе следующие строки:    * user\_id – ключевой идентификатор пользователя;    * user\_login – логин пользователя; | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 38 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | * user\_email – почта пользователя; * email\_verified – строка, указывающая подтверждена ли почта поль- зователя; * token – случайная строка длиною 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;  1. 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; 2. 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 прого- лосовавшего в данном голосовании;    * voting\_id – идентификатор голосования из таблицы voting, в котором уже проголосовал данный пользователь; | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 39 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1. results – таблица для подсчета голосов; содержит в себе следующие стро-   ки:   * + 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, указыва- ющий к какому голосованию относится данный вариант ответа.   3.3.2 Описание разработанных php-скриптов  Общая структура проекта представлена на рисунке 3.10.  P646#yIS1  Рисунок 3.10 – Общая структура проекта | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 40 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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.  P649#yIS1  Рисунок 3.11 – Логическая связь скриптов веб-приложения  Скрипт connection.php используется для подключения к базе данных и при- сутствует во всех скриптах, которые взаимодействуют с ней.  Скрипт index.php является ядром проекта. Он отрисовывает основную раз- метку страницы сайта. В данном скрипте происходит проверка присутствия дан- ных о пользователе в cookie. Если они имеются, то выводится приветствие и кнопки «Создать голосование» и «Выйти», а если данных нет, то есть пользова- | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 41 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | тель не авторизировался, то будут доступны кнопки «Войти» и «Зарегистриро- ваться».  При первом посещении сайта пользователю необходимо на нём зарегистри- роваться, за это отвечает скрипт register.php.  Регистрация пользователя происходит с подтверждением почты, которую он введёт. Для этого отправляется письмо на указанную почту с ссылкой, содер- жащей в себе случайно сгенерированный токен. После перехода по этой ссылке аккаунт пользователя активируется.  Пароль пользователя хешируется посредством алгоритма Blowfish и сохра- няется в базе данных в виде хеша, так как хранить пароль в открытом виде небез- опасно.  Также осуществляется проверка на соответствие введенных паролей, на со- ответствие 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 пользова- теля по ссылке, которая приходит ему на почту. В скрипте confirmed.php проис- ходит сравнение токена из ссылки с тем токеном, который был записан у пользо- вателя в базе данных при регистрации. Если они совпадают, то выходит сообще- ние о подтверждении почты.  Скрипт login.php, отвечающий за авторизацию пользователя, сначала прове- ряет 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 введен- ного пароля. Если пароль неверный, то выходит сообщение с ошибкой. Если па- роль верный, то формируется уникальный хеш для данной сессии и происходит проверка, стоял ли флаг «Запомнить меня» при авторизации. Это означает при-  вязку данной сессии к IP-адресу пользователя, то есть пока пользователь не разо- | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 42 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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 пропус- кается. Также в данном скрипте реализована защита от перебора пароля с помо- щью отправки специального письма пользователю, на аккаунт которого пытались зайти, введя трижды неправильные данные. Пользователю необходимо будет пе- рейти по ссылке, отправленной в письме, чтобы суметь в дальнейшем авторизо- ваться на сайте.  В конце скрипта устанавливаются сессионные cookie и происходит переад- ресация 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, ко- торая была отправлена ему на почту после трех неудачных попыток входа в его аккаунт.  В скрипте csrf\_protection.php реализована защита от CSRF-атак путем со- здания специального токена сессии, который в дальнейшем привязывается к каж- дой форме отправки данных.  В скрипте new\_voting.php осуществляется отправка данных из формы со- здания 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 коллек- ции созданных голосований. Затем они поочередно выводятся на страницу сайта. В каждом голосовании происходит скрытие кнопки «Проголосовать» в случаях, если пользователь не авторизовался, кончилось время для голосования или поль- зователь уже проголосовал.  При истечении времени, отведенного на голосование, строка с датой и вре- менем окончания голосования заменяется на надпись «Голосование заверши- лось». Также по завершению голосования происходит вывод итогов в виде полос  в процентном соотношении напротив каждого варианта ответа. Справа от полос | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 43 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | отображается количество голосов за данный вариант ответа, а ниже приводится статистика, сколько из зарегистрированных пользователей проголосовали в дан- ной теме.  Cкрипт send\_vote.php отвечает за отправку голоса за выбранный вариант. В скрипте осуществляется определение идентификатора голосования и варианта от- вета. Затем к данному варианту ответа прибавляется голос, а идентификатор про- голосовавшего 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 шифрова- ния отправляемых на клиентские системы данных. Сертификат SSL находится в открытом доступе для всех, кто запрашивает этот контент. Его можно использо- вать 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/" | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 44 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | W:\modules\http\Apache\_2.4-PHP\_7.2-7.4\bin\openssl x509 -req - sha256 -days 5475 -in server.csr -extfile config.txt -CA rootCA.crt -CAkey rootCA.key -CAcreateserial -out server.crt  Данный файл генерирует ключи и сертификаты. Пояснение используемых команд:   * *openssl*: это базовый инструмент командной строки для создания и управ- ления сертификатами OpenSSL, ключами и другими файлами; * *req*: данная субкоманда указывает, что мы хотим использовать управление запросами подписи сертификатов X.509 (CSR). X.509 — это стандарт инфра- структуры открытых ключей, используемый SSL и TLS для управления ключами и сертификатами. Вы хотим создать новый сертификат X.509, и поэтому исполь- зуем эту субкоманду; * *x509*: это дополнительно изменяет предыдущую субкоманду, сообщая утилите, что мы хотим создать самоподписанный сертификат, а не сгенерировать запрос на подпись сертификата, как обычно происходит; * *sha256*: указывает, что необходимо использовать хеш-функцию из семей- ства алгоритмов SHA-2 с размером хеша 256 бит; * *newkey rsa:2048*: указывает, что мы хотим генерировать новый сертификат и новый ключ одновременно. Мы не создали требуемый ключ для подписи сер- тификата на предыдущем шаге, и поэтому нам нужно создать его вместе с серти- фикатом. Часть rsa:2048 указывает, что мы создаем ключ RSA длиной 2048 бит. * *nodes*: этот параметр указывает OpenSSL пропустить опцию защиты сер- тификата с помощью пароля. Для чтения этого файла при запуске сервера без вмешательства пользователя нам потребуется Apache. Кодовая фраза может предотвратить это, поскольку нам придется вводить ее после каждого перезапус- ка; * *days 5475*: данный параметр устанавливает срок, в течение которого сер- тификат будет считаться действительным. Здесь мы устанавливаем срок действия на 15 лет; * *keyout*: эта строка указывает OpenSSL, где мы разместим создаваемый за-   крытый ключ; | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 45 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | * *out*: данный параметр указывает OpenSSL, куда поместить создаваемый сертификат; * *extfile config.txt*: указывает, что необходимо открыть файл config.txt с кон- фигурацией; * *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](http://www.voting.ru/)  Рассмотрим назначение строк:   * *nsComment*: строка, содержащая комментарий, который будет отображать- ся при просмотре сертификата; * *basicConstraints*: указывает, относится ли сертификат к центру сертифика- ции. 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 серти- фиката; * *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 следу- ющим 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.  P722#yIS1  Рисунок 3.12 – Состояние подключения до настройки SSL-сертификата  На рисунке 3.13 изображено состояние подключения после настройки SSL- сертификата.  P726#yIS1  Рисунок 3.13 – Состояние подключения после настройки SSL-сертификата Теперь подключение между клиентом и сервером защищено, значит SSL-  сертификат установлен правильно. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 47 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | * 1. Оценка защищенности разработанного веб-приложения      1. Демонстрация защитных мер при регистрации и авторизации пользо- вателя   При регистрации пользователей используется хеширование паролей функ- цией Blowfish. Таким образом, в базе данных пароль хранится не в открытом ви- де, а в виде, так называемого, отпечатка (рисунок 3.14).  P735#yIS1  Рисунок 3.14 – Хранение паролей в базе данных в виде хешей  При авторизации пользователя происходит сравнение хеша введенного па- роля с хешем, хранящимся в базе данных.  Также после регистрации пользователю приходит письмо на почту, в кото- ром 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.  P740#yIS1  Рисунок 3.15 – Пример содержимого письма при регистрации пользователя | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 48 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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).  P743#yIS1  Рисунок 3.16 – Содержимое полей в базе данных до подтверждения почты  После того как пользователь перейдет по ссылке, поле email\_verified при- нимает значение «*1*», а token – «*NULL*» рисунок (3.17).  P747#yIS1  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 зай- ти в аккаунт станет невозможно. Пример содержимого такого письма приведен ниже (рисунок 3.18). | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 49 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P751#yIS1  Рисунок 3.18 – Пример содержимого письма при попытке взлома аккаунта  Принцип работы аналогичен с подтверждением почты при регистрации. Пе- ременная «*verified\_code=g8Sz5RAWfu*» также заносится в базу данных. После то- го, 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, ко- торый записывается в данные сессии. Выведем сгенерированный токен на стра- нице сайта с помощью команды «*echo "<br>"."\_SESSION['csrf\_token'] = ".$\_SESSION['csrf\_token']."<br>";*» (рисунок 3.19).  P759#yIS1  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 то- кен сессии. И затем, при любой отправке формы, система сверяет токен, пропи- | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 50 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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, ели попробовать скомпрометировать токен, записанный в скрытом блоке (рисунок 3.20), то при отправке формы выйдет сообщение об ошибке (рисунок 3.21).  P768#yIS1  Рисунок 3.20 – Подделка сессионного токена  P770#yIS1  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). | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 51 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | P778#yIS1  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 всплываю- щее окно с данными из cookie (рисунок 3.24).  P782#yIS1  Рисунок 3.23 – Встроенный в разметку страницы скрипт  P784#yIS1  Рисунок 3.24 – Пользовательские данные из cookie | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 52 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Добавим защиту пользовательских cookie посредством флага HttpOnly. Те- перь посредством скрипта нельзя получать эти значения, во всплывающем окне пропадают пользовательские данные (рисунок 3.25).  P787#yIS1  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 бу- дет строка с текстом, в которой используются мнемоники (рисунок 3.26).  P791#yIS1  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 защищен- ности. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 53 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1. Безопасность жизнедеятельности    1. Постановка задачи   В данном разделе необходимо рассмотреть следующие вопросы по безопас- ности жизнедеятельности:   * характеристика условий труда при работе с ПК; * эргономические требования к рабочему месту пользователя; * требования охраны труда офисных работников; * пожарная безопасность.   4.2 Характеристика условий труда при работе с ПК  Согласно ст. 209 ТК РФ, условиями труда является совокупность факторов производственной среды и трудового процесса, оказывающих влияние на работо- способность и здоровье работника. Вредным производственным фактором, в свою очередь, является такой производственный фактор, воздействие которого на ра- ботника может привести к его заболеванию. [[20](#_bookmark51)]  Трудовой кодекс обязывает работодателей обеспечить безопасные условия и охрану труда работников на каждом рабочем месте (ст. 212 ТК РФ). Регулярная работа за компьютером сопровождается постоянным влиянием множества вред- ных для здоровья факторов. Зачастую специалисты, проводящие больше 12 часов в день за компьютером, со временем начинают страдать от профессиональных за- болеваний. Поэтому для работников, которые работают с персональной электрон- но-вычислительной машиной (ПЭВМ) очень важна правильная организация рабо- чего места. В таблице 4.1 приведены требования к условиям труда согласно Сан- ПиН 2.2.2/2.4.1340-03. В таблице 4.2 приведены временные допустимые уровни ЭМП, создаваемых ПЭВМ на рабочих местах, а в таблице 4.3 – визуальные пара- метры ВДТ, контролируемые на рабочих местах. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 54 |
| *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 м2 и с ВДТ на базе плоских дискретных экранов (жидкокристаллические, плазменные) – 4,5 м2. |
| 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. |

*Взам. инв. № Инв. № дубл.*

*Подпись и дата*

*Изм.*

*Инв. № подл.*

*Лист*

*№ докум*

*Подпись Дата*

# *ФАЭС.10.05.02.55.ПЗ*

*Лист*

*Подпись и дата*

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 блесткость от источников освещения, при этом яркость светящихся поверхностей (окна, светильники и др.), находящихся в поле зрения, должна  быть не более 200 кд/м2. |
| 5 | Следует ограничивать отраженную блесткость 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/м2 и яркость  потолка не должна превышать 200 кд/м2. |
| 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 |

*Взам. инв. № Инв. № дубл.*

*Подпись и дата*

*Изм.*

*Инв. № подл.*

*Лист*

*№ докум*

*Подпись Дата*

# *ФАЭС.10.05.02.55.ПЗ*

*Лист*

56

*Подпись и дата*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Таблица 4.2 - Временные допустимые уровни ЭМП, создаваемых ПЭВМ на рабочих местах [18]  Таблица 4.3 - Визуальные параметры ВДТ, контролируемые на рабочих местах [18]  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 производи- тельность 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 рабо-  чих мест пользователей ПЭВМ сведены в таблицу 4.4. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 57 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |
| --- | --- | --- |
| 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 маг-  нитного 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 |

|  |  |  |
| --- | --- | --- |
| 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 x 1E(-4L), где L - проектное расстояние  наблюдения, мм |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Таблица 4.4 – Общие требования к организации рабочих мест пользователей ПЭВМ [18]  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] | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 58 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |
| --- | --- |
| № | 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 |
| 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. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Продолжение таблицы 4.5  4.4 Требования охраны труда офисных работников  Согласно ст. 209 ТК РФ, требованиями охраны труда являются государ- ственные нормативные требования охраны труда, в том числе стандарты безопас- ности труда, а также требования охраны труда, установленные правилами и ин- струкциями по охране труда. [[20](#_bookmark51)]  Требования охраны труда офисных работников сведены в таблицу 4.6. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 59 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |
| --- | --- |
| № | 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:   * ширину и глубину поверхности сиденья не менее 400 мм; * поверхность сиденья с закругленным передним краем; * регулировку высоты поверхности сиденья в пределах 400 - 550 мм и углам наклона вперед до 15 град, и назад до 5 град.; * высоту опорной поверхности спинки 300 +-20 мм, ширину - не менее 380 мм и радиус кривизны горизонтальной плоскости - 400 мм; * угол наклона спинки в вертикальной плоскости в пределах +-30 градусов; * регулировку расстояния спинки от переднего края сиденья в пределах 260 - 400 мм; * стационарные или съемные подлокотники длиной не менее 250 мм и шириной - 50 - 70   мм;   * регулировку подлокотников по высоте над сиденьем в пределах 230 +-30 мм и внутреннего расстояния между подлокотниками в пределах 350 -500 мм. |
| 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. |

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. |

*Взам. инв. № Инв. № дубл.*

*Подпись и дата*

*Изм.*

*Инв. № подл.*

*Лист*

*№ докум*

*Подпись Дата*

# *ФАЭС.10.05.02.55.ПЗ*

*Лист*

60

*Подпись и дата*

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 накапливания 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. |

*Взам. инв. № Инв. № дубл.*

*Подпись и дата*

*Изм.*

*Инв. № подл.*

*Лист*

*№ докум*

*Подпись Дата*

# *ФАЭС.10.05.02.55.ПЗ*

*Лист*

61

*Подпись и дата*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Продолжение таблицы 4.6  4.5 Пожарная безопасность  Каждый сотрудник независимо от занимаемой должности обязан знать и строго выполнять правила пожарной безопасности, не допускать действий, кото- рые могут привести к пожару. Основные причины пожаров на предприятиях - не- осторожное обращение с огнем, оставленные без присмотра электроприборы, проведение с нарушениями требований правил пожарной безопасности огневых, строительных и других пожароопасных работ, курение в не установленных ме- стах, использование легковоспламенямых веществ, нарушение технологий.  Работодатели обязаны обеспечить полное, своевременное и неукоснитель- ное 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](#_bookmark48)]  Работодатель или лицо, на которого возложено проведение работ по пожар- ной безопасности в организации, обязан:   * назначить лиц, ответственных за пожарную безопасность в структурных подразделениях; * квалифицировать все рабочие места по категориям взрывоопасной и по- жарной опасности; | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 62 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |
| --- | --- |
| № | 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. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | * разработать и утвердить инструкции пожарной безопасности и планы эва- куации в случае пожара; * организовывать проведение противопожарных инструктажей и занятий по пожарной безопасности с ответственными лицами; * приобрести и своевременно обновлять средства пожаротушения.   Работники допускаются к работе на объекте только после прохождения обучения мерам пожарной безопасности. Требования к организации обучения ме- рам пожарной безопасности работников организаций определены Приказом МЧС России от 12 декабря 2007 г. N 645.  Основными видами обучения работников организаций мерам пожарной безопасности являются:   * противопожарный инструктаж; * пожарно-технический минимум.   Порядок и сроки проведения противопожарного инструктажа и прохожде- ния пожарно-технического минимума определяются руководителем организации. Обучение мерам пожарной безопасности осуществляется в соответствии с норма- тивными документами по пожарной безопасности.  Проверка знаний требований пожарной безопасности руководителей, спе- циалистов и работников организации осуществляется по окончании обучения по- жарно-техническому минимуму с отрывом от производства и проводится квали- фикационной комиссией, назначенной приказом (распоряжением) руководителя организации, состоящей не менее чем из трех человек.  В состав квалификационной комиссии входят руководители и штатные пе- дагогические работники обучающих организаций и по согласованию специалисты федеральных органов исполнительной власти, органов исполнительной власти субъектов Российской Федерации, органов местного самоуправления, органов государственного пожарного надзора.  В обязательном порядке должны быть разработаны инструкции о мерах противопожарной безопасности, а объекты обеспечены первичными средствами  пожаротушения. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 63 |
| *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, подручны- ми средствами; 2. подготовить к эвакуации материальные ценности, документацию; 3. слушать распоряжения руководителя отдела, организованно покинуть здание; 4. рассмотреть вариант эвакуации через запасные выходы, пожарную лест- ницу, соседние помещения; организовать встречу подразделений пожарной охра- ны; 5. при невозможности покинуть здание (задымление, высокая температура) плотно закрыть дверь помещения, уплотнить тканью щели, вентиляционные от- верстия, открыть окно и ждать пожарных. Стоит запомнить, что при задымлении над полом воздух более чист. Это может помочь при эвакуации и ожидании по- мощи.   4.6 Выводы по разделу  В данной главе были рассмотрены характеристики условий труда при рабо- те с ПК и эргономические требования к рабочему месту пользователя. Определе- ны требования охраны труда офисных работников, а также рассмотрены правила пожарной безопасности. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 64 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1. Технико-экономическое обоснование работы    1. Постановка задачи   Целью выпускной квалификационной работы являлась разработка веб- приложения для защищенного электронного голосования. Веб-приложение явля- ется программным кодом, который, согласно ст. 1259 ГК РФ, относится к объек- там 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](#_bookmark47)]  Таблица 5.1 – План разработки программного продукта | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 65 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование этапов | Виды работ | Исполнитель (должность, квалифика-  ция) | Количество исполните- лей |
| Анализ предмет- ной области раз- работки | Определение объекта разра-  ботки | Студент | 1 |
| Анализ основных угроз и уяз-  вимостей | Студент | 1 |
| Разработка модели наруши- теля информационной без-  опасности | Студент | 1 |
| Проектирование | Планирование архитектуры  веб-приложения | Студент | 1 |
| Выбор языковых и программ-  ных средств разработки | Студент | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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 предпочти- тельным будет использование метода экспертных оценок.  В этом методе для каждого этапа требуется экспертным путем определить три оценки трудоемкости, в днях: | | | | | | | | |  |
| *Подпись и дата* |  | * наиболее вероятная величина затрат, 𝑚𝑚𝑖𝑖 ; * наиболее возможная величина затрат, 𝑏𝑏𝑖𝑖 [[15](#_bookmark47)].   На основании экспертных оценок средняя величина для ai, mi и bi определя- ется по формуле (5.1):  3Трук + 2Тавт  Т� = , (5.1)  5  где Т� – среднее время, полученное на основании экспертных оценок;  Трук – оценка затрат времени, данная руководителем;  Тавт – оценка затрат времени, данная автором проекта.  Результаты расчета средней оценки затрат времени на разработку про- граммного продукта приведены в таблице 5.2. | | | | | | | | | | |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  | |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | | | *Лист* | |
|  | |  |  |  |  | 66 | |
| *Изм.* | | *Лист* | *№ докум* | *Подпись* | *Дата* |

– наименее возможная величина затрат, 𝑎𝑎𝑖𝑖 ;

|  |  |
| --- | --- |
|  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Таблица 5.2 – Время, затраченное на разработку программного продукта  На основе средних оценок рассчитываются математическое ожидание и от- клонение по каждому этапу разработки программного продукта. Формула расчета математического ожидания для i-гo этапа:  𝑎𝑎𝑖𝑖 + 4𝑚𝑚𝑖𝑖 + 𝑏𝑏𝑖𝑖  𝑀𝑀𝑂𝑂𝑖𝑖 = 6 , (5.2)  где 𝑀𝑀𝑂𝑂𝑖𝑖 – математическое ожидание для i-гo этапа;  𝑎𝑎𝑖𝑖 , 𝑚𝑚𝑖𝑖 , 𝑏𝑏𝑖𝑖 – средние значения.  Стандартное отклонение для каждого этапа разработки программного про- дукта определяется по формуле:  b𝑖𝑖 − a𝑖𝑖  𝐺𝐺𝑖𝑖 = 6 , (5.3)  где 𝐺𝐺𝑖𝑖 – стандартное отклонение по i-му этапу.  Зная математическое ожидание по каждому этапу, рассчитываем общую ве- личину математического ожидания в целом по программному продукту:  𝑀𝑀𝑂𝑂 = � 𝑀𝑀𝑂𝑂𝑖𝑖, (5.4)  где MO – общая величина математического ожидания. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 67 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Этапы разработки программного продукта | Наименее возможная величина затрат  (𝑎𝑎𝑖𝑖 ), дни | | | Наиболее вероятная величина затрат  (𝑚𝑚𝑖𝑖 ), дни | | | Наиболее возможная величина  затрат (𝑏𝑏𝑖𝑖 ), дни | | |
| Тавт | Трук | Т� | Тавт | Трук | Т� | Тавт | Трук | Т� |
| 1.Анализ предметной  области разработки | 7 | 5 | 5,8 | 10 | 8 | 8,8 | 16 | 14 | 14,8 |
| 2. Проектирование | 13 | 12 | 12,4 | 17 | 15 | 15,8 | 20 | 19 | 19,4 |
| 3. Разработка | 22 | 19 | 20,2 | 26 | 23 | 24,2 | 29 | 26 | 27,2 |
| 4. Тестирование и  отладка | 4 | 6 | 5,2 | 6 | 8 | 7,2 | 8 | 10 | 9,2 |
| 5. Внедрение | 3 | 2 | 2,4 | 4 | 3 | 3,4 | 5 | 4 | 4,4 |

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Стандартное отклонение G в целом по программному продукту рассчитыва- ется по следующей формуле:  𝐺𝐺 = �� 𝐺𝐺2, (5.5)  𝑖𝑖  где 𝐺𝐺 –стандартное отклонение;  𝐺𝐺𝑖𝑖 – стандартное отклонение по i-му этапу.  На основе расчетов математического ожидания (5.4) и стандартного откло- нения (5.5) рассчитываем коэффициент вариации – коэффициент согласованности мнения экспертов. Коэффициент вариации рассчитывается по формуле:  𝐺𝐺𝑖𝑖  𝑣𝑣𝑖𝑖 = 𝑀𝑀𝑂𝑂 , (5.6)  𝑖𝑖  где 𝑣𝑣𝑖𝑖 – коэффициент вариации по i-му этапу [[15](#_bookmark47)]. Все произведенные расчеты сведены в таблицу 5.3.  Таблица 5.3 – Затраты на разработку программного продукта  В итоге коэффициент вариации равен 0,039 и не превосходит 0,33. Поэтому мнения экспертов считаются согласованными. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 68 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Средняя величина затрат по этапам, | | | Матем. ожидание (МОi, дни) | Станд. отклонение  (Gi, дни) | Коэффициент вариации  (vi) |
| Этапы | дни | | |
| разработки | Наименее | Наиболее | Наиболее |
| программного продукта | возможная  величина затрат | вероятная  величина затрат | возможная  величина затрат |
|  | (ai, дни) | (mi, дни) | (bi, 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 |

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 5.3 Расчет себестоимости и цены программного продукта  Себестоимость программного продукта – это все виды затрат, понесенные при разработке продукта. Чтобы определить себестоимость разработки применя- ется метод экспертных оценок.  Себестоимость программного продукта определяется по формуле (5.7):  З  С = ∙ 𝑘𝑘 ∙ 𝑘𝑘ТЕР ∙ 𝑘𝑘ПР ∙ (t1 + t2) ∙ (1 + kН) + 8 ∙ t3 ∙ CМ + 8 ∙ t4 ∙ CИ, (5.7)  𝑚𝑚  где З – среднемесячная заработная плата php-разработчика, З = 30000;  𝑘𝑘ТЕР – территориальный коэффициент, 𝑘𝑘ТЕР = 1,2 (для НСО);  𝑘𝑘ПР – коэффициент премии, 𝑘𝑘ПР = 1;  *k* – коэффициент, учитывающий страховые взносы (фонды пенсионного, социального и медицинского страхования), 𝑘𝑘 = 1,3;  *m* – количество рабочих дней в месяце, 𝑚𝑚 = 22;  kН – коэффициент, учитывающий накладные расходы (отопление, освеще- ние, уборка и т. д.), kН = 0,4;  𝑡𝑡1 – время, затраченное разработчиком на разработку требований к про- грамме, т.е. подготовительное время, которое необходимо потратить, чтобы преступить к написанию программы и отладки программы, чел./дни;  𝑡𝑡2 – сборка устройства, составление алгоритма в программе, время, затра- ченное на написание и отладку программы, чел./дни;  𝑡𝑡3 – время, затраченное на разработку программы с использованием машин- ного времени, чел./дни;  𝑡𝑡4 – время работы в сети интернет, дни;  CИ – стоимость 1 часа работы в сети интернет, руб. (оценивается через або- нентскую плату);  CМ – стоимость одного часа машинного времени.  Для расчета стоимости одного часа машинного времени, необходимо опре- делить затраты на эксплуатацию ПК за год по следующей формуле: | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 69 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |
| --- | --- |
|  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Зэл + За + Зкомпл + Зпр  𝐶𝐶м = 𝑇𝑇 . (5.8)  общ  Общее время работы компьютера за год составляет:  𝑇𝑇общ = 22 ∗ 12 ∗ 8 = 2112 (часов)  Затраты на электроэнергию за год работы (на данный момент тариф 𝐶𝐶эл со- ставляет 2,68 руб. за кВт/ч):  Зэл = 𝑇𝑇общ ∗ 𝐶𝐶эл ∗ 𝑃𝑃, (5.9)  где 𝑃𝑃 – потребляемая мощность ноутбука по паспортным данным в час, 𝑃𝑃 = 135  Вт/ч.  По (5.9) затраты на электроэнергию за год работы составляют:  Зэл = 2112 ∗ 2,68 ∗ 0,135 = 764,1 (руб. )  Амортизационные отчисления в год определяются как процент отчисления на амортизацию от первоначальной стоимости основных производственных фон- дов. Процент отчисления на амортизацию, согласно ст. 258 НК РФ, составляет 34-50% от первоначальной стоимости ПК (компьютер относится ко второй груп- пе имущества со сроком полезного использования свыше 2 лет до 3 лет включи- тельно). Затраты на ПК определяются по формуле:  За = С ∗ Пр, (5.10)  где С – стоимость ноутбука, руб.;  Пр – процент отчисления на амортизацию, Пр = 40%.  Получим:  За = 55000 ∗ 0,4 = 22000 (руб. )  Затраты на комплектующие материалы составляют:  Зкомпл = 5000 (руб. )  Прочие расходы составляют 5% от общей суммы затрат:  0,05 ∗ (Зэл + За + Зкомпл)  Зпр = 0,95 . (5.11) | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 70 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | По (5.11) прочие расходы равны:  0,05 ∗ (764,1 + 22000 + 5000)  Зпр = 0,95 = 1461,27 (руб. )  По формуле 5.8 стоимость одного часа машинного времени равна:  764,1 + 22000 + 5000 + 1461,27  𝐶𝐶м = 2112 = 13,84 (руб. )  Тариф на услугу интернет составляет 635 руб. в месяц, следовательно, сто- имость 1 часа работы в сети интернет равен:  635  Cи = 30 = 21,2 (руб. )  Заключительным этапом расчета является распределение ранее рассчитан- ной трудоемкости (таблица 5.3) по 4 направлениям:   * 𝑡𝑡1 включает первые два этапа: анализ предметной области разработки и проектирование:   𝑡𝑡1 = 9,3 + 15,83 = 25,1 (дней)   * 𝑡𝑡2 включает этапы: разработка, тестирование и отладка и внедрение:   𝑡𝑡2 = 24,03 + 7,2 + 3,4 = 34,6 (дней)   * 𝑡𝑡3 включает время работы ПК для разработки программы:   𝑡𝑡3 = 50 (дней)   * 𝑡𝑡4 включает время использования интернета для разработки программы:   𝑡𝑡4 = 45,5 (дней)  Наконец, итоговая себестоимость программного продукта составляет:  30000  С = ∙ 1,3 ∙ 1,2 ∙ 1 ∙ (25,1 + 34,6) ∙ (1 + 0,4) + 8 ∙ 50 ∙ 13,84 + 8 ∙ 45,5 ∙ 21,2  22  = 191050,25 (руб. )  В случае, если программный продукт будет доработан и реализован на рын- ке, следует рассчитать цену по следующей формуле:  𝑃𝑃  Ц = 𝐶𝐶 ∗ (1 + ), (5.12)  100  где 𝐶𝐶 – себестоимость разработки программы, руб;  𝑃𝑃 – рентабельность, руб. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 71 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |
| --- | --- |
|  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Определим цену программного продукта, при условии, что значение рента- бельности равно 20%:  20  Ц = 191050,25 ∙ �1 + 100� = 229260,3 (руб. )  Цена с учетом налога на добавленную стоимость находится по формуле:  ЦНДС = Ц ∗ КНДС, (5.13)  где Ц – цена программного продукта;  КНДС – коэффициент, учитывающий ставку налога на добавленную стоимость (НДС), КНДС = 1,20. [[15](#_bookmark47)]  Цена с учетом налога на добавленную стоимость составит:  ЦНДС = 229260,3 ∗ 1,20 = 275112,36 (руб. )  5.4 Выводы по разделу  В данном разделе была определены и рассчитаны трудоемкость и длитель- ность работ, а также рассчитаны себестоимость и цена программного продукта. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 72 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |
| --- | --- |
|  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Заключение  В результате выполнения выпускной квалификационной работы была до- стигнута поставленная цель и ее задачи.  В первой главе было определено назначение разрабатываемого веб- приложения. Также выявлены актуальные риски и угрозы: недостатки аутентифи- кации, 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, Ja- vaScript, 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 оце- нена защищенность разработанного веб-приложения, в следствие чего можно ска- зать, 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* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 73 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Список литературы   1. Open Server Panel // Open Server. Лучшая панель управления сервером для   Windows. – URL: https://ospanel.io/ (дата обращения: 05.11.20).   1. OWASP Top Ten 2017 // OWASP – 2017. – URL: https://owasp.org/www- project-top-ten/2017/ (дата обращения: 19.10.20). 2. SSL-сертификаты бывают разные // Kaspersky daily. – URL: https://[www.kaspersky.ru/blog/certificates-are-different/20227/](http://www.kaspersky.ru/blog/certificates-are-different/20227/) (дата обращения: 01.11.20). 3. TLS и SSL: Необходимый минимум знаний // MNorin.com. – URL: https://mnorin.com/tls-ssl-neobhodimy-j-minimum-znanij.html/ (дата обращения: 01.11.20). 4. Wolf, P. Introducing Electronic Voting: Essential Considerations / P. Wolf, R. Nackerdien, D. Tuccinardi. – Stockholm.: Bulls Graphics, 2011. – 36 p. 5. Банк данных угроз безопасности информации // ФСТЭК России. – URL: https://bdu.fstec.ru/threat/ (дата обращения: 25.10.20). 6. Буя, П.М. Защита информации в телекоммуникационных системах / П.М. Буя // Модель нарушителя информационной безопасности. – 2016. – №4. – С. 7-15. 7. ГОСТ Р 50922-2006. Защита информации. Основные термины и опреде- ления. – М.: Стандартинформ, 2008. – 8 с. 8. ГОСТ Р 53114-2008. Защита информации. Обеспечение информационной безопасности в организации. Основные термины и определения. – М.: Стандар- тинформ, 2009. – 16 с. 9. ГОСТ Р ИСО 9241-151-2014. Эргономика взаимодействия человек- система. Часть 151. Руководство по проектированию пользовательских интерфей- сов сети Интернет. – М.: Стандартинформ, 2019. – 46 с. 10. Ерохина, О.В. Технологии электронного голосования в России / О.В. Ерохина // Вестник университета. – 2019. – № 11. – С. 5-11. 11. Иванов, М.А. Хеш-функции. Теория, применение и новые стандарты   (часть 1) / М.А. Иванов, А.В. Стариковский. – 2017. – 31 с. | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 74 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1. Киселев, Ю.А. Алгоритм AES – Пример современного симметричного криптопреобразования: Методические указания к лабораторной работе / Ю.А. Ки- селев, О.Е. Александров. – Екатеринбург: кафедра молекулярной физики УГТУ- УПИ. – 2016. – 28 с. 2. Методика определения угроз безопасности информации в информацион- ных системах: методический документ // ФСТЭК России. – 2015. – С. 43. 3. Мухина, И.С. Технико-экономическое обоснование проектных решений при выполнении выпускных квалификационных работ: Учебное пособие / И.С. Мухина – Новосибирск: СибГУТИ, 2020. – 85 с. 4. Постановление Правительства РФ от 25.04.2012 N 390 (ред. от 23.04.2020) «О противопожарном режиме» (вместе с «Правилами противопожар- ного режима в Российской Федерации»). 5. Руководство по PHP. Безопасное хеширование паролей // PHP. – URL: https://[www.php.net/manual/ru/faq.passwords.php#faq.passwords.bestpractice/](http://www.php.net/manual/ru/faq.passwords.php%23faq.passwords.bestpractice/) (дата обращения: 08.11.20). 6. Руководство по PHP. Что такое PHP? // PHP. – URL: https://[www.php.net/manual/ru/intro-whatis.php/](http://www.php.net/manual/ru/intro-whatis.php/) (дата обращения: 28.10.20). 7. СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы. 8. Трудовой кодекс Российской Федерации от 30.12.2001 № 197-ФЗ (ред. от 09.11.2020) // Собрание законодательства РФ. – 07.01.2002. 9. Уязвимости и угрозы веб-приложений в 2019 году // Positive Technologies  * 2018. – URL: https:/[/w](http://www.ptsecurity.com/ru-ru/research/analytics/web-)w[w.ptsecurity.com/ru-ru/research/analytics/web-](http://www.ptsecurity.com/ru-ru/research/analytics/web-) vulnerabilities-2020/ (дата обращения: 20.10.20).   1. Эмм, Д. Главные риски в онлайн-голосовании: мнение охотника за кибе- ругрозами / Д. Эмм // Information Security/ Информационная безопасность – 2018.   –№ 1. – URL: <http://lib.itsec.ru/articles2/Oborandteh/glavnye-riski-v-onlayn-> golosovanii-mnenie-ohotnika-za-kiberugrozami/ (дата обращения: 15.10.20).   * 1. Язык программирования PHP // Depix. – URL:   https://depix.ru/articles/yazyk\_programmirovaniya\_php/ (дата обращения: 30.10.20). | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 75 |
| *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">  <img src="assets/images/logo.png" alt="Логотип" width="150" height="150">  </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> | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 76 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | </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"> | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 77 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Нажимая кнопку "Зарегистрироваться", Я безоговорочно соглашаюсь с <a href="#">правилами сайта</a>.  </label>  </div>  </form>  </div>  </div>  <div class="modal" id="modal\_create">  <div class="modal\_container modal\_container\_create">  <span class="modal\_close" id="modal\_close\_create">&times;</span>  <h3>Создание нового голосования</h3>  <form action="includes/new\_voting.php" method="POST" class="modal\_create\_form" name="modal\_create\_form">  <div class="form\_element">  <label for="voting\_theme\_new">Тема  голосования: </label>  <input type="text" id="voting\_theme\_new" name="voting\_theme\_new" required>  </div>  <div class="form\_element">  <label for="voting\_content\_new">Описание:  </label>  <textarea name="voting\_content\_new" id="voting\_content\_new" cols="30" rows="10" placeholder="Введите текст..." required></textarea>  </div>  <div class="form\_element">  <label for="voting\_left\_time">Осталось дней:  </label>  <input type="text" id="voting\_left\_time" name="voting\_left\_time" required>  </div>  <div class="voting\_options">  <p>Варианты голосования:</p>  <div id="voting\_options">  <div class="form\_element voting\_option">  <label for="voting\_option\_1">1.</label>  <input type="text" id="voting\_option\_1" name="voting\_option\_1" required>  </div>  <div class="form\_element voting\_option">  <label for="voting\_option\_2">2.</label>  <input type="text" id="voting\_option\_2" name="voting\_option\_2" required>  </div>  </div>  </div>  <div class="add\_del\_option">  <span id="add\_voting\_option">&#43;</span>  <span id="delete\_voting\_option">&ndash;</span>  </div>  <input type="hidden" name="csrf\_token" value="<? echo $csrf\_token?>">  <input type="submit" name="submit" value="Создать голосование" id="create\_new\_voting\_button"> | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 78 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | </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 = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPRQSTUVWXYZ0123456789";  $code = "";  $chars\_len = strlen($chars) - 1; while (strlen($code) < $length) {  $code .= $chars[random\_int(0, $chars\_len)];  }  return $code;  } | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 79 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | $new\_login = $\_POST['new\_login'];  $new\_email = $\_POST['new\_email'];  $new\_password = $\_POST['new\_password'];  $new\_password\_repeat = $\_POST['new\_password\_repeat'];  // Проверка сложности пароля  if (strlen($new\_password) < 4 or !preg\_match("/[A-Z]+/",  $new\_password) or !preg\_match("/[0-9]+/", $new\_password)) {  print\_r("Пароль должен:<br>- содержать только латинские сим- волы;<br>- быть длиною более 3-х символов;<br>- содержать хотя бы од- ну цифру и заглавную букву.");  exit();  }  // Проверка на совпадение введенных паролей  if ($new\_password != $new\_password\_repeat) {  print\_r("Пароли не совпадают!");  exit();  }  $errors = [];  // Проверка логина на допустимые символы  if (!preg\_match("/^[a-zA-Z0-9]+$/", $new\_login)) {  $errors[] = "Логин должен состоять только из букв английского алфавита и цифр";  }  // Проверка логина на допустимую длину  if (strlen($new\_login) < 3 or strlen($new\_login) > 30) {  $errors[] = "Длина логина должна быть от 3 до 30 символов";  }  $new\_login\_slash = mysqli\_real\_escape\_string($connection,  $new\_login);  $new\_email\_slash = mysqli\_real\_escape\_string($connection,  $new\_email);  // Запрос в БД, чтобы узнать, имеются ли такие же логин и/или почта  $login\_check\_query = mysqli\_query($connection, "SELECT `user\_id` FROM `users` WHERE `user\_login` = '".$new\_login\_slash."' OR `us- er\_email` = '".$new\_email\_slash."'");  if (mysqli\_num\_rows($login\_check\_query) > 0) {  $errors[] = "Пользователь с таким логином или почтой уже су- ществует";  }  // Если ошибок нет, то пользователь создается в БД. В противном случае выводится список ошибок и соединение с БД прерывается  if (count($errors) == 0) {  // Хеширование пароля с использованием алгоритма Blowfish  $new\_password\_hash = password\_hash($new\_password, PASS- WORD\_BCRYPT);  $new\_password\_hash\_slash = mysqli\_real\_escape\_string($connection, $new\_password\_hash);  $token = generateCode(30); | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 80 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | $token\_slash = mysqli\_real\_escape\_string($connection, $to-  ken);  // Переменная $headers для Email заголовка  $headers = "From: [<mail@voting.ru>](mailto:mail@voting.ru)\r\n";  // Сообщение для Email  $message = "  Чтобы подтвердить Email, перейдите сюда:  [http://voting.ru/includes/confirmed.php?token=".$token."](http://voting.ru/includes/confirmed.php?token) ";  // Добавление данных о пользователе в БД  mysqli\_query($connection, "INSERT INTO `users` SET `us- er\_login` = '".$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,  $headers)) {  // Если да, то выводит сообщение  echo "Проверьте свою почту для подтверждения аккаунта";  }  exit();  }  else {  print\_r("<b>При регистрации произошли следующие ошиб- ки:</b><br>");  foreach($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`, `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` = | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *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, $que- ry\_user\_data['user\_id'])."'");  if (mysqli\_num\_rows($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 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 = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPRQSTUVWXYZ0123456789";  $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`, `us- er\_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. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 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'], $que- ry\_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 `us- er\_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); | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 83 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | // Сообщение для Email  $message = "  Обнаружена попытка взлома аккаунта, перейдите сюда: [http://voting.ru/includes/verified\_user.php?verified\_code=".$verified](http://voting.ru/includes/verified_user.php?verified_code)  \_code." ";  // Отправка письма пользователю, которого пы-  таются взломать  mail($query\_login\_data['user\_email'], "Под- твердите Ваши действия на сайте", $message, "From:  <[mail@voting.ru](mailto:mail@voting.ru)>\r\n");  // Заносим в БД сгенерированный код, который  прийдет в письме  mysqli\_query($connection, "UPDATE `users` SET  `verified\_login\_code` = '".$verified\_code\_slash."' WHERE `user\_id` = '".$query\_login\_data['user\_id']."'");  print\_r("Не удается зайти в аккаунт! Вам на почту отправлено письмо с дальнейшими действиями.");  } else {  print\_r("Вы ввели неправильный ло-  гин/пароль");  mysqli\_query($connection, "UPDATE `users` SET  `err\_login\_number` = `err\_login\_number`+1 WHERE `user\_id` = '".$query\_login\_data['user\_id']."'");  }  }  }  }  }  }  ?>  *login\_check.php*:  <?php  // Соединямся с БД  include("connection.php");  if (isset($\_COOKIE['id']) and isset($\_COOKIE['hash'])) {  // Запрашиваем IP пользователя, id которого указан в cookie  $query = mysqli\_query($connection, "SELECT \*,INET\_NTOA(`user\_ip`) AS user\_ip FROM `users` WHERE `user\_id` = '".mysqli\_real\_escape\_string($connection, $\_COOKIE['id'])."' LIMIT 1");  $userdata = mysqli\_fetch\_assoc($query);  // Если данные не совпадают, то есть на данный аккаунт пытаются зайти с другого IP, то удаляются cookie и выходит сообщение об ошибке  if(($userdata['user\_hash'] !== $\_COOKIE['hash']) or ($userdata['user\_id'] !== $\_COOKIE['id'])  or (($userdata['user\_ip'] !== $\_SERVER['REMOTE\_ADDR']) and ($userdata['user\_ip'] !== "0.0.0.0"))) {  setcookie("id", "", time() - 3600\*24\*30\*12, "/"); | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 84 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 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.* |  |
|  |  |  |  |  | *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);  $voting\_content\_new = htmlspecial- chars($\_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"))."', | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 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,  $last\_voting\_id);  /\*----- Добавление вариантов созданного голосовании в таблицу results  \*/  foreach (array\_keys($\_POST) as $key) {  if (strpos($key, 'option') !== false) { mysqli\_query($connection, "INSERT INTO `results` SET `op-  tion\_name` = '".mysqli\_real\_escape\_string($connection, htmlspecial- chars($\_POST[$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>  <img src="assets/images/your\_vote.png" alt="Ваш голос важен!" width="200" height="200">  </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']); | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 87 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | // 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']/$sum\_votes)\*100)?>"></progress>  </td>  <td>  &nbsp;&nbsp;<? echo "(".$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> | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 88 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | <? else:  $option\_number = 0; foreach($query\_options as $option):  $option\_number++; ?>  <label for="option\_<?echo $op-  tion['result\_id']?>">  <input type="radio" id="option\_<?echo  $option['result\_id']?>" name="voting\_<?echo $voting['voting\_id']?>" value="option\_<?echo $option\_number?>">  <span><?echo $op-  tion['option\_name']?></span>  </label>  <? endforeach ?>  <? endif ?>  </div>  <?  // Если пользователь авторизован, срок голосования еще не истек и данный пользователь еще не голосовал в этом голосовании, то выводится кнопка Проголосовать  if (!empty($\_COOKIE['id']) and strtotime(date("d.m.Y H:i:s")) < $voting\_end\_datetime\_sec and mysqli\_num\_rows($query\_voted\_users) == 0): ?>  <input type="hidden" name="csrf\_token" value="<? echo  $csrf\_token?>">  <input type="submit" id="voting\_sub\_1" name="submit" val- ue="Проголосовать">  <? endif ?>  </form>  <div class="countdown">  <?  // Если время для голосования еще не вышло, то выводится надпись о времени и дате окончания голосования, в противном случае, выводится надпись о завершении голосования  if (strtotime(date("d.m.Y H:i:s")) < $vot- ing\_end\_datetime\_sec):?>  Голосование закончится:&nbsp;&nbsp;  <span class="datetime">  <?  $voting\_end\_datetime = date("d.m.Y в H:i:s",  $voting\_end\_datetime\_sec);  echo $voting\_end\_datetime;  ?>  </span>  <? else: ?>  Голосование завершилось  <? endif ?>  </div>  </div>  </div>  <? endforeach  ?>  *send\_vote.php*:  <?php session\_start(); | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 89 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *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 порядкого 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: /");  }  ?> | | | | | | |
| *Подпись и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подпись и дата* |  |
| *Инв. № подл.* |  |
|  |  |  |  |  | *ФАЭС.10.05.02.55.ПЗ* | *Лист* |
|  |  |  |  |  | 90 |
| *Изм.* | *Лист* | *№ докум* | *Подпись* | *Дата* |