## Введение

В данном техническом задании представлены требования к дизайну, безопасности и функциональности веб-приложения, которое будет служить инструментом для автоматизации процессов управления в спортивном клубе «Спортсила».

Цель данного приложения — улучшить взаимодействие между сотрудниками клуба, оптимизировать планирование тренировок, учёт спортсменов и тренеров, а также формирование отчётности. Для достижения этой цели необходимо разработать интуитивно понятный и простой дизайн, а также провести тщательное тестирование приложения, чтобы исключить возможные ошибки.

Разрабатываемое приложение будет использоваться исключительно внутри спортивного клуба «Спортсила».

## 1. Основания для разработки

Приказ по СПб ГБПОУ «Политехнический колледж городского хозяйства» № \_\_\_\_\_\_ от «\_\_» \_\_\_\_\_\_\_ 2024 года.

## 

## 2. Назначение разработки

### Веб-приложение предназначено для автоматизации следующих задач:

1. Безопасная и понятная регистрация и авторизация участников: спортсменов и тренеров.
2. Понятное управление расписанием тренировок: запись, отмена, назначение.
3. Надежное и безопасное хранение и отображение информации о спортсменах: личные данные, доступность для тренировок.
4. Надежное и безопасное хранение и отображение информации о тренерах: личные данные, назначенные спортсмены.
5. Понятное управление данными о матчах спортивного клуба.
6. Надежное и безопасное предоставление администратору полного контроля над системой.

### 

## 3. Требования к программе

### **3.1. Требования к функциональным характеристикам**

1. Система должна поддерживать безопасную и понятную регистрацию и авторизацию пользователей трех видов: спортсмены, тренеры и администратор.
2. Система должна обеспечивать возможность записи спортсменов на тренировки с указанием даты и времени, и надежного сохранения этих данных.
3. Система должна отображать четкое и понятное расписание тренировок для спортсменов и тренеров.
4. Тренеры должны иметь возможность назначать спортсменов на тренировки.
5. Администратор должен иметь возможность управлять данными спортсменов и тренеров (добавление, редактирование, удаление).
6. Администратор должен иметь возможность управлять расписанием тренировок.
7. Система должна надежно хранить и безопасно отображать информацию о матчах спортивного клуба.
8. Система должна обеспечивать удобный и интуитивно понятный пользовательский интерфейс.
9. Система должна быть безопасна и надежна для всех видов пользователей.
10. Система должна организовано и надежно хранить полученные данные.
11. Система должна четко и без ошибочно отрабатывать свои функциональные возможности.

### **3.2. Требования к надёжности**

1. Устойчивость к отказам: система должна обеспечивать непрерывную работу, оставаясь устойчивой к сбоям в работе как программного, так и аппаратного обеспечения.

2. Резервное копирование данных: система должна иметь надежные механизмы для резервного копирования информации, включая данные пользователей (спортсменов, тренеров, администраторов) и записи о тренировках, чтобы предотвратить их утрату.

3. Проверка входных данных: система обязана осуществлять проверку входных данных на корректность и целостность, что поможет предотвратить ошибки и потенциальные уязвимости, снижая риски для безопасности.

4. Время восстановления после отказа: в случае сбоя система должна восстанавливать свою работу не дольше чем за 15 минут, что обеспечит минимизацию времени простоя и доступность услуг для пользователей.

5. Мониторинг системы: Система должна иметь встроенные средства мониторинга, которые отслеживают состояние всех компонентов в режиме реального времени и оповещают администраторов о проблемах.

6. Необходимость журналирования: Все ключевые события, включая ошибки и системные сбои, должны записываться в журнал для последующего анализа и улучшения функциональности системы.

7. Автоматическое обновление: Система должна поддерживать автоматические обновления для обеспечения актуальности программного обеспечения и устранения известных уязвимостей.

8. Предотвращение конфликта данных: Система должна иметь механизмы, препятствующие конфликту данных, особенно при одновременном доступе нескольких пользователей к одной и той же информации (например, при записи на тренировки).

9. Тестирование на отказоустойчивость: Регулярные тестирования на отказоустойчивость должны проводиться, чтобы убедиться в способности системы нормально функционировать при нагрузках и в условиях исключительных ситуаций.

10. Обратная связь от пользователей: Система должна поддерживать механизм обратной связи, позволяя пользователям сообщать о проблемах и сбоях для быстрой их диагностики и устранения.

**3.3. Условия эксплуатации**

1. Требования к серверу: Система должна работать на сервере с поддержкой PHP 8.1 и MySQL 8.0 для обеспечения совместимости и производительности.

2. Доступность через веб-браузер: Система должна быть доступна через любой современный веб-браузер, включая Google Chrome, Yandex Browser, Mozilla Firefox, Safari и Microsoft Edge, без необходимости установки дополнительных плагинов.

3. Температура окружающего воздуха: Рабочая среда должна поддерживать температуру окружающего воздуха в пределах от +10°C до +35°C для оптимальной работы серверного оборудования и устройств пользователей.

4. Относительная влажность: Условия эксплуатации должны обеспечивать относительную влажность от 20% до 80% для предотвращения повреждений электронных компонентов.

5. Квалифицированное обслуживание: Обслуживание системы должно осуществляться квалифицированным специалистом, знакомым с технологией и функционалом системы.

6. Электропитание: Система должна иметь бесперебойное питание (ИБП) для защиты от перебоев в электроснабжении и защиты данных от потери.

7. Безопасность сети: Система должна эксплуатироваться в защищенной сети с установленными средствами защиты (например, фаерволы и антивирусное программное обеспечение) для предотвращения внешних атак.

8. Регулярные обновления программного обеспечения: Обновления операционной системы и программного обеспечения должны проводиться регулярно для поддержания безопасности и производительности.

9. Доступность вне офиса: Система должна поддерживать возможность доступа через защищенные каналы для работы удаленных пользователей.

10. Соответствие стандартам: Система и сервер должны соответствовать действующим стандартам и нормативам (например, ISO/IEC 27001, ГОСТ Р ИСО/МЭК 27001-2021) по безопасности информации и эксплуатации.

**3.4. Требования к составу и параметрам технических средств**

1. Веб-сервер: Необходимо использовать сервер Hestia, обеспечивающий надежное хостинг-окружение с поддержкой PHP 8.1 для выполнения веб-приложений.

2. Сервер базы данных: Сервер MySQL версии 8.0 должен быть настроен для обеспечения эффективного хранения и обработки данных, с возможностью масштабирования и резервного копирования.

3. Клиентские устройства: Современные компьютеры и мобильные устройства с доступом в интернет и современными веб-браузерами (такими как Google Chrome, Yandex Browser, Mozilla Firefox, Safari, Microsoft Edge) для оптимального взаимодействия с веб-приложением.

4. Безопасность от уязвимостей: Система должна быть защищена от SQL-инъекций и других уязвимостей, включая внедрение кросс-сайтовых скриптов (XSS). Для этого необходимо использовать безопасные методики программирования и регулярные аудиты кода.

5. Шифрование паролей: Пароли пользователей должны храниться в зашифрованном виде с использованием алгоритма password\_hash с солью и bcrypt для защиты от несанкционированного доступа и утечек данных.

6. Адаптивный дизайн: Интерфейс системы должен быть разработан с использованием адаптивного дизайна (responsive design), чтобы обеспечить полноценное использование на различных устройствах (компьютерах, планшетах, мобильных телефонах), а также с возможностью изменения размера окна браузера.

**3.5. Требования к информационной и программной совместимости**

1. Совместимость с веб-браузерами: Система должна быть совместима с популярными веб-браузерами, такими как Chrome, Firefox, Safari, Edge и Yandex, для обеспечения широкого доступа и удобства пользователей.

2. Использование технологий: Система должна использовать PHP версии 8.0 и MySQL версии 8.0 для обеспечения современных возможностей разработки и управления данными, что гарантирует высокую производительность и безопасность. Система должна быть защищена от SQL-инъекций и других уязвимостей.

### 3. Адаптивный дизайн: Интерфейс системы должен быть разработан с использованием адаптивного дизайна (responsive design), чтобы обеспечить полноценное использование на различных устройствах (компьютерах, планшетах, мобильных телефонах), а также с возможностью изменения размера окна браузера.

### 4. Удобство пользовательского интерфейса: Дизайн пользовательского интерфейса должен быть интуитивно понятным и простым, обеспечивая легкий доступ к основным функциональным возможностям системы, а также предоставляя ясные инструкции и подсказки для пользователей.

### 5. Доступность и поддержка: Система должна соответствовать стандартам доступности (например, WCAG - Web Content Accessibility Guidelines), чтобы обеспечить возможность использования для людей с ограниченными возможностями.

6. Многоязычность: Система должна поддерживать несколько языков для интерфейса, обеспечивая возможность выбора языка пользователями в зависимости от их предпочтений.

### 7. Документация и обучение: Необходимо предоставить четкую и доступную документацию по использованию системы, а также проводить обучение пользователей для повышения понимания всех доступных функций и возможностей.

### 

## 4. Требования к программной документации

Программная документация должна соответствовать стандартам ЕСПД и включать:

* 1. Техническое задание
  2. Описание программы
  3. Руководство пользователя
  4. Программа и методика испытаний

## 5. Стадии и этапы разработки

**5.1. Стадия разработки «Технический проект»**

* Разработка технико-экономического обоснования;
* Разработка структуры и определение формы представления входных и выходных данных;
* Разработка алгоритма решения задачи;
* Разработка структуры программы;
* Окончательное определение конфигурации технических средств;
* Разработка пояснительной записки;
* Согласование и утверждение технического проекта.

**5.2. Стадия разработки «Рабочий проект»**

* Программирование и отладка программы;
* Разработка программных документов в соответствии с требованиями ГОСТ 19.101 – 77;
* Разработка и согласование программы и методики испытаний;
* Проведение предварительных приемо-сдаточных испытаний;
* Корректировка программы и программной документации по результатам испытаний.

**5.3. Стадия разработки «Внедрение»**

* Подготовка и передача программы и программной документации для сопровождения;
* Оформление и утверждение акта о передаче программы на сопровождение.

## 6. Мероприятия по информационной безопасности

# 1. Защита от SQL-инъекций: Использование подготовленных запросов для предотвращения несанкционированного доступа к базе данных и защиты данных пользователей. Регулярное тестирование на уязвимости.

# 2. Защита от межсайтовых скриптов (XSS): Экранирование вывода пользовательских данных и использование Content Security Policy (CSP) для минимизации уязвимостей, связанных с внедрением вредоносного кода.

# 3. Хеширование паролей: Использование алгоритма bcrypt или аналогичного для безопасного хеширования паролей пользователей с добавлением соли, что делает их практически не поддающимися восстановлению.

# 4. Регулярное обновление программного обеспечения: Программное обеспечение, включая библиотеки, плагины и базы данных, должно регулярно обновляться для устранения известных уязвимостей и снижения рисков.

# 5. Внедрение системы контроля доступа на основе ролей (RBAC): Определение и применение ролей для пользователей системы, что позволяет ограничить доступ к данным и функциям, основываясь на их ролях и обязанностях.

# 6. Мониторинг и аудит безопасности: Установление системы мониторинга и регулярного аудита безопасности, что позволяет выявлять и реагировать на нарушения безопасности и подозрительные активности.

# 7. Шифрование данных: Шифрование чувствительных данных как в состоянии покоя, так и при передаче, для защиты информации от несанкционированного доступа при хранении и передаче через сеть.

# 8. Защита каналов связи: Использование протоколов шифрования (например, SSL/TLS) для защиты данных, передаваемых между клиентом и сервером, что предотвращает перехват информации.

# 9. Обучение сотрудников: Проведение регулярных тренингов и обучающих сессий для сотрудников по вопросам информационной безопасности, включая выявление фишинговых атак и правильное обращение с конфиденциальной информацией.

# 10. План защиты от инцидентов: Разработка и внедрение плана реагирования на инциденты безопасности, который включает в себя процедуры для выявления, анализа и восстановления после инцидентов.

## 7. Порядок контроля и приема

Контроль и прием программы будут осуществляться путем тестирования функциональности и безопасности приложения. Тестирование включает в себя unit-тесты, интеграционное тестирование, тестирование безопасности, Функциональное тестирование, нефункциональное тестирование и тестирование пользовательского интерфейса (UI).

## 8. Перечень графических материалов

1. Диаграмма компонентов системы.
2. Диаграмма последовательности.
3. ER-диаграмма базы данных.
4. Макеты основных экранов (прототипы интерфейса).