# СЦЕНАРИЙ ЗАЩИТЫ

sharix-platform-user

1. Добрый день уважаемая комиссия! Меня зовут Маркусь Евгений Олегович. Представляю вашему вниманию дипломную работу на тему «Разработка веб-приложения для сервиса «Друг-спортсмен» компании ООО «ШЭРИКС»». Компания «ШЭРИКС» занимается разработкой сервисов. Они берут стандартный шаблон «ShariX Open» и адаптируют его к задаче.

2. Проблема компании,состоит в том, что их сервис Друг-спортсмен построен на старой версии «ShariX Open» и его необходимо полность переписать. Целью является разработка веб-интерфейса системного администратора и обработчиков, что позволит компании увеличить клиентскую базу, повысить лояльность клиентов, обеспечить работоспособность мобильного приложения, обеспечить возможность администрирования посредством веб-приложения.

3. Для достижения цели были сформулированы задачи, которые представлены на слайде.

4. Добраться до админ-панели действующих аналогов не представляется возможным, так как модель функционирования существующих аналогов это не предусматривает, поэтому в данной дипломной работе рассматривались готовые админ-панели фреймворков, на основе которых можно было бы построить данное решение.

На слайде приведены рассмотренные решения. Они не подходят по лицензии, а также по причине того, что не являются специфичным решением для данной задачи (нет специфики для тренировок)

5. Было проведено исследование потребностей компании, основная потребность – это автоматизация бизнес процессов.

6. По итогу проведения анализа аналогов оказалось, что аналоги платные, с закрытым исходным кодом и не подходят по стеку.

7. Спроектированы структуры данных, необходимые для обеспечения работоспособности веб-сервиса, такие как: спортсмен, события, виды спорта, предпочитаемые виды спорта, заявка на события, место проведения спортивного мероприятия и другие.

8. Использованные инструменты разработки: OpenLDAP, HTML, CSS, JS, Bootstrap 5, Python, ECMAScript, Bash, Django, PostgreSQL, VSCode, Vim, Graphviz, jabber, converse.

9-10. Разработаны: модели, шаблоны, представления, формы, таблицы, html-шаблоны для визуального представления веб части приложения. Также было проведена доработка backend части ShariX Open, чтобы реализация приложения была более полноценной. Не всё, что представлено на слайде было сделано, поскольку по сценариям работы сервиса - это не предусмотрено.

11. Разработка интеграция ShariX Open с LDAP

12. Созданы обработчики для автоматизации обработки заявок.

13. Выполнено тестирование сервиса с помощью программы postman.

14. Было создано руководства на корпоративном вики портале, включая установку сервиса, настройку обработчиков, ДОБАВИТЬ.

15. На слайде представлены затраты функционирования без использования информационной системы.

16. Экономическая часть приведена для сервиса в целом, потому что невозможно достоверно установить экономическую эффективность части общей системы, так как без разработки система не будет функционировать.

Расчет подразумевает сравнение функционирования сервиса без ИС и с ИС. Модель без ИС, конечно, в современном мире нецелесообразна и несет в себе смысл иллюстрации для расчета эффекта от автоматизации процессов.

Полученные значения стоимости разработки и функционирования сервиса соответствуют стоимости разработки и функционирования типовых решений компании.

17. Затраты функционирования с использованием информационной системы.

18. Представлен срок окупаемости информационной системы в виде графика. Красным выделены затраты с учётом разработки и внедрения ИС, синим траты без разработки и внедрения ИС. Начиная с 10 месяца наблюдается снижение затрат, окуп ИС начинается с 13 месяца

19. Рабочий сервис доступен по адресу testopen.sharix-app.org.

20. Заключение.

# Вопросы и ответы

№1 Сможешь про взаимодействие в команде разработчиков, про гит, про вики?

В своей работе я активно использую распределенную систему контроля версий Git для организации совместной разработки. Основные практики включают работу с ветками, где каждая новая функциональность или исправление реализуются в отдельной ветке. Для интеграции изменений в основную ветку я применяю команду git merge, которая объединяет коммиты из функциональной ветки в целевую, создавая коммит слияния и сохраняя историю изменений. При возникновении конфликтов слияния я использую команды git log и инструменты для анализа изменений, чтобы понять причины конфликтов и корректно их разрешить. Также регулярно применяю git pull для синхронизации локального репозитория с удалённым, что обеспечивает актуальность кода и минимизирует риски конфликтов.

Для документирования и стандартизации процессов в проекте я веду внутреннюю вики, где создаю и поддерживаю подробные инструкции по установке и запуску проекта, а также руководства по конфигурации окружения. В этих статьях описываются необходимые зависимости, шаги сборки, параметры запуска и рекомендации по устранению типичных ошибок.

## №2 Какая СУБД выбрана для реализации БД и почему?

Для реализации базы данных выбрана PostgreSQL — мощная и надежная объектно-реляционная СУБД с открытым исходным кодом, которая поддерживает сложные запросы, транзакции и масштабируемость, обеспечивает высокую производительность и отказоустойчивость, совместима с разными языками программирования и не требует лицензий, что делает её универсальным и экономичным решением для различных проектов.