

**Техническое задание на разработку нового сайта**

**для Общественного фонда «SOS Детские деревни Кыргызстана»**

# 1. Цель разработки

Разработка современного, функционального и инклюзивного веб-сайта под ключ, с учетом актуальных требований фандрайзинга, цифровой безопасности, UX/UI и доступности.
Новый сайт должен:
- Повысить узнаваемость бренда и прозрачность деятельности организации.
- Привлечь ежемесячные/разовые денежные пожертвования от частных лиц и компаний.
- Обеспечить удобную навигацию для всех категорий пользователей, включая людей с ограниченными возможностями.

- Иметь яркий и интерактивный дизайн.
- Служить платформой для проведения фандрайзинговых кампаний
- Быть интегрированным с CRM, платежными системами, аналитикой и соцсетями.

# 2. Основной функционал

2.1. Информирование населения об услугах и деятельности фонда, включая публикацию материалов в форматах PDF, DOCX, PPT, Excel.
2.2. Поддержка публикации видео с YouTube, новостей, отчетов, историй успеха и обратной связи.
2.3. Мультиязычность (RU/KG/EN) — значок глобуса вместо флагов.

2.4. Поддержка слабовидящих: переключение на черно-белый режим, доступность ALT-текстов, навигация с клавиатуры.
2.5. Адаптивность (мобильные телефоны, планшеты, ПК).
2.6. Защищённый HTTPS, капча (v2 или v3), Web Application Firewall.
2.7. Интеграция с:

* CRM Salesforce (включая неполные формы);
* Платежными системами (платежная система будет уточнена позднее и другие, возможность добавлять новые платежные системы в будущем);
* Google Analytics и Яндекс.Метрика;
* Социальными сетями (встроенные ленты);
* Mailchimp (подписка на email - рассылку).

**Дополнительно:**

1. Возможность временной приостановки работы сайта – включение режима обслуживания (отображении одной информативной страницы.
2. Интеграция на сайт аналитических средств (н. Google аналитика).
3. Возможность вставки иконок в текст:

|  |
| --- |
| Рис. Пример вставки иконок |
| C:\Users\AdisovaS\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\702966C5.tmp |

# 3. Модуль пожертвований

3.1. Полная и мини-форма доната с многошаговым процессом (выбор суммы, периодичности, данных и способа оплаты).
3.2. Без регистрации, с проверкой обязательных полей.
3.3. Поддержка всех популярных местных и зарубежных способов оплаты.

3.4. Интеграция модуля платежной системы на сайт.
3.5. Страница благодарности и сообщения при неудачной транзакции.
3.6. Экспорт всех данных в CRM и/или Excel, CSV.
3.7. Отдельная страница «Помочь сейчас» с несколькими вкладками.
3.8. Интеграция с системой подписок (ежемесячно, ежеквартально и т.д.).
3.9. Уведомления при сбоях, группировка доноров, защита от ботов.

3.10. Всплывающее окно «Помочь сейчас»

3.11. Настройка доступа к модулю пожертвований для определенных пользователей через систему управления правами. У других пользователей данный раздел отображаться не должен.
3.11. Предоставить возможность просмотра списка всех пожертвований с сортировкой по статусу, дате начала, сумме и периодичности.

* 1. Должен вестись учет всех типов событий пожертвований**. Должны регистрироваться все события**.
	2. Реализовать уведомления (1-2 человека) о сбоях в рекуррентных платежах. С возможностью самостоятельной настройки получателей.
	3. Включить возможность выгрузки данных о пожертвованиях в формате CSV и Excel.
	4. Возможность временно приостановить платежи - редирект на страницу заглушку.
	5. Интеграция с “Salesforce”. Кнопка отправить в “Salesforce” информации с пожертвования.
	6. Автоматическая отправка email после каждого успешного/ рекуррентного платежа (для обоих сторон).
	7. Реализовать защиту от ботов (капчу v2 или v3) при проведении пожертвований (платежей). Чтобы ввод капчи был обязательным при осуществлении операций.
	8. Отмена, приостановка рекуррентных платежей.
	9. Три статуса платежей: незавершенные «Unfinished» (ввели данные, но не завершили процесс), «Fail» (ввели данные, но либо было недостаточно средств, либо что-то случилось) и успешные «Success».
	10. Генерация уникального кода для каждого пожертвования (12 символов, буквы и цифры, латиница).
	11. Хранение данных о пожертвованиях в зашифрованном виде.

**Пример реализации модуля:**

|  |
| --- |
| Рис. Модуль – «Пожертвования»  |
| C:\Users\AdisovaS\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\F69B4560.tmp  |
| Рис. Детализация пожертвования  |
| C:\Users\AdisovaS\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\6688B26E.tmp  |
| Настройка экспорта.   |
| C:\Users\AdisovaS\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\BA86B8AC.tmp  |
| Рис. Сотрировка в моделе «Порежтвования»  |
|  C:\Users\AdisovaS\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\67071F9A.tmp  |
| Рис. Группировка пожертвований  |
| C:\Users\AdisovaS\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\52D89AB8.tmp  |

# 4. Обратная связь

* 1. На сайте должна быть реализована система для получения и обработки обратной связи от пользователей.
	2. Отправка уведомлений на заданную почту. Возможность настройки куда отправлять.
	3. Наличие защиты от ботов (капча v2 или v3).

# 5. Специальные лендинги

5.1. Возможность создавать специальные лендинги (к примеру: Орозо Айт, Новый год, Марафон (с калькулятором км и суммы), Подготовка к школе, Международный день детей и др.)
5.2. Каждый лендинг содержит:
- Заголовок, изображение, мини-форму, объяснительный текст, визуальные блоки, форму оплаты, кнопку помощи.

# 7. Эргономика и дизайн

* 1. Интерфейс должен быть выполнен в корпоративных цветах (согласно брендбуку).
	2. Обеспечить удобную навигацию для всех категорий пользователей, включая людей с ограниченными возможностями (к примеру, дислексия, слабовидящие).
	3. Интерфейс веб-сайта должен быть рассчитан на использование:
* персональных рабочих станций,
* переносных компьютеров (ноутбуков) или планшетных компьютеров,
* мобильных телефонов,
	1. Управление веб-сайтом должно осуществляться с помощью набора экранных меню, кнопок, значков и прочих графических элементов.
	2. Клавиатурный режим ввода должен использоваться при заполнении и/или редактировании текстовых и числовых полей экранных форм.
	3. Надписи экранных форм, сообщения, выдаваемые пользователю, должны быть выполнены на кыргызском, русском и английском языках, в зависимости от выбранной языковой версии.
	4. Исключения могут составлять только системные сообщения, не подлежащие переводу. Цифровая информация должна быть отражена арабскими цифрами.
	5. Web-интерфейс должен проектироваться с учетом требований унификации:
* интерфейс взаимодействия пользователей с веб-сайтом должен быть интуитивным, интерактивным и выдержан в корпоративных цветах (брендбук есть);
* отображение разделов, страниц, вкладок и списков должно отражать логику работы с этими элементами в процессе деятельности;
* неиспользуемые элементы не должны отображаться;
* необходима возможность разграничения прав и ограничения возможностей пользователей и администраторов в административной части;
* система должна быть хорошо защищенной и устойчивой к взломам и атакам;
* веб-сайт должен быстро загружаться на всех устройствах.
* интерфейс веб-сайта должен быть направлен на минимизацию производимых пользователем операций при переходе из одного раздела в другой, из одной страницы сайта в другой, от одной операции к другой (то есть должен быть эргономичным).

# Технические требования

* 1. Сайт должен работать на безопасном протоколе на https.
	2. Отсуствие числовых ID в URL (лучше UUID или хеши).
	3. Капча должна быть настроена надлежащим образом чтобы обеспечивать защиту от автоматических ботов.
	4. В качестве движка сайта не должны использоваться широко популярные CMS (такие как: рапример, WordPress, Joomla и т.д.). Приветствуется создание сайта на фреймворке.
	5. Не рекомендуется использовать Docker.
	6. Реализовать авторизацию к серверу только по SSH ключу.
	7. Не должно быть чего-то скрытого или не описано (например, зашитых учетных записей и паролей).
	8. Разработчик обязан предоставить детализированное техническое задание (ТЗ), основанное на настоящем брифе.
	9. **При заключении договора, к нему должно быть приложено согласованное с нами детализированное техническое задание.**
	10. Документ должен описывать весь цикл разработки сайта: от проектирования и выбора технологий до тестирования, внедрения и поддержки.
	11. В ТЗ необходимо отразить архитектуру решения, функциональные и нефункциональные требования, этапы разработки, подход к безопасности, интеграции с внешними системами и план сопровождения.
	12. Документ подлежит согласованию с заказчиком до начала основной фазы работ.
	13. Тех. поддержка по сайту в течение 3 лет, включая устранение ошибок, обновление и добавление компонентов, резервное копирование.
	14. Техническая поддержка по сайту должна включать не только устранение ошибок в работе сайта, но и обновление ПО на котором он работает (плагины, серверное ПО, фреймаорк и т.д.).
	15. Разработчики должны быть готовы не только к выполнению стандартных задач по технической поддержке, обновлению и сопровождению сайта, но и к реализации новых функциональных решений и улучшений по запросу заказчика.
	16. Это включает в себя разработку и внедрение инновационных функций, адаптацию под изменяющиеся потребности проекта, а также участие в создании нестандартных решений, направленных на развитие и модернизацию веб-сайта.
	17. Детали разработки новых функциональных решений и улучшений будут оговариваются и оплачиваться отдельно.
	18. Поддержка пользователей (09:00-18:00, пн-пт), реакция на запросы до 30 минут.
	19. Подробная ежемесячная отчетность.
	20. Разграничение приоритетов заявок (консультации, ошибки, сбои).
	21. Передача всех доступов, лицензионных ключей, исходников и инструкций (если таковые будут иметься).
	22. Обеспечение резервного копирования веб-сайта (не менее 3-4 копии).
	23. Отсутствие не протестированных или уязвимых плагинов на момент публикации веб-сайта.
	24. Необходимо обеспечить обязательное ведение журнала событий с указанием каждого события в Системе.
	25. Доступ к журналам должен настраиваться только для определенных пользователей.
	26. Длительность хранения журналов должна настраиваться, но быть не менее 4 месяцев.
	27. Необходимо обеспечить недоступность изменения записей журнала для всех пользователей, в том числе и учетной записи администратора.
	28. Логирование должно включать в себя запись всех событий административного характера:
* Создание/удаление страниц, публикация материала,
* Вход пользователей, удаление/создание пользователей;
* Запись всех событий, относящиеся к загрузке/выгрузке данных, в том числе и обрабатываемые в автоматическом режиме;
* Запись сведений о произошедших ошибках в системе или процессе загрузки/выгрузки данных;
* Запись все событий, относящиеся к изменению параметров Системы.

8.27 Sitemap.xml должен содержать только страницы, предназначенные для индексирования. В файл не должны попадать административные разделы, страницы с персональными данными, формы пожертвований, API и другие приватные URL. При необходимости sitemap может быть разделен на несколько частей с раздельным управлением доступа. Файл должен быть валиден и протестирован с помощью Search Console.

8.28 Хостинг: помощь в выборе или интеграция с текущим.

# Приемка работ:

Приёмка осуществляется поэтапно: после завершения каждого ключевого этапа (дизайн, функционал, интеграции и т.д.) подрядчик предоставляет доступ к тестовой версии сайта. Заказчик проверяет результат и предоставляет обратную связь в течение 3–5 рабочих дней.

Финальная приёмка проводится после устранения всех замечаний и завершения тестирования. Разработчик предоставляет итоговую версию сайта, исходный код, инструкции и документацию. Работа считается принятой, если сайт соответствует требованиям ТЗ, работает корректно на всех устройствах и соответствует критериям безопасности и доступности.

По завершении работ подписывается акт приёмки, после чего начинается этап технической поддержки.

#  Тестирование

Тестирование должно производиться по следующим параметрам:

**10.1** **Функциональное тестирование**

На данном этапе выполняется проверка, что весь заявленный функционал работает корректно, а именно:

* Навигация по сайту (меню, кнопки, ссылки)
* Формы (отправка, валидация, обработка ошибок)
* Регистрация / вход / восстановление пароля
* Поиск (если есть)
* Корзина / заказ (если интернет-магазин)
* Интеграции с платежами, внешними API, CRM и т.д.
* Мультиязычность (если предусмотрена)
* Элементы сайта должны быть ровными и аккуратными, футер «прибитым».

**10.2** **Кроссбраузерное и кроссплатформенное тестирование.**

На этом этапе выполняется проверка отображения и работы сайта в разных браузерах и на разных устройствах:

* Браузеры: Chrome, Firefox, Safari, Edge (актуальные версии).
* Устройства: десктопы, планшеты, смартфоны (iOS/Android).
* Разрешения экрана: от 320px до 1920px+.
	1. **Тестирование UI/UX**

На этом этапе выполняется проверка соответствия дизайну и удобству:

* Сайт соответствует дизайн-макетам (например, из Figma).
* Адаптивная верстка не «ломается».
* Все шрифты, отступы, цвета — как в ТЗ/дизайне.
* Удобство пользования интерфейсом.
	1. **Производительность (Performance Testing)**.

На этом этапе оценивается скорость загрузки и оптимизация:

* Время загрузки страниц (например, проверить через Google PageSpeed или Gtmetrix или Lighthouse)
* Оптимизация изображений
* Минификация CSS/JS
* Lazy loading для изображений и видео
	1. **Безопасность.**

На этом этапе обязательно выполняются проверки характерные для сайтов с формами, личными данными, платежами:

* HTTPS (SSL-сертификат установлен)
* Проверка XSS, CSRF, SQL-инъекций (можно использовать сканеры или аудит)
* Защита административной панели
* Сложные пароли, защита от брутфорса
* Тестирование корректности работы капчи
* Проверка защиты от IDOR.
* robots.txt с запрещёнными User-Agent AI-ботов, настройка оганичений доступа к чувствительным директориям.
* Безопасная реализация sitemap.xml.

**10.6 Тестирование бэкенда**

Проверка работы серверной части:

* Если етсь API, то работает ли корректно.
* Ошибки обрабатываются правильно (возвращаются коды 4xx/5xx).
* Логи ошибок доступны и пишутся в журналы.
* Резервное копирование настроено.

**10.7 Контент и SEO**

* Все тексты, изображения, мета-теги загружены корректно
* Уникальные Title, Description
* Есть robots.txt и sitemap.xml
* Настроена микроразметка (Schema.org, если нужно)

**10.8 Email-уведомления**

Проверка отправки писем:

* Письма приходят
* Не попадают в спам
* Правильный адрес отправителя, корректное оформление.

**10.9 Регрессионное тестирование (если были доработки)**

* Проверка, что после новых изменений ничего не сломалось в уже работающем функционале.

**10.10 Какой доступ нам нужен:**

* Доступы: SFTP, админка, база данных, панель хостинга.
* Инструкции по запуску и поддержке (если сайт сложный).
* Резервная копия на момент сдачи.
* Отчет о тестировании (если тестировали сами).

**10.11 Сайт считается принятым, если:**

* Пройдено тестирование по всем пунктам
* Все баги устранены или согласованы как допустимые
* Подписан акт приёма-передачи

# Дополнительно

* 1. Требуется защита от взломов и атак, включая установку Web Application Firewall **(WAF).**
	2. Перенос существующих материалов корпоративного сайта (<https://soskyrgyzstan.kg/>) на новый.
	3. Если предполагается настройка/использование сервисов, то она должна быть сделана только с использованием корпоративной почты.

# Примерная структура сайта:

Аналогично сайту SOS Children’s Villages Armenia: <https://sos-kd.am/index.php/ru>

# 1. ****Главное меню****

* **КТО МЫ**
	+ О нас
	+ Совет попечителей
	+ Наша команда
	+ Аудиторские отчеты
* **ЧТО МЫ ДЕЛАЕМ**
	+ Альтернативный уход
	+ Превенция
	+ Адвокация
	+ Гуманитарная деятельность
	+ Психическое здоровье
* **ПОДДЕРЖИТЕ ДЕТЕЙ**
	+ Как я могу помочь детям?
	+ Кампании
* **НАШИ ДРУЗЬЯ**
	+ Партнеры
	+ Типы наших партнеров
	+ Клуб Друзей SOS
* **НОВОСТИ**
	+ Новости и объявления
* Удивительные истории
* Познакомьтесь с нашими фейсерами
* Часто задаваемые вопросы
* Сообщить о нарушении
* Вакансии
* Связаться с нами
* Соглашение о пожертвовании, соглашение об обработке персональных данных
* Политика конфиденциальности, 404, сервисная(е) страница(ы)

# Главная страница

* **Баннер с призывом к пожертвованию**
* **Краткое описание миссии фонда**
* **Статистика и достижения**
* **Отзывы партнеров**
* **Ссылки на социальные сети**
* **Кнопка "Пожертвовать сейчас"**

# Дополнительные элементы

* **Многоязычность**: Русский, Английский, Армянский
* **Интеграция с социальными сетями**: Facebook, LinkedIn, Instagram, YouTube
* **Форма пожертвования**: Возможность выбрать сумму, периодичность, способ оплаты
* **Подписка на новости и обновления**
* **Интерактивная карта с географическим охватом деятельности фонда**

|  |
| --- |
|  Рис. Пример реализации структуры и меню управления сайта |
| * **Dashboard**
* **Contacts**
* **Our team**
* **Pages**
* **FAQ’s**
* **Addresses**
* **Donations**
* **Supports**
* **Testimonials**
* **Campaigns**
* **Success stories**
* **Faces**
* **Locations**
* **Audits**
* **Careers**
* **Subscribers**
* **Events и т.д.**
 | C:\Users\AdisovaS\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\DCEA9B5.tmp |

# Примеры реализации

Пример реализации модуля «Компании»:

|  |
| --- |
|  Рис. Модуль «Компании» |
| C:\Users\AdisovaS\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\C51C7A8B.tmp  |

**Еще примеры:**

* [**https://sos-decijasela.rs/**](https://sos-decijasela.rs/) **- интерфейс, дизайн**
* [**https://sos-kd.am/**](https://sos-kd.am/)**- предпочитаемый вариант (структура сайта, функционал)**

# Требование к процессу разработки веб-сайта

* 1. Использование разработчиками безопасных методов кодирования, соответствующих используемому языку программирования и среде разработки.
	2. Если разработчики сами разрабатывают какое-либо ПО, они должны убедиться, что явная проверка ошибок выполняется и документируется для всех входных данных, включая размер, тип данных и допустимые диапазоны или форматы.
	3. Разработчики должны убедиться, что версия всего используемого программного обеспечения, полученного за пределами организации, все еще поддерживается его разработчиком или соответствующим образом укреплена в соответствии с рекомендациями разработчика по безопасности.
	4. Должны использоваться только актуальные и проверенные компоненты сторонних производителей для программного обеспечения, разработанного организацией.
	5. Использование только стандартизованных, принятых в настоящее время и прошедших всестороннюю проверку алгоритмов шифрования.
	6. Необходимо, чтобы все сотрудники, занимающиеся разработкой программного обеспечения, прошли обучение по написанию безопасного кода для своей конкретной среды разработки и обязанностей.
	7. Должны применяться инструменты статического и динамического анализа для проверки соблюдения практик безопасного кодирования программного обеспечения, разработанного внутри компании.
	8. Должен быть установлен процесс приема и рассмотрения сообщений об уязвимостях программного обеспечения, включая предоставление средств для связи внешних организаций с вашей группой безопасности.
	9. Должны поддерживаться отдельные среды для производственных и непроизводственных систем. Разработчики не должны иметь неконтролируемый доступ к производственным средам.
	10. Для приложений, которые зависят от базы данных, должны быть использованы стандартные шаблоны конфигурации для усиления. Все системы, являющиеся частью критически важных бизнес-процессов, также должны быть протестированы