Разработка сайта и приложения QuickBaa

Цель: разработка приложения и сайта QuickBaa.

Что предстоит сделать:

* Разработать дизайн сайта
* Реализовать функционал сайта
* Разработать дизайн приложения
* Разработать и реализовать протокол передачи данных по локальной сети разработанный и оптимизированный специально для передачи данных по локальной сети
* Разработка Android-приложения
* Разработка Windows-приложения
* Тестирование приложения на разных платформах, проверка технической реализации

Сайт:

Является платформой для временного хранения и обмена файлов

Страницы:

- главная: функция загрузки файлов на сайт с помощью drag&drop или выбором файла без регистрации или авторизации с последующей генерацией ссылки и QR-кода.

- "о наших продуктах":

Является страницей загрузки приложения QuickBaa для разных платформ

- "админ-панель":

страница только для администраторов с историей загрузки всех пользователей и настройками

Каждый файл загруженный на сайт должен хранится на сервере в течении двух часов. Должен иметься выбор темной и светлой темы с последующим сохранением в браузере пользователя.

Реализация сайта должна проводится на CSS, HTML, JavaScript. Серверная часть должна обеспечить бесперебойную работу сервиса при большой нагрузке пользователей

Приложение на Android и Windows:

Является сервером и клиентом реализующий протокол с помощью, которого пользователей может отправлять и получать данные между устройствами, где тоже стоит приложение посредством локальной сети

Должно иметь переключатель светлой/темной темы и выбором нескольких цветовых тем. На Android должна иметься функция включения/выключения использования динамического цвета (dynamic colors) на Android 12+

Должен поддерживаться android от Android 7 до последней на данный момент версии ОС Android

Задача приложение это обеспечить пользователя возможностью принимать и отправлять файлы, а так же сканировать локальную сеть на другие запущенные приложения. Сервер приложения должен быть всегда запущен на фоне пока пользователь находится в локальной сети чтобы обеспечить бесперебойное получение файлов от доверенных устройств. Приложение должно обеспечить поддержку различных типов форматов файлов и типов данных и удобный интерфейс для их отправки

В приложении должны быть меню:

- Выбора файла или другого типа данных, к примеру отправка ссылки, текста, или буфера обмена.

- Список доступных устройств для отправки

- История получения и отправки данных: отображает какой тип данных куда и во сколько отправили, при возможности отображая контент

- Настройки: настройки для обычных пользователей и "Расширенные". В настройках пользователь может указать отображаемое имя устройства, тему интерфейса, настройку папку для загрузок различных типов файлов. Расширенные настройки позволяют продвинутым пользователям редактировать различные параметры протокола, такие как на сильное изменение его версии, изменение размеров буферов, настройки порта сервера и редактирование других настроек протокола и сервера

- Страница передачи файлов и др. данных с помощью Bluetooth или WiFi Direct

Главная цель разработки User Experience (UX): Сократить время от желания пользователя отправить файл на конкретное устройство до самой отправки. Можно при этом использовать контекстные меню, меню “Поделится” на Android и др. без нужды открытия приложения а также возможность использования всех функций протокола предоставляя пользователю полные возможности

Приложение должно уметь само открывать ссылки в браузере, копировать текст по желанию и тд. для минимизации усилий пользователя на устройстве - получателе

Основные задачи протокола и его реализации:

Обеспечить нахождение других приложений в сети с помощью UDP Multicast и/или сканирования IP-адресов и портов на которых может запуститься сервер работающий в каждом приложении.

В каждой реализации приложения QuickBaa должен работать TCP-сервер который позволяет ему одновременно принимать данные и контактировать с другими серверами запущенные в приложениях на других устройствах.

Сервер приложения должен работать на одном из TCP портов от 7005 до 7030 включительно. Если серверу не удалось запустится на одном этих портов и пользователь не указал в настройках альтернативный порт то приложение должно считаться недееспособным на вкладке передачи данных по локальной сети. В Android среде сервер приложения должен работать только когда устройство подключено к WiFi-сети или на нем запущена точка доступа чтобы минимизировать потребление ресурсов.

Типы данных: Когда сервер отправляет данные другому устройству, до данных должен указываться их тип (не формат файла), к примеру "файл" или "ссылка" или "просто текст" и так далее. С помощью типизации данных сервер и приложение должно иметь возможность расширение поддерживаемых типов данных с помощью различных расширений.

Доверенные устройства: Доверенным устройство считается тогда когда уникальный идентификатор отправителя содержится в его списке доверенных устройств. Когда отправитель данных входит в этот список, устройство может принимать данные от отправителя, а если его там нету то отказывается принимать данные. Таким образом только доверенным устройствам можно отправлять файлы. Доверенные устройства должны добавляться пользователем с помощью кодов или QR кодов с разных экранов устройств.

Шифрование между устройствами: Каждое устройство должно иметь отдельные ключи шифрования между друг другом. Шифрование должно обеспечить защиту данных от перехвата таким образом что расшифровать данные сможет только получающая сторона.

Когда устройство находится в локальной сети, оно должно быть готово в любой момент принимать данные от доверенных устройств без предварительного соединения.

Асинхронность/мульти-поточность: Сервер в приложении должен быть готов принимать неограниченное количество одновременных загрузок или отправок

Batch (множественные) загрузки и отправки: Протокол должен быть оптимизирован для передачи множества файлов за раз

Опознавание устройств: Приложения должны иметь уникальный идентификатор устройства для ведения различных таблиц и т.д. в то же время предоставляя другим устройствам имя который может выбрать пользователь.

Обратная совместимость: Наряду с другой информацией о устройстве, приложение должно сообщать свою версию используемого протокола и реализовывать обратную совместимость.

Поддержка передачи вне локальной сети: Протокол должен иметь возможность использоваться с помощью WiFi Direct и Bluetooth

Реклама и прибыль:

Веб-сайт и приложение должно иметь функцию подписки которая позволяет пользователю:

На сайте:

Отправлять объемы данных в несколько раз больше

В Android-приложении:

Отключить рекламу, пересылать папки, возможность открывать ссылки при их получении

Также приложение на Android и веб - сайт должно иметь выделенное место в интерфейсе для отображения рекламы

Используемые инструменты и библиотеки для реализации:

Разработчики могут использовать любые библиотеки и инструменты, которые соответствуют нескольким критериям:

* Не замедляет разработку
* Разработчику не требуется дополнительная квалификация или обучение для использования этой библиотеки или инструмента
* Позволяет реализовать задуманный дизайн протокола и дизайн интерфейса
* Библиотека не является низко-производительной на целевой платформе
* Является бесплатной

Однако, основными инструментами разработки приложения на Android должны быть:

* IDE IntelliJ IDEA или Android Studio
* Языки программирования Java/Kotlin
* Библиотеки Jetpack Compose для реализации дизайна интерфейса
* Библиотеки Compose Material 3

Команды, реализующие проект:

1. Команда UI/UX дизайнеров, обеспечивающая дизайн и логику поведения интерфейса приложения и веб-сайта
2. Команда Java/Kotlin Программистов
3. Команда Программистов для реализации функционала сайта
4. Команда программистов, имеющих опыт с разработкой настольных приложений на Windows

Этапы разработки и спринты:

Спринт 1:

Длиться 2 недели

Команда 1 разрабатывает интерфейс веб-сайта и приложений. Параллельно с ними команда 3 реализует функционал веб-сайта. Команды 2 и 4 разрабатывают теоретическую реализацию протокола и изучают, выбирают инструменты для разработки

Спринт 2:

Длиться 3 недели

Команды 2 и 4 разрабатывают реализацию протокола и интерфейса на двух платформах с помощью выбранных средств и инструментов. Параллельно с разработкой проводится тестирование готовых функций, при надобности проводится пересмотр реализации и дизайна.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Наименования Заказчика | Наименование Услуги | Краткая Характеристика | Единица Измерения | Количество,  Объём | Цена за единицу | Планируемый срок  оказания | Срок оказания услуги |
| Проектная Команда | UI/UX дизайн интерфейса | Разработка дизайна и логики интерфейса приложения и веб-сайта | Спринт | 3 недели | - | Спринт 1 и Спринт 2 | 3 недели |
| Проектная Команда | Программирование Java/Kotlin | Разработка реализации протокола и интерфейса на платформе | Спринт | 3 недели | - | Спринт 1 и Спринт 2 | 3 недели |
| Проектная Команда | Функционал веб-сайта | Реализация функциональности веб-сайта | Спринт | 2 недели | - | Спринт 1 | 2 недели |
| Проектная Команда | Разработка для настольных приложений | Разработка реализации протокола и интерфейса для настольных приложений на Windows | Спринт | 3 недели | - | Спринт 1 и Спринт 2 | 3 недели |