

Техническое задание на разработку

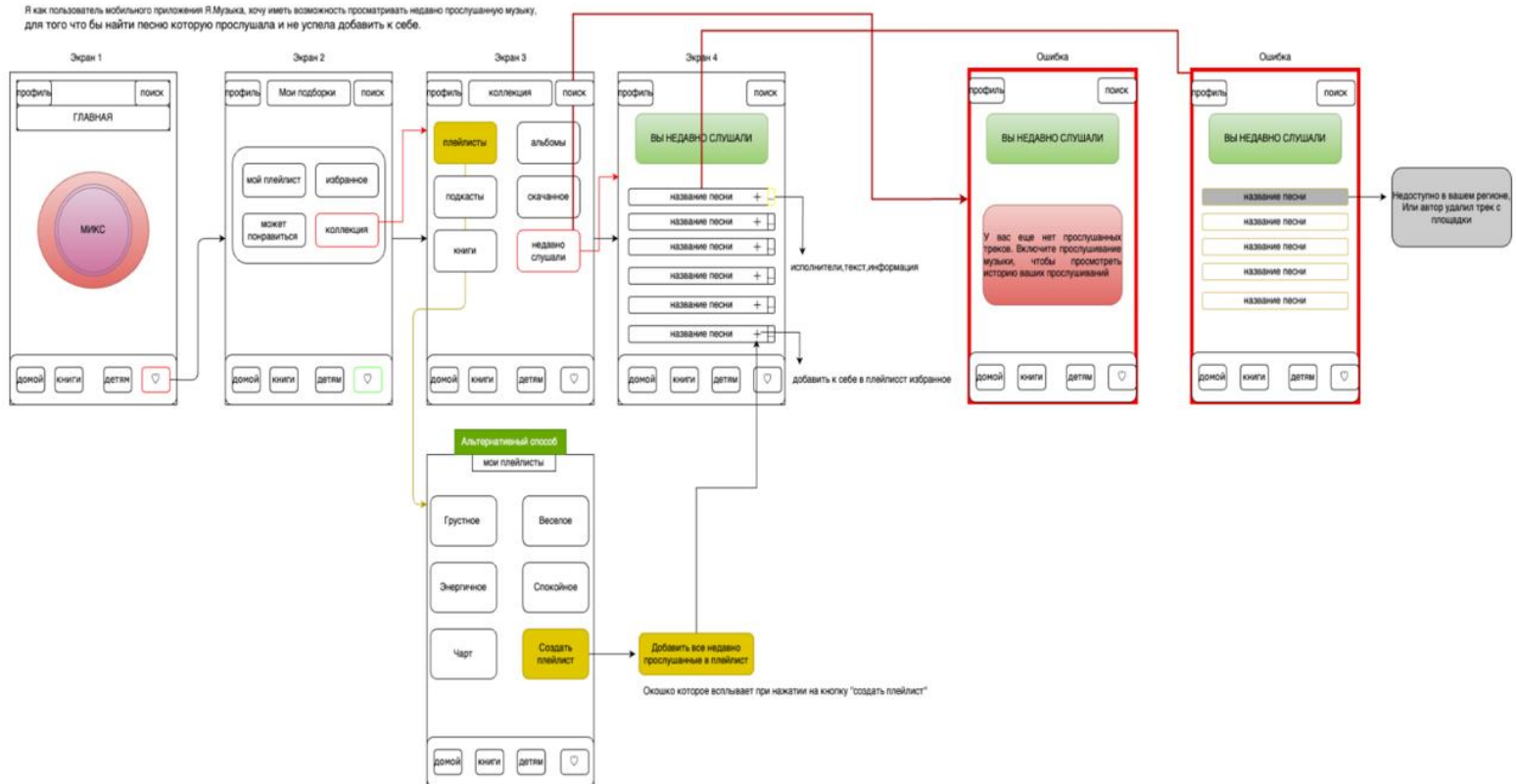
Бизнес требования:

1. **Название продукта** – приложение для прослушивания Яндекс Музыка
2. **User Story:**

Я как пользователь мобильного приложения Яндекс музыка, хочу иметь возможность найти песню для того, чтобы просматривать недавно прослушанную музыку, которая мне понравилась.

3. Макеты

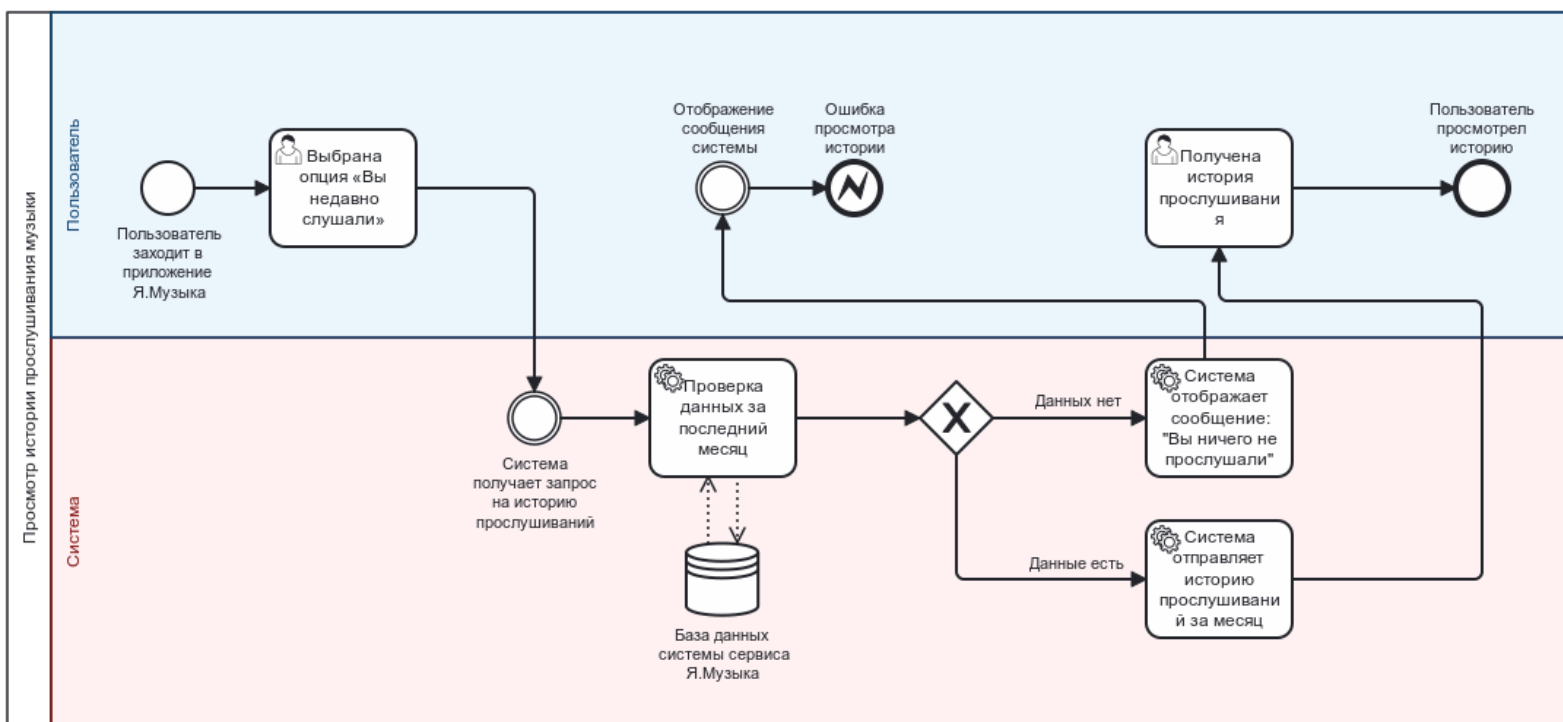
Я как пользователь мобильного приложения Яндекс Музыка, хочу иметь возможность просматривать недавно прослушанную музыку, для того что бы найти песню которую прослушала и не успела добавить к себе.



4. Use Case

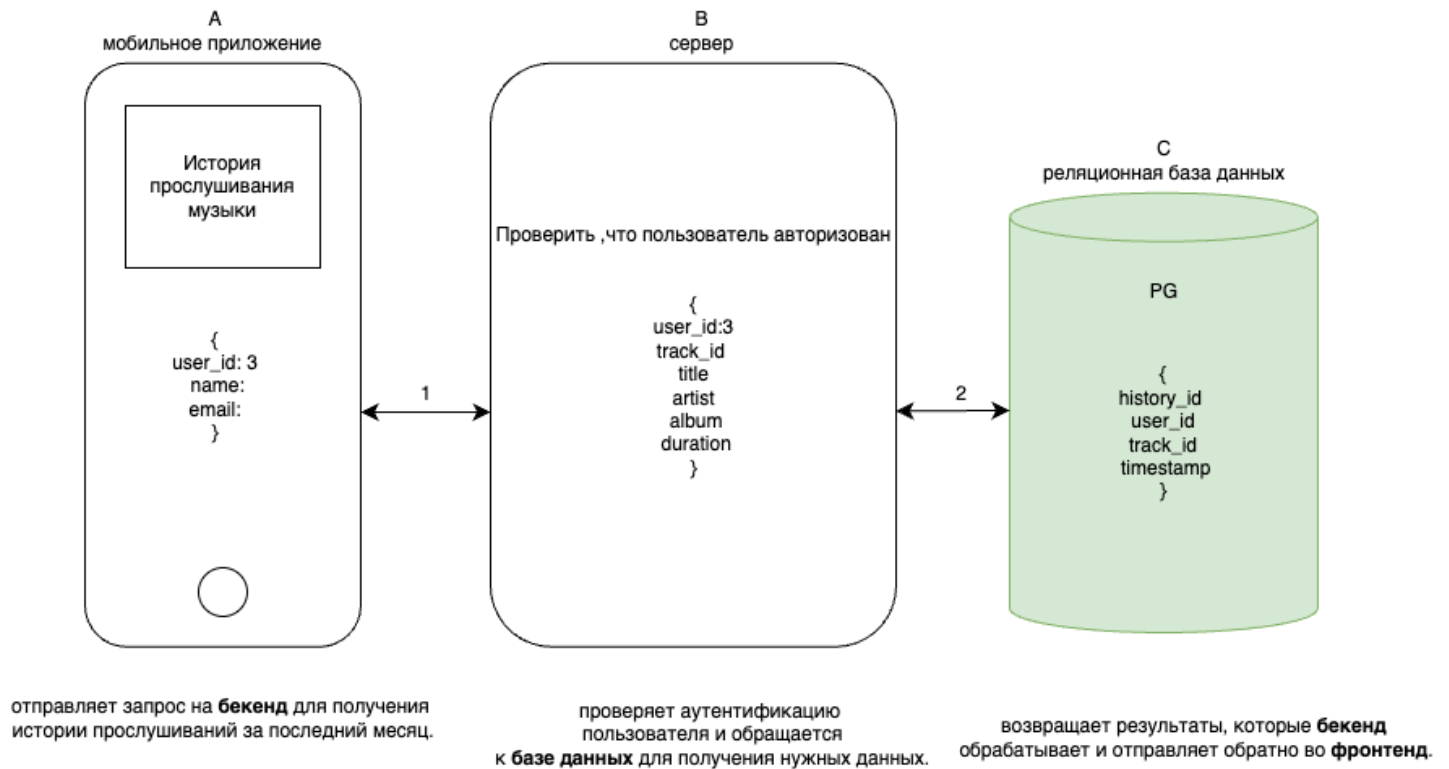
Заголовок	Просмотр истории прослушивания музыки
Актеры	Пользователь
Предусловие	Клиент авторизован в приложении Я.Музыка и у него есть доступ в интернет
Ограничения	Система сохраняет информацию за месяц, с каждым новым месяцем список сбрасывается
Триггер	Клиент заходит в приложение Я.Музыка
Основной сценарий	<ol style="list-style-type: none"> 1. Система отображает экран главной страницы с различными рекомендациями (экран 1) 2. Пользователь нажимает кнопку раздела с его коллекцией обозначенную "♥" 3. Система переносит нас на экран "мои подборки" с различными разделами (экран 2) 4. Пользователь выбирает раздел "коллекция" 5. Система отображает коллекцию пользователя с тем, что ему может понравиться (экран 3) 6. Пользователь нажимает кнопку "недавно слушали" 7. Система отображает список музыки которую слушал пользователь (экран 4) <p>Критерий успеха: пользователь смог посмотреть недавно прослушанную музыку</p>
Альтернативный сценарий	<p>6a Пользователь нажимает "создать плейлист" (экран 3)</p> <p>7a система предлагает добавить недавно прослушанные в плейлист с возможностью выбрать весь список сразу</p> <p>Пользователь выбирает добавить недавно прослушанные в плейлист</p> <p>----- переход к шагу 7 основного сценария</p>
Исключительный сценарий	<ol style="list-style-type: none"> 1. Пользователь не слушал музыку за последние 30 дней 2. Исполнитель удалил трек с площадки и его нельзя прослушать

5. BPMN Диаграмма



Функциональные требования:

1. Архитектура



1- протокол передачи данных http/https
2- драйвер JDBC для соединения

2. Модель данных



Объект Пользователь

Родительская сущность	Атрибут	Описание
User	last name	Фамилия пользователя
	first name	Имя пользователя
	email	Адрес электронной почты
	phone number	Номер телефона
	listening history	История прослушивания

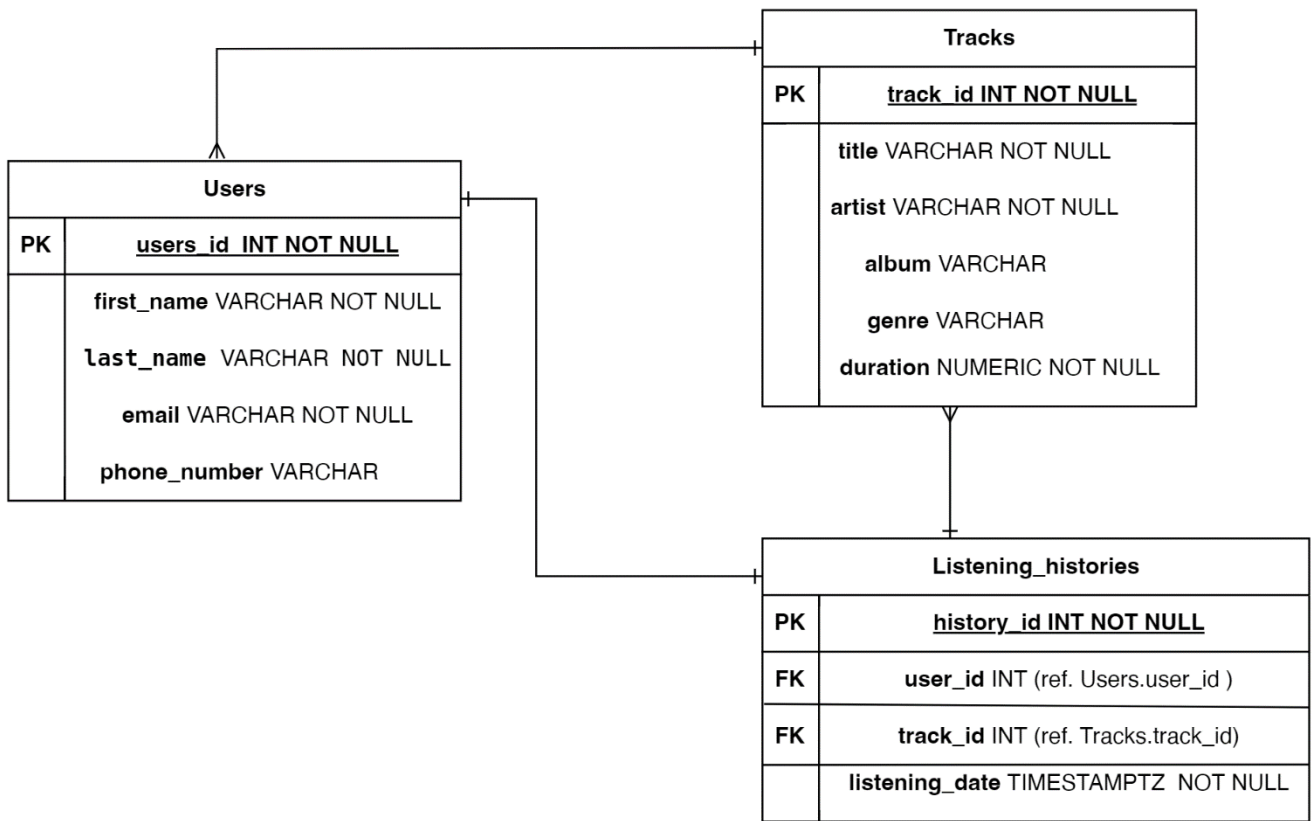
Объект история прослушивания

Дочерняя сущность	Атрибут	Описание
Listening History	track	трек который был прослушан
	listening date	Дата и время когда был прослушан конкретный трек
	User	Пользователь, который прослушал трек

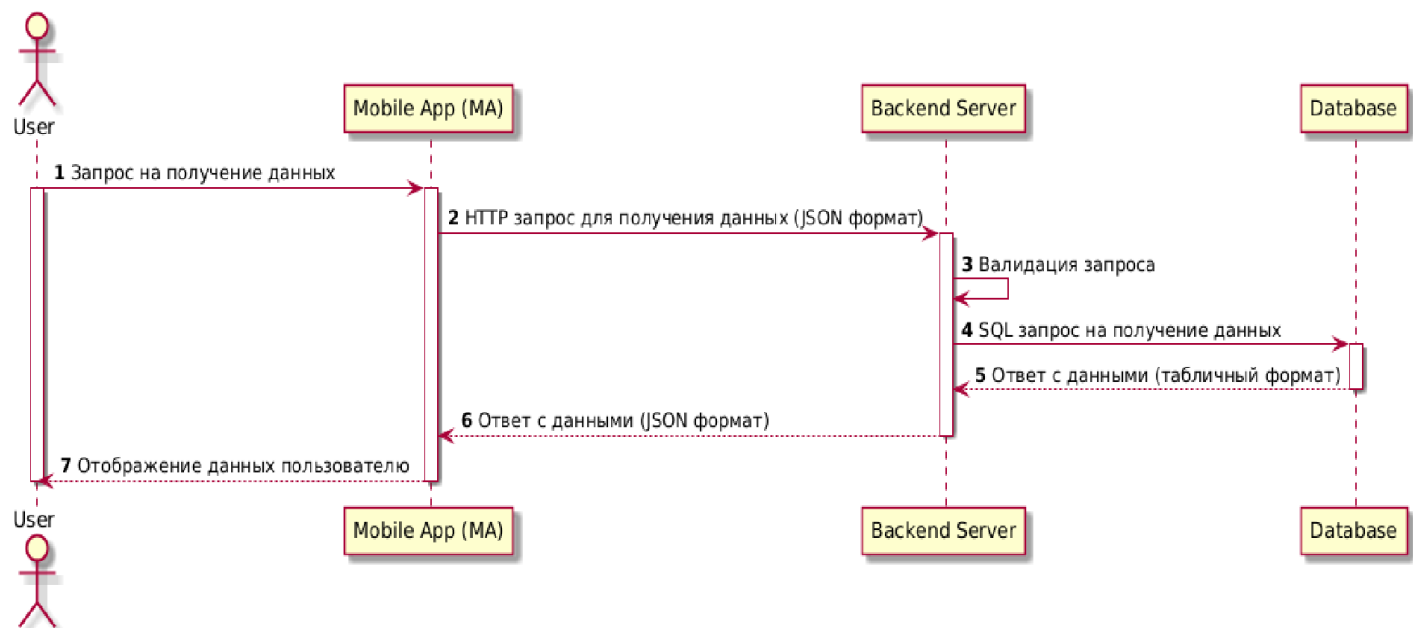
Объект трек

Дочерняя сущность	Атрибут	Описание
Track	title	Название трека
	artist	Имя или псевдоним исполнителя
	album	Название альбома, в который входит трек
	genre	Жанр трека
	duration	Продолжительность трека

3. ERD



4. Диаграмма последовательности



1	Пользователь отправляет запрос на получение данных в мобильное приложение.
2	Мобильное приложение отправляет HTTP запрос на сервер для получения данных в формате JSON.
3	Сервер выполняет валидацию полученного запроса, проверяя корректность и полноту данных.
4	Сервер отправляет SQL запрос в базу данных для получения необходимой информации.
5	База данных возвращает ответ серверу с данными в табличном формате.
6	Сервер обрабатывает данные из базы и отправляет их обратно в мобильное приложение в формате JSON.
7	Мобильное приложение отображает полученные данные пользователю, завершая процесс взаимодействия.

5. REST – табличный вид

REST. Табличный вид

GET /v1/music/history

Получение истории прослушивания музыки за последний месяц.

Request

Название параметра	Тип данных	Описание	Обязательность параметра
userId	int	Уникальный идентификатор пользователя	да

Response

Название параметра	Тип данных	Описание	Обязательность параметра
userId	int	Уникальный идентификатор пользователя	да
tracks	array of objects	Массив объектов, каждый из которых представляет один трек	да
track	object	Объект, содержащий информацию о конкретном треке	да
title	string	Название трека	да
artist	string	Имя исполнителя	да
listenDate	string	Дата и время прослушивания трека	да

6. Swagger

GET /v1/music/history Получение истории прослушивания музыки за последний месяц

Возвращает массив объектов с информацией о треках, прослушанных за последний месяц.

Parameters Try it out

Name	Description
userid <small>required</small>	Уникальный идентификатор пользователя
Integer (query)	<input type="text" value="userid"/>

Responses

Code	Description	Links
200	Успешный ответ	No links

Media type:

Controls: Accept: header

Example Value | Schema

```
{  "userId": 123,  "tracks": [    {      "title": "Imagine",      "artist": "John Lennon",      "listenDate": "2024-12-22T15:30:00Z"    }  ]}
```

POST /v1/music/history Добавление информации о прослушивании трека

Добавляет новый трек в историю пользователя.

Parameters Try it out

No parameters

Request body required

Example Value | Schema

```
{  "userId": 123,  "track": {    "title": "Imagine",    "artist": "John Lennon",    "listenDate": "2024-12-22T15:30:00Z"  }}
```

Responses

Code	Description	Links
201	Трек успешно добавлен	No links

Media type:

Controls: Accept: header

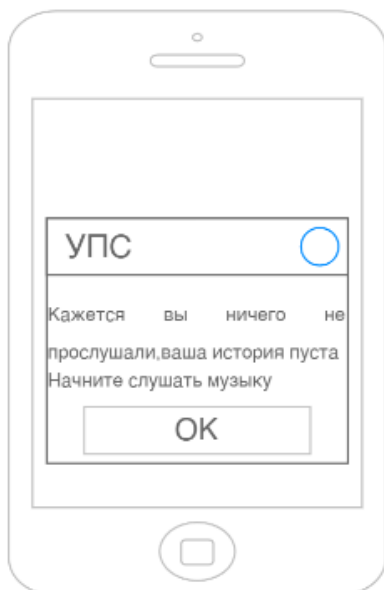
Example Value | Schema

```
{  "message": "Track successfully added."}
```

Критерии приемки и нефункциональные требования

1. Критерии приемки:

Кейс номер:1



Функциональность: Пользователь может просматривать недавно прослушанную музыку за последний месяц.
Дано: Пользователь не слушал музыку через приложение за последний месяц.
Когда: Пользователь нажимает кнопку "Недавно слушали".
Тогда: Система отображает сообщение: "Ваша история прослушиваний пуста. Начните слушать музыку!"

Кейс номер:2



Функциональность: Пользователь может просматривать недавно прослушанную музыку за последний месяц.
Дано: Пользователь находится на экране "Коллекция".
Когда: Пользователь нажимает кнопку "Недавно слушали", но происходит ошибка соединения с сервером.
Тогда: Система отображает сообщение: "Ошибка загрузки данных. Попробуйте позже."

Кейс номер:3



Функциональность: Пользователь может просматривать недавно прослушанную музыку за последний месяц.

Дано: Пользователь не авторизован в системе.

Когда: Пользователь нажимает кнопку "Недавно слушали".

Тогда: Система отображает сообщение: "Авторизуйтесь, чтобы получить доступ к вашей истории прослушиваний."

Кейс номер:5



Функциональность: Пользователь может просматривать недавно прослушанную музыку за последний месяц.

Дано: Пользователь не обновлял приложение до последней версии.

Когда: Пользователь пытается открыть раздел "Недавно слушали".

Тогда: Система отображает сообщение: "Обновите приложение для доступа к этой функции."

2. Нефункциональные требования

Требования доступности:

1. Функция просмотра истории прослушивания должна быть доступна из приложения 99,5% времени в течение месяца.
2. В случае отсутствия истории приложение должно отображать сообщение об отсутствии данных и предлагать релевантный контент.

Требования производительности:

1. Время загрузки истории прослушивания для пользователя должно составлять не более 1,5 секунд при наличии 100 треков в базе данных.
2. Приложение должно корректно обрабатывать до 500 запросов на просмотр истории в минуту от разных пользователей.
3. При использовании функции фильтрации и сортировки треков отклик системы должен занимать не более 2 секунд.

Требования совместимости:

1. Функция должна корректно работать на устройствах с операционными системами Android 8.0+ и iOS 13.0+.
2. Функция должна сохранять данные в локальном кэше приложения для повторного быстрого доступа при отсутствии интернета.

Требования к безопасности:

1. История прослушивания должна быть доступна только авторизованным пользователям.
2. Все данные (включая историю прослушивания) должны передаваться через защищённое соединение (HTTPS).
3. Данные в локальном кэше приложения должны быть зашифрованы.
4. Доступ к истории прослушивания должен быть защищён через механизм авторизации токенами (OAuth 2.0).

Требования к устойчивости:

1. Приложение должно корректно обрабатывать отсутствие соединения с сервером, отображая локально сохранённую историю, если таковая имеется.
2. В случае ошибки сервера пользователь должен видеть сообщение о технических неполадках с предложением повторить запрос позже.