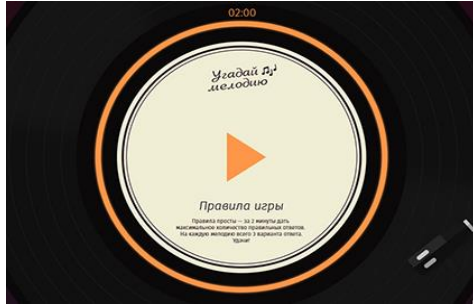


# Угадай мелодию

## Техническое задание



Угадай мелодию — это онлайн-игра, в которой игроку нужно угадывать исполнителей, названия и жанры определённых мелодий.

## Описание функциональности

### Страницы приложения

- Приложение состоит из нескольких последовательно переключающихся экранов: `Welcome/Game (/)`, `Sign In (/login)`, `Result (/result)`, `GameOver (/lose)`.
- В ходе игры пользователь переходит от первого экрана к последнему.
- Часть экранов отвечает за игровой процесс: экран начала игры и экран игрового шага.
- Другая часть экранов отвечает за вспомогательные сценарии, такие как авторизация.

### Приветственный экран

- На приветственном экране показываются правила игры и кнопка старт для начала новой игры.

### Игровые экраны

- По ходу игры пользователю нужно ответить на все предложенные вопросы. При этом даётся возможность ошибиться три раза за игру.
- При каждой ошибке в верхнем правом углу высвечивается красная нота.
- Если игрок ошибается третий раз, ему засчитывается поражение и он перемещается на экран проигрыша.
- В левом верхнем углу экрана появляется ссылка на приветственный экран. Нажатие на эту ссылку возвращает пользователя на приветственный экран.

## Виды игровых экранов

### Угадай исполнителя

- При переходе на этот экран композиция начинает проигрываться автоматически.
- Пользователю предлагается выбрать одного исполнителя из нескольких предложенных вариантов.

### Угадай жанр

- При переходе на этот экран первая композиция начинает проигрываться автоматически.
- В этом режиме пользователю даётся четыре композиции, каждую из которых можно запустить самостоятельно.
- Среди предложенных песен нужно выбрать только те, которые относятся к определённому жанру.
- В каждый момент времени может играть не более одной композиции.
- Правильный ответ засчитывается только в случае, когда пользователь правильно выбрал все композиции.
- Если он допускает хотя бы одну ошибку, ответ считается неправильным.

### Экран победы

- Для просмотра результатов игры пользователю требуется выполнить аутентификацию (см. взаимодействие с сервером).
- После успешного прохождения игры пользователю показывается экран с результатами и предложением сыграть ещё раз.
- При нажатии на кнопку `Попробовать ещё раз` пользователь сразу попадает на первый вопрос с тем же набором вопросов, что и в прошлый раз.

### Экран поражения

- На экране поражения отображается причина проигрыша: сообщение о том, что кончились попытки.
- На экране есть кнопка `Попробовать ещё раз` чтобы начать игру с начала.

## Взаимодействие с сервером

- Все необходимые данные загружаются с сервера.
- Сервер доступен по адресу: `https://4.react.pages.academy/guess-melody`.
- В случае ошибки запроса к серверу отображается информационное сообщение. Дизайн сообщения остаётся на усмотрение разработчика.
- Сервер принимает данные в виде JSON объекта.
- Запросы должны предоставлять доступ к кукам. В случае если запросы отправляются через `axios`, то должен быть проставлен параметр `withCredentials: true`.

## Структуры данных

- QuestionArtist

```
{
  "answers": [
    {
      "artist": "Quincas Moreira",
      "picture": "https://htmlacademy-react-2.appspot.com/guess-melody/static/artist/Quincas_Moreira.jpg"
    },
    {
      "artist": "Density & Time",
      "picture": "https://htmlacademy-react-2.appspot.com/guess-melody/static/artist/Density_n_Time.jpg"
    },
    {
      "artist": "Endless Love",
      "picture": "https://htmlacademy-react-2.appspot.com/guess-melody/static/artist/Endless_Love.jpg"
    }
  ],
  "song": {
    "artist": "Quincas Moreira",
    "src": "https://htmlacademy-react-2.appspot.com/guess-melody/static/music/Blue_Whale.mp3"
  },
  "type": "artist"
}
```

- QuestionGenre

```
{
  "answers": [
    {
      "genre": "reggae",
      "src": "https://htmlacademy-react-2.appspot.com/guess-melody/static/music/Addis_Ababa.mp3"
    },
    {
      "genre": "electronic",
      "src": "https://htmlacademy-react-2.appspot.com/guess-melody/static/music/Azure.mp3"
    },
    {
      "genre": "country",
      "src": "https://htmlacademy-react-2.appspot.com/guess-melody/static/music/Whaling_City.mp3"
    },
    {
      "genre": "reggae",
      "src": "https://htmlacademy-react-2.appspot.com/guess-melody/static/music/Skanada.mp3"
    }
  ],
  "genre": "reggae",
  "type": "genre"
}
```

- User

```
{
```

```
"email": "Oliver.conner@gmail.com",  
"password": "12345678"  
}
```

- AuthInfo

```
{  
  "email": "Oliver.conner@gmail.com",  
  "id": 1  
}
```

## Маршруты

### GET /questions

Получение списка вопросов.

- Коды ответов:

- 200 OK

Пример:

- Request:

- URL: GET /questions

- Response:

- Status: 200 OK

- Body: массив, содержащий элементы типа `QuestionArtist` или `QuestionGenre`

### POST /login

Авторизация пользователя на сервере. В случае успешного запроса в куку записывается токен, по которому в дальнейшем происходит авторизация. Если авторизация на сервере не проходит, то возвращается 401 ошибка при запросах на приватные части сайта.

Пример:

- Request:

- URL: POST /login

- Body: структура `User`

- Response:

- Status: 200 OK

- Body: структура `AuthInfo`

- Request:

- URL: POST /login

- Body: структура `User`

- Response:

- Status: 400 Bad Request

## GET /login

Проверка статуса авторизации пользователя.

Пример:

- Request:
  - URL: GET /login
- Response:
  - Status: 200 OK
  - Body: структура `AuthInfo`
- Request:
  - URL: GET /login
- Response:
  - Status: 401 Unauthorized (в случае если не пройдена авторизация)