# Техническое задание: Телеграм бот для проведения аудио-экскурсий.

**/start:**

Приветствие, описание формата.

Привет 👋

Меня зовут Ситик! Петербург скрывает от тебя много тайн!

Хотел бы их разгадать? Предлагаю тебе отправится в путешествие по культурной столице России в новом формате.

▪️ Уникальные путешествия и квесты, которые можно получить только через наш клуб.

▪️ Онлайн формат для индивидуальных прогулок без группы, гида и привязки к дате.

Скорее жми «НАЧАТЬ ПРИКЛЮЧЕНИЕ» ✨

Кнопка: НАЧАТЬ ПРИКЛЮЧЕНИЕ (переход в главное меню).

**Главное меню:**

Текст: “Главное меню”.

Кнопки: Подробнее, Тарифы, Выбрать экскурсию (выпадающий список).

**Подробнее:**

Санкт-Петербург был основан Петром I в 1703 году. Он назвал этот город в честь своего покровителя, апостола Петра. С самого начала Санкт-Петербург стал центром культуры, науки и искусства. Город пропитан тайнами и мифами, которые складывались множество столетий.  
  
Если ты хочешь разгадать тайны Санкт-Петербурга, то наши путешествия - это именно то, что тебе нужно. Мы предлагаем уникальную возможность познакомиться с историей и культурой города, а также увидеть все его главные достопримечательности, разгадать загадки, открыть множество его секретов.  
  
Кроме того, наша экскурсия - это отличная возможность попробовать новый формат путешествий. Тебе нужен только смартфон и наушники. Ты можешь проходить маршрут в несколько этапов, в собственном ритме, останавливать и продолжать экскурсию в любой удобный момент. Ты не привязан к группе и можешь наслаждаться красотами города в индивидуальном формате.  
  
Путешествуя с SAINT CITY, ты получишь незабываемые впечатления и прекрасно проведешь время в одном из самых красивых городов мира.

**Тарифы:**

Описание тарифов.

|  |  |  |  |
| --- | --- | --- | --- |
|  | ЛЮБИТЕЛЬ | ТУРИСТ | ГУРУ |
| Кол-во э | 1 | 3 | 5 |
| период | 30 | 30 | 90 |
| стоимость | 299 | 807,3 | 1270,75 |

Тариф открывает доступ к одной или нескольким экскурсиям на ограниченный период времени (30, 90 дней). По окончании периода бот должен оповестить об этом.

Кнопки: Тариф 1, Тариф 2, Тариф 3

**Тариф N:**

Оплата тарифа.

**Экскурсия N:**

Описание экскурсии: текст, фото.

Кнопки: Выбрать экскурсию, Другие экскурсии (выпадающий список), Главное меню.

**Выбрать экскурсию:**

После выбора экскурсии необходимо проверить что тариф по экскурсии оплачен. Если нет – перейти к оплате тарифа.

При успешной проверке на оплату:

Для доступа к экскурсии Напишите /exc1 (например)

**Сама экскурсия:**

В любой момент экскурсии можно выйти в главное меню командой /stop.

Доступ к данному блоку только после оплаты тарифа.

Текст, Аудио, Фото, Местоположение.

**Текст, аудио, фото для примера экскурсии.**

Кнопка: ДВИГАЕМСЯ.

**ДВИГАЕМСЯ:**

Фото, Местоположение следующей точки.

**Пример фото.**

Кнопка: НА МЕСТЕ.

**НА МЕСТЕ:**

Аудио, Фото.

Опционально квиз: ответы в виде кнопок. Можно выбирать ответ пока не получится правильный.

ВОПРОС. Оглянитесь вокруг и найдите лицо героини греческих мифов. Что это за женщина?

Афродита / Сирена / **Медуза Горгона** / Фемида

После квиза – дополнительно Текст, Фото.

Также следующее местоположение, фото.

Для организации таблицы, в которой нужно хранить прогресс пользователей по разным экскурсиям, вы можете использовать следующую структуру:

1. \*\*Таблица Users (Пользователи):\*\*

- user\_id (Primary Key) - уникальный идентификатор пользователя

- user\_name - имя пользователя или другие данные, связанные с пользователем

2. \*\*Таблица Excursions (Экскурсии):\*\*

- excursion\_id (Primary Key) - уникальный идентификатор экскурсии

- excursion\_name - название экскурсии или другие данные, связанные с экскурсией

3. \*\*Таблица UserProgress (Прогресс пользователя):\*\*

- user\_id (Foreign Key к Users) - ссылка на пользователя

- excursion\_id (Foreign Key к Excursions) - ссылка на экскурсию

- progress - значение прогресса пользователя для данной экскурсии (например, 0 или 1)

Пример SQL-создания таблиц:

```sql

CREATE TABLE Users (

user\_id INT PRIMARY KEY,

user\_name VARCHAR(255)

);

CREATE TABLE Excursions (

excursion\_id INT PRIMARY KEY,

excursion\_name VARCHAR(255)

);

CREATE TABLE UserProgress (

user\_id INT,

excursion\_id INT,

progress INT,

PRIMARY KEY (user\_id, excursion\_id),

FOREIGN KEY (user\_id) REFERENCES Users(user\_id),

FOREIGN KEY (excursion\_id) REFERENCES Excursions(excursion\_id)

);

```

В этой структуре, таблица UserProgress содержит информацию о прогрессе пользователей для каждой экскурсии. Составной первичный ключ (user\_id, excursion\_id) гарантирует уникальность комбинаций пользователь-экскурсия, а внешние ключи обеспечивают связь с таблицами Users и Excursions.

Когда пользователь делает прогресс в какой-либо экскурсии, вы можете вставить новую строку в таблицу UserProgress с соответствующими значениями user\_id, excursion\_id и progress. Такой подход позволяет эффективно хранить и обрабатывать данные о прогрессе пользователей по разным экскурсиям.