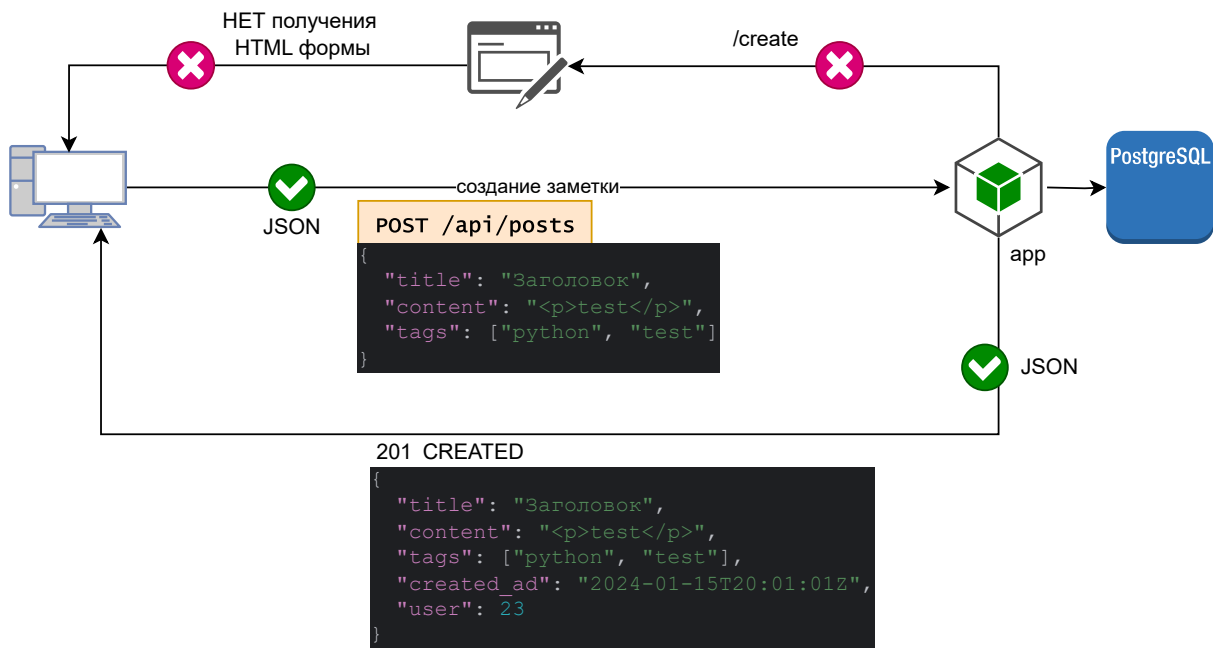
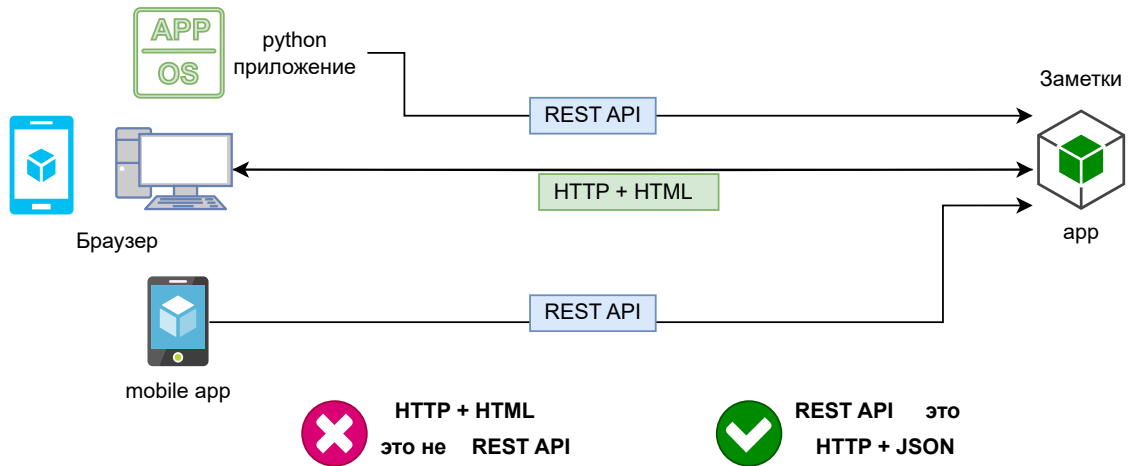


REST (Representational State Transfer) — это архитектурный стиль, который используется для построения веб-сервисов.



Клиент-серверная архитектура

Система делится на клиентов и серверы. Клиенты и серверы независимы друг от друга и могут развиваться и изменяться независимо.





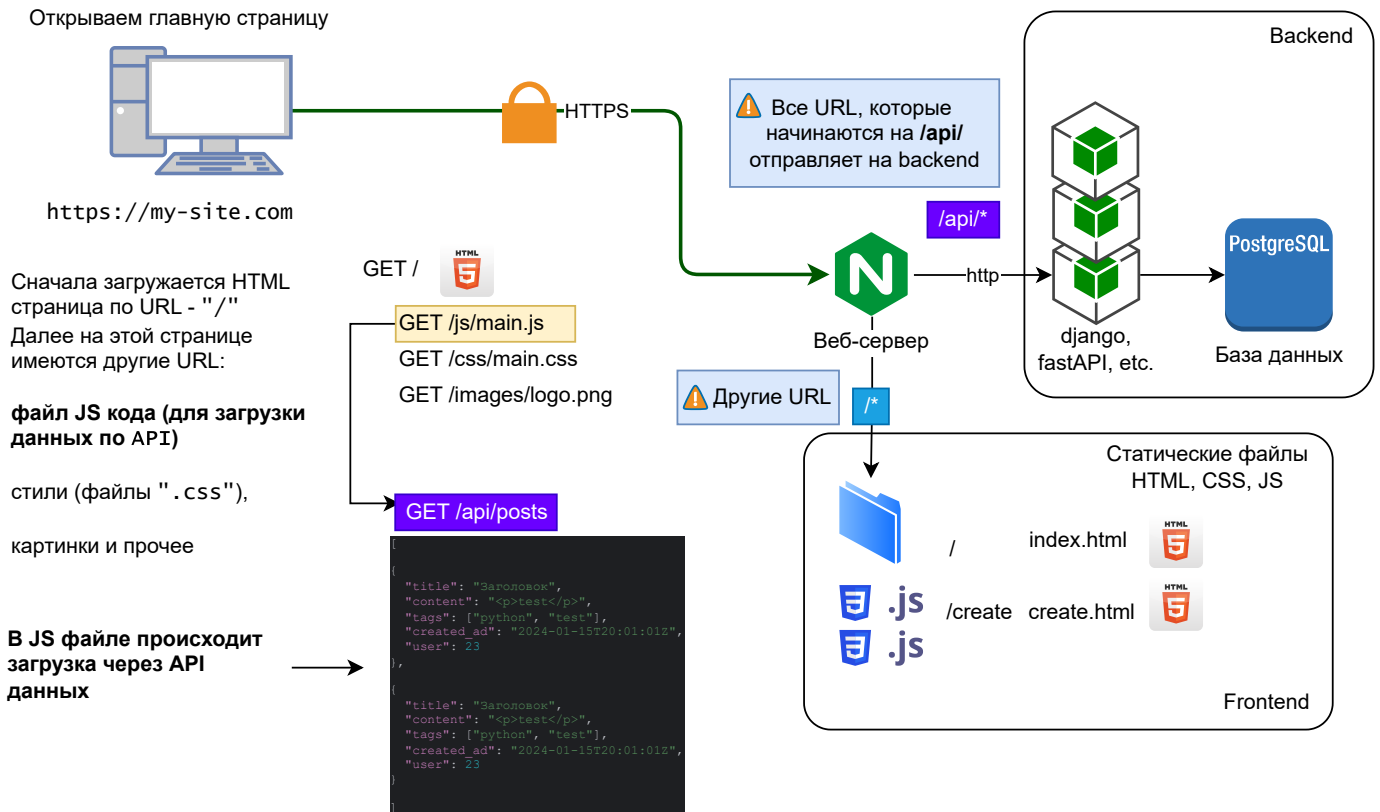
Каждый запрос - первый (stateless)

Каждый запрос от клиента к серверу должен содержать всю необходимую информацию для выполнения запроса. Сервер не должен хранить состояние клиента между запросами.



Многоуровневая архитектура

Архитектура может быть составной из различных уровней, где каждый уровень выполняет определенные функции. Каждый уровень не знает о деталях реализации других уровней.



`https://my-site.com`

GET `/create`



GET `/js/create.js`

GET `/css/create.css`

GET `/images/logo.png`

POST `/api/posts`

```
{
  "title": "Заголовок",
  "content": "<p>test</p>",
  "tags": ["python", "test"]
}
```

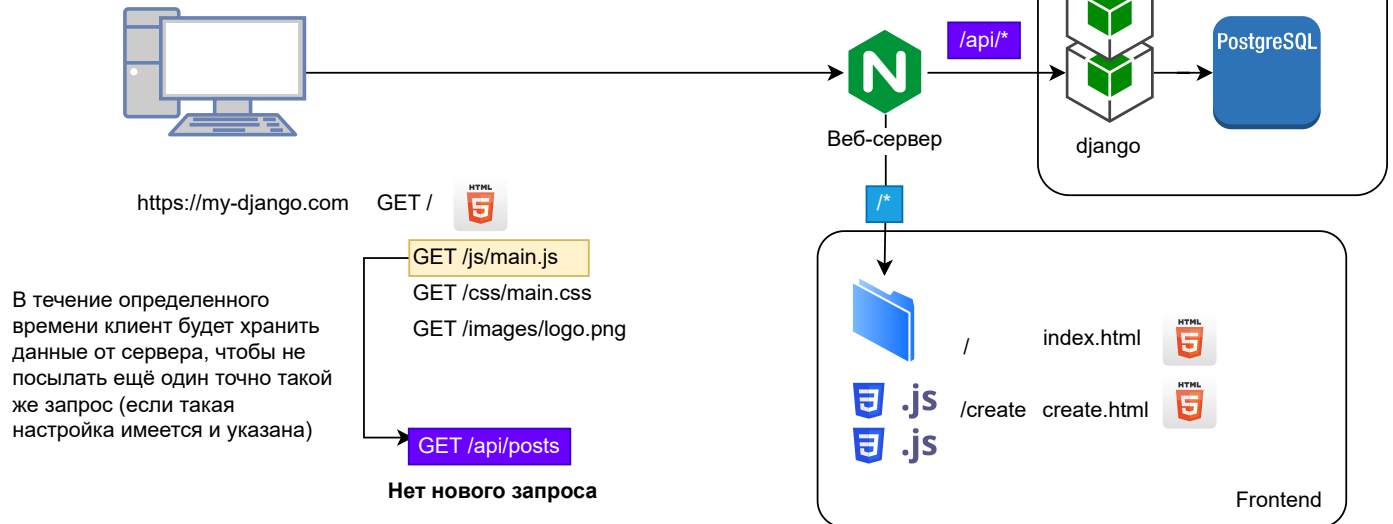
201 CREATED

```
{
  "title": "Заголовок",
  "content": "<p>test</p>",
  "tags": ["python", "test"],
  "created_at": "2024-01-15T20:01:01Z",
  "user": 23
}
```



Кэширование

Клиенты могут кэшировать ответы сервера для улучшения производительности. Серверы должны указывать, могут ли ответы быть кэшированы или нет.

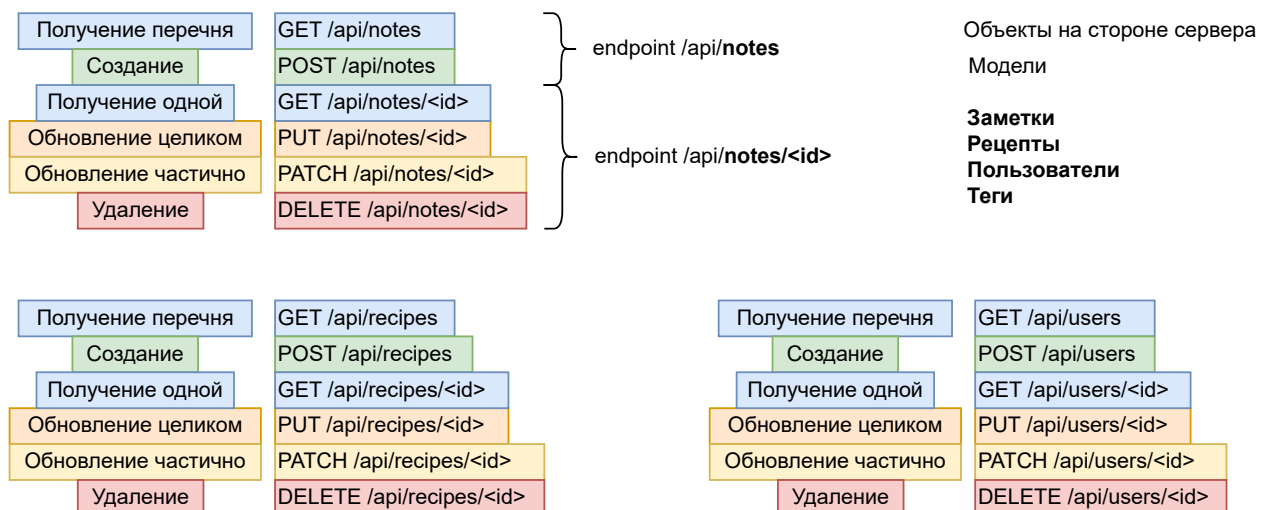


Единоеобразие интерфейса

Идентификация ресурсов: Каждый ресурс (например, документ, изображение или сервис) идентифицируется уникальным URI (Uniform Resource Identifier).

Манипуляция ресурсами через представление: Клиенты могут изменять состояние ресурсов, отправляя представления ресурсов на сервер.

Самоописываемые сообщения: Каждое сообщение содержит достаточно информации для понимания, как обрабатывать сообщение. Это позволяет серверу и клиенту быть независимыми.



БЕЗОПАСНЫЕ МЕТОДЫ HTTP

GET, OPTIONS, HEAD

ОПАСНЫЕ МЕТОДЫ HTTP

POST PUT PATCH DELETE