

Упрощенно HTTP

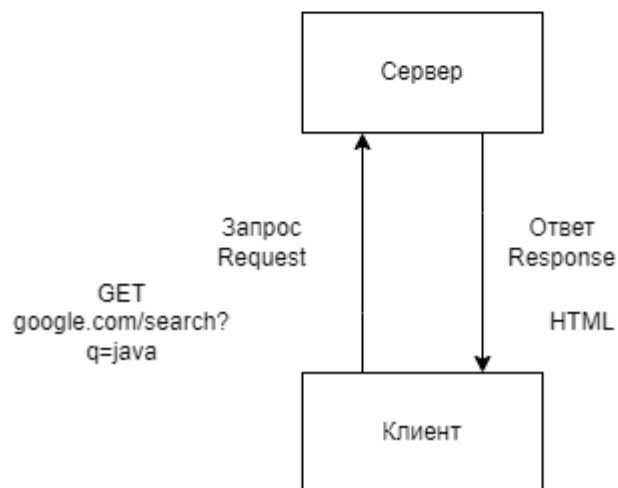
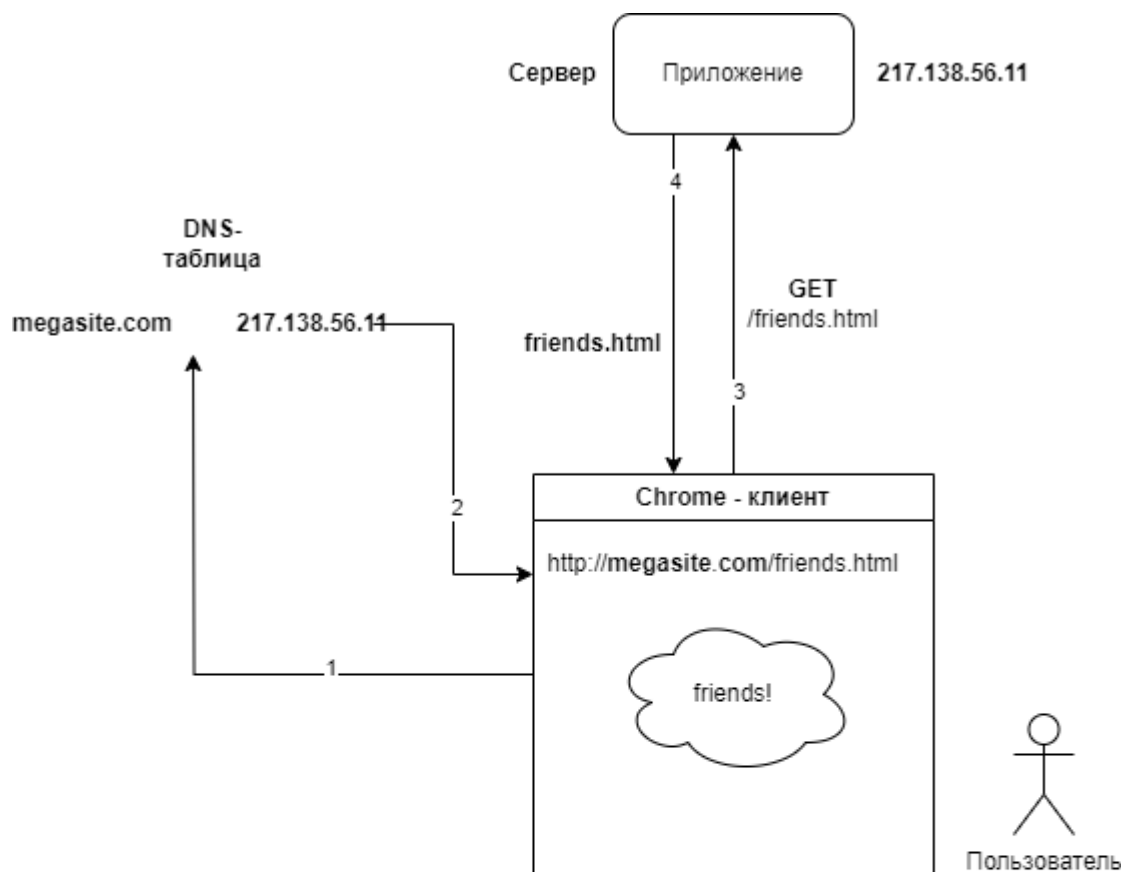
- HTML - Hyper Text Markup Language
- HTTP - Hyper Text Transfer Protocol
- HTTP - протокол, который описывает взаимодействие между клиентом и сервером. Нужен для передачи HTML в том числе.
- DNS-таблица - таблица в сети интернет, доступная всем провайдерам. В ней перечислены доменные имена и соответствующие им сервера.
- Сервер - выделенный компьютер, у которого есть публичный IP-адрес и по нему можно обратиться
- Клиент - какое-либо приложение (мобильное приложение, браузер, desktop-приложение), может отправлять запросы серверу.
- HTTP - протокол, который позволяет клиенту отправлять запрос серверу и получать ответ от сервера.

Структура HTTP-запроса

- URL - адрес, по которому мы хотим обратиться.
- Метод запроса - глагол, который позволяет серверу понять "семантику" запроса. Т.е. его назначение.
 - GET - получение информации от сервера
 - POST - отправка данных на сервер (например при регистрации - используется этот тип запроса)

Структура HTTP-ответа

- Статус ответа
 - 404 - не найдено, не существует, Not found
 - 200 - все хорошо
 - 500 - произошла ошибка на сервере
 - 401 - Не авторизован
 - 403 - Запрещено
 - 405 - Метод не поддерживается сервером (Method not allowed)
- Тело ответа
 - HTML-страница
 - Видео-файл
 - Картинка
 - Т.д.



Пример запроса-ответа

GET <https://www.google.com/search?q=java> - поисковый запрос в GOOGLE

- URL - `https://www.google.com/search`
- Method - GET
- Параметр запроса - q, значение параметра java
- Тело запроса (только у POST)

Ответ:

- Статус - 200
- Тело ответа - HTML-страница с результатами поиска

Forms

- Тег `<form>` - позволяет отправлять запросы на сервер из вашей HTML-страницы

```
<form action="https://google.com/search">
  <label for="search-input">Можно найти все!</label>
  <input type="text" id="search-input" name="q" placeholder="Enter
query..." />
  <input type="submit" value="Search" />
</form>
```

- `action` - URL-страницы, на которую отправляем запрос. Если не указывать `action` запрос уходит на тот же URL, под которым открыта страница
- `method` - метод запроса (get, post), по умолчанию - get
- `<label>` - метка для вашего input (текст, который будет отображаться рядом)
 - `for` - к какому input следует привязать метку
- `<input>` - элемент для ввода каких-либо данных.
 - `type` - тип input, по умолчанию - обычный текст (text)
 - `id` - идентификатор input
 - `name` - название input, используется в качестве названия параметра запроса
 - `placeholder` - текст, который будет отображаться в input по умолчанию
- `<input type="submit">` - тип input, который представляет собой кнопку для отправки запроса
 - `value` - текст, который отображается на кнопке
- `<input type="password">` - тип input, который скрывает значения, вводимые пользователем
- `<input type="email">` - тип input, который проверяет, ввели ли вы корректный email (не очень хорошо проверяет)
- `<input type="radio">` - тип input, который позволяет выбрать один вариант из нескольких. Группа таких вариантов определяется параметром `name`. У инпутов одной группы они должны иметь одно название.
- `<input type="checkbox">` - тип input, который позволяет выбрать несколько вариантов из нескольких.
- `<select>` - представляет собой список для выбора одного варианта
- `` - позволяет назначить какие-либо атрибуты фрагменту HTML, например назначить CSS-класс части текста
- `<input type="color">` - тип input для выбора цвета

- `<input type="range"` - тип input для выбора диапазона
- `<input type="date"` - тип input для выбора даты

Разница между GET и POST

- При использовании GET-запросов все параметры размещаются в URL-запросе
- Это удобно при работе, например, с поиском. Потому что потом можно в истории браузера просто посмотреть, какие вы запросы на поиск делали и по ним перейти еще раз.
- Это очень не удобно, если вы хотите какие-то данные "скрыть", а не светить их в истории, URL и т.д.
- Решение - использовать для таких запросов метод POST, который все данные отправляет в теле запроса.