Базові поняття мережевих технологій

№ уроку: 25 **Курс:** Manual QA

Засоби навчання: Браузер, Microsoft Office

Огляд, мета та призначення уроку

Метою цього уроку є детальне ознайомлення основними базовими поняттями мережевих технологій.

Вивчивши матеріал даного заняття, учень зможе:

- Розбиратися з чого складається http запит та відповідь.
- Знати, що таке URL, URI, URN
- Працювати з http методами.
- Розуміти навіщо потрібні cache та cookies.
- Знати, що таке IP, DNS.

Зміст уроку

- URL, URI
- HTTP request, HTTP response
- HTTP methods
- Cache, Cookie
- IP, DNS
- IPv4, IPv6

Резюме

- **«Клієнт — сервер» (англ. client–server)** — обчислювальна або мережева архітектура, в якій завдання або мережеве навантаження розподілені між постачальниками послуг, які називаються серверами, та замовниками послуг, які називають клієнтами.

- **HTTP** – це протокол, в якому описано правила передачі даних в Інтернеті. Він допомагає браузеру завантажувати веб-сторінки, а серверу отримати інформацію, яку користувач ввів на сайті.

- **HTTPS** — це той самий протокол, але з надбудовою безпеки.

- **Уніфікований ідентифікатор ресурсу (URI)** – це компактна послідовність символів, що ідентифікує абстрактний або фізичний ресурс.

- URI можна додатково класифікувати як локатор, ім'я або те й інше. Термін «уніфікований покажчик ресурсу» (URL) відноситься до підмножини ідентифікаторів URI, які, крім ідентифікації ресурсу, надають засоби визначення розташування ресурсу шляхом опису його основного механізму доступу (наприклад, його мережевого «місцеположення»).

- URI можна класифікувати як локатори (URL), імена (URN) або те й інше. Універсальне ім'я ресурсу (URN) функціонує як ім'я людини, а уніфікований покажчик ресурсу (URL) нагадує поштову адресу цієї людини. Іншими словами: URN визначає ідентичність елемента, а URL надає метод його пошуку.

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- URL-адреса складається з таких частин:
    - **Схема**: URL-адреса – це протокол, який потрібно використовувати для доступу до ресурсу. Крім добре відомих HTTP та HTTPS, ви можете використовувати безліч інших схем.
    - **Домен**: ця частина вказує на сервер, на якому розміщено ресурс. Це може бути доменне ім'я або IP-адреса.
    - **Порт**: це порт протоколу, на який надсилається запит на доступ до ресурсу. Зазвичай він опускається, що означає, що слід використовувати порт протоколу за промовчанням.
    - **Шлях**: це шлях ресурсу на хостинг-сервері.
    - **Параметри**: це необов'язкова додаткова інформація, яку надає хост-сервер.
    - **Якір**: ця частина становить певну частину всередині ресурсу. Його також називають фрагментом.

HTTP запит та відповідь – текстові повідомлення. Вони мають однакову структуру:

a. Рядок запиту/статусу. У першому рядку міститься основна інформація. Для запиту це HTTP-метод, шлях та версія протоколу. Для відповіді – статус: версія протоколу, код відповіді та опціональне пояснення. Значення йдуть саме в такому порядку, і поділяються пробілами.

b. Заголовки. Кожен наступний (до порожнього) рядок представляє заголовок. Заголовки – метаінформація повідомлення у форматі ключ-значення. Назва відокремлюється від значення символом.

c. У заголовках приходять кодування, розмір тіла, дані про кешування, тощо. Стандартний набір доступних заголовків описаний у специфікації. Користувач має право вигадувати власні заголовки - їх прийнято забезпечувати префіксом X-. Для запиту обов'язковий заголовок Host.

d. Тіло (опціональне). Відокремлюється від заголовків порожнім рядком. Можливість запиту/відповіді мати тіло може бути обмежена використовуваним HTTP-методом та кодом статусу. У тілі міститься будь-яке корисне навантаження. Наприклад, відповідь на HTTP запит сайту містить HTML-код сторінки, який і відмальовується браузером.

- Код відповіді (стану) HTTP показує, чи було успішно виконано певний HTTP запит. Коди згруповані у 5 класів:
    - Інформаційні 1xx
    - Успішні 2xx
    - Перенаправлення 3xx
    - Клієнтські помилки 4xx
    - Серверні помилки 5xx

- HTTP визначає методи, щоб вказати бажану дію, яка має бути виконана з ідентифікованим ресурсом.

Метод	Опис
GET	Дозволяє запитати певний конкретний ресурс. Додаткові дані можуть бути надіслані через рядок запиту (Query String) у складі URL

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

	(наприклад ?param=value)
POST	Дозволяє надіслати дані на сервер. Підтримує надсилання різних типів файлів, серед яких текст, PDF-документи та інші типи даних у двійковому вигляді. Зазвичай метод POST використовується при надсиланні інформації (наприклад, заповненої форми логіну) та завантаженні даних на веб-сайт, таких як зображення та документи.
HEAD	Сервер у відповідь на запит повертає заголовок і тіло, в якому міститься запитуваний ресурс. Даний метод при використанні його в запиті дозволить отримати тільки заголовки, які сервер повернув би при отриманні GET-запиту до того ж ресурсу. Запит з використанням даного методу зазвичай проводиться для того, щоб дізнатися розмір ресурсу, що запитується перед його завантаженням.
PUT	Використовується для створення/оновлення ресурсів на сервері .
DELETE	Дозволяє видалити існуючі ресурси на сервері .
OPTIONS	Дозволяє запитати інформацію про сервер, у тому числі інформацію про HTTP-методи, що допускаються до використання на сервері.
PATCH	Дозволяє внести часткові зміни до зазначеного ресурсу за вказаним розташуванням.

- **Безпечні методи**. Метод запиту безпечний, якщо запит з використанням цього методу не впливає на сервер. Методи GET, HEAD, OPTIONS та TRACE визначені як безпечні. Іншими словами, безпечні методи призначені лише для читання. Однак вони не виключають побічних ефектів, таких як додавання інформації про запит у файл журналу або списання коштів з рекламного рахунку, оскільки вони не запитуються клієнтом за визначенням.

- Навпаки, методи POST, PUT, DELETE, CONNECT та PATCH небезпечні. Вони можуть змінювати стан сервера або мати інші наслідки, такі як надсилання електронної пошти.

- Незважаючи на прописану безпеку GET-запитів, на практиці їх обробка сервером технічно не обмежена. Недбале або навмисно неправильне програмування може призвести до того, що запити GET спричинять нетривіальні зміни на сервері. Це не рекомендується через проблеми, які можуть виникнути, коли веб-кешування, пошукові системи та інші автоматичні агенти вносять ненавмисні зміни на сервер. Наприклад, веб-сайт може дозволити видалення ресурсу за допомогою URL-адреси, такої як https://example.com/article/1234/delete, який при довільному видаленні, навіть з використанням GET, просто видалити статтю.[59] На правильно закодованому веб-сайті для цієї дії буде потрібний метод DELETE або POST.

- **Ідемопотентні методи**. Метод запиту є ідемпотентним, якщо кілька ідентичних запитів із цим методом мають той самий ефект, що й один такий запит. Методи PUT та DELETE, а також безпечні методи визначені як ідемпотентні. Безпечні методи тривіально ідемпотентні, оскільки вони не повинні впливати на сервер; методи PUT та DELETE, тим часом, є ідемпотентними, оскільки послідовні ідентичні запити ігноруватимуться. Веб-сайт може, наприклад, налаштувати кінцеву точку PUT для зміни зареєстрованої адреси електронної пошти користувача. Якщо ця кінцева точка налаштована правильно, будь-які запити, які просять змінити адресу електронної пошти користувача на ту саму адресу електронної пошти, яка вже записана, наприклад. дублювати запити після успішного запиту не матиме жодного ефекту. Так само запит на ВИДАЛЕННЯ певного користувача не буде мати жодного ефекту, якщо цей користувач вже був видалений.

- Навпаки, методи POST, CONNECT та PATCH не обов'язково є ідемпотентними, і тому багаторазове надсилання одного і того ж запиту POST може ще більше змінити стан сервера

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

або мати додаткові наслідки, такі як надсилання кількох електронних листів. У деяких випадках це бажаний ефект, але в інших випадках може статися випадково. Користувач може, наприклад, ненавмисно відправити кілька POST-запитів, знову натиснувши кнопку, якщо йому не буде надана чітка інформація про те, що обробляється перше клацання. Хоча веб-браузери можуть відображати діалогові вікна попереджень, щоб попередити користувачів у деяких випадках, коли перезавантаження сторінки може призвести до повторного надсилання запиту POST, зазвичай веб-додаток повинен обробляти випадки, коли запит POST не слід надсилати більше одного разу.

- Зверніть увагу, що ідентифікація методу не залежить від протоколу або веб-сервера. Цілком можливо написати веб-додаток, в якому (наприклад) вставка в базу даних або іншу неідемпотентну дію запускається GET або іншим запитом. Однак зробити це всупереч рекомендаціям може призвести до небажаних наслідків, якщо користувач агент припускає, що повторення одного і того ж запиту безпечно, хоча це не так.

- **Методи, що кешуються**. Спосіб запиту кешується, якщо відповіді на запити за допомогою цього методу можуть бути збережені для повторного використання в майбутньому. Методи GET, HEAD та POST визначені як кешовані.

- Навпаки, методи PUT, DELETE, CONNECT, OPTIONS, TRACE та PATCH не кешуються.

- **IP-адреса** – це унікальна адреса в мережі, необхідна для знаходження, отримання та передачі інформації від одного комп'ютера (вузла) до іншого.

- **DNS (англ. Domain Name System "система доменних імен")** – комп'ютерна розподілена система для отримання інформації про домени.

Закріплення матеріалу

- З чого складається http request?
- Чи є будь-який URN так само і URI? А чи будь-який URI є URL?
- Які існують найвідоміші http методи?
- Для чого потрібні cache, cookie?
- Що таке IP та DNS?

Самостійна діяльність учня

Завдання 1
Дізнайтеся вашу зовнішню IP адресу. Відкрийте адмін сторінку роутера за адресою IP.

Завдання 2
Відкрийте наступний сайт (https://petstore.swagger.io/#/) і надішліть кілька запитів усіх типів запитів. Отримайте різні результати – не лише успіхи

Завдання 3
Проаналізуйте URL нижче.
https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL#basics_anatomy_of_a_url

- Чи це URI?
- Визначте та підпишіть, які компоненти є цієї у URL-адреси
- Чи наявні всі можливі компоненти URL у прикладі? Якщо ні, то яких не вистачає? Доповніть приклад відсутніми елементами як вони могли б виглядати.

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

Рекомендовані ресурси

What is a URL?
https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL#basics_anatomy_of_a_url

Is there any relation between URI and URL?
https://www.quora.com/Is-there-any-relation-between-URI-and-URL

URL, URI, URN: What's the Difference?
https://auth0.com/blog/url-uri-urn-differences/

Усі статус коди
https://pbs.twimg.com/media/CuP3rMkWEAEDSac?format=jpg&name=large

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com