

4F

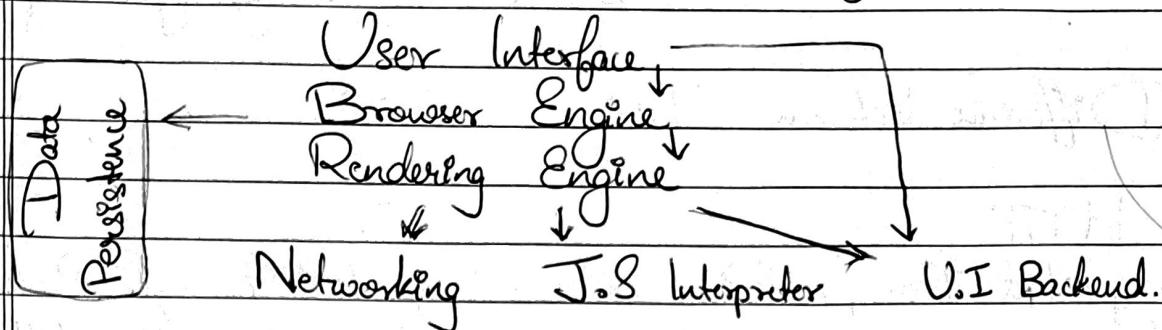
YASH SARANG - I 11AD

A

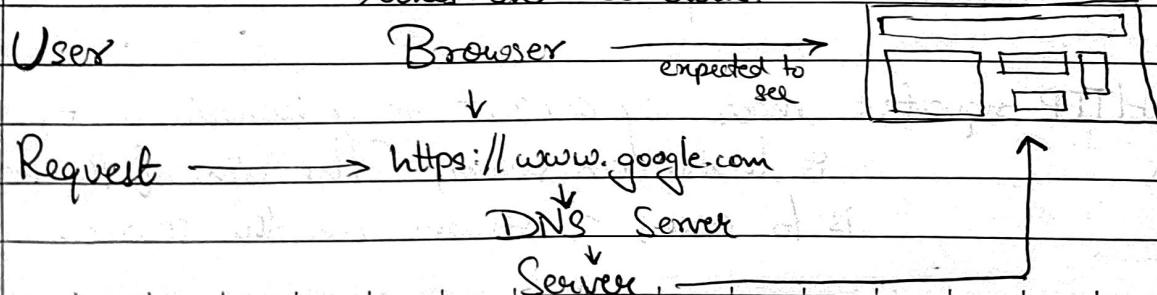
Shrey

Web Computing Assignment 1.

Q. Write in brief about the working of web browser.



- The User Interface: include the address bar, back/forward button, bookmarking menu, etc. part of the browser display except the window where you see the requested page.
- The Browser Engine: Maintains actions between the UI and the rendering engine.
- The Rendering Engine: Responsible for displaying requested content.
- Networking: For Network Calls such as HTTP requests, using different implementations for different platforms behind a platform.
- UI Backend: Used for drawing basic widgets like combo boxes and windows.



STAND P. pmravindra PP.

vii) Javascript interpreter - Used to parse and execute Javascript code.

viii) Data persistence (storage) - The storage used by the browser to save data locally e.g. Cookies.

2. Difference between HTTP and HTTPS.

HTTP

HTTPS.

- 1. HyperText Transfer Protocol
 - 2. URL begins with http://
 - 3. Uses port number 80.
 - 4. Insecure.
 - 5. Works at Application layer.
 - 6. Encryption is absent
 - 7. Does not require any certificate
 - 8. Faster
 - 9. Does not improve search ranking
- HyperText Transfer Protocol Secure
 - URL begins with https://
 - Uses port number 443
 - Secured.
 - Works at Transport layer.
 - Encryption is present.
 - requires SSL certificate.
 - Slower.
 - search ranking is improved.

3. How does HTTP send and receive requests?

HTTP works as a request-response protocol between a client and server. A client sends an HTTP request to server, then the server returns a response to the client. The response contains status information about the request and may also contain the requested content.

HTTP requests: made by a client to a normal host, which is located on server. The aim of the request is to access a resource on the source.

HTTP response: made by a server to client. The aim of response is to provide client with resource requested or inform the client that the action has been carried out.

4) What is URT, URL and URN? Write brief about URL.

→ URI - Uniform Resource Identifier.

It contains URL and URN.

URN stands for Uniform Resource Name.

They are globally unique persistent identifiers assigned within defined namespaces.

URL stands for Uniform Resource Locator.

Scheme: Every URL begins with the scheme. It tells the browser what type of address it is, so the browser connects to it correctly.

Domain name: most prominent part of web address.

Different pages on same site will continue to use the same domain name.

File path: Tells the browser to load a specific page.

Parameters: cannot be understood by a human, but it is critical information for the server.

Anchor: Tells the browser to scroll or load a specific part of page. Begins with a hashtag # and is used to direct browser to specific part of a very long page.

5) What is XML? How is it different from HTML?

XML stands for Extensible Markup Language and is a text based markup language.

XML

1. Designed to carry data
2. Tags are not pre-defined
3. Forms a tree structure
[root to branches]
4. Dynamic
5. Order is required
6. Space is preserved
7. Proper nesting is to be done

HTML

1. Designed to display data.
2. Tags are pre-defined.
3. Does not have a tree like structure.
4. Static
5. No order required.
6. Space gets truncated
7. Proper nesting is not required.

6) Difference between JSON and XML.

JSON

1. Object created has some type.
2. Supported data types are strings, numbers, booleans, etc.
3. Does not have capacity to display data.
4. Has no tags.
5. Can use array to represent data.
6. Data oriented.
7. Less secure than XML.
8. File size is smaller.
9. Easy to parse.
10. Quicker to read and write.

XML

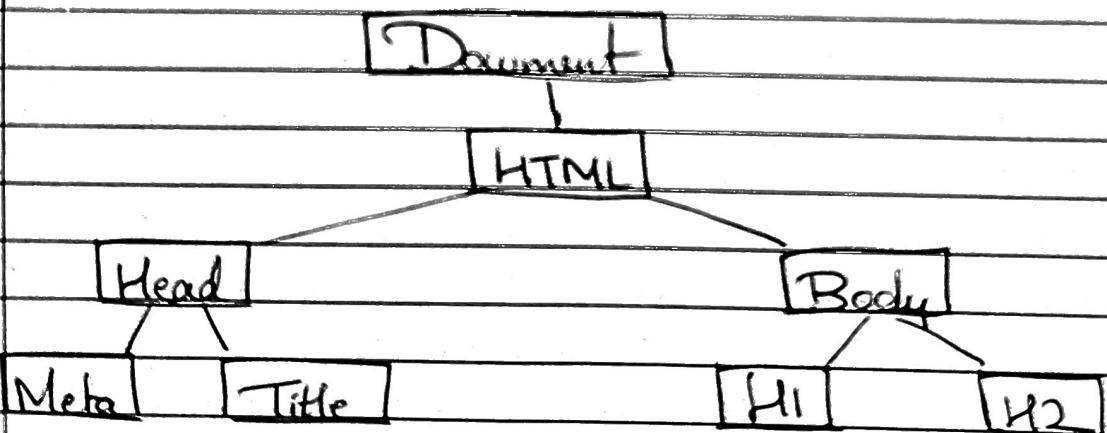
1. Does not have any type.
2. Data is in string format.
3. Has capacity to display data.
4. Has start & end tags.
5. Does not have array concept.
6. Document oriented.
7. More secure than JSON
8. File size is larger.
9. Difficult to parse.
10. Takes time to read & write because learning curve is higher.

7) What are the different data types in JSON?

- 1. String
- 2. Number
- 3. Object
- 4. Array
- 5. Boolean
- 6. Null

8) What is document object model?

- DOM is a way to represent the webpage in a structured hierarchical way so that it will become easier for programmers and users to glide through the document.
- We can easily access and manipulate tags.
- Using DOM, javascript gets access to HTML as well as CSS of web page and can also add behavior to the HTML elements.
- It is an API that represents and interacts with HTML or XML document.



9) What is an API? What is REST API?

→ API stands for Application Programming Interface which defines the rules that you must follow to communicate with the software systems.

REST API (Representational State Transfer) is a software architecture that imposes conditions on how an API should work. It was initially created as a guideline to manage communication on a complex network like internet. It supports high performing and reliable communication.

10) Explain TLS handshake.

→ TLS is a data privacy and security protocol implemented for secure communication over internet. It usually encrypts communication between server & client.

• A Transport layer security (TLS) connection is established via handshake.

- 1) With a TLS enabled service, a sender sends a clientHello (information about client)
- 2) Server responds with serverHello message and then chooses a cipher suite from list in clientHello message. The server also transmits its Digital certificate and a final serverHelloDone
- 3) Client validates the certificate. Client then sends ClientKey Exchange message. Here client chooses a key exchange mechanism to securely establish a shared secret with server. Client also needs to send change Cipher Spec indicating that it is switching to secure communication now, which is finally followed by Finished message for indicating a successful handshake.
- 4) Server replies with Change Cipher Spec and an encrypted Finished message once shared secret is received.

2)

HTTP

10. Vulnerable to hackers
11. Does not need SSL
12. Operates at level

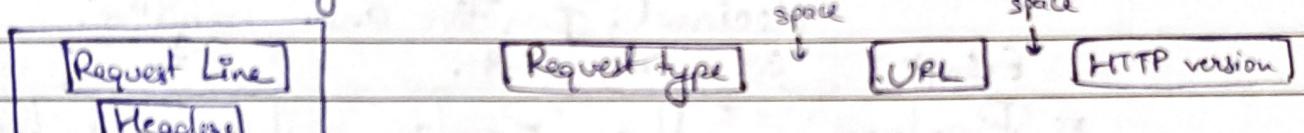
HTTPS

- Highly secure as data is encrypted
- needs SSL certificate.
- Does not have separate protocol.
- Operates using HTTP but encrypted TLS/SSL connection.

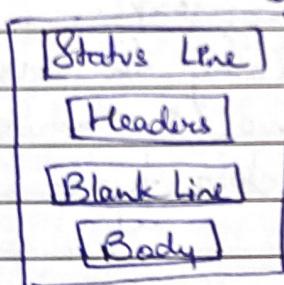
3)

HTTP request, response diagrams.

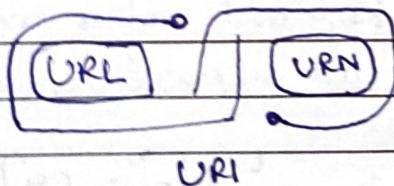
Request Message → Request line.



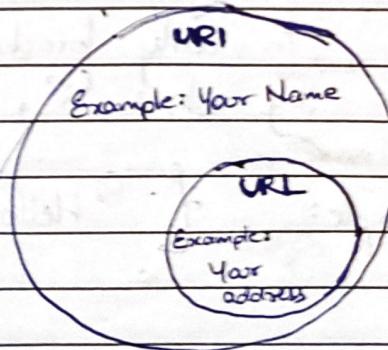
Response Message



4)



URI



7)
→

- JSON stands for Javascript Object Notation.
- It supports data structures such as arrays and objects

Basic data types supported by JSON are:

1) Strings: Characters that are enclosed in single or double quotation marks.
Example: "Yash", "Hehe Bwoi!"

2) Number: A number could be integer or decimal, positive and negative.
Example: 39, 420, 6.9.

3) Booleans: The Boolean value could be either true or false without any quotation marks.

Example: FALSE, TRUE.

4) Null: Here, null means nothing without any quotation marks.
Example:

5) Array: Lists that are represented by the square brackets and the values have commas in between them. They may contain mix. data types.
Example: [1, 2, 3, 4].

6) Objects: Objects are JSON dictionaries that are enclosed in curly brackets. Keys and values are separated by colons ':', pairs are separated by commas.

Example: { "Hello": 23, "No": 69 }

6)

JSON

XML

⇒ i) file size is smaller as compared to XML.

XML file size is larger as compared to JSON

ii) Can be parsed by a standard javascript function. Has to be done with respect to their programming language.

7)

→

In order for a REST API to be useful, it has to confirm to these criteria:

i) Uniform interface.

Fundamental to the design of any RESTful web service. It indicates that the server transfers information in a standard format.

(d)

The formatted resource is called a representation in REST. The format can be different from the internal representation of the resource on the server application.

For example, the server can store data as text but send it to an HTML representation format.

ii) Statelessness.

In REST architecture, statelessness refers to a communication method in which the server completely completes every client request independently of all previous requests.

Clients can request resources in any order, and every request is stateless or isolated from other requests.

vii) Layered system

In a layered system architecture, the client can connect to other authorized intermediaries between the client and server, and it will still receive responses from the server.

Servers can also pass on requests to other servers.

viii) Cachability.

RESTful web services support caching, which is the process of storing some responses on the client or on an intermediary to improve server response time.

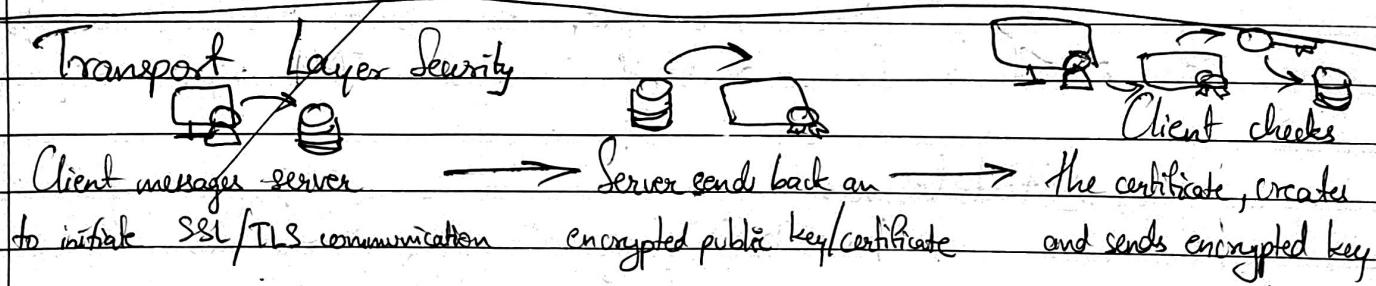
For example, suppose that you visit a website that has common header and footer images on every page.

vix) Code on demand.

In REST architectural style, servers can temporarily extend or customize client functionality by transforming software programming code to the client.

For example, when you fill a registration form on any website, your browser immediately highlights any mistakes you make, such as incorrect phone numbers.

vi) Transport Layer Security



Client decrypts the content
completing the SSL/TLS
handshake

Server decrypts the key and delivers encrypted
content with key to the client (communication
fails if certificate is invalid)