



**Vidyavardhini's College of Engineering and Technology**  
**Department of Artificial Intelligence & Data Science**

AY: 2025-26

Class:	T.E.	Semester:	V
Course Code:	CSC502	Course Name:	WEB COMPUTING

Name of Student:	SHRUTI GAUCHANDRA
Roll No. :	16
Assignment No.:	01
Title of Assignment:	Apply the fundamentals web technologies for creating web applications.
Date of Submission:	23/07/25
Date of Correction:	28/07/25

**Evaluation**

Performance Indicator	Max. Marks	Marks Obtained
Completeness	5	5
Demonstrated Knowledge	3	3
Legibility	2	2
Total	10	10

Performance Indicator	Exceed Expectations (EE)	Meet Expectations (ME)	Below Expectations (BE)
Completeness	5	3-4	1-2
Demonstrated Knowledge	3	2	1
Legibility	2	1	0

Checked by

Name of Faculty : Ms. KSHITIJ A GHARAT

Signature : *Rohit*

Date : 28/7/25

CSC502.1

Apply fundamental web technologies for creating web application.

Q1

Explain how DNS works and the process it follows to resolve domain names to IP addresses.

- ① DNS stands for Domain Name System.  
② It allows human to use easily recognizable names instead of complex numeric IP addresses.  
③ DNS resolution is the process of translating a hostname (e.g. 192.168.1.1) (e.g. www.example.com) into its corresponding IP address (e.g. 192.168.1.1).  
④ Each device on the Internet has a unique IP address similar to a street address used to locate a specific house.  
⑤ When a user enters a domain in the browser, DNS translates it into an IP address, enabling the browser to find and load the correct website.

Working of DNS:

(i) User Input:

User enters a website address into a web browser.

(ii) Local Cache Check:

The browser first checks its local cache to see if it has recently looked up the domain.

If it finds the corresponding IP address, it uses directly without querying external servers.

(iii) DNS resolver Query:

If the IP isn't in the local cache, the computer

sends a request to a DNS resolver.

The resolver is typically provided by your Internet Service Provider (ISP) or network settings.

(iv) Root DNS Server:

The resolver sends the request to a root DNS server. The root server doesn't know the exact IP address for the website but knows which Top-level-Domain (TLD) server to query based on the domain's extension.

(v) TLD Server:

The TLD server for the extension directs the resolver to the authoritative DNS server.

(vi) Authoritative DNS Server:

This server holds the actual DNS records, including the IP address for of the website's server.

It sends this IP address back to the resolver.

(vii) Final Response:

The DNS resolver sends the IP address to your computer, allowing it to connect the website's server and load the page.

Q2. Compare the two protocols used for formatting and transmitting the messages over the internet with respect to encryption, authentication, integrity and application.



Aspect

HTTP	HTTPS
① HTTP stands for Hyper-Text Transfer Protocol	① HTTPS stands for Hypertext Transfer Protocol Secure.
② HTTP uses port number 80 for communication.	② HTTPS uses 443 port number for communication.
③ HTTP is considered to be unsecure	③ HTTPS is considered to be secure.
④ Encryption is absent in HTTP	④ Encryption is present in HTTPS.
⑤ HTTP does not use data hashtags to secure data	⑤ HTTPS still have the data before sending it and return it to its original state and on the receiver side.
⑥ In HTTP data is transferred in plaintext	⑥ In HTTPS data transferred in ciphertext.

- ⑦ There is no mechanism to verify the identity of website.
- ⑧ It uses digital certificates to authenticate the website.
- ⑨ Data can be modified or corrupted during transmission without detection.
- ⑩ It ensures data integrity. Any tampering is detected and rejected.
- ⑪ It is used for general browsing where security is not a concern.
- ⑫ It is used for secure applications like banking, online shopping, login forms and data transfer.

Q3. Identify and explain the different methods to retrieve the resource, create new resource, update the resource, modify the resource and delete the resource while using the REST style.

- 
- ① Representational State Transfer (REST) is an architectural style that outlines a set of rules for designing web services.
  - ② It uses standard HTTP methods to perform operations on resources, which are identified by URLs.
  - ③ Different methods using the REST style are:
    - (i) Retrieve a resource:  
HTTP method : GET
      - (a) It is used to fetch / read data from the server.
      - (b) It is safe and idempotent.
      - (c) eg :  
GET /users/101 returns details of user with ID 101.
    - (ii) Create a new resource:  
HTTP method : POST
      - (a) It is used to create a new resource on the server.
      - (b) It sends data in the request body.
      - (c) eg :  
~~POST /users~~ creates a new user with the given data.
    - (iii) Update a resource:  
HTTP method : PUT
      - (a) It is used to replace an existing resource entirely.
      - (b) It requires complete data.

(c) eg: PUT | Users | 101 replaces user 101's data with the new data.

(iv) Modify a resource:

HTTP method: PATCH

- (a) It is used to partially update a resource.
- (b) It is only the modified fields are sent.
- (c) eg: PATCH | Users | 101 updates only specified fields of user 101.

(v) Delete a resource:

HTTP method: DELETE

- (a) It is used to remove a resource from the server.
- (b) eg: DELETE | Users | 101 deletes user 101.