1.17 CLIENT SERVER WEB COMMUNICATION

Web is just a form of client/server interaction. *Web Server* and *Web Client* are the two main elements in all web applications. The Web Client sends requests and the Web Server listens to those requests and responds/accomplishes the required task. The Web Server and Web Client communicate with each other through a common protocol.

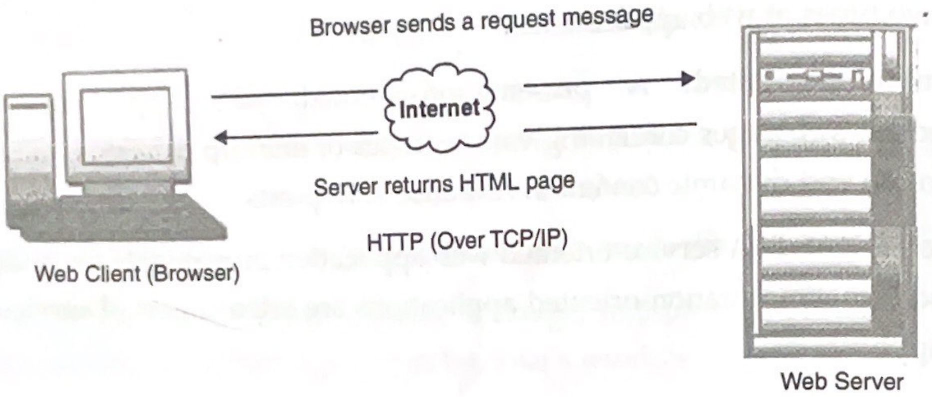

Fig. 1.13 (Client-Server Web Communication)

The request-response communication between a *Web Client* and a *Web Server* illustrated in **Fig. 1.13** is as explained below:

1. A Web client (or browser) sends requests to a Web server. Every retrievable piece of information on the Web is identified by a URL, which includes the hostname, its location, port and the protocol used to get it.

2. For example, in the URL http://www.yahoo.com/docs/index.html, the communication protocol is HTTP; the hostname is www.yahoo.com. The port number takes default number, which is TCP port 80 for HTTP. The path and file name for the resource to be located is "/docs/index.html".
3. The Web server is responsible for document storage, manage and retrieval. The Web server interprets the request message, and returns the document requested (or an error message) back to the requesting client.
4. The Web browser interprets and presents the document. The browser is responsible for document presentation.

The language that Web clients and servers use to communicate with each other is called the *Hypertext Transfer Protocol (HTTP)*. All Web clients and servers must be able to speak HTTP in order to send and receive hypermedia documents.

Now, with the rise of client-side technologies like- *Java, ActiveX, JavaScript,* and *DHTML,* it is possible to perform a great deal of computation from within the browser.

A web developer who develops web-based application must choose where to host the logic of the program (client side or server side) and which technology should be used to do it.

Now, with the rise of numerous technologies available for Web programming, web developers can focus on building their web applications rather than thinking about how the applications they are trying to build are going to work.

1.18 COMMON GATEWAY INTERFACE (CGI)

The *Common Gateway Interface,* or CGI, is a set of standards that define how information is exchanged between the web server and a custom script.

CGIs are separate programs that run on a web server. The CGI programs provide an interface between the web server and web content. CGI programs are usually written in following two types:

(i) **Programming languages:** C, C++, FORTRAN and Visual Basic

(ii) **Scripting Language:** Perl, Java, UNIX Shell and TCL.

Web server is the middleware to receive request from the Web client and pass it to CGI script, and send information back to the Web Client. The CGI script is an extension of HTML that performs actions such as form processing, image maps or database interconnectivity on information provided by web server.

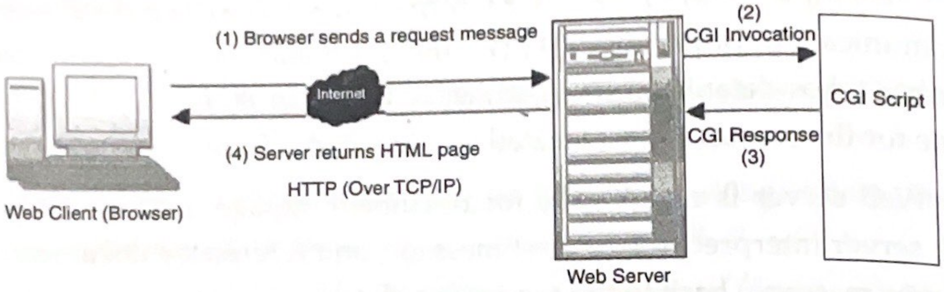

Fig. 1.14 (Client-Server process in HTTP using CGI)

The processing steps for a Web Server and CGI illustrated in **Fig. 1.14** are explained below:

1. User enters the input into the web-based form under form's different fields and clicks the submit button.
2. Browser packs the input entered in form fields according to the CGI specifications and sends requests to a Web server through Hyper Text Transfer Protocol (HTTP).
3. Web Server process the request, extracts the program and pass information on to the CGI.
4. The CGI accepts the information and communicates with any resources necessary to process the request, such as database or file server.
5. The CGI returns the result (or an error message) of the program as HTML back to the Web server.
6. Web server takes the output (standard output) of the program and sends HTML back to the browser.