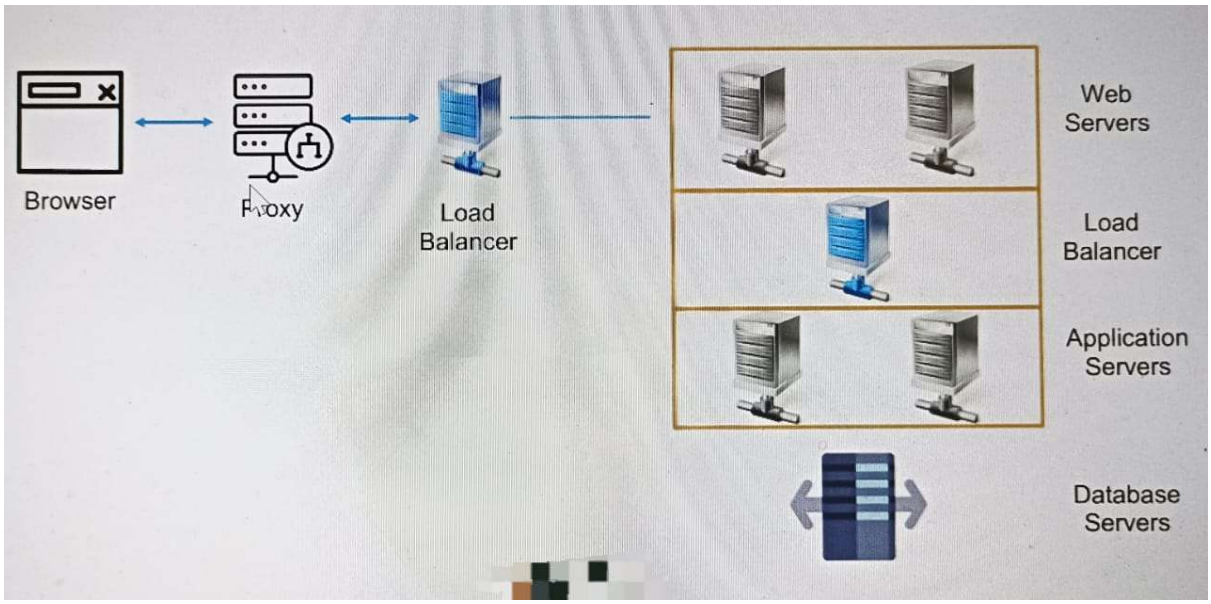
WEB ARCHITECTURE



A Web server is a program that uses HTTP (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. Dedicated computers and appliances may be referred to as Web servers as well.

A load balancer is a device that distributes network or application traffic across a cluster of servers. Load balancing improves responsiveness and increases availability of applications. A load balancer sits between the client and the server farm accepting incoming network and application traffic and distributing the traffic across multiple backend servers using various methods such as Round Robin.

An application server is a type of server designed to install, operate and host applications and associated services for end users, IT services and organizations. It facilitates the hosting and delivery of high-end consumer or business applications.

Database management systems (DBMSs) are specially designed software applications that interact with the user, other applications, and the database itself to capture and analyse data. A general-purpose DBMS is a software system designed to allow the definition, creation, querying, update, and administration of databases.

```
JS index.js > server.listen() callback
1   const http=require('http');
2   const port=8000;
3
4   const server = http.createServer();
5
6   server.listen(port,function(err){
7       if(err){
8           console.log(err);
9           return;
10      }
11
12      console.log("Server is up and running on port: ", port);
13  });
14
15  // 127.0.0.1 --> localhost
```

Note: Use 'npm init' command to setup the package.

package.json contains the metadata and project dependencies of Node project.

var http = require('http'); → create an instance of HTTP module in Node.js file

index.js | index.html | package.json

index.js > requestHandler > fs.readFile('./index.html') callback

```
1  const http=require('http');
2  const port=8000;
3  const fs = require('fs')  // to read or write from a file
4  function requestHandler(req, res){
5      console.log(req.url);
6      res.writeHead(200,{'content-type':'text/html'}); // sends a Response HTTP Status Code 200 to the
       browser
7
8      fs.readFile('./index.html', function(err,data){
9          if(err){
10             console.log('error',err);
11             res.end('<h1>Error!</h1>');
12         }
13
14         return res.end(data);
15     });
16
17     // res.end('<h1>Gotcha!</h1>');
18 }
19 const server = http.createServer(requestHandler);
20
21 server.listen(port,function(err){
22     if(err){
23         console.log(err);
24         return;
25     }
26     console.log("Server is up and running on port: ", port);
27 });
28 // 127.0.0.1 --> localhost
```

'appendFile' function of 'fs' module used for creating the file if it does not yet exist.

nodemon is a tool that helps develop node js based applications by automatically restarting the node application when file changes in the directory are detected.

npm install -g nodemon → to install nodemon globally

index.js > requestHandler > fs.readFile() callback

```
1  const http=require('http');
2  const port=8000;
3  const fs = require('fs')  // to read or write from a file
4  function requestHandler(req, res){
5      console.log(req.url);
6      res.writeHead(200,{'content-type':'text/html'}); // sends a Response HTTP Status Code 200 to the
       browser
7      let filepath;
8      switch(req.url){
9          case '/':
10             filepath='./index.html';
11             break;
12         case '/profile':
13             filepath='./profile.html';
14             break;
15         default:
16             filepath='404.html';
17     }
18     fs.readFile(filepath, function(err,data){
19             if(err){
20                 console.log('error',err);
21                 return res.end('<h1>Error!</h1>');
22             }
23
24             return res.end(data);
25         })
26 }
27 const server = http.createServer(requestHandler);
28
```