WIM LAB EXP - 3

Name : Aditi Chhajed
Reg.No : 221081009
Branch : IT
Course Instructor : Prof. Nikhil Handa

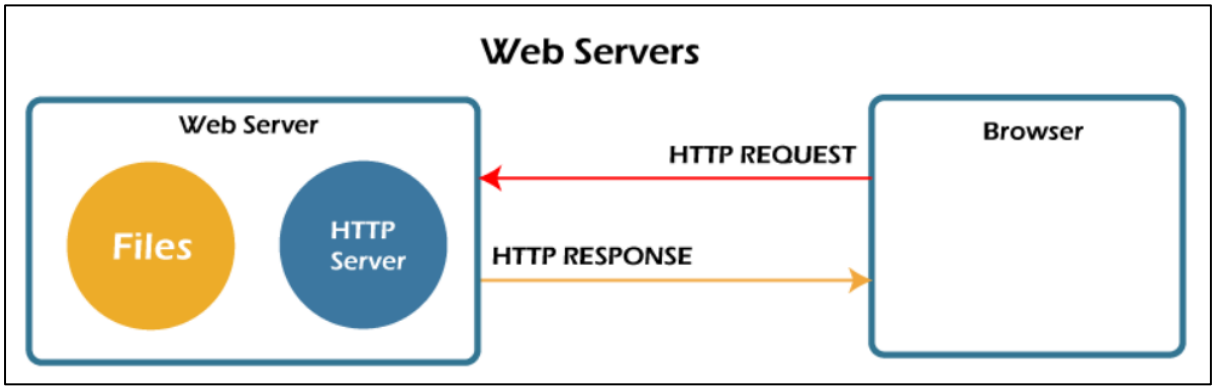
AIM :

1. Explore Webserver architecture, and installation process.
2. Create installation guide.
3. Deploy your webpage on your installed Webserver

THEORY:

Part 1: *Explore Webserver architecture, and installation process.*

a) Webserver

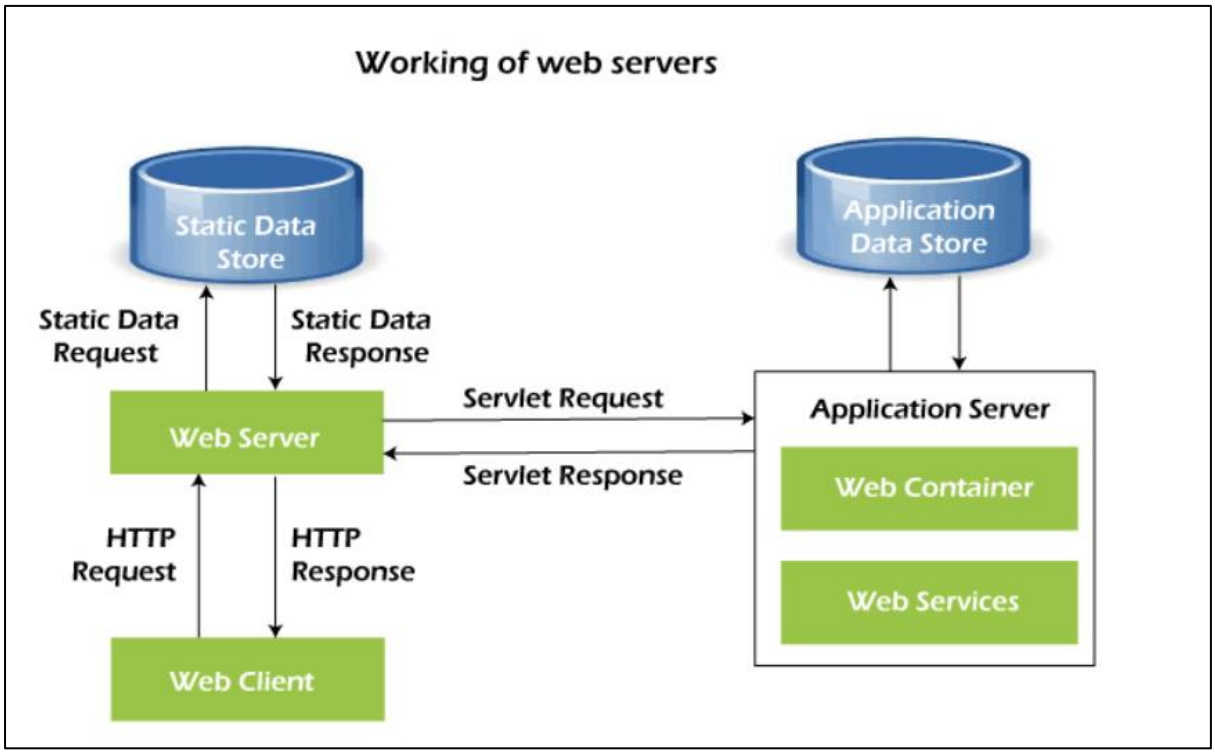


A web server is a dedicated computer responsible for running websites sitting out on those computers somewhere on the Internet. They are specialized programs that circulate web pages as summoned by the user. The primary objective of any web server is to collect, process and provide web pages to the users.

It plays a critical role in the client-server model of the World Wide Web, where clients (typically web browsers) request web pages and resources, and servers respond to these requests by delivering the requested content.

Web servers operate on the Hypertext Transfer Protocol (HTTP), which is the foundation of data communication on the World Wide Web. When you enter a website's URL into your browser, it sends an HTTP request to the web server hosting that website, which then sends back the web page you requested, allowing you to view it in your browser.

Working of Webservers:



The term web server can denote server hardware or server software, or in most cases, both hardware and software might be working together.

1. ***On the hardware side**, a web server is defined as a computer that stores software and another website raw data, such as HTML files, images, text documents, and JavaScript files. The hardware of the web servers are connected to the web and supports the data exchange with different devices connected to the Internet.*
2. ***On the software side**, a web server includes server software accessed through website domain names. It controls how web users access the web files and ensures the supply of website*

content to the end-user. The web server contains several components, including an HTTP server.

*Below is the **step by step** working of what happens whenever a web browser approaches the web server and requests a web file or file:*

1. *First, any web user is required to **type the URL of the web page in the address bar** of your web browser.*
2. *With the help of the URL, your **web browser will fetch the IP address of your domain name** either by converting the URL via DNS (Domain Name System) or by looking for the IP in cache memory. The IP address will direct your browser to the web server.*
3. *After making the connection, the **web browser will request for the web page from the web server** with the help of an HTTP request.*
4. *As soon as the web server receives this request, it immediately **responds by sending back the requested page** or file to the web browser HTTP.*
5. *If the web page requested by the **browser does not exist or if there occurs some error in the process**, the web server will return an error message.*
6. *If there occurs no error, the browser will successfully display the webpage.*

Examples of web server uses are

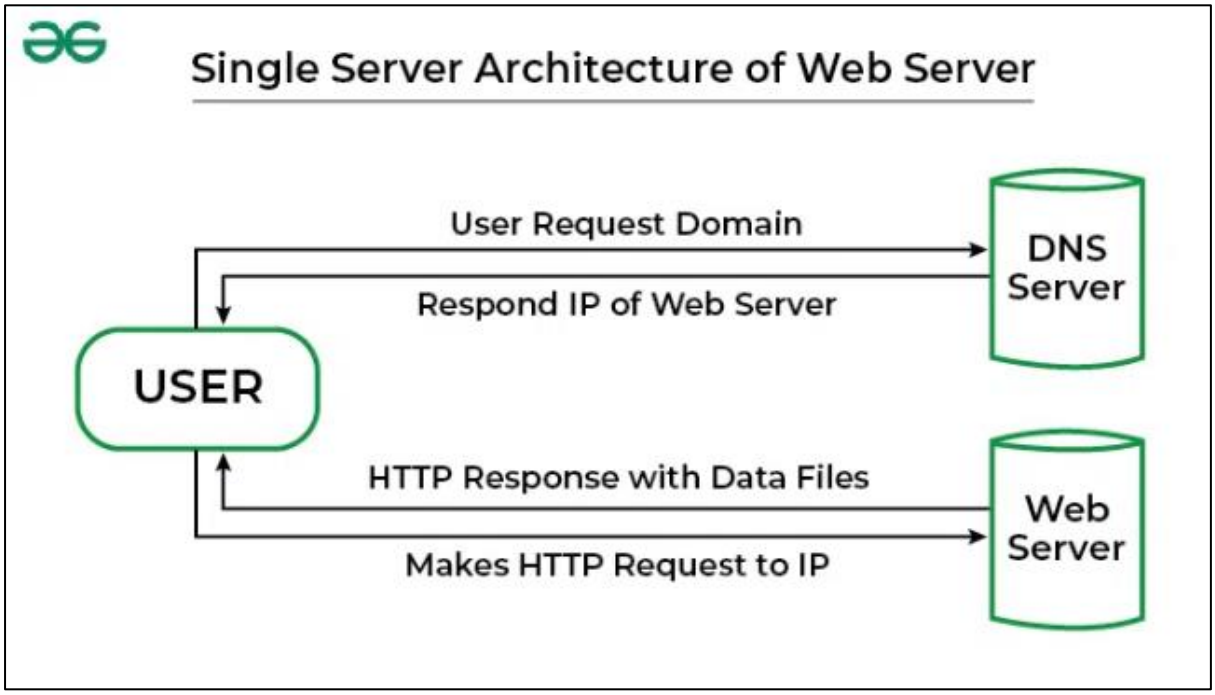
- *sending and receiving mails on Internet by using SMTP (Simple Mail transfer Protocol);*
- *fetching requests for File Transfer Protocol (FTP) files; and*
- *designing, developing, and publishing websites.*

b) Webserver Architecture:

Web server architecture refers to the structure and design of web servers, outlining how they handle incoming requests and deliver web content. There are two main approaches to web server architecture:

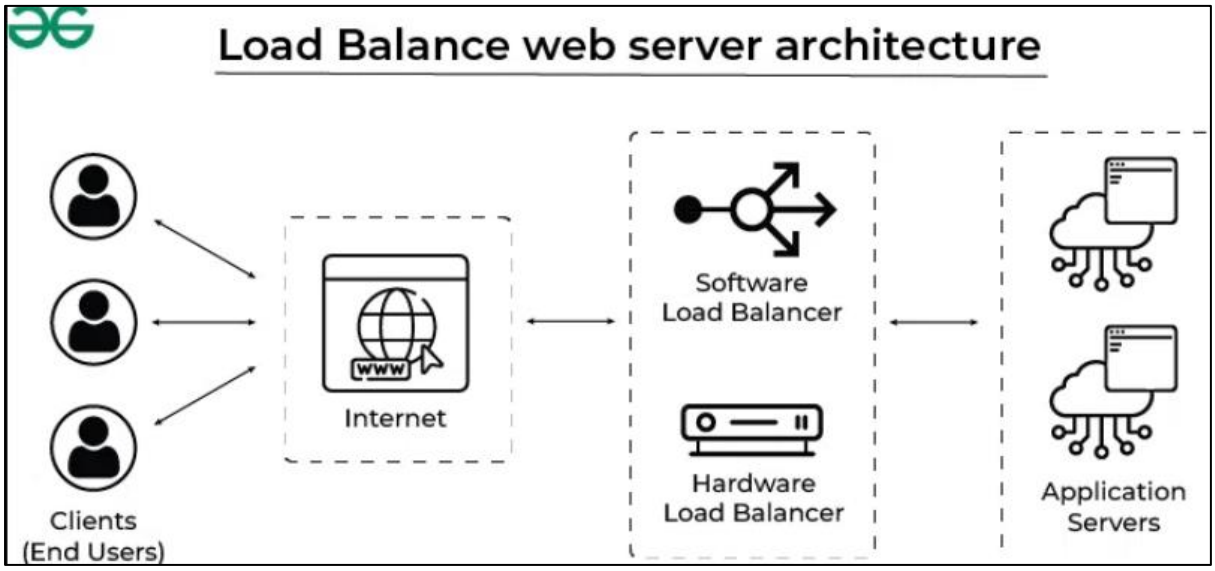
- ***Single-Tier (Single Server) Architecture:***

In a single-tier architecture, a single server is responsible for both processing requests and serving web content. This is suitable for small websites or applications with low traffic. However, it has limitations in terms of scalability and fault tolerance. If the server goes down, the entire service becomes unavailable.



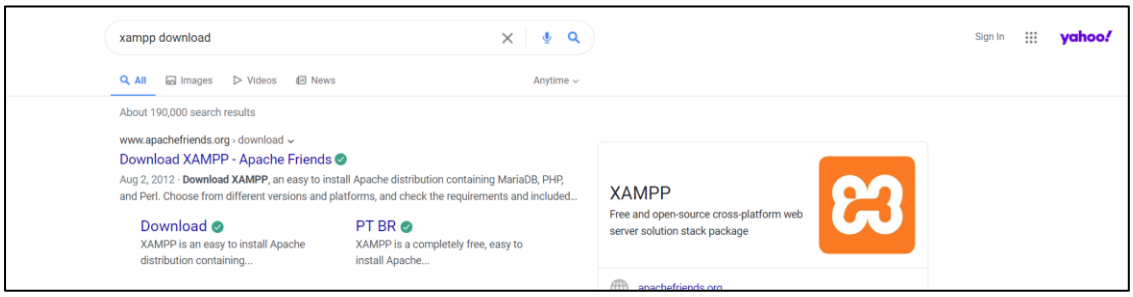
- ***Multi-Tier (Load-Balanced) Architecture:***

In a multi-tier architecture, multiple servers are used to distribute the workload and ensure high availability. This approach often involves load balancers that evenly distribute incoming requests across a cluster of web servers. Each server can serve web content independently, and if one server fails, the load balancer redirects traffic to healthy servers, ensuring uninterrupted service.

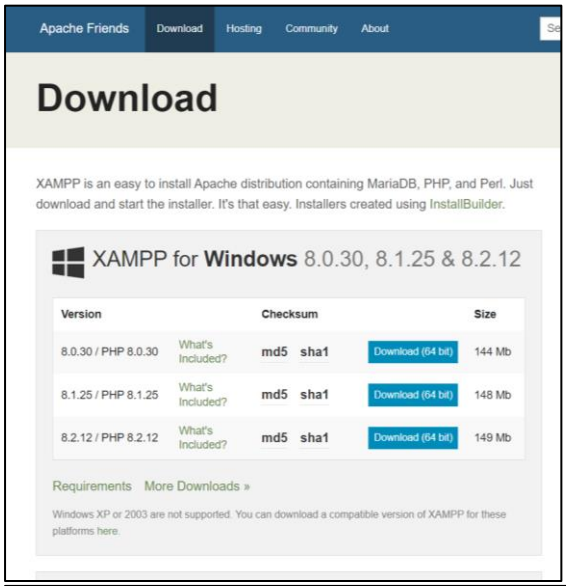


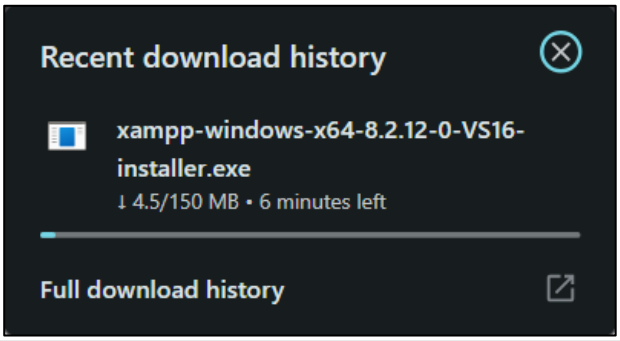
Part 2: Installation guide.

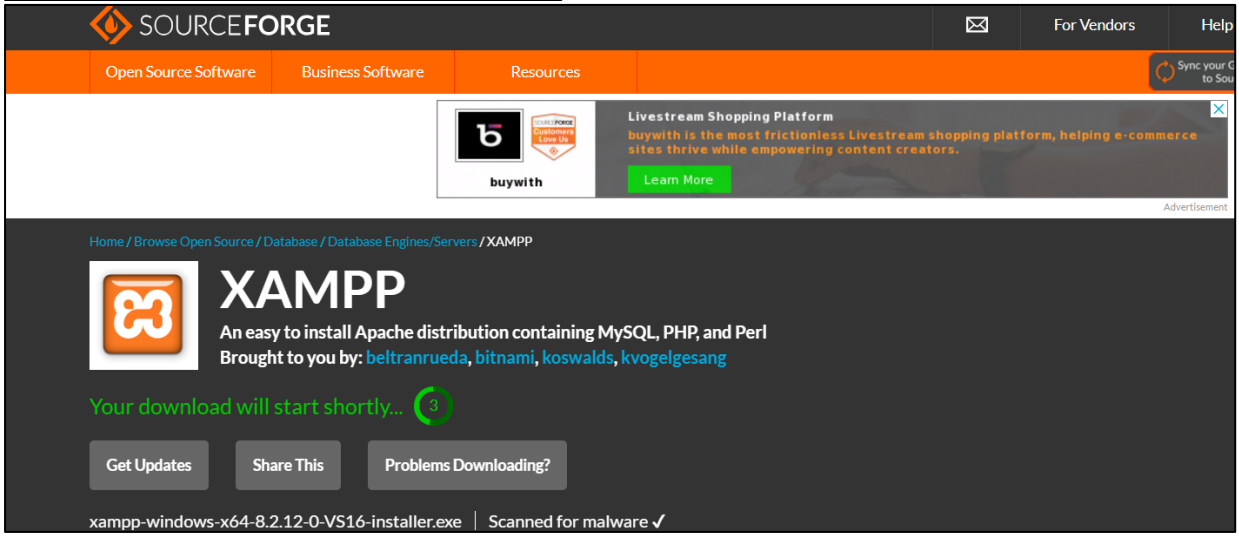
a)Search “xampp installation” on google.



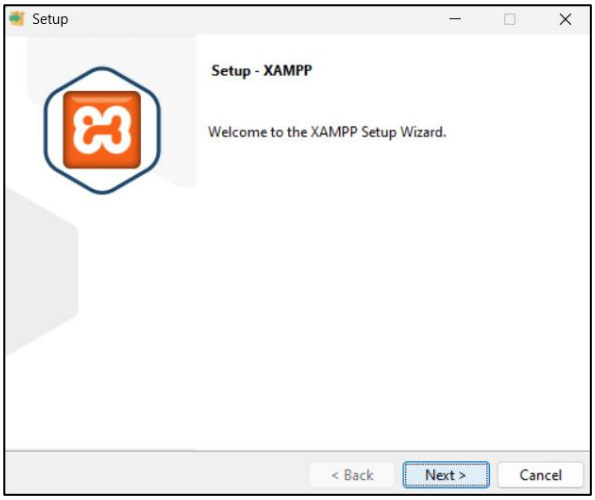
b)Download the latest version

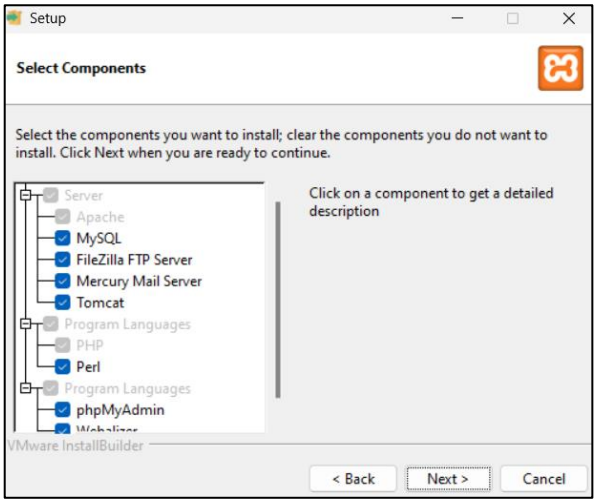


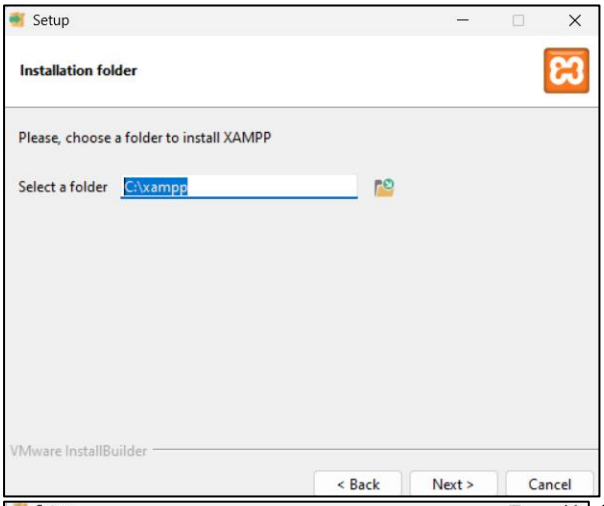


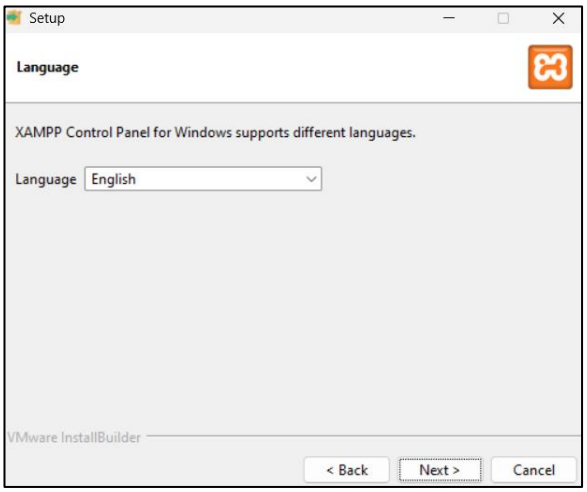


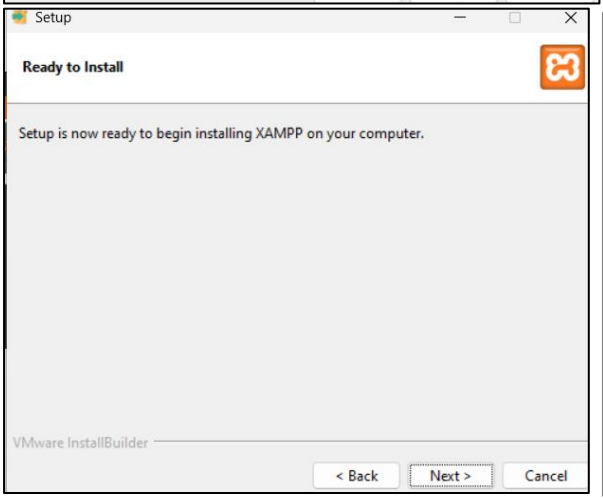
c)Setup configuration

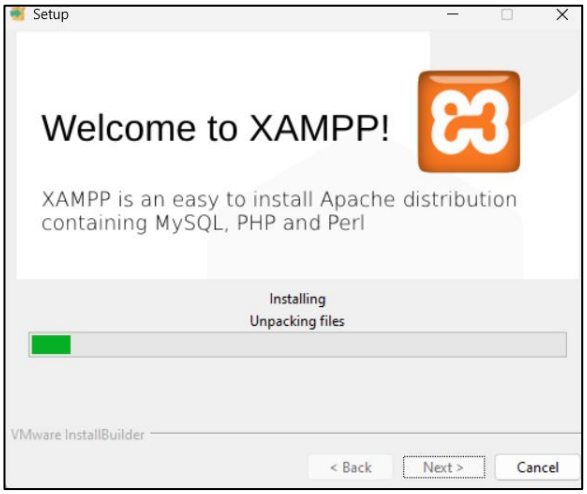


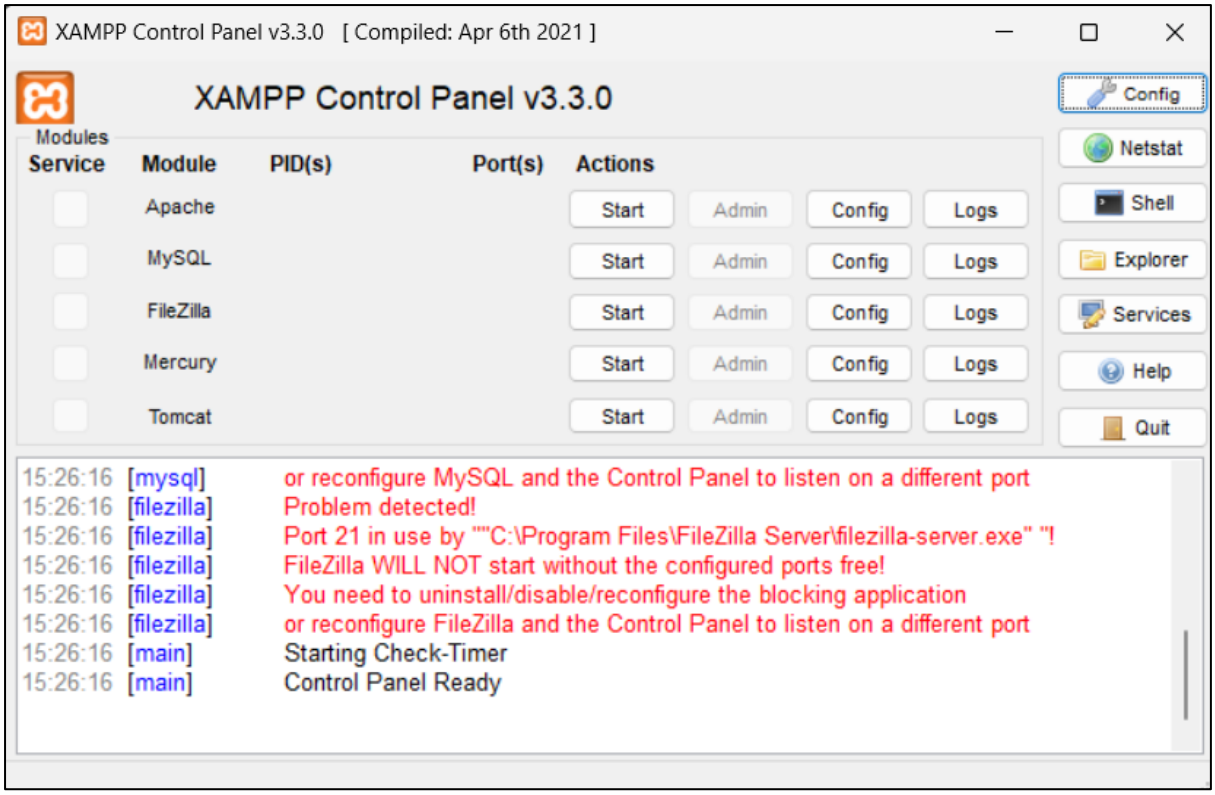






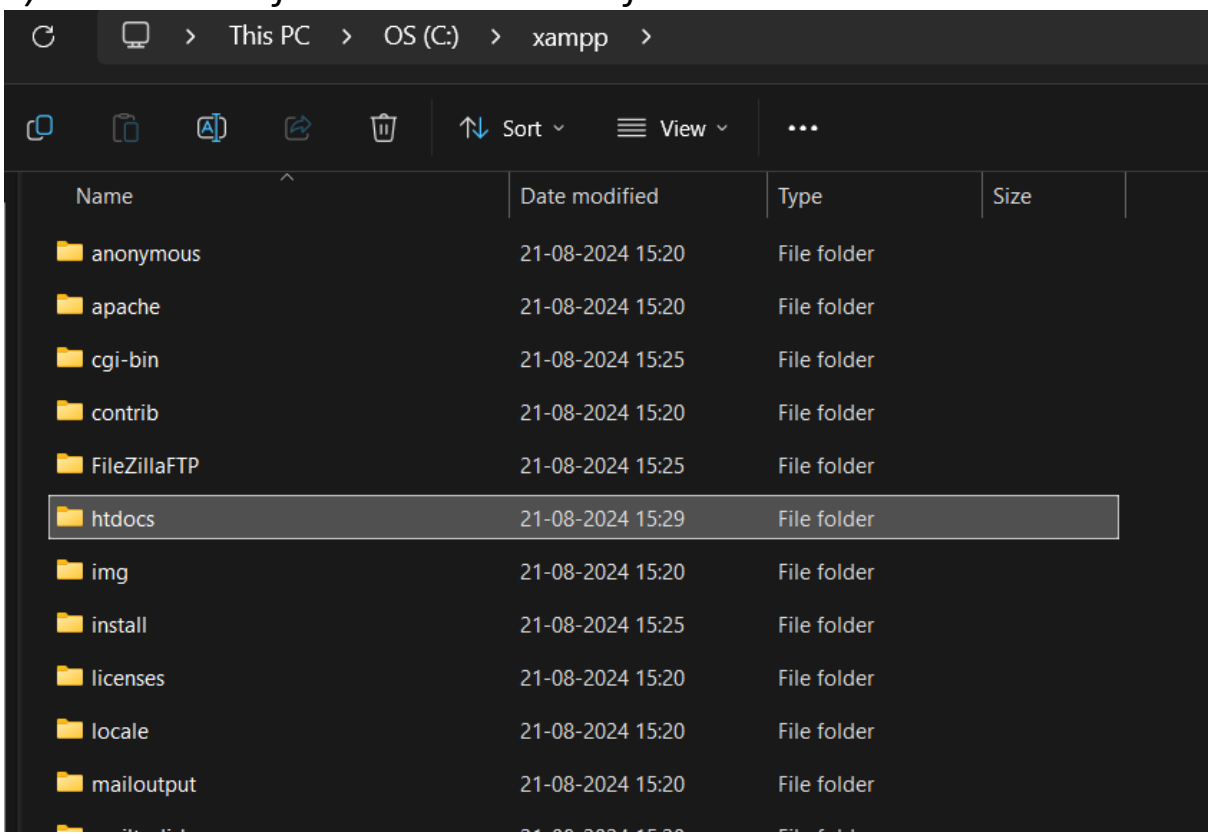




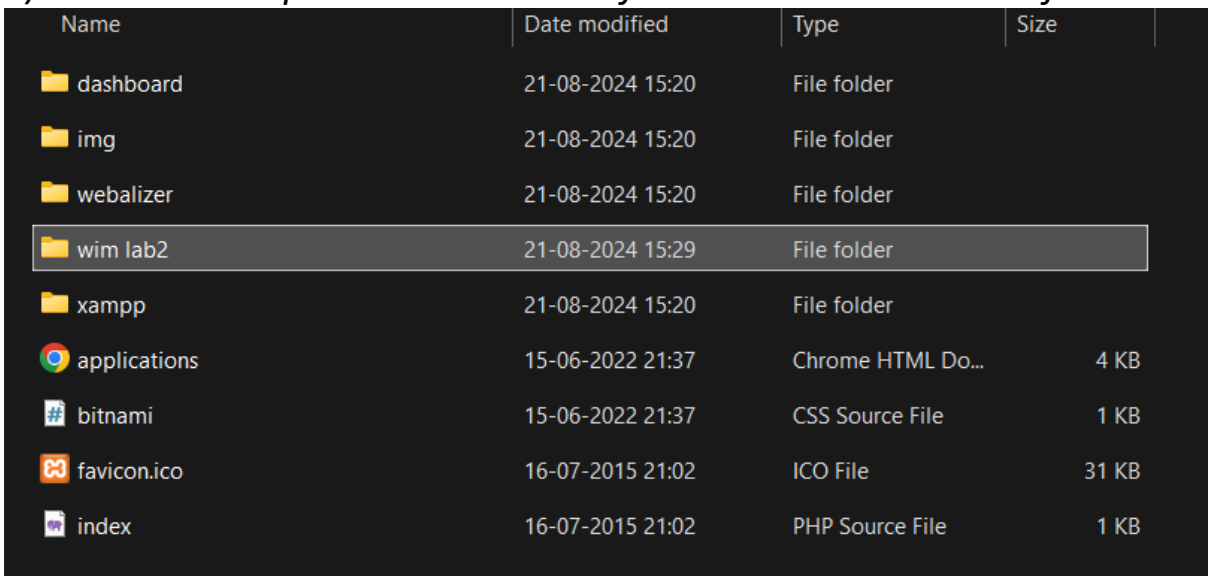


Part 3: *Deploy your webpage on your installed Webserver*

a)Move the lab files to the "htdocs" folder.

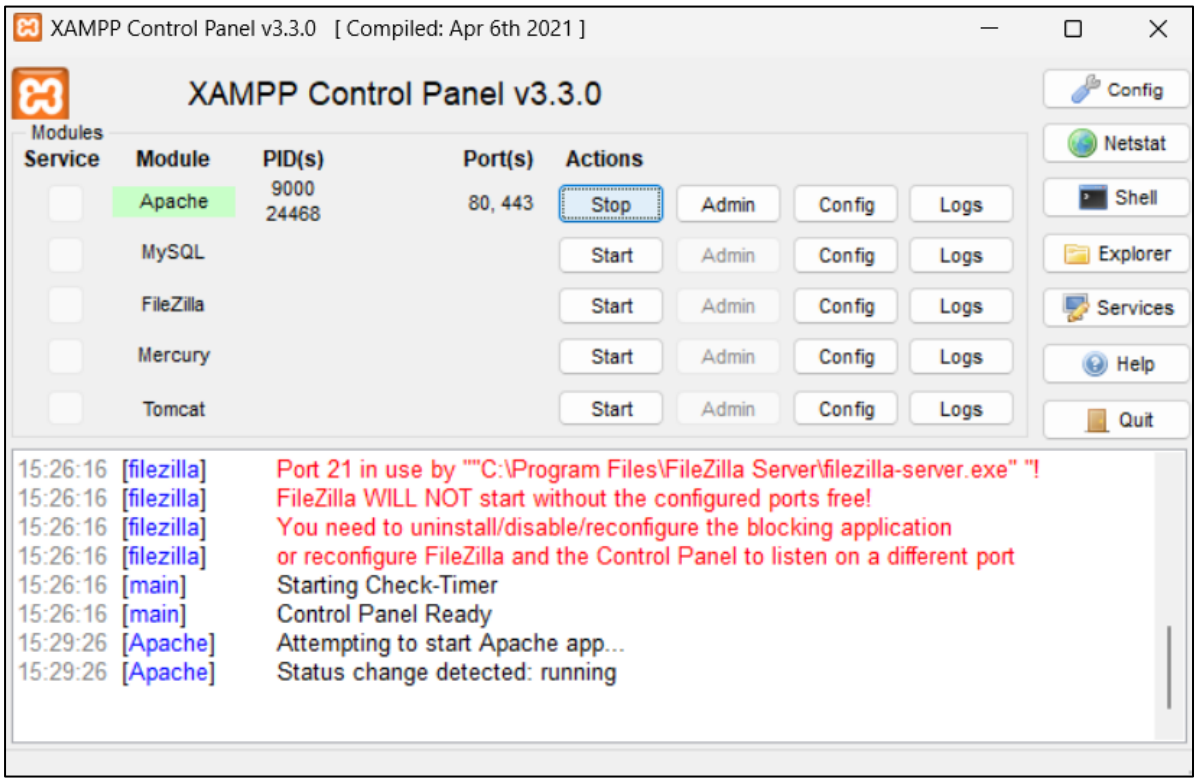


b)Here we have put the "wim lab2" folder which has the lab files.

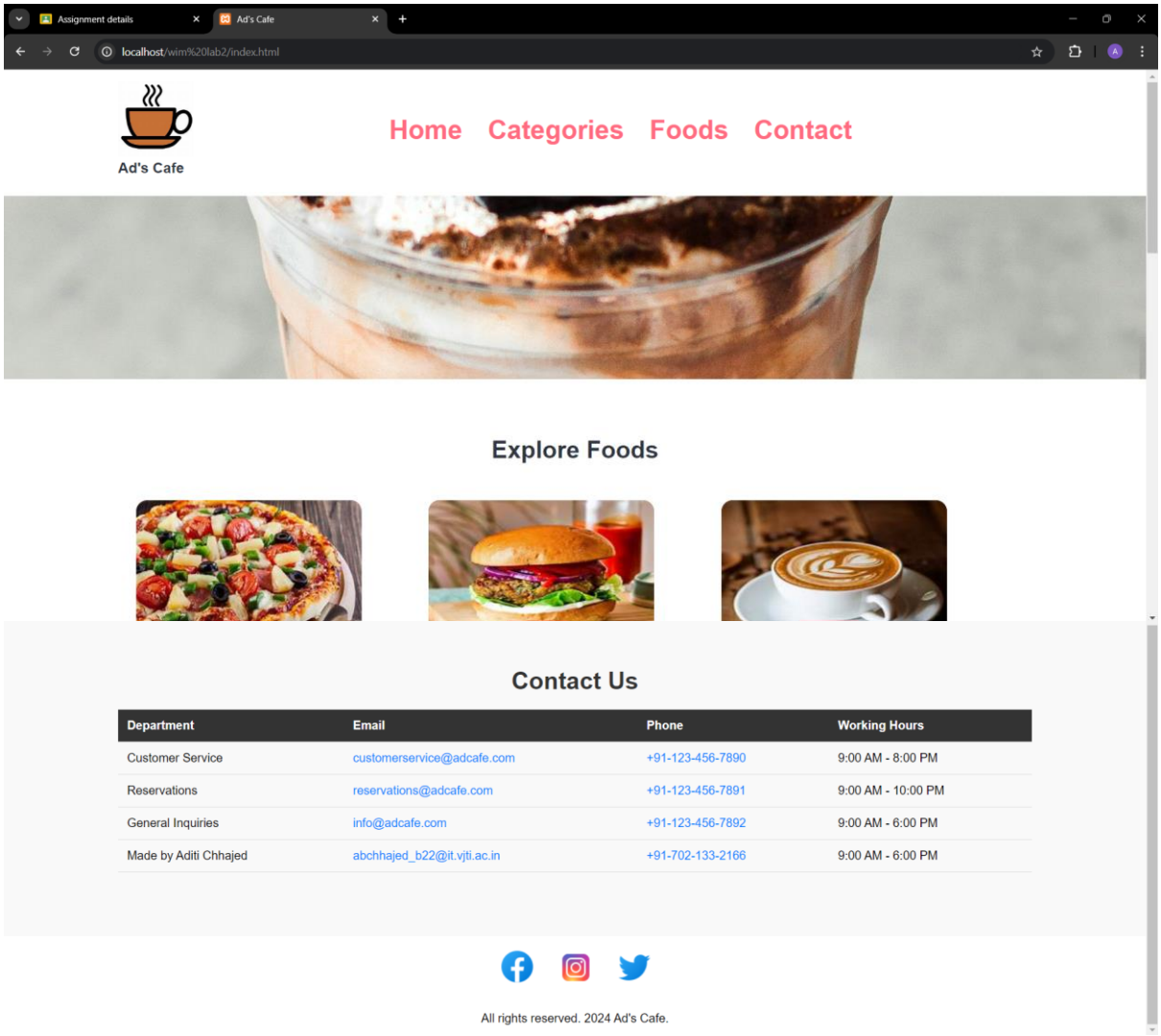


Name	Date modified	Type	Size
dashboard	21-08-2024 15:20	File folder	
img	21-08-2024 15:20	File folder	
webalizer	21-08-2024 15:20	File folder	
wim lab2	21-08-2024 15:29	File folder	
xampp	21-08-2024 15:20	File folder	
applications	15-06-2022 21:37	Chrome HTML Do...	4 KB
bitnami	15-06-2022 21:37	CSS Source File	1 KB
favicon.ico	16-07-2015 21:02	ICO File	31 KB
index	16-07-2015 21:02	PHP Source File	1 KB

c) Now we start the Apache server.



d) Now we go to http://localhost/wim%20lab2/



CONCLUDING REMARKS:

In conclusion, this experiment made us familiar with webservers, types of webserver architecture (single-tier and multi-tier), working of a webserver, and working of DNS (Domain Name System). Then, we installed Xampp which contains Apache Server and then deployed a webpage onto the Xampp Apache webserver.