

SUBJECT – Desktop Application Development-II

Faculty Name – Ankit Rami

Unit-1

Introduction of Client – Server Process

AMIT ACADEMY FOR COMPUTER EDUCATION

Nr Vardayini Mataji Temple, Rupal, Gandhinagar-382630

Email – amitacademy1117@gmail.com

Mobile No - 8460467193

YouTube Link – https://www.youtube.com/@ankitramijoinar

Instagram Link – https://www.instagram.com/amitacademy17/

Facebook Link –

https://www.facebook.com/aramitacademy/

What is Client-Server Architecture?

- The Client-server model is a distributed application structure that partitions task or workload between the providers of a resource or service, called servers, and service requesters called clients.
- ✓ In the client-server architecture, when the client computer sends a request for data to the server through the internet, the server accepts the requested process and deliver the data packets requested back to the client. Clients do not share any of their resources. Examples of Client-Server Model are Email, World Wide Web, etc
- Client Server Architecture is a computing model in which the server hosts, delivers, and manages most of the resources and services to be consumed by the client. This type of architecture has one or more client computers connected to a central server over a network or internet connection. This system shares computing resources. Client/server architecture is also known as a networking computing model or client/server network because all the requests and services are delivered over a network.

How the Client-Server Model works?

Client: When we talk the word Client, it mean to talk of a person or an organization using a particular service. Similarly in the digital world a Client is a computer (Host) i.e. capable of receiving information or using a particular service from the service providers (Servers).

 Servers: Similarly, when we talk the word Servers, It means a person or medium that serves something. Similarly in this digital world a Server is a remote computer which provides information (data) or access to particular services.

Diagram-1



Diagram-2

The Client-Server Model



Page | 3 Design By Ankit Rami Contact Details - +91 8460467193 | Email- amitacademy1117@gmail.com

4 Client server architecture example

- ✓ Here are some of the client server model architecture examples from our daily life. Hope it helps you to understand the concept better.
- Mail servers: Email servers are used for sending and receiving emails. There are different software that allow email handling.
- File servers: File servers act as a centralized location for files. One of the daily life examples to understand this is the files that we store in Google Docs. The cloud services for Microsoft Office and Google Docs can be accessed from your devices; the files that you save from your computer, can be accessed from your phone. So, the centrally stored files can be accessed by multiple users.
- Web servers: Web servers are high-performance computers that host different websites. The server site data is requested by the client through high-speed internet.
- Components of client server architecture:
 - Essentially, three components are required to make client server architecture work. The three components are workstations, servers, and networking devices. Let us, now, discuss them in detail:
 - Workstations: Workstations are also called client computers. Workstations work as subordinates to servers and send them requests to access shared files and databases. A server requests information from the workstation and performs several functions as a central repository of files, programs, databases,

and management policies. Workstations are governed by server-defined policies.

- Servers: Servers are defined as fast processing devices that act as centralized repositories of network files, programs, databases, and policies. Servers have huge storage space and robust memory to deal with multiple requests, approaching simultaneously from various workstations. Servers can perform many roles, such as mail server, database server, file server, and domain controller, in client server architecture at the same time.
- Networking devices: Now that we know about the roles that workstations and servers play, let us learn about what connects them, networking devices. Networking devices are a medium that connects workstations and servers in client server architecture. Many networking devices are used to perform various operations across the network. For example, a hub is used for connecting a server to various workstations. Repeaters are used to effectively transfer data between two devices. Bridges are used to isolate network segmentation.

How does client server architecture work?



Page | 5 Design By Ankit Rami Contact Details - +91 8460467193 | Email- amitacademy1117@gmail.com

<u>AMIT ACADEMY</u> Desktop Application Development-II

- The user enters the uniform resource locator (URL) of the website or file and the browser sends a request to the domain name system (DNS) server.
- The DNS server looks for the address of the web server and the DNS server responds with the IP address of the web server.
- After the DNS server responds, the browser sends over an HTTP or HTTPS request to the web server's IP, which was provided by the DNS server.
- The server then sends over the necessary files of the website.
- Finally, the browser renders the files and the website is displayed.

Types of client server architecture

1-tier architecture



✓ In this category of client server architecture, the architecture contains all kinds of settings, such as configuration setting and marketing logic, on a single device.

Page | 6 Design By Ankit Rami Contact Details - +91 8460467193 | Email- amitacademy1117@gmail.com

- ✓ While the diversity of services offered by 1-tier architecture makes it one of the reliable sources, handling such architecture is difficult. This is primarily due to the data variance.
- ✓ It often results in replication of work. 1-tier architecture consists of several layers, such as presentation layer, business layer, and data layer, that are combined with the help of a unique software package.
- ✓ The data present in this layer is usually stored in local systems or on a shared drive.

4 2-tier architecture



This architecture has the best environment. In this architecture, the user interface is stored on the client's side and the database is stored on the server, while database logic and business logic is maintained either on the client's side or on the server's side.

Page | 7 Design By Ankit Rami Contact Details - +91 8460467193 | Email- amitacademy1117@gmail.com

✓ The 2-tier architecture is faster in comparison to the 1tier architecture; this is because the 2-tier architecture does not have any intermediary between the client and the server. It is often utilized to avoid confusion between clients. One of the popular examples of 2-tier architecture is the online ticket reservation system.

3-tier architecture



✓ Unlike 2-tier architecture that has no intermediary, in 3tier client server architecture, a middleware lies between the client and the server. If the client places a request to fetch specific information from the server, the request will first be received by the middleware. It will then be dispatched to the server for further actions. The same pattern will be followed when the server sends a response to the client. The framework of 3-tier architecture is categorized into three main layers, presentation layer, application layer, and database tier.

✓ All three layers are controlled at different ends. While the presentation layer is controlled at the client's device, the middleware and the server handle the application layer and the database tier respectively. Due to the presence of a third layer that provides data control, 3tier architecture is more secure, has invisible database structure, and provides data integrity.





N-tier architecture is also called multi-tier architecture. It is the scaled form of the other three types of architecture. This architecture has a provision for locating each function as an isolated layer that includes presentation, application processing, and management of data functionalities.

Comparison of Client – Server? Client

- A client can be a device or a machine.
- A client program runs on the local machine, requesting service from the server. A client program is a finite program means that the service is started by the user and terminates when the service is completed. For instance, web browser.
- A client device is a machine that the end-user uses to access the web. Examples of clients are smartphones, desktops, laptops, etc.
- It is simple and less powerful and used for simple tasks. It has a basic hardware configuration. Clients are categorized into thin client, fat client, and hybrid client. Thin client is lightweight and relies on the resources of the host computer. A fat client (or thick client) lightly relies on the server and provides rich functionality. A hybrid client is the combination of the characteristics of a thin client and a thick client.

✓ Server

- A server is like a computer program, which is used to provide functionality to other programs. It can be any computerized process called by a client to distribute the work and share the resources.
- It receives and responds to requests made over a network. Server receives the request from the client for a web document, and it sends the requested information to the client's computer.

- A device can be both a client and a server at the same time, as an individual system has the ability to provide resources and use them from another system in one go. In a single machine, there can be multiple servers.
- Server has high efficiency and performance. Simultaneous multiple-user login and request processing are supported in servers. Some of the complex tasks like fulfilling client requests, storing and processing large datasets, data analysis are common for servers.



Based on	Client	Server
Basic functionality	Client relies on the services of server, and generates requests for various services.	Server authorizes the client's requests and facilitates them with the requested services.
Configuration	The configuration of client systems is simple. Their tasks are limited to generating requests. It has a basic hardware configuration.	The configuration of the server is more complex and sophisticated. Server has advanced hardware configuration.
Efficiency	The efficiency of client is limited.	The performance of server is high, and they are highly efficient.
Tasks	The common tasks for client are simple and mostly include requesting services.	The complex tasks like fulfilling client requests, storing and processing large datasets, data analysis are common for server.
Switch off	The client systems can be switch off without any fear.	Switching off servers may be disastrous for client systems that continuously request the services.
Login Support	There can be single user logins.	Server support multiple user login and request processing simultaneously.
Examples	Examples of clients are smartphones, desktops, laptops, etc.	Examples of servers are web servers, file servers, database servers, etc.

Characteristics of a server:

- \checkmark It is initially passive (or slave, waiting for a query);
- It is listening, ready to respond to requests sent by clients;
- ✓ When a request comes, he treats it and sends a response.

Characteristics of a client:

- ✓ It is the first active (or master);
- ✓ Sends requests to the server;
- \checkmark It expects and receives responses from the server.

Desktop base Client server process



- A Desktop Application is a software program that is used to run on a personal computer, machine, or laptop. Required applications or software are installed on a stand-alone machine and the installed application is accessible by only one user at a time. Hence, desktop application testing is also known as stand-alone application testing.
- ✓ Few desktop applications are MS-Office, Calculator, Adobe Acrobat Reader, etc.

Characteristics

- These applications are highly-developed to execute the applications on particular operating systems.
- No internet connectivity is required to work on desktop applications

Types of Testing to perform

- Below are the different testing techniques to be performed:
 - Usability Testing
 - Functional Testing
 - Load Testing (Back-end Testing)
 - Performance Testing
 - Security Testing
 - Regression Testing
 - Compatibility Testing (Software/Hardware)

4 Merits & Demerits

✓ Merits:

- Doesn't require any internet connection.
- Can be accessed very quickly.
- Hacking these applications is impossible because of high security.

✓ Demerits:

- Confined to only one user to access the desktop application at a time.
- It is not possible to share the data.
- High maintenance is required as these applications need some storage to install.

Client-Server Application Process



Page | 14 Design By Ankit Rami Contact Details - +91 8460467193 | Email- ankitramiblog@gmail.com

- ✓ As the name suggests, the Client-Server application consists of two systems, one is the Client and the other is the Server. Here, the client and server interact with each other over the computer network.
- ✓ In Client-Server application testing, the client sends requests to the server for specific information and the server sends the response back to the client with the requested information. Hence, this testing is also known as two-tier application testing.
- Few client-server applications are Email, Web Server, FTP, etc

Characteristics

- ✓ In the Client-Server applications, '.exe' file is available on each client machine whereas the application is loaded on the server machine.
- Internet connectivity is required to work on the Client-Server.
- The GUI feature is available on both client and serverside machines.

4Types of Testing to Perform

- ✓ Below are the different testing techniques to be performed:
- ✓ GUI Testing (UI Testing)
- ✓ Functional Testing
- ✓ Manual Testing
- ✓ Configuration Testing
- ✓ Load Testing(Back-end Testing)

Page | 15 Design By Ankit Rami Contact Details - +91 8460467193 | Email- ankitramiblog@gmail.com

- ✓ Performance Testing
- ✓ Interoperability Testing
- ✓ Compatibility Testing (Software/Hardware)

Merits & Demerits

Merits:

- ✓ All the data is located at the central location (Server), hence data is protected easily by providing access to authorized users.
- ✓ Data can be accessed efficiently even if the server is located away from the clients.

Demerits:

- ✓ If multiple clients send requests to the same server in parallel, it leads to congestion in the network because the server may get overloaded.
- ✓ Due to any reason, if a server fails then none of the client requests can be fulfilled and this leads to the disruption of the client-server network.
- The cost of setting up and maintaining the server is high.

Web Application Process



Page | 16 Design By Ankit Rami Contact Details - +91 8460467193 | Email- ankitramiblog@gmail.com

- Testing Web Applications is a bit different and complicated when compared with Desktop Application testing and Client-Server application Testing.
- As the name hints, any application which is accessed over the internet through a browser is known as a web application. Testing performed on these applications is known as Web Application testing.
 Web Applications are launched using a URL in a browser. Web testing is performed on different browsers and in different operating systems.
- ✓ Here browser sends the request to the web server and the web server sends the same request to the database and the database sends the response to the web server and the web server redirects the same response to the browser. Hence, this is also known as three-tier application testing.
- Few Web Applications are Yahoo, Facebook, Twitter, etc.

Characteristics

- \checkmark Internet connectivity is required to work on the web.
- \checkmark The GUI feature is available on Web Applications.
- Can be accessed from anywhere with no location constraints.

Types of Testing to Perform

- Below are the different testing techniques to be performed:
- ✓ GUI Testing (UI Testing)
- ✓ Functional Testing
- ✓ Manual Testing
- ✓ Security Testing
- ✓ Exploratory Testing
- ✓ Load Testing (Back-end Testing)
- ✓ Performance Testing
- ✓ Interoperability Testing
- ✓ Browser Compatibility Testing
- \checkmark Storage and data volume testing

Merits & Demerits

Merits:

- ✓ Platform independent.
- Improves user experience and makes sure all functionalities are working fine.
- Ensure a high-quality web application as an end result.
- \checkmark Can be accessed by multiple users at a time.

Demerits:

- ✓ Internet connectivity is required.
- ✓ Browser dependency.
- \checkmark Applications can't be accessed if the server is down.

Comparison Table

Desktop Application Testing	Client-Server Application Testing	Web Application Testing
Applications which are installed on single machine	Applications which are installed on both client and server	Applications which are URL - driven and executed on different web browsers
Desktop applications are accessed by only one user	Client-Server applications can be accessed by multiple users but limited number	Web applications can be accessed by any number of user without any limit across the internet
Testing is performed on stand-alone machine or single machine	Testing is performed on 2-tier application	Testing is performed on 3-tier application
Testing environment is Desktop	Testing environment is intranet	Testing environment is web browsers
No authentication required as only one user is accessing the application	Client-Server applications access restricted to limited users hence username/password is provided to access the application	Web applications can be accessed by all users
Examples of Desktop applications: MS Office, AutoCAD	Examples of Client-Server applications: Email, Web Server, FTP	Examples of Web applications: Facebook, Twitter

Concept of Front-End and Back-End Front-end

- ✓ Front-end development focuses on the user-facing side of a website.
- ✓ Front-end development focuses on the user-facing side of a website. Front-end developers ensure that visitors can easily interact with and navigate sites by using

Page | 19 Design By Ankit Rami Contact Details - +91 8460467193 | Email- ankitramiblog@gmail.com

programming languages, design skills, and other tools. They produce the drop-down menus, layouts, and designs for websites.

- Front-end Developers use computer programming languages like JavaScript, Hypertext Markup Language (HTML), and Cascading Style Sheets (CSS) to design websites. Each language serves a unique purpose.
 HTML lays out the site's content and structure, CSS adds design features, and JavaScript creates advanced interactive features. Less commonly, front-end Developers know other programming languages like Python, PHP, or Ruby.
- These developers need expertise using frameworks and libraries like jQuery, Bootstrap, AngularJS, and EmberJS. Frameworks ensure that content appears correctly on all devices, while libraries condense code into smaller, time-saving packages. Front-end Developers may also use graphic design programs like Photoshop or Sketch, along with code editing tools like Notepad or Eclipse.
- Front-end development requires technical expertise and creativity to ensure websites' user interfaces look correct and function properly. Front-end Developers work with back-end developers, designers, and user experience analysts.

 A front-end developer typically needs a bachelor's degree in web design, programming, computer science, or graphic design. Some individuals develop these skills through previous work, bootcamps, or projects to find employment without pursuing formal education.



Back-end

- ✓ Back-end developers focus on the server side of websites.
- ✓ Back-end developers focus on the server side of websites. They use technical skills to perform the behind-the-scenes work that creates a website's structure and overall functionality, allowing a site's front end to exist. These professionals create a site's

operations, databases, and application programming interface (API).

- The back end includes an application, server, and database. Computer users often do not see the elements of back-end development, which remain below the hood of a website.
- ✓ Back-end developers need advanced experience in server-side programming languages like Java, Python, and Ruby to build applications. Students can learn or brush up on specific computer programming languages by completing bootcamps. For example, see our page on the best Python bootcamps.
- ✓ Back-end devs use tools like SQL Server and Oracle to store, organize, and change data. Employers often require expertise with PHP frameworks, version control software, and debugging back-end systems and applications. Back-end devs collaborate with front-end developers, management, and business stakeholders to understand each project's goals.
- Most employers require back-end devs to hold bachelor's degrees in computer science, programming, or web development. Some back-end devs can find employment without earning four-year degrees by learning through relevant work experience or bootcamps.

 With additional training, education, or certifications, some back-end developers can translate their experience into higher-paying careers as software engineers. Below, we explain some skills that back-end developers usually possess.



Page | 23 Design By Ankit Rami Contact Details - +91 8460467193 | Email- ankitramiblog@gmail.com

Comparison of Front-end and Back-end

FRONTEND	VERSUS	BACKEND

Frontend	Backend
Frontend refers to the client-side of the application.	Backend refers to the server-side of the application.
It is the part of the website users can see and interact with.	It constitutes everything that happens behind the scenes.
It typically includes everything that attributes to the visual aspects of websites.	It generally includes a web server that communicates with a database to serve requests that the frontend presents.
It forms the basis of what users can touch and experience on their web browsers.	It is the brain of the website that is never visible to the end users.
The essentials of frontend web development include HTML, CSS, and JavaScript.	The essentials of backend development include Ruby, Python, Java, .Net, etc.
Front End • Markup and web languages such as HTML, CSS and Javascript • Asynchronous requests and Ajax • Specialized web editing software • Image editing • Accessibility • Cross-browser issues • Search engine optimisation	 Back End Programming and scripting such as Python, Ruby and/or Perl Server architecture Database administration Scalability Security Data transformation Backup

Page | 24 Design By Ankit Rami Contact Details - +91 8460467193 | Email- ankitramiblog@gmail.com

Comparison of Front-end and Back-end

Front-End Developer vs. Back-End Developer



Page | 25 Design By Ankit Rami Contact Details - +91 8460467193 | Email- ankitramiblog@gmail.com

Difference between peer-to-peer network and client server architecture

Client server architecture	Peer-to-peer architecture
It has specific clients and servers.	There is no differentiation between clients and servers.
It has centralized data management.	It has its own data and applications.
The purpose is to share information.	Its main goal is to maintain connection among peers.
Data is provided only in response to a request.	In this network, peers have the authority to request as well as provide a service.
It is suitable for small as well as large networks.	It is suitable for less users, less than 10 devices.

Advantages and disadvantages of client-server architecture

Advantages	Disadvantages
The centralized network has complete leverage to control the processes and activities.	If the primary server goes down, the entire architecture is disrupted.
All devices in the network can be controlled centrally.	It is expensive to operate because of the cost of heavy hardware and software tools.
Users have the authority to access any file, residing in the central storage, at any time.	This architecture requires particular OSs related to networking.
It provides a good user interface, easy file finding procedure, and management system for organizing files.	Too many users at once can cause the problem of traffic congestion.
Easy sharing of resources across various platforms is possible.	It requires highly technical stuff, such as server machines, for maintenance of the network.

Page | 26 Design By Ankit Rami Contact Details - +91 8460467193 | Email- ankitramiblog@gmail.com

4 Reference Link

- 1. <u>https://www.softwaretestinghelp.com/difference-between-desktop-client-server-and-web-testing/</u>
- 2. <u>https://intellipaat.com/blog/what-is-client-server-architecture/</u>
- 3. https://www.geeksforgeeks.org/frontend-vs-backend/

Any Query Contact Us

Faculty Name- Ankit Rami

Email – ankitramiblog@gmail.com

Contact No - +91 8460467193

Website - amit.arinfoway.com



Subscribe Our YouTube Channel

https://www.youtube.com/channel/UCWbJh2iQ8w-8nrU0Xpjpw7g

Page | 27 Design By Ankit Rami Contact Details - +91 8460467193 | Email- ankitramiblog@gmail.com