



1.3.4 Client-side vs Server-side Processing

Client-side processing

Client-side processing is when processing is carried out on the **client device**, usually within the user's web browser.

The web server sends files such as HTML, CSS and JavaScript to the client. The browser then processes them.

Examples

- Applying CSS styles to a webpage
- Running JavaScript
- Checking that a form has been completed
- Changing the content or appearance of a webpage
- Responding to user interaction

For example, JavaScript can check whether a user has entered a surname into a form before the form is submitted.

Advantages of client-side processing

- **Reduces server workload** because some processing is performed by the client's device.
- **Faster response** for some interactions because the browser does not need to contact the server.
- Can provide **interactive webpages**.
- Can continue to perform some processing without repeatedly communicating with the server.

Disadvantages of client-side processing

- The client's device may have **limited processing power**.
- JavaScript can be **disabled** by the user.
- Client-side code can potentially be **modified or bypassed** by the user.
- Processing is performed on the user's device rather than being centrally controlled.





Server-side processing

Server-side processing is when processing is carried out on the **web server**.

The client sends a request to the server. The server processes the request and sends the result back to the client.

Examples

- Accessing a database
- Checking whether a product is in stock
- Processing an online order
- Authenticating a username and password
- Processing sensitive information
- Generating webpage content from stored data

For example, when a customer checks whether an item is in stock, the server can query the database and return the result.

Advantages of server-side processing

- **Sensitive data can be kept on the server** rather than being sent to the client.
- Processing can be performed on powerful **server hardware**.
- Database access can be controlled centrally.
- Users cannot directly access or modify the server-side code.
- Processing is consistent because it is controlled by the server.

Disadvantages of server-side processing

- Creates **additional workload on the server**.
- Requires a **network connection** to communicate with the server.
- A large number of requests can slow the server down.
- The response may be slower because data has to travel between the client and server.

