

## 1. Front-end

사용자가 직접 보고, 상호 작용하는 웹 사이트나 애플리케이션의 일부분

## 2. Back-end

사용자가 직접 볼 수 없고, 상호 작용할 수 없는 웹 사이트나 애플리케이션의 일부분

## 3. Full-stack

Front-end와 Back-end를 모두 포함하는 웹 사이트나 애플리케이션의 전체 스택

## 4. DOM (Document Object Model)

웹 문서를 위한 프로그래밍 인터페이스로, 프로그램이 문서 구조, 스타일 및 내용을 변경할 수 있도록 록 페이지를 나타낸다.

## 5. API (Application Programming Interface)

프로그램이 쉽게 데이터에 접근하거나, 쉽게 데이터를 주고 받을 수 있도록 제공하는 것을 목적으로 한다.

## 6. Encryption

무단 액세스를 방지하기 위해 정보 또는 데이터를 코드로 변환하는 프로세스로, 암호화라고 한다.

## 7. Decryption

암호화된 코드를 다시 원래 형태로 변환하는 프로세스로, 복호화라고 한다.

## 8. HTTP (Hypertext Transport Protocol, Hypertext Transfer Protocol)

월드 와이드 웹(WWW)에서 사용되는 데이터 전송 프로토콜

## 9. HTTPS

기본적으로 HTTP와 동일하지만, 웹 페이지와 주고 받는 데이터를 보호하기 위해 암호화 방법을 사용한다.

## 10. CI (Continuous Integration) / CD (Continuous Delivery) 시스템

애플리케이션이나 웹 사이트에 코드 변경 사항을 보다 자주, 안정적으로 전달하기 위한 목적의 시스템

## 11. UI (User Interface)

UI 개발자는 애플리케이션이나 웹 사이트의 인터페이스를 쉽게 사용하고 탐색할 수 있도록 만들기 위해 노력한다.

## 12. UX (User Experience)

UX 개발자는 애플리케이션이나 웹 사이트의 디자인을 방문자, 이용자들에게 더 매력적으로 보일 수 있도록 하는 데 중점을 둔다.

## 13. Serverless computing

Back-end 서비스 사용에 제공되는 방법으로, Serverless 공급자는 사용자들에게 기본 인프라 구조에 대한 걱정 없이 코드를 작성하고 배포할 수 있도록 한다. 서버리스 공급자로부터 Back-end 서비스를 받는 회사는 서버리스 공급자의 계산에 따라 요금이 부과된다. 그리고 서비스가 자동으로 확장(Auto-Scaling)되기 때문에 고정된 양의 대역폭 또는 서버의 개수를 확보하기 위해서 비용 지불을 지불하거나 예약할 필요가 없다. 물리적인 서버는 여전히 사용되지만 개발자는 이를 인식할 필요가 없어진다.

## 14. Backend 작업

작업에서 요구하는 라이브러리들이나 프레임워크가 Back-end에서만 작동되어 Front-end에서 수행할 수 없거나 수행하기에는 시간이 많이 걸리거나 메모리를 많이 사용하게 되는 경우, Back-end에서 수행된다.

#### 15. Request / Response

Front-end와 Back-end가 서로 통신하는 방식으로 Front-end Script는 Back-end에 데이터를 요청(Request)하고, Back-end는 Front-end에 응답(Response)으로 데이터를 보내줄 수 있다.

#### 16. Cache

캐시는 필요할 때마다 동일한 데이터를 계속해서 로드하는 시간을 줄이기 위해 브라우저 또는 컴퓨터에 데이터를 임시로 저장하는 장소

#### 17. Command-line

컴퓨터 운영체제에 직접 명령을 입력하기 위한 인터페이스로, PHP 서버 시작, 디렉토리 생성, 스크립트 실행, 파일 권한 변경 등 운영 체제에서 작업을 수행하는 가장 직접적이고 빠른 방법

#### 18. VM (Virtual Machine)

가상머신으로, 컴퓨팅 환경을 소프트웨어로 구현하여 사용되는 컴퓨터 시스템이다. 즉, 컴퓨터 시스템을 에뮬레이션하는 소프트웨어라 할 수 있다.

물리적 하드웨어 시스템에 구축되어 자체 CPU, 메모리, 네트워크 인터페이스 및 스토리지를 갖추고 가상 컴퓨터 시스템으로 작동하는 가상 환경으로 웹 사이트 호스팅 또는 데이터 저장과 같은 작업을 수행할 수 있다.

#### 19. AWS (Amazon Web Services)

VM은 물론 파일 저장공간과 같은 다양한 서비스를 제공하는 현재 가장 큰 제공업체

#### 20. DB (Database)

구조화된 데이터의 모음으로, 주로 대용량의 사용자 데이터를 보관하고자 할 때 사용된다.

#### 21. XML (eXtensible Markup Language)

데이터를 전달(저장 및 전송)을 목적으로 설계된 도구로, 태그로 싸인 정보이다. 소프트웨어 및 하드웨어에 독립적이고, 데이터의 확장이 용이하기 때문에 인터넷을 통해 데이터를 배포하는 데 자주 사용되며, IT 시스템에서 중요한 역할을 한다.

#### 22. JSX (JavaScript XML)

JavaScript의 확장 문법으로, XML과 유사하게 생겼으며, 바벨을 사용하여 일반 JavaScript 형태의 코드로 변환된다. JSX를 사용하면 ReactJS에서 HTML을 작성할 수 있다.

#### 23. JSON (JavaScript Object Notation)

사람들이 읽을 수 있는 데이터 교환을 위해 명시적으로 설계된 가벼운 텍스트 기반의 개방형 표준 형식이다. 언어 독립적인 형식으로, 거의 모든 종류의 언어와 프레임워크, 라이브러리를 지원한다.

#### 24. Firewall

미리 정해진 보안 규칙에 따라 들어오고 나가는 트래픽을 모니터링하고 제어하는 네트워크 보안 시스템

#### 25. IP Address

인터넷 프로토콜을 사용하여 네트워크로 통신하는 각 컴퓨터를 식별하는 고유한 문자열

#### 26. IPv4 (IP version 4)

10진수 표기법으로 표현되는 32비트 정수로 가장 일반적인 IP 주소이다.

#### 27. IPv6 (IP version 6)

IPv4의 개선된 버전으로, 더 많은 주소를 제공할 뿐더러 네트워크 연결 공급자의 변경 시 라우터 알림, 네트워크 번호 재부여, 주소 구성을 단순화하는 기능도 가지고 있다.

## 28. Proxy Server

클라이언트와 서버 사이에서 중간 역할을 하는 서버로, 다른 IP 주소를 통해 웹 요청을 할 수 있다. VPN에서 주로 사용되며, 웹에서 자신의 위치와 같은 개인 정보 등을 숨기려는 경우에 유용하게 사용된다.

## 29. VPN (Virtual Private Network)

가상 사설망으로, 인터넷 연결과 온라인 개인 정보를 보호한다. 또한 그리고 Public Wi-Fi 핫스팟을 안전하게 사용할 수 있게 한다. 데이터를 위한 암호화된 터널을 생성하고, Proxy Server를 사용하여 IP 주소를 숨겨서 온라인 개인 정보를 보호하는 형태이다.

## 30. Nginx (engine-ex)

웹 요청을 처리하는 데 사용되는 오픈 소스 웹 서버 소프트웨어

## 31. Apache

Nginx와 유사한 웹 서버 소프트웨어로, 가장 널리 사용된다.

## 32. Jamstack

웹을 더 빠르고, 더 안전하고, 더 쉽게 확장할 수 있도록 설계된 아키텍처이다. ReactJS와 같은 개발자가 선호하는 많은 툴과 워크플로를 기반으로 만들어졌다. 사전 렌더링과 분리를 핵심 원칙으로 하여, 웹 사이트나 애플리케이션을 보다 훌륭히 제공할 수 있게 한다.

## 33. Sudo (SuperUser DO)

제한된 파일 및 작업에 액세스하는 데 사용된다. 기본적으로 Linux 시스템에서 사용자의 권한을 일시적으로 상승시켜 사용자가 루트 사용자로 로그인하지 않고도 민감한 작업을 완료할 수 있도록 한다.

## 34. Web Scraping

외부 웹 사이트에서 데이터를 추출하는 프로세스로, 가장 인기있는 웹 스크래핑 라이브러리에는 Puppeteer, Selenium, BeautifulSoup가 있다.

## 35. Git

파일 및 디렉토리의 변경 사항을 추적하기 위해 사용되며, 일반적으로 협업을 통해 소프트웨어를 개발할 때 개발자 간의 작업 조정에 사용된다.

## 36. Cron jobs

Back-end에 의해 일정 간격으로 실행되도록 예약된 작업으로, 주로 사용자가 Front-end를 통해 생성한 Back-end 작업을 관리하는 데 사용된다.

## 37. RWD (Responsive Web Design)

디스플레이 화면에서 자동으로 적용되어 모든 장치와 화면 크기에서 웹 사이트를 렌더링 또는 표시할 수 있는 현대적인 웹 디자인 접근 방식

## 38. MySQL

SQL(Structured Query Language)을 기반으로 오라클에서 개발한 관계형 데이터베이스 관리 시스템(RDBMS)이다.

## 39. NoSQL

높은 기능과 규모 맞춤형 성능으로 개발에 용이한 비관계형 데이터베이스이다. RDBMS의 규칙을 따르지 않으므로 기존에 사용되던 전형적인 SQL문을 사용하지 않는다. NoSQL 시스템에는 MongoDB와 Neo4J가 있다.

## 40. 환경 변수 (Environment variable)

일반적으로 운영 체제를 통해 프로그램 외부에서 값이 설정되는 변수이다. 이를 설정하면 프로그램을 통해 변수를 정의하고 재정의할 필요가 없다.

#### 41. React

가장 가장 인기있는 JavaScript Front-end 프레임워크. 페이스북에서 개발하였으며, 대화형 사용자 인터페이스를 구축하는 데 사용된다.

#### 42. Angular

TypeScript를 기반으로 하는 개발 플랫폼으로, 확장 가능한 웹 애플리케이션 구축을 위한 component 기반 프레임워크와 라우팅, Form 관리, 클라이언트-서버 통신 등 다양한 기능을 갖추고 있는 통합 라이브러리 그리고 코드 개발, 구축, 테스트, 업데이트를 위한 개발자 도구를 포함하고 있다.

#### 43. Docker

애플리케이션을 빠르게 구축, 테스트, 배포할 수 있는 소프트웨어 플랫폼이다. 컨테이너라는 표준화된 단위에 라이브러리, 시스템 도구, 코드, 런타임을 포함하여 소프트웨어를 패키징한다. 어떤 환경에도 애플리케이션을 빠르게 배포, 확장할 수 있으며 코드 실행 여부도 알 수 있다.

#### 44. Kubernetes (K8s)

구글에서 제공한 이식 가능하고, 확장 가능한 오픈 소스 플랫폼으로 컨테이너화 된 워크로드 및 서비스를 관리하기 위한 목적으로 사용된다.

#### 45. Rust

성능과 안전, 특히 동시성 안전과 메모리 관리를 위해 고안된 프로그래밍 언어로 정적으로 유형이 지정되어 있다.

#### 46. TensorFlow

머신 러닝을 위한 종단 간 (end-to-end) 오픈 소스 플랫폼이다. 포괄적이고 유연한 도구, 라이브러리, 커뮤니티 리소스를 가짐에 때문에 최첨단의 머신 러닝 기반 애플리케이션을 쉽게 구축하고 배포할 수 있다.

#### 47. Solidity

정적으로 유형이 지정된 중괄호 프로그래밍 언어로 이더리움에서 실행되는 블록체인 기반 계약 개발을 위해 설계되었다.

#### 48. Laravel

PHP에서 가장 널리 사용되는 프레임워크이다.

#### 49. Go

구글에서 처음 개발한 프로그래밍 언어로, C와 유사한 구문을 가진 정적 언어이다. 가비지 컬렉션, 데이터 타입 안전성, 동적 데이터 타입 지정은 물론 가변 길이 배열, 맵과 같은 많은 내장 데이터 타입까지 제공한다. 또한 풍부한 표준 라이브러리를 제공한다.

#### 50. jQuery

빠르고 작고, 기능이 풍부한 JavaScript 라이브러리이다. 여러 브라우저에서 작동하고 사용도 쉬운 API를 사용하여, HTML 문서 탐색 및 조작, 이벤트 처리, 애니메이션 및 AJAX와 같은 작업을 훨씬 더 간단하게 만든다.

#### 51. Ajax (Asynchronous JavaScript And XML)

서버에 데이터를 로드하고 웹 페이지의 일부를 선택적으로 업데이트하는 수단이다. 기본적으로 Ajax는 브라우저에 내장되어 있는 XMLHttpRequest(XHR) 객체를 이용하여, 페이지를 막거나 사용자의 이용을 방해하지 않고 웹 서버와 비동기적으로 정보를 주고 받는다. Ajax는 매우 대중적이며 이를 사용하지 않는 애플리케이션을 찾을 수 없을 정도이다. 대규모 Ajax 기반 온라인 애플리케이션의 대표적인 예는 Youtube, Facebook, Google Docs, Gmail 등이 있다.

#### 52. NodeJS

V8 엔진에서 실행되고, 웹 브라우저 외부에서 JavaScript 코드를 실행하는 오픈 소스, 크로스 플랫폼, Back-end JavaScript 런타임 환경이다. 개발자가 JavaScript를 사용하여 명령줄 도구를 작성하고, 페이지가 사용자의 웹 브라우저로 전송되기 전에 서버 측 스크립트에서 동적 웹 페이지 콘텐츠를 생성할 수 있게 한다. 이로써 서버 측과 클라이언트 측에 다양한 프로그래밍 언어를 사용하지 않고 JavaScript라는 단일 언어를 이용하여 웹 애플리케이션을 개발할 수 있도록 통합하는 “JavaScript Everywhere” 패러다임을 나타내게 된다.

#### 53. NPM

NPM에는 두 가지가 있다. 첫 번째는 오픈 소스인 Node.js 프로젝트 게시를 위한 온라인 저장소이다. 이는 패키지 설치, 버전 관리 및 종속성 관리를 지원한다. 두 번째는 이 온라인 저장소와 상호작용하기 위한 명령줄 유틸리티이다. 수많은 Node.js 라이브러리와 애플리케이션이 npm에 게시되며 계속해서 더 많은 것들이 추가되고 있다. 설치하려는 패키지가 있는 경우 단일 명령으로 쉽게 설치할 수 있다.

#### 54. CSS (Cascading Style Sheets)

HTML과 같은 마크업 언어로 작성된 문서의 표현을 위해 사용되는 언어이다. HTML, JavaScript와 함께 World Wide Web의 초석 기술이 된다. 레이아웃, 색상, 글꼴을 포함하고 있으며, 화면과 내용을 분리할 수 있도록 설계되었다. 이러한 분리는 더 많은 유연성과 제어를 제공한다. 관련 CSS를 별도의 파일에 지정하여 여러 웹 페이지가 형식을 공유할 수 있도록 하여, 복잡성과 반복을 줄이고, 페이지 로드 속도까지 개선한다.

#### 55. Git

모든 파일들의 변경 사항을 추적하는 데 사용되는 소프트웨어로, 일반적으로 소프트웨어 개발 과정에서 공동으로 개발하는 프로그래머 간의 작업 조정에 사용된다.

#### 56. AWS (Amazon Web Services)

개인, 회사 및 정부에 종량제 방식으로 주문형 클라우드 컴퓨팅 플랫폼과 API를 제공하는 Amazon의 자회사이다. 클라우드 컴퓨팅 웹 서비스는 다양한 추상 기술 인프라와 분산 컴퓨팅을 제공한다. 대표적인 서비스는 EC2(Amazon Elastic Compute Cloud)로, 사용자는 인터넷을 통해 가상 컴퓨터 클러스터를 마음대로 사용할 수 있게 된다.

#### 57. Google Cloud

GCP(Google Cloud Platform)은 Google에서 제공하는 클라우드 컴퓨팅 서비스로, Google 검색, Gmail, Google Drive, Youtube와 동일한 인프라에서 동작한다. 또한 컴퓨팅, 데이터 저장 및 분석, 머신 러닝을 포함한다.

#### 58. Microsoft Azure

Microsoft에서 운영하는 클라우드 컴퓨팅 서비스로, Microsoft의 데이터 센터 내 애플리케이션 관리를 목적으로 한다. SaaS(Software as a Service), PaaS(Platform as a Service), IaaS(Infrastructure as a Service)를 제공하고 타사 소프트웨어 및 시스템을 포함한 다양한 프로그래밍 언어, 도구, 프레임워크를 지원한다.

#### 59. Selenium

Python에서 웹 브라우저와의 상호 작용을 자동화하는 데 사용되는 패키지이다. 주로 봇을 만들거나, 웹 스크래퍼를 만드는 데 사용된다.

#### 60. Django

무료로 제공되는 Python 웹 프레임워크로, 빠른 개발과 깨끗하고 실용적인 디자인을 추구한다. 번거로운 웹 개발을 대부분 처리하므로 애플리케이션 작성에 집중할 수 있게 한다.