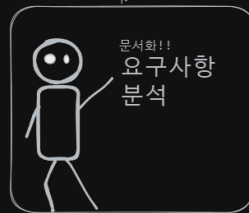


# 웹개발의 기본

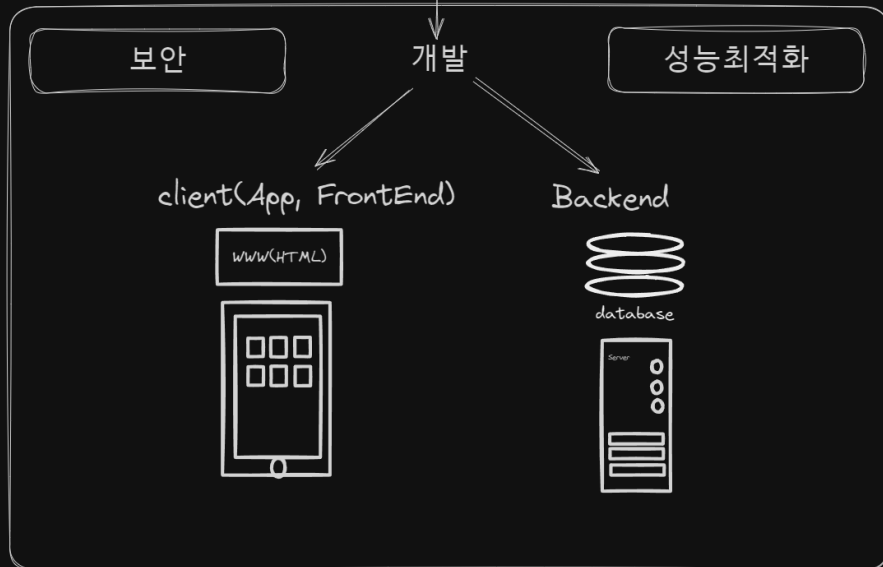
## 1. 웹개발 이해

#그림으로이해



요구사항  
문서화

시스템 설계  
문서화



배포

문서화, 교육

## 1. 요구 사항 분석 및 명세화:

1. 고객 또는 프로젝트 이해 관계자와의 소통
2. 요구 사항을 이해
3. 문서화
4. 명확한 개발 목표설정

## 2. 시스템 설계 및 아키텍처:

1. 시스템의 전체 구조를 설계
2. 사용할 기술 스택을 결정
3. 데이터베이스 및 서버 구조를 설계

## 3. 개발

### 1. 프론트엔드 개발:

1. 웹 애플리케이션의 사용자 인터페이스(UI)를 개발
2. HTML, CSS, JavaScript 등을 사용
3. 웹 페이지를 디자인하고 구현

### 2. 백엔드 개발:

1. 서버 측의 로직을 개발
2. 데이터베이스와의 상호 작용을 관리
  1. 데이터베이스 설계 및 구축
  2. 데이터의 관리 및 유지보수를 담당

## 4. 테스트 및 디버깅:

1. 코드를 테스트
2. 버그를 찾아 수정
3. 소프트웨어의 품질관리

## 5. 보안:

1. 웹 애플리케이션의 보안을 유지
2. 사용자 데이터를 안전하게 관리
  1. 보안 프로토콜 및 방어 기술을 구현
  2. 사용자 정보관리

## 6. 성능 최적화:

1. 웹 사이트 또는 애플리케이션의 성능을 향상
  1. 로딩 시간을 줄이기
  2. 데이터 처리 속도를 향상

## 7. 지속적 통합 및 배포 (CI/CD):

1. 코드의 변경 사항을 지속적으로 통합

2. 테스트
3. 안정적인 배포
4. 유지보수

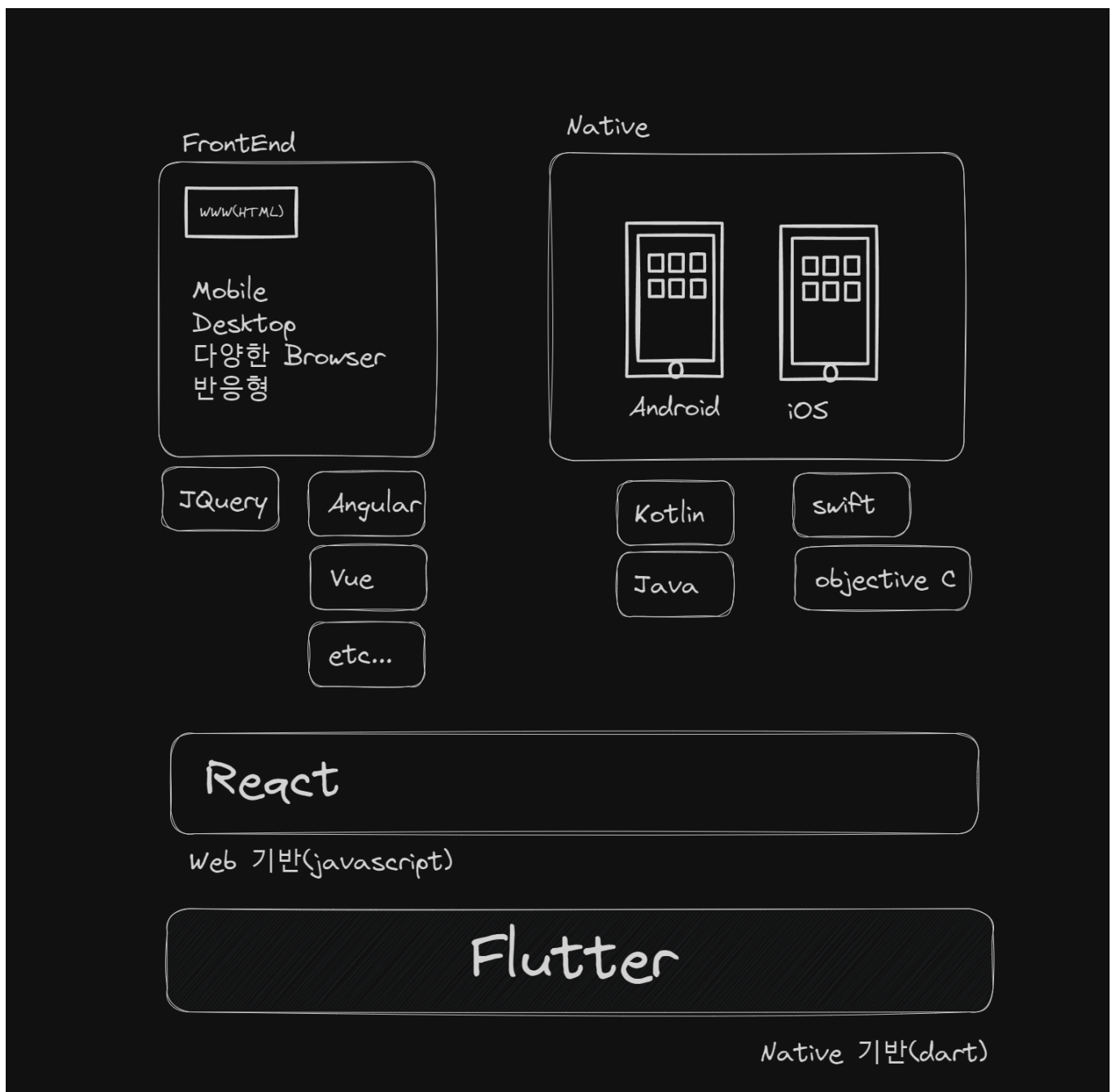
#### 8. 문서화 및 협업:

1. 코드, 시스템 구조, API 등을 문서화
2. 유지보수 및 협업을 위한 교육

위의 내용은 웹 개발 프로젝트에서 매우 중요한 업무단위이다.

#### 2. 클라이언트 개발의 이해

#그림으로이해



프론트엔드 개발은 주로 웹을 이야기하며 클라이언트에는 "앱"도 포함된다. 웹이나 앱이나 사용자가 직접 상호작용하는 부분으로, 웹 애플리케이션의 사용자 인터페이스(UI)와 사용자 경험(UX)을 개발하는 것을 말한다.

## 특징:

### 1. 사용자 인터페이스(UI) 개발:

1. HTML, CSS, JavaScript 등을 사용
2. 웹 페이지의 구조, 스타일, 동작을 개발
3. 또는 Android, iOS와 같은 단말기의 Native 기능으로 구현
4. 웹은 속도가 느리나 유연하고 앱은 유연하지 못하나 막강한 기능을 제공

### 2. (웹)브라우저 호환성 관리:

1. 다양한 브라우저에서 일관된 사용자 경험을 제공
2. 크로스 브라우징 이슈를 해결해야 함

### 3. (웹)반응형 및 모바일 최적화:

1. 다양한 디바이스 및 화면 크기에 대응하여 반응형 웹 디자인을 구현
2. 모바일 기기에 최적화된 경험을 제공

### 4. 웹 성능 최적화:

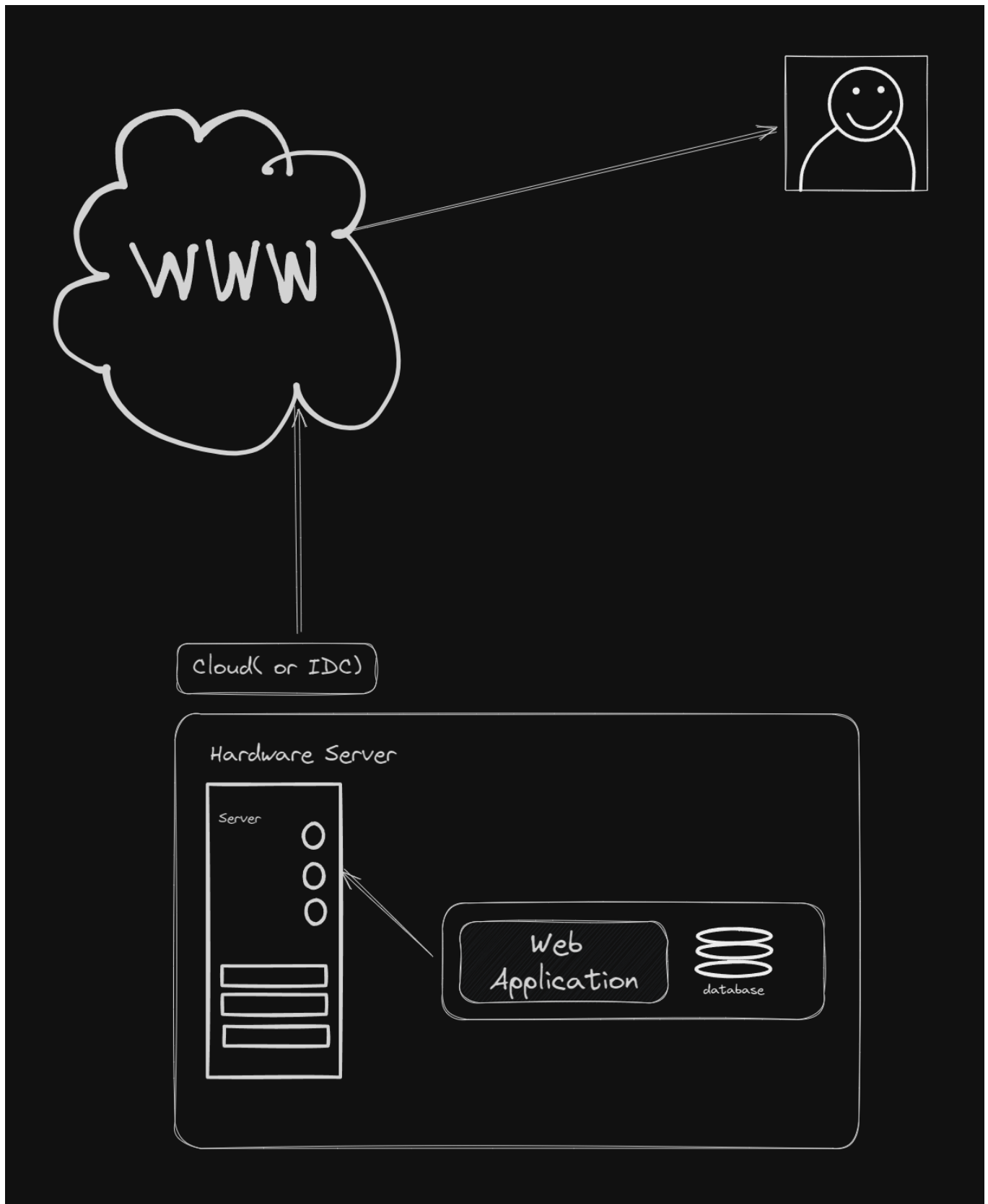
1. 웹 페이지의 로딩 시간을 최소화
2. UX 최적화 작업

### 5. 앱 개발(Native):

1. iOS와 Android는 별개의 프로그램
2. 두 개의 플랫폼을 한 개의 개발환경에서 개발하는 솔루션도 있음
  1. React Native(하이브리드)
  2. Flutter(네이티브)
3. Native 프로그래밍은 속도와 기능 면에서 압도적이다.
4. 시간이 오래 걸린다.
5. 개발자 구하기 쉽지 않다.

### 3. 서버 개발의 이해

#그림으로이해



백엔드 개발은 웹 애플리케이션의 서버 측에서 동작하는 부분으로, 데이터 처리, 비즈니스 로직, 보안 등을 담당한다.

### 1. 서버 측 로직 개발:

1. 사용자 요청에 대한 데이터 처리

2. 비즈니스 로직을 구현합니다.

**2. 데이터베이스 관리:**

1. 데이터의 저장, 검색, 수정, 삭제 등을 관리
2. 데이터베이스 시스템을 설계하고 조작합니다.

**3. API 개발:**

1. 프론트엔드와의 상호작용을 위한 API를 설계
2. 개발
3. 클라이언트와 데이터 통신을 처리

**4. 인증 및 보안:**

1. 사용자 인증
2. 권한 부여
3. 데이터 보호
4. 웹 애플리케이션의 안전성을 유지

**5. 성능 및 확장성:**

1. 서버의 성능을 최적화
2. 대규모 트래픽에 대응

서버 프로그래밍이 가장 중요함. 서비스의 뼈대이자 심장임

**4. 서버의 종류**

웹 서버의 종류:

**1. Apache HTTP Server (Apache):**

1. 가장 널리 사용되는 오픈 소스 웹 서버
2. 다양한 OS에서 동작

**2. Nginx:**

1. 경량이면서 높은 성능을 제공하는 웹 서버
2. Node.js나 Python으로 개발한 웹 사이트에서 필수
3. 빠른 응답 속도를 위해 사용

**3. Microsoft Internet Information Services (IIS):**

1. 마이크로소프트에서 개발한 웹 서버
2. Windows 서버 환경

**4. Lighttpd:**

1. 경량이면서 빠른 속도를 제공하는 웹 서버
2. 고성능 및 확장성이 필요한 경우에 사용

DBMS의 종류:

### 1. MySQL:

1. 가장 널리 사용되는 오픈 소스 관계형 데이터베이스 관리 시스템(RDBMS)
2. 간단한 웹 사이트
3. 대규모 웹 사이트

### 2. PostgreSQL:

1. 고급 기능과 확장성을 제공하는 오픈 소스 RDBMS
2. 데이터의 안정성과 일관성

### 3. MongoDB:

1. NoSQL 데이터베이스
2. 유연한 스키마 디자인과 확장성
3. 대용량 및 실시간 데이터 처리

### 4. Microsoft SQL Server:

1. 마이크로소프트가 개발한 RDBMS
2. Windows 환경에서 주로 사용
3. 기업용 애플리케이션에 적합

클라우드 서비스의 종류:

### 1. Amazon Web Services (AWS):

1. 아마존이 제공하는 클라우드 컴퓨팅 플랫폼
2. 컴퓨팅, 스토리지, 데이터베이스, 네트워크 등 다양한 서비스를 제공

### 2. Microsoft Azure:

1. 마이크로소프트의 클라우드 컴퓨팅 플랫폼
2. 인프라스트럭처부터 플랫폼 서비스까지 다양한 서비스를 제공

### 3. Google Cloud Platform (GCP):

1. 구글의 클라우드 컴퓨팅 플랫폼으로
2. 컴퓨팅, 스토리지, 빅데이터, 기계 학습 등의 서비스를 제공

### 4. IBM Cloud:

1. IBM이 제공하는 클라우드 컴퓨팅 플랫폼
2. 하이브리드 및 멀티 클라우드 환경 지원

### 5. 웹개발 기본단어 및 FAQ

서버 개발자와 소통을 위한 단어상식

용어	설명
HTTP	하이퍼텍스트 전송 프로토콜, 웹 서버와 클라이언트 간의 통신에 사용
URL	통합 자원 식별자, 웹 리소스의 위치 및 식별 정보를 나타내는 주소



용어	설명
<b>GET</b>	HTTP 메서드 중 하나로, 서버로부터 정보를 요청하는 역할
<b>POST</b>	HTTP 메서드 중 하나로, 서버에 정보를 제출하거나 전송하는 역할
<b>Response Code</b>	서버의 응답 상태를 나타내는 HTTP 상태 코드
<b>Header</b>	HTTP 요청 또는 응답에 대한 부가 정보를 포함하는 부분
<b>Body</b>	HTTP 요청 또는 응답의 본문으로, 실제 데이터가 담겨 있는 부분
<b>Request</b>	클라이언트가 웹 서버에 보내는 HTTP 요청 메시지
<b>Response</b>	웹 서버가 클라이언트에게 보내는 HTTP 응답 메시지
<b>Cookie</b>	클라이언트 측에서 서버에게 전송되는 정보를 저장하는 작은 데이터 조각
<b>Session</b>	클라이언트와 서버 간의 상태를 유지하기 위한 일시적인 데이터 저장 공간
<b>SSL/TLS</b>	보안 소켓 계층 및 전송 계층 보안으로, 데이터 암호화 및 보호를 제공하는 프로토콜
<b>WebSocket</b>	양방향 통신을 지원하는 프로토콜로, 실시간 데이터 전송에 사용
<b>API</b>	응용 프로그램 간 인터페이스로, 데이터 교환 및 상호 작용을 위한 규약
<b>REST</b>	Representational State Transfer, 웹 아키텍처의 한 형식으로, 자원을 표현하고 상태를 전송하기 위한 규칙
<b>JSON</b>	JavaScript Object Notation, 경량의 데이터 교환 형식으로, 클라이언트와 서버 간에 데이터를 교환하는 데 자주 사용
<b>AJAX</b>	Asynchronous JavaScript and XML, 비동기적으로 서버와 통신하여 웹 페이지를 동적으로 업데이트하는 기술

## 프론트 엔드 개발자와 소통을 위한 필수용어

용어	설명
<b>HTML</b>	HyperText Markup Language, 웹 페이지의 구조와 콘텐츠를 정의하는 마크업 언어
<b>CSS</b>	Cascading Style Sheets, 웹 페이지의 스타일 및 레이아웃을 디자인하는 스타일 시트 언어
<b>JavaScript</b>	웹 페이지의 동적 동작을 담당하는 스크립트 언어
<b>DOM</b>	Document Object Model, HTML 및 XML 문서의 구조화된 표현으로, JavaScript와 같은 스크립트 언어로 접근할 수 있는 인터페이스를 제공
<b>AJAX</b>	Asynchronous JavaScript and XML, 비동기적으로 서버와 통신하여 웹 페이지를 동적으로 업데이트하는 기술
<b>Responsive Design</b>	반응형 디자인, 다양한 디바이스 및 화면 크기에 대응하여 웹 페이지의 레이아웃을 조정하는 디자인 기법

용어	설명
<b>UI/UX</b>	User Interface/User Experience, 사용자 인터페이스 및 사용자 경험을 개선하기 위한 디자인 및 개발
<b>Framework</b>	개발자가 더 빠르고 효율적으로 웹 애플리케이션을 개발하기 위한 도구나 라이브러리 집합
<b>Library</b>	특정 기능을 수행하기 위한 코드나 함수의 집합으로, 개발자가 재사용할 수 있도록 제공됨
<b>Responsive Design</b>	반응형 디자인, 다양한 디바이스 및 화면 크기에 대응하여 웹 페이지의 레이아웃을 조정하는 디자인 기법
<b>Cross-Browser Compatibility</b>	다양한 웹 브라우저에서 웹 페이지가 일관된 모습으로 동작하는 것을 보장하는 기술
<b>Accessibility</b>	웹 콘텐츠에 대한 접근성을 고려하여 모든 사용자가 정보에 쉽게 접근할 수 있도록 하는 디자인 및 개발 원칙
<b>SEO</b>	Search Engine Optimization, 검색 엔진 최적화로, 웹 페이지가 검색 결과에서 높은 순위를 차지하도록 하는 기술 및 전략
<b>Version Control</b>	코드 변경 이력을 관리하고 협업을 용이하게 하는 도구나 시스템
<b>Git</b>	분산 버전 관리 시스템으로, 코드의 변경 사항을 추적하고 협업을 관리하는 데 사용됨
<b>Repository</b>	Git에서 코드와 이력을 저장하는 저장소로, 여러 개발자가 함께 작업할 수 있는 공간
<b>Pull Request</b>	변경된 코드를 병합하기 위해 다른 개발자에게 검토를 요청하는 GitHub 등의 협업 플랫폼에서 사용되는 용어

백엔드 프로그래밍에 사용되는 개발환경(언어)

언어	설명
<b>Java</b>	대규모 엔터프라이즈 애플리케이션에 많이 사용되며, 안정성과 확장성이 뛰어남
<b>Python</b>	간결한 문법과 다양한 라이브러리로 빠르게 개발할 수 있으며, 데이터 과학 및 웹 개발에 널리 사용
<b>PHP</b>	동적 웹 페이지를 생성하는 데 주로 사용되며, 대표적으로 WordPress 등의 CMS에 사용됨
<b>Node.js</b>	JavaScript 런타임 환경으로, 비동기적인 이벤트 처리를 기반으로 한 빠른 웹 서버 개발에 적합
<b>C#</b>	Microsoft의 개발 플랫폼인 .NET Framework를 사용하여 웹 애플리케이션을 개발하는 데 사용
<b>Go</b>	Google에서 개발한 프로그래밍 언어로, 높은 성능과 병행성을 제공하여 백엔드 개발에 사용됨 - 블록체인에서 많이 사용됨

프론트엔드 프로그래밍에 사용되는 개발환경(언어)

언어	설명
<b>HTML</b>	웹 페이지의 구조를 정의하는 마크업 언어
<b>CSS</b>	웹 페이지의 스타일과 레이아웃을 디자인하는 스타일 시트 언어
<b>JavaScript</b>	웹 페이지의 동적 동작을 담당하는 스크립트 언어
<b>TypeScript</b>	JavaScript에 타입 시스템을 추가한 프로그래밍 언어로, 코드의 안정성을 향상 시킴
<b>SCSS/Sass</b>	CSS의 확장된 문법으로, 변수, 중첩 규칙, 믹스인 등을 제공하여 CSS 코드를 더욱 효율적으로 작성할 수 있게 함
<b>Vue.js</b>	사용자 인터페이스를 구축하기 위한 프로그레시브 프레임워크로, 반응형 및 단일 페이지 애플리케이션을 개발하기 위해 사용됨
<b>React</b>	사용자 인터페이스를 구축하기 위한 자바스크립트 라이브러리로, 페이스북에서 개발되었으며, 단일 페이지 애플리케이션을 위한 UI 구성 요소를 제공함
<b>Angular</b>	Google에서 개발한 프레임워크로, 단일 페이지 애플리케이션을 개발하기 위한 기능 및 도구를 제공함
<b>Bootstrap</b>	반응형 웹 디자인 및 프론트엔드 개발을 위한 가장 인기 있는 HTML, CSS, JavaScript 프레임워크 중 하나
<b>Tailwind CSS</b>	구성 가능한 CSS 프레임워크로, 클래스 기반 디자인 시스템을 제공하여 HTML 마크업과 CSS를 더욱 유연하게 관리할 수 있음