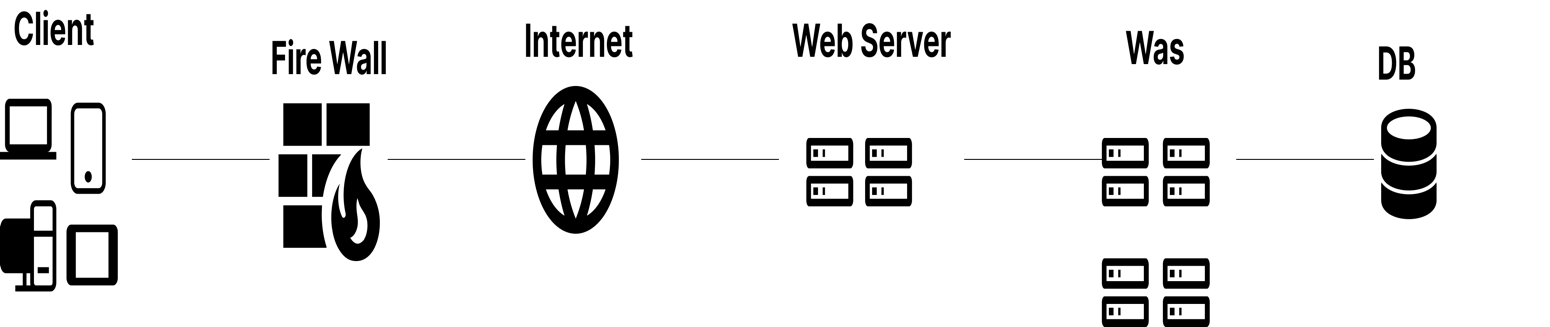
**TERM PROJECT 1**

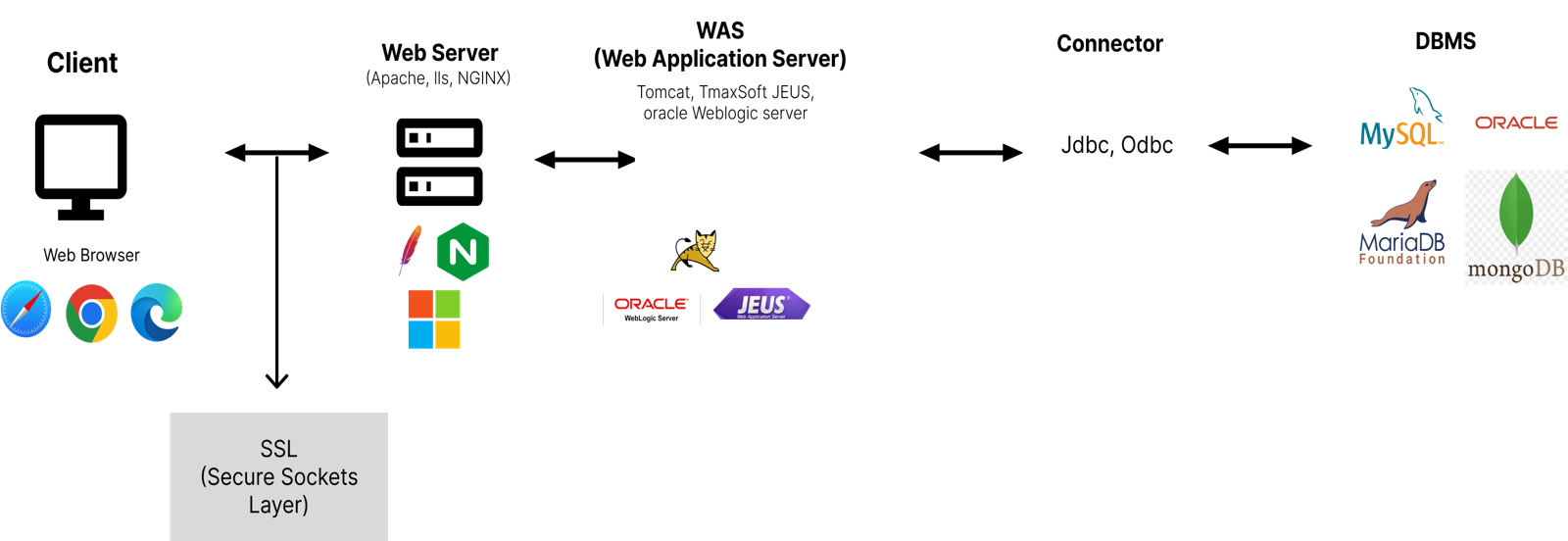
|  |  |
| --- | --- |
| **사 원 명** | **석하림** |
| **입 사 일** | **2023.07.17** |
| **담 당 자** | **김민진 주임** |
| **수습 기간** | **2023. 07. 17 ~ 2023. 10 16** |

**목차**

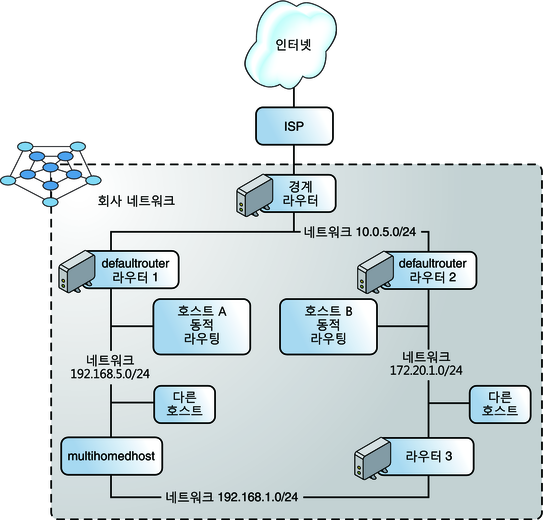
1. **WEB 시스템 구성도**
   1. **H/W 구성도**
   2. **S/W 구성도**
   3. **N/W 구성도**
   4. **웹 시스템 기술요소**
2. **WEB 시스템 기술요소 설명**
   1. **클라이언트측 기술요소**
   2. **서버측 기술요소**
3. **WEB 시스템 구조와 논리**
   1. **WEB 시스템 구조**
   2. **WEB 시스템 논리**
4. **WEB 시스템 개발 방법론**
   1. **정의**
   2. **정보공학 방법론**
   3. **객체지향 개발론**
   4. **CBD 개발론**
   5. **애자일 개발론**
5. **WEB 시스템 구성도**
   1. **H/W 구성도**



* 1. **S/W 구성도**



* 1. **N/W 구성도**



* 1. **웹 시스템 기술요소**

|  |  |
| --- | --- |
| 기술요소 | 내용 |
| Internet | 인터넷(Internet)이란 여러 통신망을 하나로 연결한다는 의미의 ‘인터 네트워크(inter-network)’라는 말에서 시작되었으며, 이제는 전 세계 컴퓨터들을 하나로 연결하는 거대한 컴퓨터 통신망을 의미한다.  인터넷은 클라이언트와 서버로 구성됭며, TCP/IP라는 기본 통신 프로토콜을 이용해 정보를 주고받는 컴퓨터 네트워크이다.  인터넷은각각 클라이언트와 서버로써 서로 연결되어 구성된 망인 컴퓨터 네트워크가 많이 모여서 이루어진 일종의 컴퓨터 네트워크 시스템이다.  즉, 인터넷이란 수많은 클라이언트 컴퓨터와 서버 컴퓨터, 그리고 이들로 구성된 네트워크들의 집합체이다. |
| Client | 클라이언트는 네트워크를 통하여 서버라는 다른 컴퓨터 시스템 상의 원격 서비스에 접속할 수 있는 응용 프로그램이나 서비스를 이용하는 사용자, 혹은 그 사용자가 사용하는 기기를 말한다. |
| Web Browser | 인터넷을 통해 검색을 할 때 주로 네이버나 다음과 같은 사이트에 접속하여 필요한 정보를 검색한다. 이 때 사이트에 접속할 수 있는 도구를 바로 '브라우저' 혹은 '웹 브라우저' 라고 부른다. 예로는 Mozilla, Firefox, Google Chrome, Microsoft Edge, Apple Safari 가 있다. |
| Server | 서버는 클라이언트에게 네트워크를 통해 정보나 서비스를 제공하는 컴퓨터 시스템으로 컴퓨터 프로그램 또는 장치를 의미한다. 쉽게 말하면 '요청을 받으면 데이터를 보내주는 기계'이다. |
| Web Server | 하드웨어 측면에서, 웹 서버는 웹 서버의 소프트웨어와 website의 컴포넌트 파일들을 저장하는 컴퓨터이다. 컴포넌트 파일에는 HTML 문서, images, CSS stylesheets, 그리고 JavaScript files가 있다. 웹 서버는 인터넷에 연결되어 웹에 연결된 다른 기기들이 웹 서버의 데이터(컴포넌트 파일들)를 주고받을 수 있도록 한다.  소프트웨어 측면에서, 웹 서버는 기본적으로 웹 사용자가 어떻게 호스트 파일들에 접근하는지를 관리한다. 가장 대표적인 HTTP 서버는 URL(Web addresses)과 HTTP(당신의 브라우저가 웹 페이지를 보여주기 위해 사용하는 프로토콜)의 소프트웨어 일부입니다.  대표적인 웹 서버로 프로그램에는 아파치, 마이크로소프트 lls, Nginx, Google Web Server, Zeus가 있다. |
| Was | 웹 서버는 정적인 컨텐츠를 클라이언트에게 제공하는 서버이다. (WAS)Web Application Server란 DB 조회 혹은 다양한 로직 처리를 요구하는 동적 컨텐츠를 제공하기 위해 만들어진 애플리케이션 서버이다. HTTP 프로토콜을 기반으로 사용자 컴퓨터나 장치에 애플리케이션을 수행해주는 미들웨어로서, 주로 데이터베이스 서버와 같이 수행된다. WAS는 JSP, Servlet 구동환경을 제공해주기 때문에 서블릿 컨테이너 혹은 웹 컨테이너로 불린다.  이러한 WAS는 웹 서버의 기능들을 구조적으로 분리하여 처리하고자 하는 목적으로 제시되었다. 분산 트랜잭션, 보안, 메시징, 쓰레드 처리 등의 기능을 처리하는 분산 환경에서 사용된다. WAS는 프로그램 실행 환경과 DB 접속 기능을 제공하고, 여러 개의 트랜잭션을 관리 가능하다. 또한 비즈니스 로직을 수행할 수 있다.  대표적인 WAS 컨테이너로는 tomcat, JBoss, WebSphere 가 있다. |
| Fire wall | 컴퓨팅에서 방화벽 또는 파이어월(firewall)은 미리 정의된 보안 규칙에 기반하여 들어오고 나가는 모든 네트워크 트래픽을 모니터링하고 제어하는 네트워크 보안 시스템을 의미한다. 일반적으로 방화벽은 신뢰할 수 있는 네트워크와 신뢰할 수 없는 네트워크 사이에 장벽을 설정한다.  주요 목표는 내부 네트워크 또는 장치와 인터넷에서 들어오는 외부 트래픽 사이에 벽을 만드는 것이다. 만드는 이유는 인터넷에 시스템을 손상시키려는 바이러스, 멀웨어, 해커 등 악성 트래픽이 많이 있기 때문이다. 네트워크 방화벽은 트래픽을 필터링하여 위협을 차단하고 무단 액세스를 방지하여 네트워크 보안을 제공하는 도구이다. |
| Switch | 네트워크 스위치는 컴퓨너 네트워크에서 데이터를 전송하고 관리하는 중요한 장비이다. 스위치는 여러 대의 컴퓨터나 네트워크 장치 간에 데이터 통신을 용이하게 하고, 효율적인 데이터 전달을 위해 트래픽을 분석하고 제어한다. 네트워크 스위치는 주로 이더넷 기반의 LAN에서 사용되며, 다양한 기능과 특성을 제공한다. |
| L2 Switch | 물리적인 MAC (Media Access Control) 주소를 기반으로 스위칭하는 장비이다. 이더넷 프레임을 검사하여 목적지 MAC 주소를 보고 포트 간에 트래픽을 전달한다. 주로 로컬 영역 네트워크 (LAN)에서 작동하며, 스위칭 속도가 빠르고 대역폭을 효율적으로 사용할 수 있습니다. |
| L3 Switch | L2 스위치의 기능뿐만 아니라 IP주소 및 라우팅 기능도 수행하는 장비이다. IP 패킷을 검사하여 목적지 IP 주소를 보고 라우팅 테이블을 기반으로 트래픽을 전달합니다. |
| L4 Switch | L2 및 L3의 스위치의 기능에 더하여 전송 계층 (Layer 4))에서의 정보도 검사하여 트래픽을 처리하는 장비이다. 주로 로드 밸런싱과 같은 트래픽 분산을 지원하며, 포트 및 프로토콜 정보를 기반으로 트래픽을 관리한다. 서버 팜과 같은 환경에서 효율적인 트래픽 분배를 제공한다. |
| SSL | SSL(Secure Sockets Layer)은 네트워크 통신에서 데이터 보안 및 인증을 제공하기 위한 암호화 프로토콜이다. 웹사이트 애플리케이션 및 다른 네트워크 기반 서비스에서 개인 정보 및 민감한 데이터의 안전한 전송을 보장하는 데 사용된다. SSL은 인터넷 보안을 강화하고 데이터 도난, 변조 및 악의적인 공격으로부터 보호하는 역할을 한다. |
| LAN | LAN(Local Area Network)는 한정된 지역 내에서 컴퓨터 및 기타 네트워크 기기들이 서로 연경되어 통신하는 네트워크를 나타낸다. LAN은 비교적 작은 지리적 영역 내에서 네트워크를 형성하며, 가정, 사무실, 학교, 공공 장소 등에서 사용되는 매우 일반적인 형태의 네트워크이다. 주로 이더넷 프로토콜을 사용하며, 보안과 관리가 상대적으로 용이하다. 컴퓨터, 프린터, 스위치, 라우터, 모뎀 등 다양한 종류의 장치들이 연결될 수 있다. |
| WAN | WAN(Wide Area Network)는 넓은 지역을 아우르는 네트워크를 나타낸다. LAN과 달리 WAN은 지리적으로 분산된 지역을 연결하고 네트워크를 형성하는데 사용되며, 인터넷과 같은 전세계적인 네트워크도 WAN의 한 예이다. 여러, 도시, 국가 대륙간 연결을 담당하며, 전용 회선, 광섬유 케이블, 위성 연결, 전화선, 무선 연결 등을 통해 연결된다. ISP와 같은 서비스 제공업채의 네트워크 인프라를 이용해 데이터를 교환한다. 대표적인 프로토콜로는 TCP/IP 프로토콜이 있다. |
| TCP/IP | TCP/IP(Transmission Control Protocol/Internet Protocol)은 네트워크 통신을 위한 프로토콜 스택으로, 현재 인터넷을 비롯한 다양한 컴퓨터 네트워크에서 사용되는 표준 프로토콜이다. TCP와 IP는 TCP/IP 프로토콜 스택의 가장 중요한 구성 요소인데, 데이터의 전송과 라우팅에 관련된 역할을 수행한다. 신뢰성있는 연결을 중시하고 계층 구조로 이루어져 있다. |
| Router | 라우터(Router)는 컴퓨터 네트워크에서 데이터 패킷을 전달하고 경로를 선택하는 장치이다. 라우터는 패킷 스위칭(Packet Switching) 기술을 사용하여 여러 개의 네트워크 간에 데이터를 전달하며, 효율적인 데이터 흐름을 조절하고 네트워크 간의 연결을 관리한다. 다양한 유형의 라우터가 있으며, 인터넷 연결과, 사설 IP주소와 공용 IP주소를변환시켜 서로 통신할 수 있게 한다. |
| IPS | IPS(Instrusion Prevention System)는 침입 방지 시스템이다. 네트워크 시스템 내부의 침입 행워를 탐지하고 차단하는 역할을 수행하는 보안 장치 또는 소프트워어이다. 네트워크와 시스템, 그리고 데이터를 악성 행위나 공격으로부터 안전하게 보호하는 역할을 수행한다. 감시하는 방법으로는 트래픽의 분석, 미리 정의된 시그니처 기반 탐지, 행위 기반 탐지, 통계 및 로깅, 가시성 및 트래픽 분석, 가상 및 클라우드 플랫폼 지원이 있다. |
| NAS | NAS시스템은 네트워크에 연결된 고용량 스토리지 장치로 이를 통해 권한이 부여된 네트워크 사용자와 클라이언트는 중앙 위치에서 데이터를 저장하고 검색할 수 있다. 기본적으로 NAS 장치는 단순히 파일을 공유하고 인증할 수 있는 몇 가지 추가 지능을 갖춘 하드 드라이브용 컨테이너이다. NAS 장치는 RAID라는 기술을 사요해 저장된 데이터를 여러 하드 디스크에 분산하고 복제할 수 있다. 이런 중복성은 드라이브에 문제가 발생했을때 데이터 복원력을 보장한다. |
| DMZ | 컴퓨터 보안에서의 DMZ는 조직의 내부 네트워크와 외부 네트워크 사이에 위치한 서브넷이다. 외부 네트워크가 DMZ로 연결할 수 있도록 허용하면서도 DMZ내의 컴퓨터는 오직 외부 네트워크에 연결할 수 있도록 한다. DMZ의 앞뒤로는 방화벽이 설치된다. 즉, DMZ는 내부 내트워크에 존재하지만 외부에서 접근할 수 있는 특수한 네트워크 영역이다. |
| DBMS | DBMS(Database Management System)는 데이터베이스를 관리하고 조작하는 소프트웨어 시스템이다. 데이터베이스는 구조화된 데이터의 모음이며, DBMS는 데이터의 저장, 검색, 수정, 삭제 등을 효율적으로 관리하기 위한 도구와 기능을 제공한다. DBMS는 다양한 종류의 데이터베이스를 생성하고 관리하며, 데이터베이스에 접근하여 작업할 수 있는 인터페이스를 제공한다. 주요 DBMS 제품으로는 Oracle, MySQL, Microsoft SQL Server, PostgreSQL, MongoDB가 있다. |

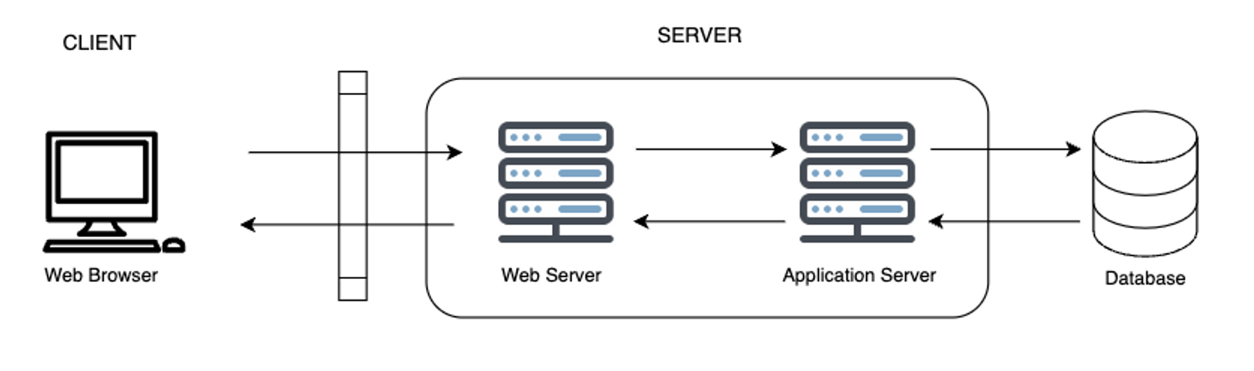
1. **WEB 시스템 기술요소 설명**
   1. **클라이언트측 기술요소**

|  |  |
| --- | --- |
| 기술요소 | 내용 |
| HTML | HTML(Hyper Text Markup Language)는 웹 페이지를 생성하고 구조화하기 위해 사용하는 마크업 언어이다. HTML은 웹 컨텐츠의 구조와 의미를 정의하며, 웹 브라우저에게 어떻게 표시되어야 하는지를 지시하는 역할을 한다. 웹 페이지의 텍스트, 이미지, 링크, 폼 등을 구성하는 요소를 정의하여 사용자에게 시각적으로 보여지는 웹 화면을 만들 수 있다. 주요 구성 요소로느 Tags, Elements, Attributes, Links, Images, Lists, Forms, Tables 등이 있다. 화면을 구조화하여 사용자에게 보여주는 역할을 하며, CSS와 JavaScript를 함께 사용해 디자인과 상호작용을 한다. |
| CSS | CSS(Cascading Style Sheets)는 웹 페이지의 스타일과 레이아웃을 정의하기 위해 사용되는 스타일 시트 언어이다. HTML이 웹 페이지의 구조와 콘텐츠를 정의하는 데 사용되는 반면, CSS는 웹 페이지의 디자인과 표현 방식을 담당한다. CSS를 사용하면 웹 개발자는 웹 페이지의 모양, 색상, 폰트, 간격 등을 제어하여 사용자에게 보다 디자인적으로 풍부한 화면을 보여준다. 요소로는 Style Ruls, Properties, Values, Units, Classes and IDs, Inheritance, Specificity, Cascading이 있다. |
| Javascript | 가장 인기있는 프로그래밍 언어 중 하나로, 웹 페이지의 동적인 기능을 구현하고 사용자와 상호작용하는데 사용된다. JavaScript는 클라이언트 쪽 스크립트 언어로서 웹 브라우저에서 실행된다. 주로 동적인 웹 페이지를 구현하는데 사용하고, DOM형식으로 웹 페이지의 요소들의 CRUD를 담당한다. 여타 프로그래밍 언어처럼 변수, 함수 등이 있고, 객체지향 개념을 지원하며, async와 await등의 비동기 프로그래밍을 지원한다. 또한 다양한 외부 라이브러리와 프레임워크가 존재하는데, 대표적으로는 jQuery, React, Angular, Vue.js 등이 있고, 심지어 서버 측 개발을 위한 Node.js에도 활용된다. |
| Plugins | 플러그인은 소프트워에의 기능을 확장하거나 추가하기 위해 사용되는 소프트웨어 모튤 또는 확장 모듈이다. 플러그인은 기본 소프트웨어의 기능을 보완하거나 커스터마이징하는 데 사용되며, 사용자에게 필요한 다양한 기능을 제공하거나 작업을 간소화하는 데 도움이 된다. 특정 소프트웨어나 애플리케이션에서는 플러그인을 추가하여 사용자가 원하는 기능을 선택적으로 활성화하거나 변경할 수 있다. 따라서 확장성이 높고, 오픈 소스의 경우에는 다양한 외부 개발자들이 참여한다. 주로 웹 브라우저, 웹 개발 프레임워크에서 찾아볼 수 있다. |

* 1. **서버측 기술요소**

|  |  |
| --- | --- |
| 기술요소 | 내용 |
| JSP | JSP(JavaSerer Pages)는 동적 웹 페이지를 개발하기 위한 자바 기반의 서버 사이트 스크립팅 기술이다. JSP는 HTML코드 안에 자바 코드를 삽입해 웹 애플리케이션의 동적인 컨텐츠를 생성하고 표현하는 데 사용된다. JSP는 서블릿 기술을 기반으로 하며, 웹 애플리케이션의 구성 요소 중 하나로 웹 브라우저와 웹 서버 사이에서 동작한다. 동적 컨텐츠 생성을 위해 HTML코 드 내에 <%%>태그를 생성해 자바 코드를 삽입할 수 있고, 서블릿으로 변환되어 처리된다. 또한 태그 라이브러리를 이용해 조건문과 반복문 등을 더 쉽게 작성할 수 있습니다. JSP는 MVC패턴에서 View를 담당하는 기술이다. |
| PHP | PHP(Hypertext Preprocessor)는 서버 측 스크립트 언어로, 주로 웹 개발에 사용되는 언어이다. PHP는 HTML 코드 안에 포함하여 웹 페이지를 동적으로 생성하고 관리하는 데 사용되며, 데이터베이스와의 상호작용, 파일 조작, 세션 관리 등 다양한 작업을 수행할 수 있다. 서버 측 스크립트 언어이기 때문에 사용자의 브라우저에 노출되지 않으며, 다양한 데이터베이스와 연동이 가능하다. 세션 및 쿠키 관리도 가능하고, 웹 폼 처리를 할 있어 사용자 입력을 DB와 비교하여 검증할 수 있다. 백엔드 개발에 적합하며 MVC 아키텍처를 따르는 프레임워크를 사용할 수도 있다. 오픈 소스이기 때문에 무료이다. |
| ASP | ASP(Active Server Pages)는 마이크로소프트가 개발한 서버 측 스크립트 기술로, 동적인 웹 페이지를 생성하고 관리하는 데 사용되는 프로그래밍 기술이다. ASP는 웹 서버에서 실행되며, 서버에서 클라이언트로 동적인 내용을 생성하여 웹 페이지를 구성하는 데 사용된다. ASP도 서버 측 스크립트 언어이다. Microsoft SQL, Oracle, MySQL등의 DBMS와 연동될 수 있으며, COM(Component Object Model)로 컴포넌트와의 통합을 지원하여 코드를 모듈화하여 재사용할 수 있다. 백엔드 개발, 사용자 입력 처리, 다양한 웹 애플리케이션 개발이 가능하다. |
| ASP.net | 마이크로소프트가 개발한 웹 응용 프로그램 및 웹 서비스를 구축하기 위한 프레임워크이다. ASP.NET은 웹 애플리케이션을 개발하기 위한 다양한 도구와 라이브러리를 제공하여 개발자가 효율적으로 웹 애플리케이션을 개발하고 관리할 수 있도록 도와준다. C#과 VB.NET등의 다양한 프로그래밍 언어를 지원한다. 또한 MVC모델을 제공하고, 다양한 리치 컨트롤과 라이브러리를 제공해 웹 페이지의 요소를 조작할 수 있다. DB연동을 지원하며, Restful API구현에 적합하다. |
| Servlet | 서블릿(Servlet)은 자바 언어를 기반으로 웹 애플리케이션을 개발하기 위한 서버 사이드 프로그래밍 기술 중 하나이다. 서블릿은 웹 서버에서 동작하며, 클라이언트의 요청을 받아 처리하고 그에 따른 응답을 생성하는 역할을 한다. 서블릿은 웹 애플리케이션의 동적인 기능을 구현하기 위해 사용되며, 주로 웹 페이지의 생성과 데이터 처리를 담당한다. 서블릿 컨테이너는 서블릿의 인스턴스를 생성하고 초기화한 후 소멸될때 까지 반복적으로 실행되는 생명주기를 가진다. HTTP통신을 하며,. 멀티스레드에서 동작한다. |

1. **WEB 시스템 구조와 논리**
   1. **WEB 시스템 구조**



웹 애플리케이션의 구조를 도식화한 이미지이다.

* 1. **WEB 시스템 논리**
  2. 웹사이트 vs 웹 애플리케이션

웹사이트(Website)와 웹 애플리케이션(Web application)은 일상에서 혼용하여 사용하기도 한다. 하지만, 엄밀히 말하면 이 둘은 서로 다른 개념이다.

* + 웹 개발 영역에서 웹사이트(Website)는 일반적으로 정적 페이지들의 집합체를 의미한다.
  + 웹사이트에서 동적 페이지를 포함하게 되면 이는 웹 애플리케이션(Web application)이 된다.

웹 애플리케이션의 특징

- 프로그램처럼 상호작용이 가능하다.

- 특정 기능을 가지고 있다 ex) 정보 검색

- 정보나 자료 등의 콘텐츠 관리 시스템과 함께 작동한다.

* + 오늘날 만들어지는 대부분의 웹 서비스들은 거의 웹 애플리케이션들이다. ex) 네이버, 페이스북 ...

웹 애플리케이션 아키텍처

웹 애플리케이션 아키텍처는 클라이언트-서버 간의 연결에 대한 설명 방법이다. 즉, 웹 애플리케이션 아키텍처는 애플리케이션 내부의 요소들이 상호 간에 어떻게 소통하는지를 설명한다.

사용자가 웹 브라우저에서 요청을 하면, 애플리케이션의 다양한 요소(브라우저, 인터페이스, 서버, 데이터베이스 등)들이 상호작용을 하게 된다.

웹 애플리케이션 아키텍처는 이러한 요소들이 상호작용을 유지할 수 있도록 서로를 연관 짓는 뼈대라고 할 수 있다.

웹 애플리케이션은 사용자의 수많은 요청에 알맞은 응답을 할 수 있어야 한다. 따라서 웹 애플리케이션 서버는 요소와 외부 애플리케이션을 서로 공유하여 설계하게 된다.

또한, 인터넷에 공개되는 순간부터 글로벌 네트워크의 대규모 트래픽에 노출될 수 있기 때문에 신뢰성, 확장성, 보안성, 견고성에 대해 고려하면서 설계해야 한다.

1. **WEB 시스템 개발 방법론**
   1. **정의**

웹개발방법론(WDM; Web Development Methodology)은 인터넷 웹사이트 개발 방법에 대한 이론으로서, 웹사이트 개발 과정, 절차, 방법, 산출물, 도구 등을 실무적인 관점에서 체계적으로 정리하고 표준화시킨 것이다. 소프트웨어의 개발 방법론은 크게 정보공학 방법론, 객체지향 방법론, CBD방법론, 애자일 방법론, 구조적 방법론으로 나눌수 있다. 5가지 방법론의 단점을 보완하거나 각각의 방법론의 장점을 따온 개발론도 있다.

* 1. **정보공학 개발론**

정의

기업 전체 , 또는 기업의 주요부분을 계획 , 분석 , 설계 및 구축에 정형화된 기법들을 상호 연관성 있게 통합 , 적용하는 데이터 중심 방법론. James Martin 이 비즈니스 시스템 즉, 정보시스템 개발을 공학적으로 접하기 위해 체계화시킨 개발 방법론이다.

특징

* + - 1. SIS(Stra Strategic Information System)에 초점을 맞춘다. 기업 중심적이다.
      2. ISP 중심적인 개발론이다.
      3. 데이터 중심적이다.
      4. 분할과 정복의 방식: Top down 방향으로 문제해결
      5. 공학적 접근
      6. 사용자 피드백 및 참여 환영
  1. **객체지향 개발론**

정의

요구분석, 설계, 구현, 테스트 및 검증 등의 소프트웨어 생명주기에 객체 지향 개념을 접목시켜 일관된 모델을 가지고 소프트웨어를 개발하는 방법론

* 1. **CBD 개발론**

정의

재사용 가능한 컴포넌트의 개발 또는 상용 컴포넌트를 조합해 어플리케이션 개발 생산성과 품질을 높이고 시스템 유지보수 비용을 최소화할 수 있는 개발 방법론.

특징

* + 모듈화: 소프트웨어 시스템을 작은 부분으로 분리하고, 각각의 부분을 개별적으로 개발하고 관리한다.
  + 재사용성: 컴포넌트들을 독립적으로 재사용 가능하도록 설계해, 개발 및 유지보수의 효율성을 높인다.
  + 의존성 관리: 각 컴포넌트 간의 의존성을 관리하여 시스템 전체의 안정성과 유지보수를 용이하게 만든다.
  1. **애자일 개발론**

정의

Agile 방법론은 프로젝트 관리와 소프트웨어 개발에서 사용되는 접근 방식 중 하나로, 빠른 변화와 적응을 강조하는 방법론이다. Agile은 기존의 폭포수(Waterfall) 모델과 달리 작은 주기의 반복을 통해 개발을 진행하며, 고객의 요구사항 변화에 신속하게 대응할 수 있도록 한다. 이는 기존 폭포수 모델의 지나친 문서화와 고객 요구사항을 일정기간으로 제한하는 단점을 보완한 것이다.

특징

* + 반복 주기(Iteration)를 통한 개발: 작은 주기를 반복하여 제품을 개발하고 개선하는 방식으로, 빠른 결과물을 제공하며 계속적인 피드백을 통해 개선한다.
  + 고객 중심 관점: 고객의 의견과 요구사항을 중요하게 생각하며, 요구사항 변경에 유연하게 대응한다.
  + 책임성: 팀원들이 스스로 조직을 운영하고 작업을 결정하는 능력을 갖도록 하여 효율성을 높인다.
  1. **구조적 개발론**

정의

구조적 방법론은 소프트웨어 개발 시, 체계적이고 구조화된 접근 방식을 강조하는 방법론으로, 시스템을 단계별로 분해하여 설계하고 개발하는 방법론이다

특징

* + 모듈화와 추상화: 시스템을 작은 모듈로 분해하고 각 모듈을 고유한 기능을 수행하도록 추상화하여 설계한다.
  + 상하위 관계: 모듈 간의 상하위 계층 구조를 통해 모듈 간의 의존성과 역할을 명확하게 정의하여, 시스템의 구조를 더욱 명확하게 한다.
  + 구조적 제어: 프로그램의 제어 흐름을 구조적으로 설계하고, 루프와 조건문 등을 효과적으로 활용하여 코드의 가독성과 유지보수성을 높인다.