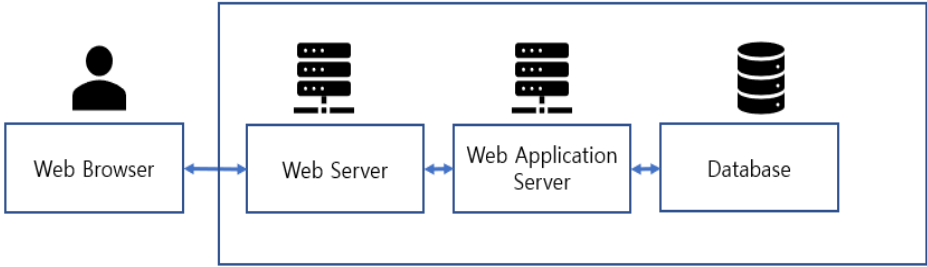


## 군 부대 행정업무 간소화 SW

<b>프로젝트명</b>	행정 간소화 SW		
<b>개발기간</b>	2019.02 ~ 2019.06 (군 복무간 개발)	<b>개발인원</b>	2명
<b>개발환경</b>	Notepad++, .net compiler(css), mssql		
<b>개발 목적</b>	군 복무간 사단장님의 명으로 행정 업무를 간소화 할 수 있는 SW를 개발하게 되었고, 군 인트라넷 망내에는 별도로 프로그램 개발 환경이 있지 않아 윈도우 운영체제에 있는 .NET Framework를 이용해 WAS와 WS를 제작하고, 웹 페이지를 바닐라 js로 구성하여 군 내부의 출타신청, 연등 신청, 휴가 관리 등의 행정 업무를 간소화 하였다.		
<b>아키텍처</b>	 <pre> graph LR     User((User)) --&gt; WB[Web Browser]     WB &lt;--&gt; WS[Web Server]     WS &lt;--&gt; WAS[Web Application Server]     WAS &lt;--&gt; DB[(Database)]     subgraph ServerGroup [ ]         WS         WAS         DB     end </pre>		
<b>개발내용</b>	출타, 연등, 휴가 관리를 위한 바닐라 js Front app 제작 사용자 접속을 위한 Web Server 제작 데이터 처리를 위한 API Server 제작 데이터베이스와 연동을 위한 ORM 라이브러리 제작		
<b>담당업무</b>	<p>사용자 접속을 위한 Web Server 제작</p> <ul style="list-style-type: none"> <li>- Browser가 HTTP 통신으로 Resource를 요청할 경우에 WS에서 해당 리소스를 응답으로 보내준다. 그 외의 요청일 경우 WS는 해당 요청을 WAS로 보내 WAS의 응답을 Browser로 보내준다.</li> <li>- WAS가 동작 하지 않을 경우, 정해진 응답 페이지를 Browser에 보내준다.</li> <li>-</li> </ul> <p>데이터 처리를 위한 API Server 제작</p> <ul style="list-style-type: none"> <li>- WS로부터 요청이 올 경우 해당 HTTP 요청을 분석하여 그에 맞는 서비스를 응답해준다.</li> <li>- 클라이언트별 세션 관리를 위한 세션 매니저</li> <li>-</li> </ul> <p>데이터베이스와 연동을 위한 ORM 라이브러리 제작</p> <ul style="list-style-type: none"> <li>- C#의 Class와 MSSQL의 Entity를 매핑해주어 SQL코드를 작성 없이 사용할 수 있도록 한다.</li> <li>- C#의 Expression Tree를 파싱하여, SQL로 해석한다.</li> <li>-</li> </ul>		

이슈	<p>1. 군대의 특성상 상급자의 기분에 따라 개발 사항이 변경되기 때문에 웹 서버를 종료하지 않고, 프로그램을 업데이트 하기 위해 어셈블리를 동적으로 로드하려 하였으나, dll이 로드되면 언로드가 불가하여 장기적으로 불필요한 메모리를 차지하게 됨.</p> <ul style="list-style-type: none"> <li>■ WAS가 패치를 진행할 경우 웹서버는 전 버전의 WAS에 연결을 하고, 패치가 완료되면 실 서버에 연결이 되도록 하여 해결하였다.</li> <li>■ 이 방법을 활용하여 로드밸런싱 구현을 하려 했으나 예하부대 전체의 트래픽으로는 성능을 확인할 수 있는 환경이 구성이 안되었다.</li> </ul> <p>2. ORM의 경우 Where, OrderBy, FirstOrDefault, Take, Skip 등의 메서드를 Expression Tree를 이용해 Enumerable객체를 반환하도록 개발 하였으나 IEnumerable의 경우 메서드가 호출 후 즉시 실행이 되어 데이터베이스 테이블을 전체 로딩 한 후, Where등의 조건 절로 다른 리스트 객체를 반환하여 불필요한 메모리까지 차지 함.</p> <ul style="list-style-type: none"> <li>■ IQueryable를 사용하면 메서드 호출 시가 아닌, 객체를 사용하는 시점에 SQL문을 생성하고 실행하기 때문에 불필요한 데이터를 조회하지 않을 수 있었다.</li> </ul>
소스코드	군 내부에서 개발하여 소스코드가 없음.
시연영상	군 내부에서 개발하여 시연영상이 없음.