

1page) 안녕하세요 네이버쇼핑 api를 활용해 상품을 검색할 수 있는 웹서버를 설계한 7조 팀장 허진환, 팀원 김태현학우입니다.

2page) 저희 발표는 프로젝트 개요로 시작하여 웹서버 주요 구성도로 저희 웹서버 구조를 그림으로 간단하게 설명드린 뒤, 순서도를 포함하여 주요 함수들의 코드를 보여드리고 마지막으로 시연한 뒤 발표 마치겠습니다.

3page) 저희 프로젝트는 가상머신 Linux Cent OS에 Express Framework를 사용한 Node.js 쇼핑몰 웹서버를 구현하는 것으로, MySQL DBMS를 활용하여 Client와 Server간 주고 받는 데이터들이 알맞게 테이블링되어 적재되도록 하였습니다.

4page) 저희 조는 웹서버 실무 능력 함양을 위해 실제 환경과 비슷하게 Linux OS를 사용하여 웹서버 환경을 구축하였으며, 클라이언트-서버 모델 이해를 기반으로 Client 요청에 따라 사용자가 필요로 하는 응답을 하도록 설계하였습니다. 또한, 데이터베이스도 마찬가지로 Client의 요청에 의한 데이터 적재/조회가 각각 post/get 매서드로 사용자에게 요청받아 구현되도록 하였습니다.

5page) 현실적 제한요소 달성결과는 다음과 같습니다. 저희는 오픈소스 운영체제인 Linux Cent OS를 기반으로 웹서버를 설계하였기에 비용이 절감되는 측면이 있으며, Linux가 상대적으로 적은 리소스를 사용하는 운영체제이므로 H/W적 비용 또한 절감된다고 말씀드릴 수 있습니다. 또한, 네트워크 침해 공격에 대해 어느정도 대응할 수 있도록 유저 데이터를 데이터베이스에 적재 시 비밀번호를 해시함수를 통해 암호화시켜 적재하도록 하였습니다.

6page) 태현

웹서버 구성도

이어서 웹서버 구성도에 대해 설명하겠습니다. 익스프레스 프레임워크 기반의 서버를 생성하고 app객체에 미들웨어, 라우팅 핸들러 설정 등을 추가했습니다. 포트넘버로는 3000번을 사용했고 해당 포트를 통해 각각의 경로에 대한 클라이언트의 요청이 들어왔을 시 get또는 post방식으로 이를 처리하고 결과를 응답합니다. 여기에서 응답은 views폴더 내의 pug파일 렌더링, 리다이렉션, 에러 전송등이 될 수 있습니다.

## DB-웹서버 통신 구성도

다음은 DB와 웹서버간 통신 구성도입니다.

클라이언트 요청 시 Main.js에서 이를 처리합니다. 필요한 경우 네이버 쇼핑 API를 활용해 상품의 정보를 조회할 수 있습니다. MySQL에서는 웹서버에서 주로 사용하는 데이터의 특성에 기반하여 items, comments, users와 같은 테이블들을 생성하였고 DB의 테이블을 기반으로 댓글, 유저데이터, 평점 등의 정보를 적재 및 조회할 수 있으며 이러한 과정을 거치면서 사용자의 요청에 맞추어 완성된 정보를 클라이언트에게 응답할 수 있습니다

다음은 순서도 및 주요파일에 관한 설명입니다.

글씨가 작아서 잘 보이지 않으실텐데 대표적인 파일, 폴더들을 몇가지 설명드리겠습니다.

첫번째로 Main.js 내부에는 로그인, 회원가입과 같은 유저 데이터 관련함수부터 검색, 결과 출력과 같은 검색관련 함수까지 요청을 처리하는 대부분의 함수가 위치해 있습니다. 네이버 쇼핑 API에 대한 접근과 app.listen()등 서버 설정에 관한 부분도 포함되어 있습니다.

두번째로 database.js 내부엔 데이터베이스 접근에 관련된 함수들이 모여 있습니다. 유저 데이터 조회, 댓글 내용 적재, 상품 평점 조회 등 웹서버에서 DB에 저장, 조회하는 함수들의 집합입니다. DB 연결에 대한 옵션부터 DB 연결, DB 접근함수 모듈화 등이 포함되어 있습니다.

세번째로 views폴더입니다.

main.js에서 app.set()을 통해 템플릿 엔진을 Pug로 설정했습니다. 따라서 뷰 폴더안에는 pug파일들이 위치해 있습니다. 클라이언트에게 특정 경로에 대한 요청이 왔을 때 상황에 맞게 pug파일들을 렌더링 할 수 있게 설정했습니다.

본 웹서버는 클라이언트의 요청이 들어왔을 시 DB와 끊임없이 데이터를 송수신 하고 올바른 응답을 할 수 있게 구성했습니다.

database.js의 대표적인 함수는 DB에서 데이터를 조회, 저장하는 get~~, save~~ 함수가 있습니다. 각각의 함수는 SELECT와 INSERT INTO 등의 쿼리문을 사용해서 DB의 데이터를 저장, 조회합니다.

유저 데이터 조회 함수는 클라이언트에게 입력 받은 username 데이터를 DB에서 조회한 후 정보가 존재하지 않을 시 ID,PW등이 담긴 user객체 형태로 반환하고 유저 데이터 적재 함수는 입력받은 users객체의 ID, PW 데이터를 DB의 users 테이블내에 적재합니다.

이를 활용한 Main.js의 대표적인 함수는 회원가입에 관한 함수입니다.

사용자에게 입력받은 데이터를 문자열로 변환 후 parse해서 ID와 PW를 저장하고 getUserFromDB 함수를 사용해서 중복계정인지 검사합니다. 해당 ID에 대한 정보가 DB에 존재하지 않을 시에 Hash알고리즘을 사용해서 암호화를 진행 후 saveUserToDB함수를 사용, 유저 정보를 DB에 적재합니다.

프로그램 시연하겠습니다.