

MERN스택 개발환경 보고서

비트캠프 KDT5기 전종현

1. MERN스택 정의
2. Node.js
3. React.js
4. Express.js
5. MongoDB
6. Docker
7. MERN스택으로 개발하기

1. MERN스택이란?

전략패턴이라고 하는 디자인패턴중 하나로 M:MongoDB, E:Express.js, R:React.js, N:Node.js 네 가지 요소를 사용하여 웹사이트를

Node.js->React.js->Express.js->MongoDB순서로 개발하는 과정을 말한다.

2.Node.js란?

Chrome V8 JavaScript엔진으로 빌드된 JavaScript 런타임이다. 즉 Node.js는 언어가 아닌 프로그램 환경이다. 노드를 통해 다양한 자바스크립트 애플리케이션을 실행할 수 있으며, 서버를 실행하는 데 많이 사용된다.

환경변수설정 방법은 노드홈페이지에 접속하여 노드 다운로드 후 압축을 풀어준다.

윈도우 + r 키를 눌러서 열기 입력창에 sysdm.cpl ,3을 입력한다.

환경변수를 눌러주고 시스템변수에 새로만들기를 클릭 후 NODE_HOME이라고 변수이름을 설정 해준 뒤 변수 값에 노드 파일이 있는 경로를 넣어준다.

그 후 확인 버튼을 누르면 환경변수설정이 완료된다. 완료 후에는 윈도우 + r 키를 눌러서 cmd실행 후 node -version 입력, 엔터를 눌렀을 때 버전이 맞는지 확인한다.

3.React.js란?

JavaScript library로, Component를 기반하여 사용자 인터페이스(UI)를 만들기 위해 사용된다.

리액트를 사용하기 위해서는 VSCode를 다운받고 터미널을 열어 npm -version을 입력해 버전이 맞게 다운 되었는지 확인한다.

그 후 VSCode를 실행을 하는데 주의해야 할 점은 관리자 권한으로 실행을 해주어야 한다.

그렇지 않을 경우 권한 에러가 발생할 수 있기 때문에 이 부분은 꼭 유의해서 실행해야 한다.

4.Express.js란?

Node.js의 프레임워크이다. 즉 Express는 Node.js를 이용하여 웹 애플리케이션을 만들기 위한 틀을 제공하는 라이브러리의 집합이며, JavaScript언어로 사용된다.

Express.js 설치 방법은 Express를 설치하고자 하는 폴더에 들어가서 마우스 우클릭 후 터미널을 실행한다.

터미널이 열리면 터미널에서 명령어를 입력해야 하는데 명령어는 mkdir폴더명-> dir-> cd.폴더명-> npx express -generator 순서로 입력을 한다.

그 후 터미널에서 여러 파일들이 생성된 후에 dir입력, 마지막으로 실행하는 code . 을 입력해 Express.js 프로그램 실행까지 잘 되었는지 확인해본다.

express프로그램이 열리면 express터미널 창에 명령어npm start 를 입력한다.

입력 했을 때 서버가 에러없이 성공적으로 연결이 되면 express설정은 완료가 된다.

5.MongoDB란?

C++로 작성된 오픈소스 문서지향 NoSQL 데이터 베이스 시스템이다. MongoDB는 동적스키마형 도큐먼트(document)들을 선호하고 이것들을 BSON이라 부른다.

몽고디비 설치 방법은 먼저 도커를 다운받는다. 도커 다운로드가 완료되면 터미널을 열고 docker pull mongo

```
docker run --name mongo -e
MONGO_INITDB_ROOT_USERNAME=root -e
MONGO_INITDB_ROOT_PASSWORD=root -d -p
27017:27017 mongo
```

docker ps 명령어를 순서대로 입력한다.

6.Docker란?

애플리케이션을 신속하게 구축, 테스트 및 배포할 수 있는 컨테이너 기반의 오픈소스 소프트웨어 플랫폼이다. Docker를 사용하면 환경에 구애받지 않고 애플리케이션을 신속하게 배포 및 확장할 수 있는 장점이 있다.

도커 설치 방법은 docker 홈페이지에 접속 후 윈도우로 도커를 다운로드 받는다.

설치가 완료되면 도커에서 안내해주는 링크로 들어가 Linux커널 업데이트 패키지를 다운한다.

패키지 설치가 끝나면 도커 터미널 창에 입력해야 할 명령어가 있는데 도커에서 안내해주는 순서대로 명령어를 붙여넣으면 설치는 끝난다.

7. MERN스택으로 개발하기

우선 프로젝트의 목적은 내가 공부했던 노트나 흔적들을 공유하고 기록하기 위해 만들었다.

공부를 기록했던 파일들이 깃허브와 노션 뿐이지만 이후 블로그나 다른 흔적들로 더 채우고 구성해 나갈 계획이다. 프로젝트는 홈화면과 로그인, 회원가입, 프로필 기능으로 구성되어 있다.

만든 폴더로는 components, pages, modules, styles이다.

먼저 사용자에게 보여주기 위한 홈 화면을 구성한다.

홈 화면 구성은 components로 이루어지기 때문에 컴포넌트 안에 각각의 맞는 폴더를 생성하여 홈, 헤더, 푸터, 네비, 레이아웃, 로그인, 회원가입, 프로필 파일들을 구성한다.

컴포넌트를 만들고 난 뒤 페이지 폴더를 생성한다. 페이지 폴더 역시 로그인, 회원가입, 프로필 파일들로 구성한다. 페이지 폴더는 컴포넌트 화면에서 받은 데이터 값을 넥스트로 넘겨준다.

페이지의 구성을 끝낸 뒤 modules를 만들어준다.

modules에서는 리덕스 사가(Redux-Saga)를 사용하는데 리덕스 사가란 리덕스의 미들웨어로 리덕스가 액션을 수행하면 redux-saga에서 디스패치하여 redux의 액션을 가로챈 뒤 액션의 역할을 수행 하고 다시 액션을 발행하여 데이터를 저장하거나 다른 이벤트를 수행하는 함수이다.

components, pages, modules의 구성이 끝나면 서버인 Express를 구성한다.

Express안에 기본적으로 생성되어있는 app.js를 server.js로 이름을 바꿔준 뒤 Express에 필요한 models, routes, services 폴더를 생성한다.

Express에서의 데이터 흐름은server.js -> routes-> services-> models-> lambdas로 흐른다.

server는 지정된 경로를 찾아 routers에게 데이터를 이동을 시킨다.

routers는 URL을 갖고 service에게 경로가 같은 URL로 데이터를 던진다.그 후 models로 이동 한다.

models에서는 password를 암호화 하고 토큰을 만들고 lambdas로 넘겨준다.

lambdas에서는 전달받은 데이터 값을 DB에 저장하고 DB에 저장된 데이터를 확인하기 위해 터미널을 실행 한 뒤

mongosh "mongodb://localhost:27017" --username "root" --password "root", show dbs,

use soccerdb, show collections, db.users.find() 명령어를 차례로 입력해 디비에 저장된 데이터 값을 확인한다. 디비에 저장된 데이터 값 까지 확인을 했다면 이제 MERN스택의 실행 순서에 맞게 가장 먼저 도커->M인 몽고디비-> Express -> React 를 차례로 실행한다.

화면이 뜨면 회원가입과 로그인을 한다.

로그인 까지 하고 나면 내가 구성한 프로필이 화면에 나타날 것이다.

아래 사진들은 홈 화면 구성과 회원가입, 로그인, 프로필 화면이다.



