

MERN스택을 활용한 프로젝트 개발 보고서

- 비트캠프 딥러닝 개발자 과정 장원중

1. MERN 스택이란?
2. 도커 개념 및 설정
3. Node 개발 환경 설정
4. React 개발 환경 설정
5. Express 개발환경 설정
6. MONGODB 개발환경 설정

1. MERN스택이란?

MERN스택을 설명하기 앞서서 MERN스택을 사용하는 이유를 알아보자면 먼저 전략패턴에 대해서 알고 있어야한다. 전략패턴이란 context를 결정하고 자신의 context에서 필요에 따라 변경이 필요한 알고리즘을 분리하고 인터페이스를 통해 외부로 분리한다.

여기서 context는 프레임워크가 설정하며 프론트단에서는(React를 사용하였음) 전략패턴을 M:MONGODB E:Express.js R:React.js N:Next.js 즉 MERN스택으로 구성한다. 이렇게 기능단을 분리해줌으로써 서비스를 분리시키고 MSA구조에서의 개발 작업을 진행하는 것을 말한다. MSA를 간단하게 설명하자면 독립적인 역할을 수행하는 작은 단위의 서비스로 분리하여 설계하는 패턴이라고 할 수 있다. 결국 기능적인 요소들, 즉 서비스를 분리시키고 결합하여 하나의 큰 구조로 형성하는 것을 말한다. 결국 구조적으로는 monolithic구조와 같지만 구조 내부의 관점에서 서비스들을 분리시키고 각각의 서비스들을 처리 할 수 있게 된다. 추가하자면 MSA와 다른구조인 모놀리식 구조에서는 서비스를 분리 시키지 않기 때문에 전략패턴인 MERN스택을 사용하지 않는다.

MERN은 스택구조(LIFO의 구조)이기 때문에 N->R->E->M 순서로 설치를 진행하며, 실행시에는 M->E->R->N의 순으로 실행하게 된다. 이 MERN이라는 4가지 요소를 사용하여 웹사이트 개발을 진행한다.

2. 도커 개념 및 도커설정

도커를 쓰는 최종적인 이유는 데이터를 메모리상에 저장하기 위해서이다. 데이터는 기본적으로 Entity(개체)이기 때문에 디스크에 저장되어야 하는데 메모리상에 저장하기 위해서 도커를 사용한다고 할 수 있다. 결국 DB시스템을 가상으로 실행시키고자 도커를 사용한다고 할 수 있다.

그렇다면 도커란 무엇일까?

도커란 애플리케이션을 신속하게 구축하고 테스트, 배포를 도와주는 소프트웨어 플랫폼이다. 소프트웨어를 컨테이너를 통해 패키징하는데 컨테이너에는 라이브러리, 시스템 도구, 코드 등 소프트웨어 실행에 필요한 모든 것이 포함되어 있다. 결국 도커는 컨테이너 환경에서 독립적으로 애플리케이션을 실행할 수 있도록 컨테이너를 만들고 관리하는 것을 도와주는 도구이다. 도커를 통해 애플리케이션을 실행하면 독립적인 환경에서 일관된 결과를 가져온다는 장점을 가지고 있다. 도커의 이미지란 도커에서 서비스 운영에 필요한 서버 프로그램, 소스코드 및 라이브러리, 컴파일된 실행 파일을 묶는 형태를 말한다. 결국 특정 프로세스를 실행하기 위한 모든 파일과 설정값을 지닌 것으로 더 이상의 설치가 필요없는 하나의 환경을 구성해 놓은 것이라고 할 수 있다. 도커 컨테이너는 이미지를 실행한 상태로 프로세스를 동작시키는 기술이라고 할 수 있다.

컨테이너를 종료시켜도 메모리상에서 삭제되지 않고 남아있으며 컨테이너를 삭제하면 컨테이너에서 생성한 파일을 삭제하는 것이라고 할 수 있다. 한 서버는 여러개의 컨테이너를 가질 수 있으며 각각의 컨테이너는 독립적으로 실행된다.

3. Node개발 환경 설정

Node.js란?

Node.js는 자바스크립트의 런타임이다. 런타임이란 프로그래밍 언어가 구동되는 환경이라고 할 수 있다. Node.js는 자바스크립트로 브라우저가 아닌 서버를 구축하고 서버에서 자바스크립트 언어가 작동되도록 해주는 런타임 환경이라고 할 수 있다. SPA구조의 웹애플리케이션에서 주로 사용된다.

개발환경 설정 방법

1. 노드설치

- 자바에서 jdk와 같다.
- 자바스크립트는 자바와 같다

실행

프로그램, 폴더, 문서, 또는 인터넷 주소를 입력하여 해당 항목을 열 수 있습니다.

열기(Q):

확인

취소

찾아보기(F)...

시스템 변수(S)

변수	값
JAVA_HOME	C:\Program Files\jdk-11
NODE_HOME	C:\Program Files\nodejs
NUMBER_OF_PRO...	8
OnlineServices	Online Services

새로 만들기(W)...

편집(E)...

삭제(L)

노드는 JDK와 개념이 같다. 그래서 HOME를 설정한다

환경 변수 편집

%SystemRoot%\system32

%SystemRoot%

%SystemRoot%\System32\Wbem

%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\

%SYSTEMROOT%\System32\OpenSSH\

C:\Program Files\Git\cmd

%JAVA_HOME%\bin

%NODE_HOME%

path에 환경변수 설정

PS C:\Users\bitcamp\Javabasic\MyProject\api-fronten> node --version

v16.14.0

PS C:\Users\bitcamp\Javabasic\MyProject\api-fronten> npm --version

8.3.1

- o 터미널 열고 node, npm 버전 확인.
- o VSCode의 터미널이 제대로 작동하는지 체크하는 과정

4. React 개발환경 설정

React를 사용하기 위해 VsCode를 <https://code.visualstudio.com/> 에서 설치하여준다. 설치 진행 중 추가작업 선택에서 체크는 모두 하고 설치를 진행한다.그리고 진행에 필요한 확장프로그램들을 설치해준다. npm install create react app을 실행하고프로젝트를생성시킨다.(프로젝트 이름 포함 설정)

Successful Created my-app at C:\Users\bitcamp\Javabasic\MyProject\api-fronten\my-app
Inside that directory, you can run several commands:

```
npm start
  Starts the development server.

npm run build
  Bundles the app into static files for production.

npm test
  Starts the test runner.

npm run eject
  Removes this tool and copies build dependencies, configuration files
  and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd my-app
  npm start

Happy hacking!
```

happy hacking! 포미엔
이래 입력하는 단계
cd my-app
npm start 입력

5. Express 개발환경 설정

Express를 설치하기 위하여 해당 Express를 설치할 폴더에서 CLI프로그램을 실행하여준다. Express를 설치할 폴더에서 폴더를 생성해주는 Linux명령어 mkdir ‘폴더명’을 실행한후 만들어진 폴더 안(cd ‘폴더명’으로 진입)에서 명령어 npx express-generator를 작성한 후 실행하면 express.js 기본설정이 완료된다.

6. MongoDB 개발환경 설정

MongoDB를 사이트에서 파일을 받아 설치하고 MongoDBCompass파일을 실행한다. MongoDB를 도커에 올려주기 위해서 터미널 실행창에서 Docker pull mongo를 통해 도커 컨테이너를 생성하고 ,

```
docker run --name mongodb_docker -e
MONGO_INITDB_ROOT_USERNAME=root -e
MONGO_INITDB_ROOT_PASSWORD=root -d -p
27017:27017 mongo(mongodb진입시 아이디와 비밀번호
설정)를 입력하여 컨테이너를 실행시켜준다.
```

7. 개인프로젝트 개발 부분

<https://github.com/JangWonJong/WJ-project>