



병원 리뷰 및 예약 웹 서비스

20181676 이동엽
20201849 임세나

목차

A table of contents

1

프로젝트(연구, 설계) 목표

2

계획대비 진행상황

3

구성원의 역할분담

4

설계 사양

5

연구 내용

6

중간 결과물

7

프로젝트 향후 계획

8

참고 자료

프로젝트(연구, 설계) 목표

병원 리뷰 및 예약 웹 서비스

현재 수많은 플랫폼에서 리뷰 서비스를 제공하지만, 아직까지 병원비를 공개하는 플랫폼은 없다.
리뷰를 통해 가격 비교를 할 수 있고, 나아가 예약 서비스와 픽업 서비스를 부가 기능으로 제공한다.
따라서, 고령화 사회에 맞춰 고령층을 포함한 모든 연령층이 사용하기 쉬운 웹 리뷰 플랫폼을 제작한다.

계획대비 진행상황

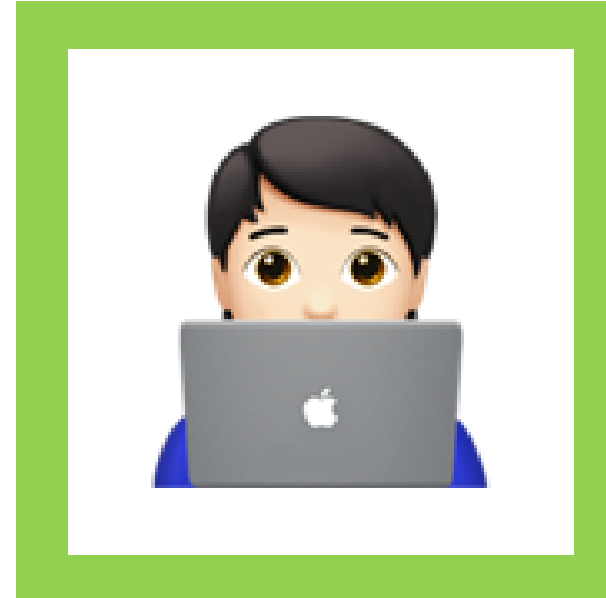
연구내용	3월	4월	5월	6월	7월	8월	9월	10월	11월
주제 선정 및 계획서 작성	🥑								
프로토타입 디자인 설계	🥑	🥑							
데이터베이스 설계	🥑	🥑							
관련 기능 설계		🥑	🥑	🥑					
페이지 기능 검증			🥑	🥑	🥑				
서버 연동 및 통합 검증				🥑	🥑	🥑			
문제점 보안 및 성능 개선						🥑	🥑	🥑	
프로젝트 결과 보고서 작성								🥑	🥑

구성원의 역할분담



20201849임세나(팀장)

- 프론트엔드 웹 서버 개발
- 주제 선정 및 계획서 작성
- 웹 페이지 프로토타입 디자인



20181676이동엽(팀원)

- 백엔드 웹 애플리케이션 서버 개발
- 회의 내용 기록 및 도큐먼트 작성
- 데이터베이스 설계

설계 사양

개발환경

| *Frontend*

- OS
 - macOS
- *Language*
 - JavaScript ES6
- *Framework*
 - React.js 5.0.1

| *Backend*

- OS
 - macOS
- *Language*
 - Java 17
- *Framework*
 - org.springframework.boot 3.0.2
- *Database*
 - h2 2.1.214

게시글 이미지 저장 구현

HTTP Avocado / board / createBoard

POST http://localhost:8080/api/boards

Params Authorization Headers (10) **Body** Pre-request Script Tests Settings

☐ none ☒ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

	Key	Value	Description
<input checked="" type="checkbox"/>	memberId	2	
<input checked="" type="checkbox"/>	title	title1	
<input checked="" type="checkbox"/>	body	body1 ↵	...
<input checked="" type="checkbox"/>	dept	DENTAL	
<input checked="" type="checkbox"/>	image	NgqqJSz_1/PNG 이미지.png ×	

Body Cookies Headers (6) Test Results

Pretty Raw Preview Visualize JSON ↕

```
1 {
2   "id": 4
3 }
```

🌐 Status: 200 OK Time: 195 ms

1

게시글 저장 후, 게시글 이미지도 저장되도록 하였음 (게시글 이미지 여러개도 가능)

```

select b1_0.member_id,b1_0.board_id,b1_0.board_body,b1_0.created_at,b1_0.dept,b1_0.board_status,b1_0.board_title,b1_0.updated_at from board b1_0 where b1_0.member_id in(?,?,?,?,?,?,?,?,?)
select b1_0.member_id,b1_0.board_id,b1_0.board_body,b1_0.created_at,b1_0.dept,b1_0.board_status,b1_0.board_title,b1_0.updated_at from board b1_0 where b1_0.member_id in(2,2,2,2,2,2,2,2,2,2)
2023-08-29T22:06:20.820+09:00 TRACE 2476 --- [qtp288608508-56] org.hibernate.orm.jdbc.bind : binding parameter [1] as [VARCHAR] - [body1]
]
2023-08-29T22:06:20.820+09:00 TRACE 2476 --- [qtp288608508-56] org.hibernate.orm.jdbc.bind : binding parameter [2] as [TIMESTAMP] - [2023-08-29T22:06:20.811352]
2023-08-29T22:06:20.820+09:00 TRACE 2476 --- [qtp288608508-56] org.hibernate.orm.jdbc.bind : binding parameter [3] as [VARCHAR] - [DENTAL]
2023-08-29T22:06:20.821+09:00 TRACE 2476 --- [qtp288608508-56] org.hibernate.orm.jdbc.bind : binding parameter [4] as [BIGINT] - [2]
2023-08-29T22:06:20.821+09:00 TRACE 2476 --- [qtp288608508-56] org.hibernate.orm.jdbc.bind : binding parameter [5] as [VARCHAR] - [WRITE]
2023-08-29T22:06:20.821+09:00 TRACE 2476 --- [qtp288608508-56] org.hibernate.orm.jdbc.bind : binding parameter [6] as [VARCHAR] - [title1]
2023-08-29T22:06:20.821+09:00 TRACE 2476 --- [qtp288608508-56] org.hibernate.orm.jdbc.bind : binding parameter [7] as [TIMESTAMP] - [2023-08-29T22:06:20.811352]
2023-08-29T22:06:20.822+09:00 INFO 2476 --- [qtp288608508-56] p6spy : #1693314380822 | took 0ms | statement | connection 7 | url jdbc:h2:mem:testdb
insert into board (board_id, board_body, created_at, dept, member_id, board_status, board_title, updated_at) values (default, ?, ?, ?, ?, ?, ?)
insert into board (board_id, board_body, created_at, dept, member_id, board_status, board_title, updated_at) values (default, 'body1
', '2023-08-29T22:06:20.811+0900', 'DENTAL', 2, 'WRITE', 'title1', '2023-08-29T22:06:20.811+0900');
2023-08-29T22:06:20.823+09:00 INFO 2476 --- [qtp288608508-56] i.w.c.d.b.application.BoardImageService : fileName: PNG 이미지.png
2023-08-29T22:06:20.827+09:00 TRACE 2476 --- [qtp288608508-56] org.hibernate.orm.jdbc.bind : binding parameter [1] as [BIGINT] - [4]
2023-08-29T22:06:20.827+09:00 TRACE 2476 --- [qtp288608508-56] org.hibernate.orm.jdbc.bind : binding parameter [2] as [VARCHAR] - [/Users/2dongyeop/Desktop/2023/08/29/1fa80876-e493-4b26-9168-07d7bc855f5f_PNG 이미지.png]
2023-08-29T22:06:20.827+09:00 INFO 2476 --- [qtp288608508-56] p6spy : #1693314380827 | took 0ms | statement | connection 7 | url jdbc:h2:mem:testdb
insert into board_image (id, board_id, saved_url) values (default, ?, ?)
insert into board_image (id, board_id, saved_url) values (default, 4, '/Users/2dongyeop/Desktop/2023/08/29/1fa80876-e493-4b26-9168-07d7bc855f5f_PNG 이미지.png');
2023-08-29T22:06:20.831+09:00 INFO 2476 --- [qtp288608508-56] p6spy : #1693314380831 | took 0ms | commit | connection 7 | url jdbc:h2:mem:testdb

```


연구 내용

로그 포맷 수정

- [날짜] [로그 레벨] [스레드명] [로깅이 발생한 클래스.메서드] - [메시지]

```
Run CapstoneDesignApplication x
Console
[2023-09-10 21:25:02:27508][WARN ][qtp2082583161-55][HttpChannel.handleException] - handleException /api/auth/login/members io.netty.channel.AbstractCha
[2023-09-10 21:25:48:73237][DEBUG ][qtp2082583161-29][SQL.logStatement] -
select
  m1_0.member_id,
  m1_0.created_at,
  m1_0.member_email,
  m1_0.member_photo_path,
  m1_0.member_nickname,
  m1_0.member_password,
  m1_0.member_phonenumber,
  m1_0.updated_at
from
  member m1_0
where
  m1_0.member_email=?

[2023-09-10 21:25:48:73239][TRACE ][qtp2082583161-29][bind.logBinding] - binding parameter [1] as [VARCHAR] - [sena@naver.com]
[2023-09-10 21:25:48:73240][INFO ][qtp2082583161-29][p6spy.logSQL] - #1694348748662 | took 0ms | statement | connection 8 | url jdbc:h2:mem:testdb
select m1_0.member_id,m1_0.created_at,m1_0.member_email,m1_0.member_photo_path,m1_0.member_nickname,m1_0.member_password,m1_0.member_phonenumber,m1_0.up
select m1_0.member_id,m1_0.created_at,m1_0.member_email,m1_0.member_photo_path,m1_0.member_nickname,m1_0.member_password,m1_0.member_phonenumber,m1_0.up
[2023-09-10 21:25:48:73340][INFO ][qtp2082583161-29][JwtTokenProvider createAccessToken] - 생성된 accessToken: eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJzZW5hQG5h
[2023-09-10 21:25:48:73341][INFO ][qtp2082583161-29][JwtTokenProvider createRefreshToken] - 생성된 refreshToken: eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJzZW5hQG5h
[2023-09-10 21:25:48:73373][INFO ][qtp2082583161-29][AuthService.loginMember] - redis : sena@naver.com님의 리프레쉬 토큰eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJzZW5hQG5h
[2023-09-10 21:25:48:73380][INFO ][qtp2082583161-29][p6spy.logSQL] - #1694348748802 | took 0ms | commit | connection 8 | url jdbc:h2:mem:testdb
```

연구 내용

| 실행 스크립트 작성

```
APP_NAME_JAR=capstone-*.jar

# 애플리케이션 실행 시점에 필요한 키
APP_JASYPT_PASSWORD={jasypt 비밀키}

# 임시 경로 : 추후 변경해야 함
APP_DIRECTORY="/home/ec2-user"
APP_PATH="$APP_DIRECTORY/$APP_NAME_JAR"

# 로그 경로
APP_LOG="/dev/null"

# 프로세스 아이디
APP_PID=$(pgrep -f $APP_NAME_JAR | head -n 1)

if [ -z "$APP_PID" ]; then
    # PID가 없으면~ 실행중이지 않은거니까 실행
    echo "project avocado start"
    nohup java -jar -Djasypt.encryptor.password=$APP_JASYPT_PASSWORD $APP_PATH > $APP_LOG 2>&1 &
else
    # PID가 있으면 이미 실행중임을 알려주기
    echo "project avocado is already started in process(id) : $APP_PID"
fi
```

중간 결과물

리뷰 상세 조회 기능 개발

React App

Swagger UI

localhost:3000/reviews/15

Naver Youtube TB board - 애자일... Swagger UI 드림코딩 아카데미 dev front tech-blog 행사 취업 아자자! hanbat etc

Avocado 리뷰 게시판

을지대학병원에서 충치 치료받은 후기 · 작성자: 임세나

을지대학병원 · DENTAL · ★5

꼼꼼한 검사 후에 치료를 진행했어요.

친절하니 좋아요.

(사람이 많아 대기 시간이 조금 길었네요 ㅠㅠ)

DevTools is now available in Korean! Always match Chrome's language Switch DevTools to Korean Don

Elements Console Sources Network Performance Memory Application >>

Filter

☐ Invert ☐ Hide data URLs ☐ Hide extension URLs

All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other ☐ Blocked requests ☐ 3rd-party requests

500000 ms 1000000 ms 1500000 ms 2000000 ms 2500000 ms

Name

main.6040002170001...

main.acf6fd24b8f00a9...

main.acf6fd24b8f00a9...

main.144f4f3e768a27d...

main.144f4f3e768a27d...

review

bundle.js

ws

manifest.json

logo192.png

reviews

15

bundle.js

ws

details

Headers

Preview

Response

Initiator

Timing

Cookies

1

{

"data": {

"writer": "임세나",

"title": "을지대학병원에서 충치 치료받은 후기",

"body": "<p>꼼꼼한 검사 후에 치료를 진행했어요.</p><p>친절하

"starPoint": 5,

"targetHospital": "을지대학병원",

"reviewReplyList": [],

"targetDept": "DENTAL",

"photoPath": "path1"

}

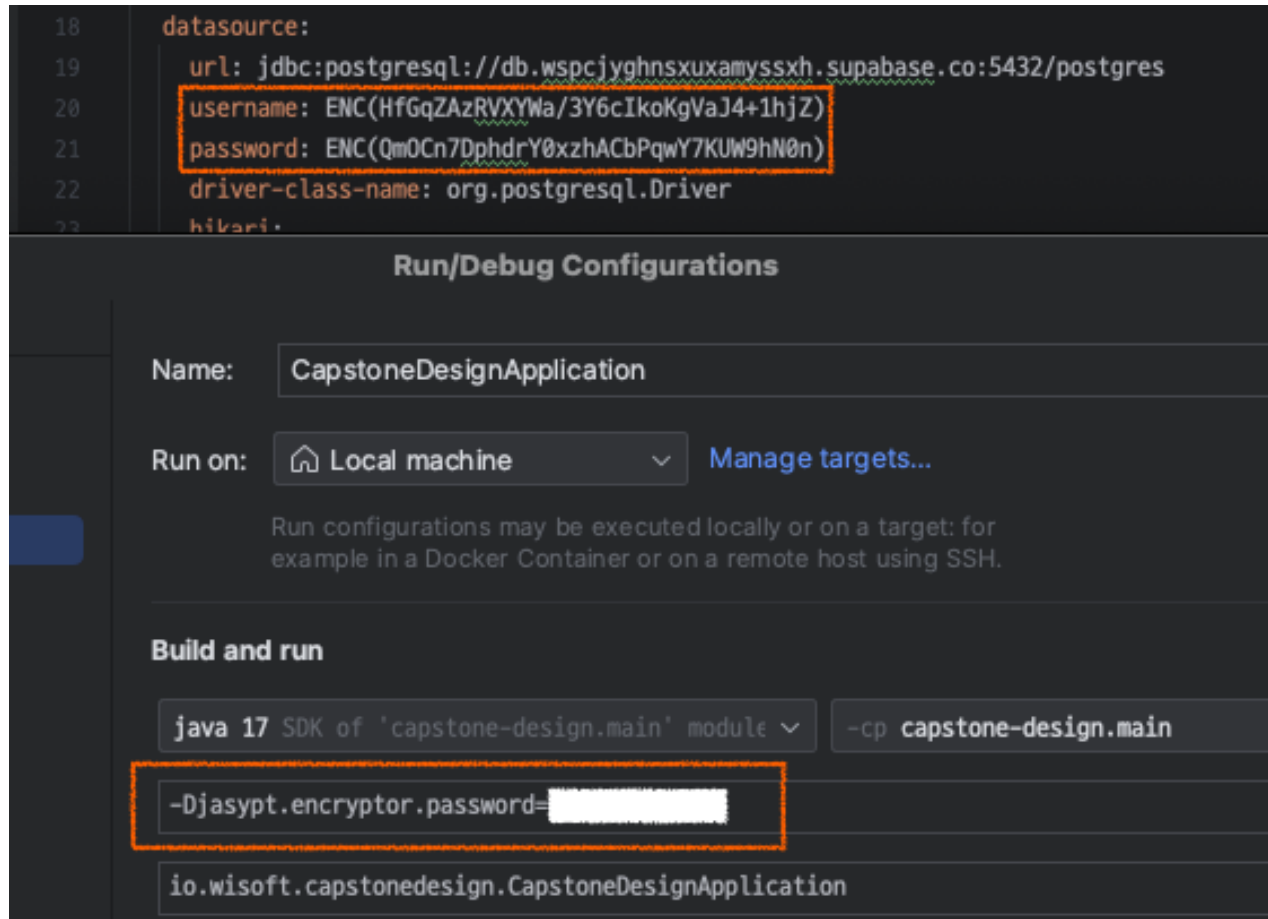
}

©Saebyeol Yu. Saebyeol's PowerPoint

중간 결과물

Springboot 설정파일 암호화

- DB 접속 정보들이 ENC() 꼴로 암호화 됨
- 이후 실행 시점에는 VM Option으로 복호화할 수 있는 password를 제공해야 애플리케이션을 실행 가능!
- → 이로써 소스코드가 노출이 되어도, 당장 DB 접속 정보를 막을 수 있고, 애플리케이션 실행조차도 못하게 막을 수 있다.



중간 결과물

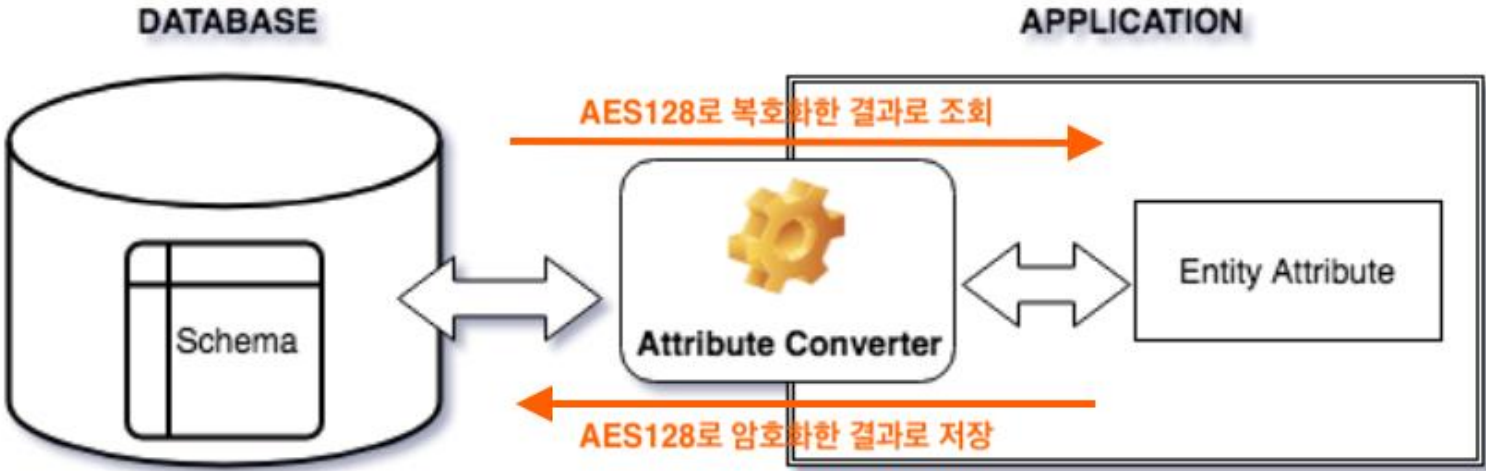
회원 및 의료진 개인정보 암호화(AES128)

member [Avocado@supabase] × before : 비밀번호만 암호화시킨 사전데이터

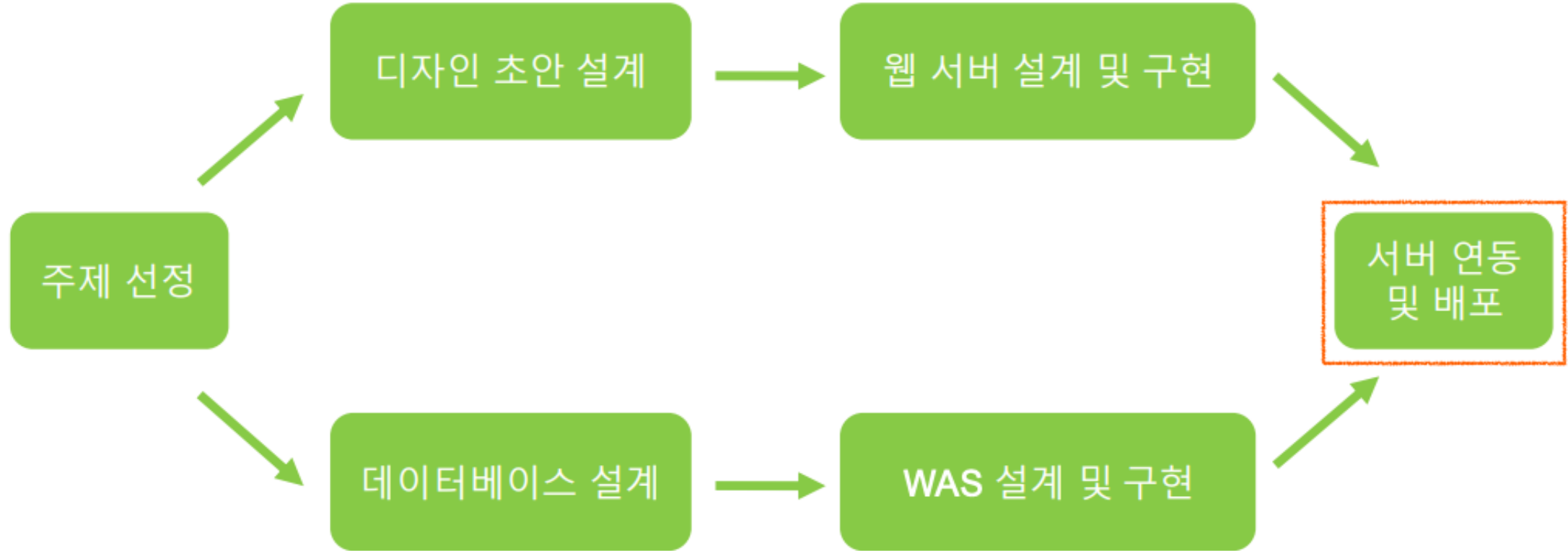
	member_email	member_photo_path	member_nickname	member_password
1	ldy_1204@naver.com	<null>	이동엽	\$2a\$10\$h0RMmUWPno8k
2	sena@naver.com	<null>	임세나	\$2a\$10\$3BoYwjR6Sih5
3	choi@naver.com	<null>	최민석	\$2a\$10\$kNCZPdmfltJc

staff [Avocado@supabase] × after : 개인정보가 모두 암호화된 사전데이터

	staff_email	staff_license_path	staff_name	staff_passw
	82e9129fc50ab3d1...	temp_license1	144d619194637a...	\$2a\$10\$0UPem4f
	0be8f20b3d8061e4...	temp_license2	1c71ed6378538c...	\$2a\$10\$yNDfoPz
	e65718cb63f8b78d...	temp_license3	e9fcf32652095f...	\$2a\$10\$I5i/mNf
	ee53c0bf8c43497192e	temp_license4	39293d42dc9304...	\$2a\$10\$L6f/wNv



프로젝트 향후 계획



참고 자료

| 참고자료

리액트를 다루는 기술 - 김민준

모던 자바스크립트 Deep Dive - 이웅모

자바 ORM 표준 JPA 프로그래밍 - 김영한

모던 자바 인 액션 - 라울 가브리엘 우르마

이펙티브 자바 - 조슈아 블록

마무리

Thanks!