



## 리액트를 활용한 소상공인을 위한 반응형 웹 인터페이스 개발에 관한 연구

### Research on Developing a Responsive Web Interface for Small Businesses Using React

이준용 (June Yong Lee)<sup>1 i</sup>

<sup>1</sup> 한국외국어대학교, 컴퓨터전자시스템공학부; [diziyong@hufs.ac.kr](mailto:diziyong@hufs.ac.kr) [diziyong1523@gmail.com](mailto:diziyong1523@gmail.com)

**한글 요약:** 규모가 큰 기업의 경우 영업 데이터베이스를 구축하여 효율적인 영업활동을 하고 있지만 영세업자들의 경우 다각적이고 복합적인 정보를 누구나 손쉽게 관리할 수 있는 인터페이스를 제공하여 소상공인들의 영업활동에 효과적인 도움을 주는 시스템이 필요하다. 본 논문에서는 매장 정보를 자동으로 수집하는 크롤링 서버에서도 구동하는 리액트 기술을 제안한다. 그리고 리액트 기술을 이용한 다양한 데이터의 효과적인 전달을 위한 마우스, 키보드 기반의 데스크탑 환경과 터치기반의 스마트폰 환경에서도 편리하게 사용할 수 있는 반응형 웹 사용자 인터페이스 구축을 제안한다. 제안하는 방식들은 리액트 기술의 비동기식 함수 처리 방식의 도입으로 기존의 자바스크립트 방식보다 대량의 데이터의 신속한 호출이 기대된다.

**핵심어:** 반응형 웹 인터페이스, 리액트, 비동기식 처리, 정보구조도 설계

본 논문은 2024 학년 2 월 졸업을 위해 제출된 한국외국어대학교 컴퓨터공학부 졸업논문이다.

지도교수: 윤일동 교수

서명: 

참여한 캡스톤 설계

설계명: 웹 크롤링을 활용한 매장 정보 수집 및 라벨링 시스템

팀원명: 이준용, 박경렬, 윤세현, 정찬

**영문 요약:** Although large corporations have established sales databases for efficient sales activities, small-scale entrepreneurs need a system that provides an interface that anyone can easily manage, offering diverse and complex information. This paper proposes the use of React technology, which can also operate on crawling servers that automatically collect store information. Furthermore, it suggests the development of a responsive web user interface that can be conveniently used in both mouse and keyboard-based desktop environments as well as touch-based smartphone environments, utilizing React technology for the effective delivery of various data. The proposed methods anticipate rapid retrieval of large volumes of data due to the adoption of asynchronous function processing in React technology, an improvement over traditional JavaScript methods.

**Keywords:** Responsive Web Interface, React, Asynchronous Processing, Information Architecture Design

## 제 1 장 서론 - Introduction

### 1.1 연구 배경

산업에서의 자영업이란 회사 등의 법인을 설립하지 않고 스스로 하는 사업을 말한다. 많은 직장인들이 퇴직, 혹은 퇴사 후 생계를 위해 쉽게 떠올리는 것이 바로 창업이다. 하지만 창업 후 3 년내에 문을 닫는 업종들의 비율이 빠르게 늘고 있다. 자영업자들은 상권, 고객 분석을 통해 실패 원인을 공부하게 된다.



Copyright: © 2023 by the authors.  
Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

이들은 가게의 위치, 주변 인구의 직업, 영업시간 등 단편적인 분석 결과가 도출될 수 있다.

자영업을 통해 실패하는 가장 큰 이유는 짧은 창업 준비기간에 있다. 중소기업청 조사 보고서에 따르면 음식점 창업자의 72.5%가 6개월 미만의 창업 준비 기간을 거쳐 가게를 연다고 정리되어 있다[1]. 이렇게 짧은 판단과 단편적인 시장 조사로 인해 창업 후 적자로 인해 문을 닫게 되는 가게들이 많다. 자영업의 성공률이 특히 낮은 이유는 낮은 진입장벽과 트렌드 업종에 집중하는 경향이 그 이유이다. 대한민국은 유난히 유행 속도가 빨라 다양한 업종들의 트렌드 역시 빠르게 변화한다. 유행하는 가게들이 우후죽순으로 생겨나다 창업에 대한 진입장벽이 낮아 가게를 쉽게 오픈할 수 있다 보니 이러한 상황이 반복되는 것이다. 그에 반대되는 것이 유행의 흐름과 관계없이 고객들이 꾸준히 유입될 아이템을 선정하는 것으로 비교적 안정적이라고 볼 수 있다. 하지만 그렇다고 해서 트렌드에 아예 무관심 할 수 없다. 메뉴나 업종 이외에도 광고 방법, 차별화된 서비스, 인테리어, 결제 및 주문 시스템 등이 유행에 뒤처지게 되면 적자를 면치 못할 것이다.

규모가 큰 기업의 경우 영업 DB 를 구축하여 효율적인 영업활동을 하고 있지만, 영세업자들의 경우 자영업이 가지는 다양한 업종과 데이터의 크기, 복합적인 조건들을 파악하기 힘들뿐만 아니라, DB 구축이 힘들고 서로 다른 매장 정보나 영업 현황을 공유할 수 없어 어려움을 겪고 있다. 다각적이고 복합적인 정보를 누구나 손쉽게 관리할 수 있는 인터페이스를 제공하여 소상공인들의 영업활동에 효과적인 도움을 주는 시스템이 필요하다.

## 1.2 연구 목적

본 연구의 목적은 영업 진행 시 영업을 진행해야 할 매장에 대한 업태와 업종 정보를 제공하고, 가독성을 높이고 검색 비용과 시간을 절감하며 다양한 영업의 활성화에 기여한다. 이를 위해 다양한 매장 정보를 자동으로 수집하는 크롤링(Crawling) 서버에서도 구동하는 리액트(React) 기술을 활용한 다양한 데이터의 효과적인 전달을 위한 마우스, 키보드 기반의 데스크탑 환경과 터치기반의 스마트폰 환경에서도 편리하게 사용할 수 있는 반응형 웹(Responsive Web) 사용자 인터페이스 구축을 제안한다.

## 1.3 연구 방법

본 논문에서 리액트(React) 기술을 이용한 크롤링 서버에서 이용가능한 반응형 웹 인터페이스를 제안한다. 반응형 웹 인터페이스 구축에 앞서 프로젝트의 개요와 산출물, 정의, 약어를 확실히 정리한다. 정해진 시간안에 무엇을 개발할 것인지와 어떤 기능을 어떻게 구현을 할 것인지 확실하게 함으로써 개발 과정 중간에 목표에 벗어나는 일을 막고자 함이다.

다음으로 계획서 작성 및 일정표 작성을 통해 검토회의 일정과 진행 방법에 대해서 논의한다. 각자 담당 업무와 직무 기술에 대해 이해하고, 지정된 일정안에 각각의 기능 구현이 완성되어야 피드백과정을 통해 최종 프로그램 시연까지 완성도 있는 반응형 웹을 개발하기 위함이다.

위와 같은 논의 과정을 문서화하고 개발환경 구축과 WBS(work Breakdown Structure)를 작성함으로써 연구의 목표와 업무 분류 체계를 시각적으로 구체화한다.

## 1.4 논문 구성

본 논문은 전체 4 장으로 구성되며 각 장에서는 다음과 같은 내용을 담고 있다. 1 장에서는 크롤링 서버를 위한 리액트 기반 반응형 웹 구축을 정의하고 연구의 목적과 필요성에 대한 연구 배경을 논의한다. 특히 자영업과 소상공인에게 크롤링 서버를 위한 반응형 웹 구축이 필요한 현실적인 이유를

기술하고 문제를 설명한다. 그리고 본 논문의 목적과 문제 해결을 위한 방법을 요약하여 제시하며 본 논문의 구성에 대해서 소개한다. 2 장에서는 본 논문의 관련 연구로서 유저 인터페이스를 구축하기전 프로그램의 전체적인 구상도를 제시한다. 3 장에서는 본 논문의 실험 연구 방법으로서 앞서 제시한 리액트 기술로 크롤링 서버를 위한 반응형 웹을 구축하는 방법을 단계적으로 설명한다. 4 장에서는 실험 연구를 통한 연구의 결과물이 제시된다. 5 장에서는 본 연구의 결론으로 수행한 연구 성과를 정리하고 반응형 웹 인터페이스 구축을 위해 본 논문에서 제시한 기법에 대해 성과와 시사점을 요약해서 제시한다. 6 장에서는 연구 수행의 결과를 통해서 분석하고 토론하는 내용을 제시한다.

## 제 2 장 관련 연구

### 2.1 시스템 구성도 및 IA 설계

웹 애플리케이션의 시스템 구성도는 전체 프로젝트 팀에게 시스템을 이해하고 관리하는 데 도움을 주는 중요한 도구이다. 시스템 구성도를 작성하면 전체 시스템의 구조와 구성 요소를 시각적으로 이해하기 쉽다. 이를 통해 개발자, 디자이너, 프로젝트 관리자, 기타 이해 관계자들이 시스템의 작동 방식을 빠르게 파악할 수 있다.

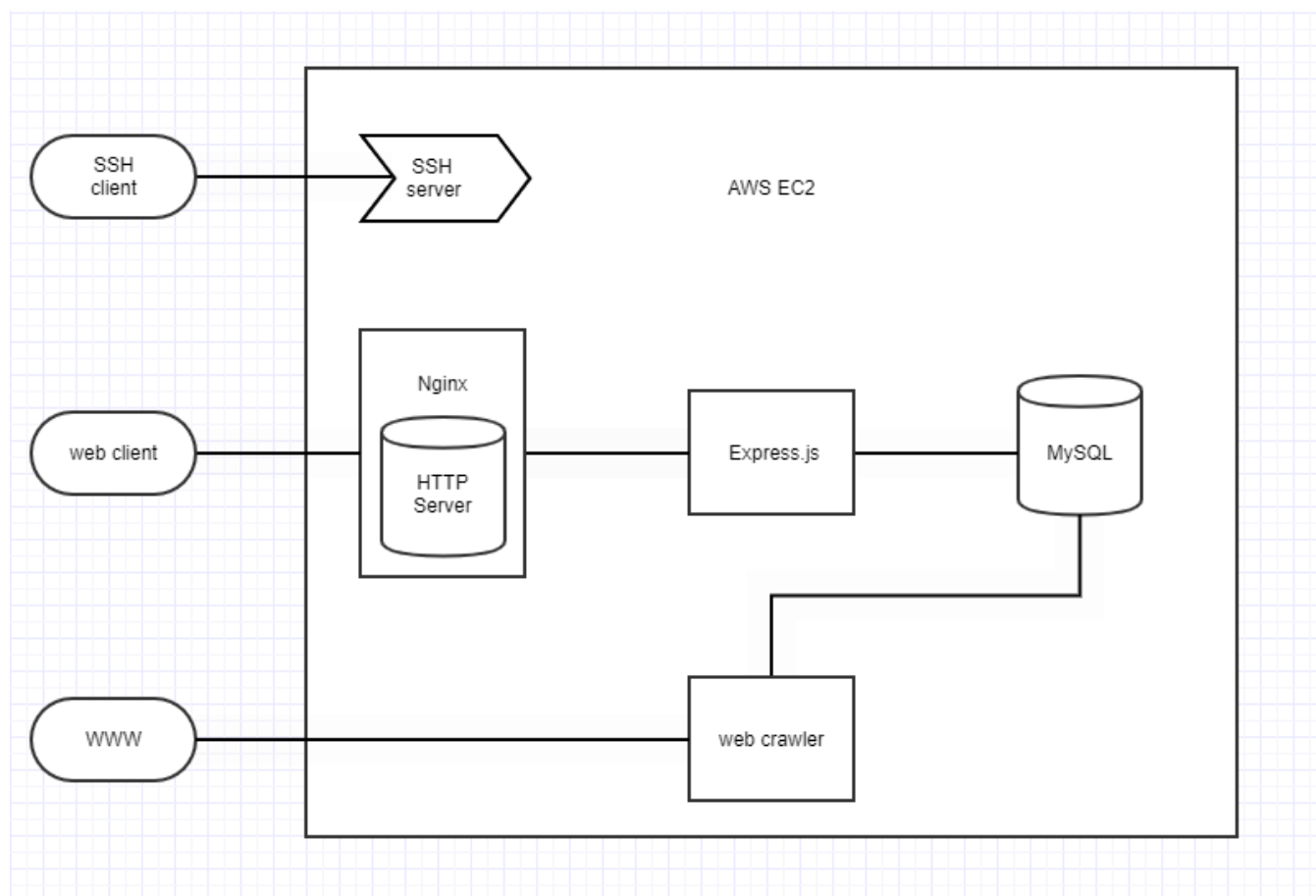


Photo by 정찬<sup>1</sup>

**그림 1.** 시스템 구성도

그림 1 은 AWS EC2 환경에서 구동하는 본 프로젝트의 시스템 구성도이다. 프로그램은 매일 정해진 시간에 웹에서 업데이트되는 크롤링된 매장 데이터들이 MySQL 데이터베이스에서 기존 데이터와의 비교를 통해 데이터 정제과정을 거친 후 저장된다. 메인 서버에서는 업데이트된 매장 데이터를 Api 서버를 통해 web client 와 통신하여 클라이언트 화면에 출력에 반영이 된다. 각각의 서로 다른 환경과 프레임워크를 연동해서 시스템이 구동하게 해주기에 AWS EC2 환경에서 연구 개발을 제안한다.

<sup>1</sup> [www.gliffy.com](http://www.gliffy.com) 웹 애플리케이션을 이용하여 시스템 구성도를 그렸다.

정보구조(IA, Information Architecture)는 서비스의 전체 그림과 흐름을 결정하는 콘텐츠 구조도이다. 이는 사용자에게 언제 어떻게 정보를 제공하지를 결정하기위해 와이어프레임 단계에서 반드시 고려해야 한다. 위 그림 2는 트리구조로 개발자가 전체 구조를 쉽게 이해할 수 있어 반응형 웹을 구축하기위해 작성한다.

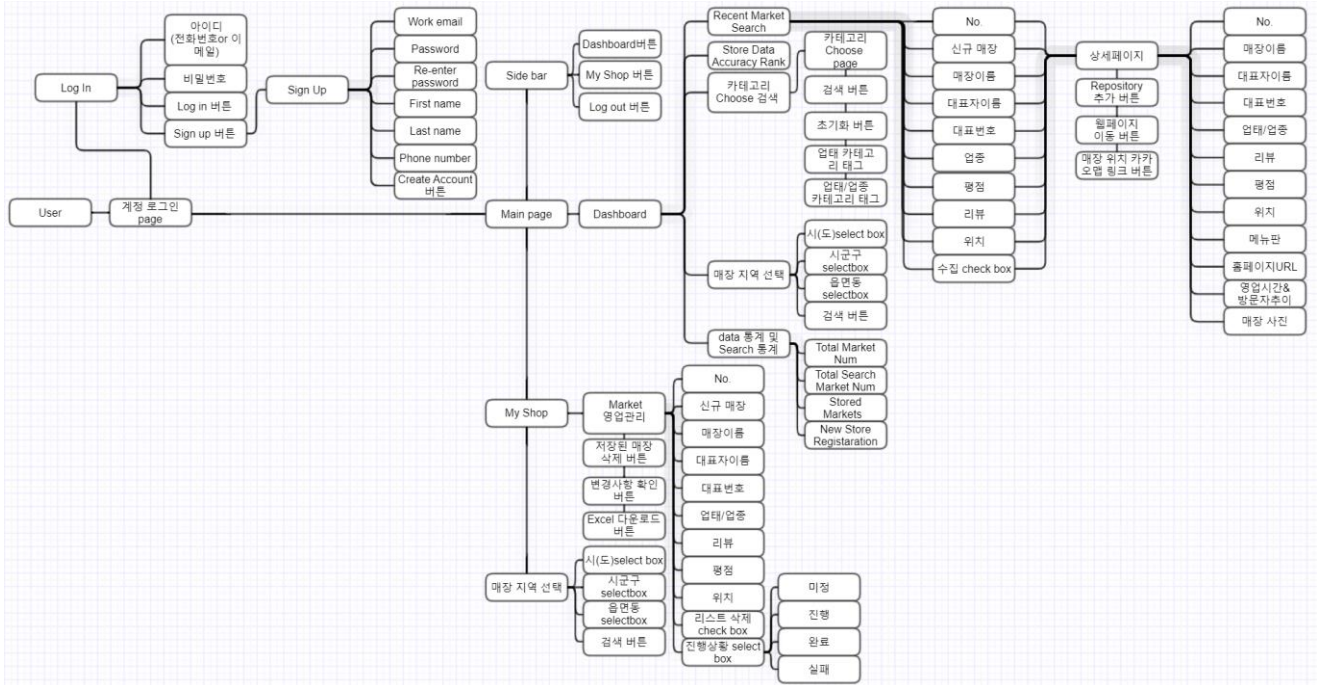


Photo by 이준용<sup>2</sup>

그림 2. IA (Information Architecture) 정보구조도

그림 2의 전체적인 구조는 로그인 페이지와 회원가입 페이지를 첫 화면으로 구성하였으며, 다음으로 Main 페이지, Dashboard 페이지, My Shop 페이지, 상세 페이지로 구성된다. 페이지별 상세 내용은 본 논문 3장에서 제시한다.

<sup>2</sup> [www.figma.com](http://www.figma.com) 웹 애플리케이션을 이용하여 정보구조도를 그린 그림이다.

## 2.2 화면 정의서 및 WBS

여러 다양한 기능들이 포함되어 있는 리액트 기반 반응형 웹을 구축하기 위해서는 개발하기 전 화면 정의서를 작성할 것을 제안한다. 정책, 기능, 화면정의를 한 번에 정의할 수 있기에 직관적이고 시각적으로 각각의 페이지의 기능과 유저 인터페이스를 어떻게 구성할 것인지 인지하기 좋다.

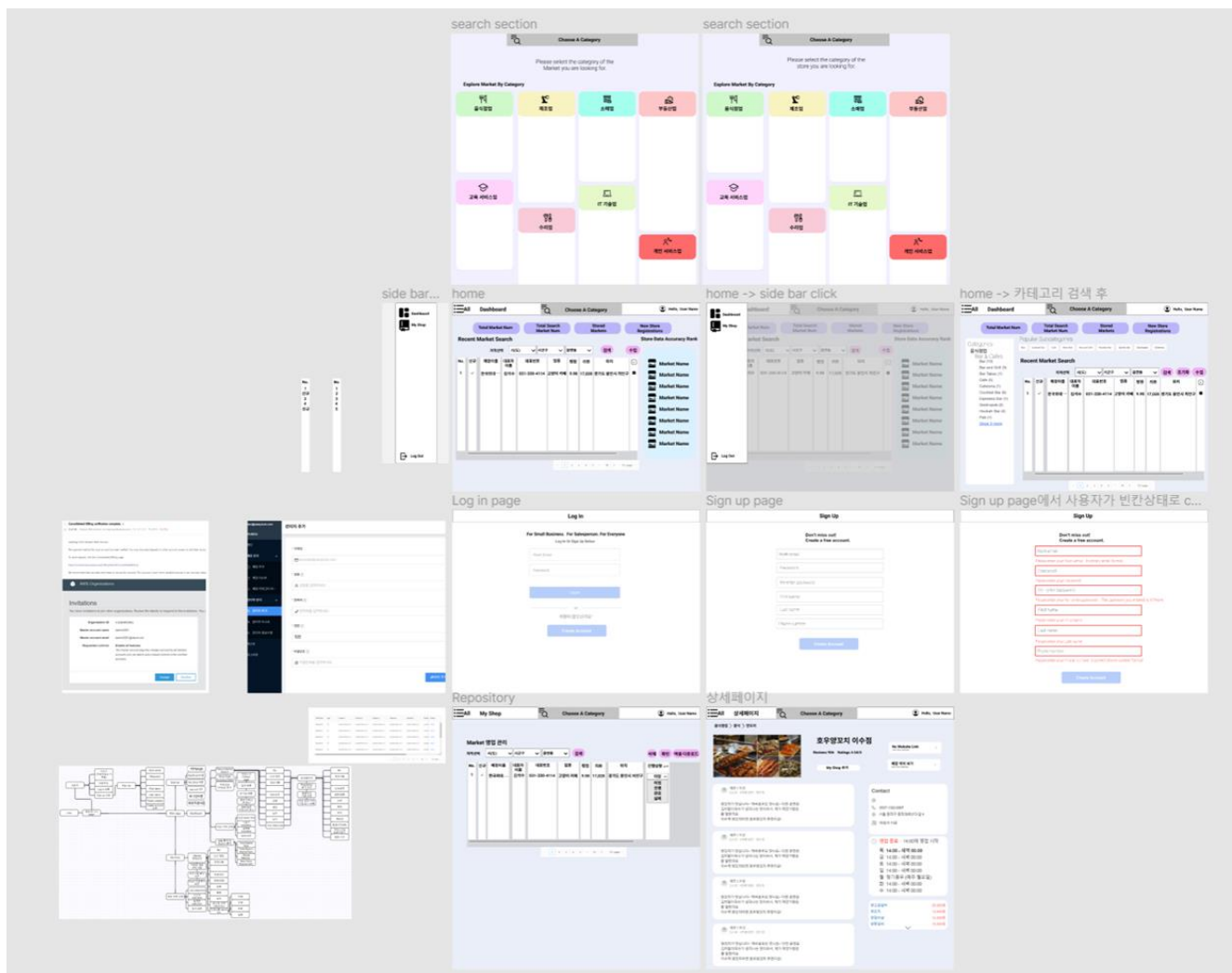


Photo by 이준용<sup>3</sup>

그림 3. Front Web Page 화면 정의서

그림 2 인 IA 설계를 바탕으로 그림 3 과 같이 화면 정의서를 작성한 그림이다.

<sup>3</sup> [www.figma.com](http://www.figma.com)

Photo by 이준용<sup>4</sup>

그림 4. Work Breakdown Structure (WBS)

그림 4 는 소프트웨어의 개발하기 위해 수행하는 작업을 계층적으로 정리한 산출물인 WBS 를 작성한 모습이다. 0 레벨은 상품명, 1 레벨은 기능(요구사항), 2 레벨은 라이프 사이클 및 상세기능 관점으로 분류하여 작성한다. 각 워크패키지(work package)마다 산출물을 정의하고 각 설계의 완료를 판단할 수 있어야 한다. 또한 WBS 는 프로젝트 계획 및 통제의 핵심이 되므로 책임과 역할을 명확히 하고 개발 업무를 작은 단계로 분할하여 추정(일정, 자원)의 정확성을 높여준다.

### 3. 실험 방법

#### 3.1 크롤링 개념 및 구현 과정

데이터 크롤링시 라이브러리는 웹 애플리케이션 테스트를 위한 프레임워크인 셀레니움(Selenium)과 Html 와 XML documents 를 파싱하는 뷰티풀 수프(Beautiful Soup)를 Chrome 환경에서 Python 을 이용해 사용한다[2]. 이때 네이버 지도(NAVER Map)와 카카오맵(Kakao Map)을 이용해 매장의 정보들을 크롤링하기 위해 검색어를 시/도와 시/군/구와 업종명으로 분류한다. 업종명은 크롤링한 업태/업종 자료들로 1,2,3 차 분류를 자동화하고, Html Tag, Xpath, CSS Selector 등을 찾아가며 검색한 데이터들을 담고 있는 경로를 분석한다. 또한 검색어의 차이와 태그 변경과 같은 예외의 경우 그에 맞게 예외처리 코드를 작성한다. 이를 통해 네이버 지도와 카카오맵의 매장들을 크롤링한다.

<sup>4</sup> [www.figma.com](http://www.figma.com)

### 3.2 Back-end 설계 및 구현 과정

크롤링된 데이터로 매장이름, 도로명 주소, 지번주소, 매장 분류 카테고리, 대표 매장 번호, 영업시간, 리뷰 및 리뷰의 개수, 평점, 메뉴 및 가격 등의 다양한 종류의 데이터를 분류하고 할당해줄 수 있는 라벨링 시스템을 구현하였다. DB 쿼리 처리시간을 1 초 이하로 제한하여 라벨링된 데이터를 저장하고 검색하는데 원활하게 도와준다.



(a)

#	이름	데이터 유형	길이/소수점	부호 없음	NULL 허용	0으로 채움	기본값	표현식
1	store_no	INT	32	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT	
2	store_name	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	"	매장 이름
3	cat_1	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	시,도
4	cat_2	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	시,군,구
5	cat_3	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	읍,면,동
6	telephone	VARCHAR	20	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	전화번호
7	addr_1	VARCHAR	21	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	업종
8	addr_2	VARCHAR	12	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	세부 업종
9	addr_3	VARCHAR	18	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	도로명 주소
10	rb_addr	VARCHAR	39	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	평점
11	rating	DECIMAL	3,2	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL	영업시간
12	business_hour	VARCHAR	200	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	출력 이미지 주소
13	homepage	VARCHAR	200	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	리뷰 숫자
14	review_count	INT	11	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL	이미지1
15	img_url1	VARCHAR	400	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	이미지2
16	img_url2	VARCHAR	400	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	이미지3
17	img_url3	VARCHAR	400	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	카카오 store_code
18	kakao_store_code	INT	32	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL	네이버 store_code
19	naver_store_code	INT	32	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL ON UPDATE CURRENT_TIMESTAMP	최근 업데이트
20	latest_update	DATETIME		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	CURRENT_TIMESTAMP	생성일자
21	create_time	DATETIME		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

(b)

Photo by 정찬<sup>5</sup>

그림 7. (a) 요구분석을 통해 필요한 데이터 추출 (b) MYSQL DB 설계

웹 크롤링된 데이터를 저장하는 관계형 데이터베이스인 MYSQL로 스키마를 구성한 결과로 그림 7(a)과 그림 7(b)이다. 관계형 데이터베이스 모델의 주요 이점은 위 그림과 같이 직관적인 데이터 표현 방법을 제공하고 관련 데이터 포인트에 쉽게 액세스할 수 있다는 점이다. 또한, 관계형 데이터베이스를 사용하는 이유는 데이터 간의 매우 복잡한 관계를 보여주는 데에도 탁월하기 때문에 데이터가 데이터베이스의 사전 정의된 관계형 스키마를 준수하는 한 더 많은 테이블의 데이터를 참조할 수 있기 때문이다.

### 3.3 Front 화면 출력 설계 및 구현 과정

프론트 화면 배포는 크롤링 후에 라벨링된 데이터들이 데이터베이스에 저장되고 매장 데이터들을 서버안에서의 API와 브라우저와 node.js를 연결하여 사용할 수 있는 프로미스(Promise) 기반 HTTP 클라이언트 라이브러리를 통하여 JSON 데이터를 요청 및 응답 데이터변환을 통해 사용자가 홈페이지와 상호작용할 수 있게 하는 것으로 구현한다.

React npm 명령어를 사용하여 기본적인 npm을 제작한 후에 node\_modules와 package.json에서 모듈(module)과 모듈에 관한 의존성(Dependency)을 관리하며 "npm install <패키지이름>" 명령어로 사용할 기본적인 모듈을 다운받고 관리한다.

<sup>5</sup> MYSQL에서 작성된 스키마 부분이다.

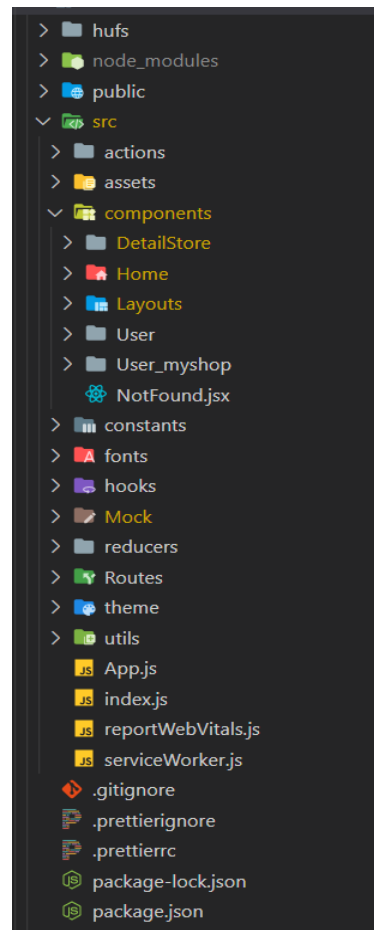


```

@emotion/styled": "^11.8.1",
"@faker-js/faker": "^6.3.1",
"@fortawesome/fontawesome-svg-core": "^6.1.1",
"@fortawesome/react-fontawesome": "^0.1.18",
"@iconify/react": "^3.2.1",
"@material-ui/icons": "^4.11.3",
"@mui/icons-material": "^5.6.2",
"@mui/lab": "^5.0.0-alpha.80",
"@mui/material": "^5.7.0",
"@mui/styled-engine-sc": "^5.6.1",
"@mui/x-data-grid": "^5.10.0",
"@mui/x-data-grid-generator": "^5.10.0",
"@mui/x-data-grid-pro": "^5.10.0",
"@testing-library/jest-dom": "^5.16.1",
"@testing-library/react": "^12.1.2",
"@testing-library/user-event": "^13.5.0",
"apexcharts": "^3.35.0",
"axios": "^0.25.0",
"change-case": "^4.1.2",
"chart.js": "^3.7.0",
"date-fns": "^2.28.0",
"formik": "^2.2.9",
"history": "^5.3.0",
"i18next": "^21.8.0",
"lodash": "^4.17.21",
"moment": "^2.29.3",
"notistack": "^2.0.4",
"numeral": "^2.0.6",
"prettier": "^2.6.2",
"prop-types": "^15.8.1",
"react": "^17.0.2",
"react-apexcharts": "^1.4.0",
"react-chartjs-2": "^4.0.1",
"react-dom": "^17.0.2",
"react-helmet": "^6.1.0",
"react-helmet-async": "^1.3.0",
"react-modal": "^3.15.1",
"react-router-dom": "^6.2.1",
"react-scripts": "5.0.0",
"react-slick": "^0.28.1",
"simplebar": "^5.3.6",
"simplebar-react": "^2.3.7",
"slick-carousel": "^1.8.1",
"source-map-explorer": "^2.5.2",
"styled-components": "^5.3.5",
"web-vitals": "^2.1.3",
"webfontloader": "^1.6.28",

```

(a)



(b)

Photo by 이준용<sup>6</sup>

그림 6. (a) 프로젝트에 사용된 모듈 패키지 의존성 (b) 화면 요구사항 기능을 기준으로 폴더별로 나누어 개발

React와 hook, axios를 통해 모든 데이터 요청 및 반환을 비동기식으로 처리한다[3]. 리엑트는 서로 다른 패키지들의 의존성을 검사 및 관리하여 취약점이 존재하는 패키지의 메이저 버전으로 변경해주기 때문에 선택하였다. 여러 폴더를 생성하여 같은 기능을 하는 파일들을 모아서 관리할 수 있기에 코드 간결화와 디버깅에도 도움이 된다.

Formik과 cookie, session을 통해 리덕스(Redux)와 리코일(Recoil) 패키지로만 가능한 비동기처리들은 최대한 모든 데이터를 가능한 많이 처리할 수 있도록 비동기식으로 처리하도록 연구한다[4]. 이는 비동기식으로 처리함으로써 네트워크 오버헤드를 줄이고 여러 사용자들이 로딩과 끊김 없이 서버에 과부하를 주지 않게 하여 안전하게 영업매장을 관리할 수 있게 개발하기 위함이다.

<sup>6</sup> VSC(Visual Studio Code)에 작성된 package.json 파일과 폴더 구성이다.

#### 4. 결과분석

위 실험을 바탕으로 결과로서 반응형 웹 화면들이다.



그림 8. (a) 로그인 페이지 구성 및 설명 (b) 로그인 페이지 반응형 기능

웹 크롤링 서버를 위한 리액트 기반 반응형 웹을 사용하려는 모든 유저는 로그인 화면에서 로그인을 수행하여야 한다. 내부 대시보드와 마이페이지에서 영업을 위해 검색되어 보여지는 매장 상세 정보들은 중요한 정보들이기 때문에 회원이 아닌 권한이 없는 유저에게 보이면 안 된다. 그렇기에 회원유저들은 로그인을 통해 들어와야 한다.

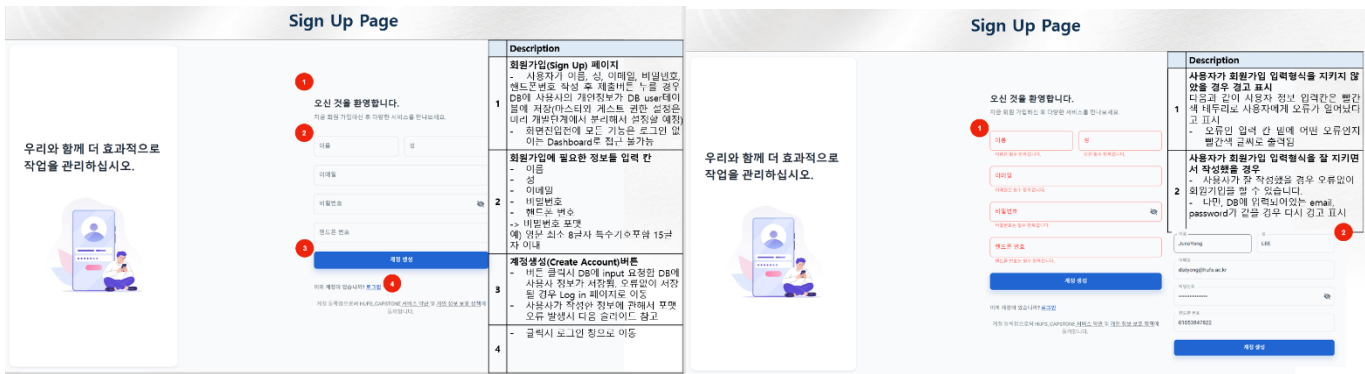
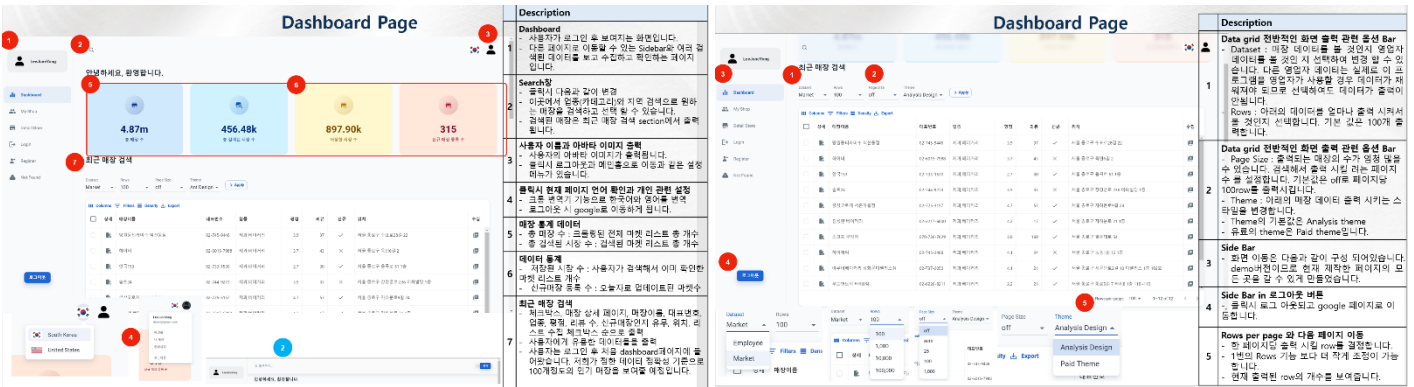


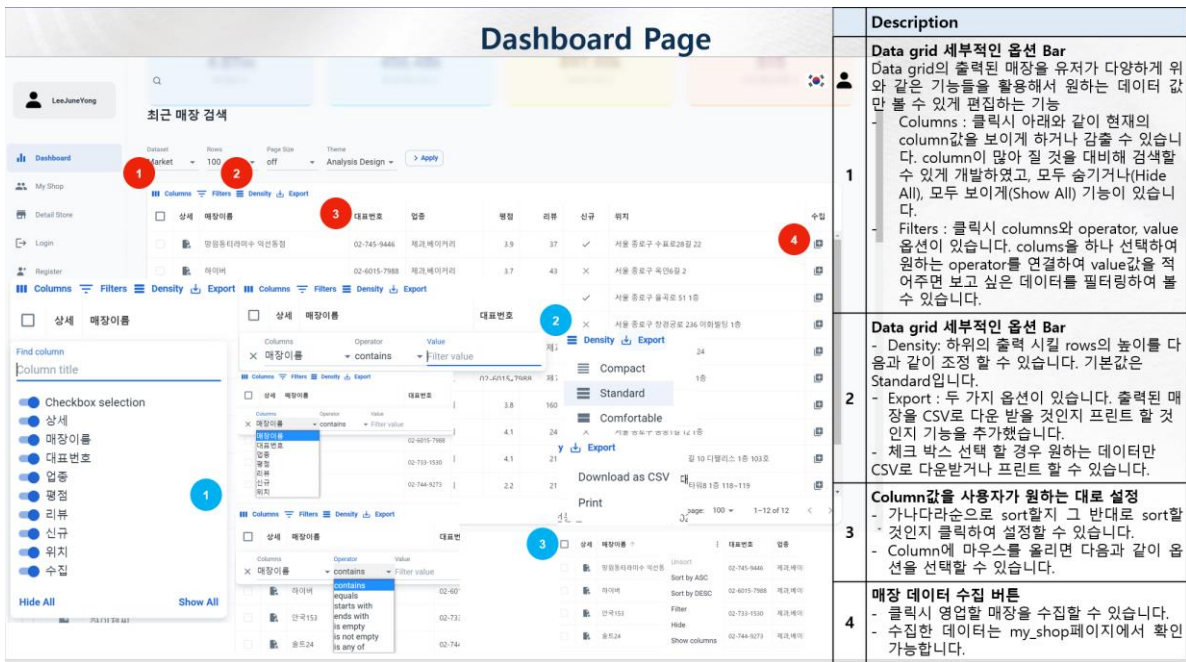
그림 9. (a) 회원가입 페이지 구성 및 설명 (b) 회원가입 페이지 반응형 기능

메인 페이지에 들어가기 위해서는 회원가입을 통해 권한을 획득해야 한다. 회원가입을 했다면 그림 8(a)에서 로그인이 가능할 것이다. 회원 가입을 통한 개인정보들은 그림 7(a)의 유저 테이블에 전송되고 저장된다.



(a) 대시보드 페이지(메인 페이지) 전체 구성

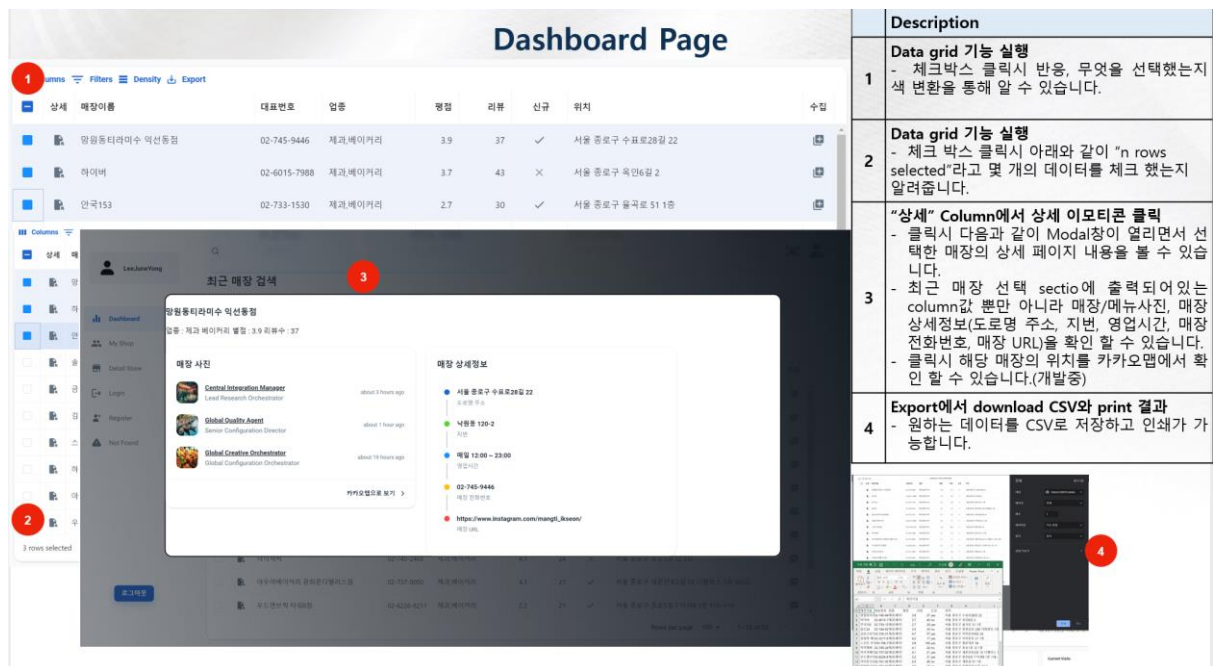
(b) 대시보드 페이지 옵션 기능 구성



(c) 대시보드 페이지에서 라벨링 데이터가 보여지는 데이터 그리드 옵션 기능 구성

그림 10.

회원가입을 하고 로그인에 성공한 유저들에게 첫번째로 보여지는 메인 페이지인 대시보드 페이지이다. 매장 정보를 한눈에 보여줄 수 있는 통계 데이터는 제일 상단에 배치했다. 대시보드 페이지에서는 각종 옵션을 선택하여 원하는 매장 데이터를 검색할 수 있게 된다. 데이터를 한눈에 잘 보여줄 수 있게 데이터 그리드 형태를 선택하였다. 행에는 데이터의 나열, 열에는 데이터의 특징으로 설정하여 많은 정보를 관적으로 볼 수 있게 된다.



**Dashboard Page**

1. Filter: Filters, Density, Export

2. 3 rows selected

3. Modal: 최근 매장 검색

4. Export: download CSV and print 결과

상세	매장이름	대표번호	업종	평점	리뷰	신규	위치	수입
<input checked="" type="checkbox"/>	망원동티라미수 익선동점	02-745-9446	제과·베이커리	3.9	37	✓	서울 종로구 수표로28길 22	
<input checked="" type="checkbox"/>	라이버	02-6015-7988	제과·베이커리	3.7	43	✗	서울 종로구 옥인6길 2	
<input checked="" type="checkbox"/>	안국153	02-733-1530	제과·베이커리	2.7	30	✓	서울 종로구 을지로 51 1층	

**Description**

1. Data grid 기능 실행  
- 체크박스 클릭시 반응, 무엇을 선택했는지 색 변환을 통해 알 수 있습니다.

2. Data grid 기능 실행  
- 체크 박스 클릭시 아래와 같이 "n rows selected"라고 몇 개의 데이터를 체크 했는지 알려줍니다.

3. "상세" Column에서 상세 이모티콘 클릭  
- 클릭시 다음과 같이 Modal창이 열리면서 선택한 매장의 상세 페이지 내용을 볼 수 있습니다.  
- 최근 매장 선택 section에 출력되어 있는 column값 뿐만 아니라 매장/매뉴사진, 매장 상세정보(도로명 주소, 지번, 영업시간, 매장 전화번호, 매장 URL)을 확인 할 수 있습니다.  
- 클릭시 해당 매장의 위치를 카카오맵에서 확인 할 수 있습니다.(개발중)

4. Export에서 download CSV와 print 결과  
- 원하는 데이터를 CSV로 저장하고 인쇄가 가능합니다.

그림 11. 대시보드에서 검색한 내용을 클릭했을 때 보여지는 상세페이지 구성 및 설명

다음은 검색된 데이터의 세부정보를 확인하려면 상세 버튼을 누르면 그림 11 과 같은 화면을 볼 수 있게 된다. 세부 정보를 열로 나열함으로써 내용이 명확하게 전달된다.



**SideBar Overlay and Responsive Web**

1. 환영하세요, 환영합니다.

2. Overlay Sidebar

3. Side Bar 구성

**Description**

1. Responsive Web  
- Dashboard화면을 전체 화면에서 줄였을 경우 Sidebar가 숨겨지고 출력되는 화면도 반응형으로 구현하여 화면크기에 맞게 축소됩니다.

2. 1번에서 설명한 아이콘 클릭시 Side Bar overlay 기능  
- 축소된 상태 에서 아이콘 클릭할 경우 다음과 같이 sidebar를 볼 수 있습니다.

3. Side Bar 구성  
- 최종 시연에서는 Dashboard와 My shop을 제외한 다른 section들은 삭제합니다.  
- 사용자가 URL를 변경하여 이와 같은 홈페이지를 강제로 이동시킬경우 404페이지가 출력됩니다.

상세	매장이름	대표번호	업종	평점	리뷰	신규
<input checked="" type="checkbox"/>	망원동티라미수 익선동점	02-745-9446	제과·베이커리	3.9	37	
<input checked="" type="checkbox"/>	라이버	02-6015-7988	제과·베이커리	3.7	43	
<input checked="" type="checkbox"/>	안국153	02-733-1530	제과·베이커리	2.7	30	
<input checked="" type="checkbox"/>	종로24	02-744-9273	제과·베이커리	3.5	33	
<input checked="" type="checkbox"/>	금강고개 서문매점	02-725-3157	제과·베이커리	4.7	57	
<input checked="" type="checkbox"/>	금강한 베이커리	02-3217-6800	제과·베이커리	4.2	17	

그림 12. 오버레이 되는 사이드 바 구성 및 설명

데이터들의 경우 다양한 특징을 가지고 있고 검색된 데이터가 출력되는 데이터 그리드 부분이 제일 중요하므로 사이드 바는 좌측 상단에 버튼을 누르면 오버레이 기능을 통해 전체 화면에서 차지하는 비율을 줄일 수 있고, 사이드바에서 다른 페이지로 이동할 수 있게 한다.

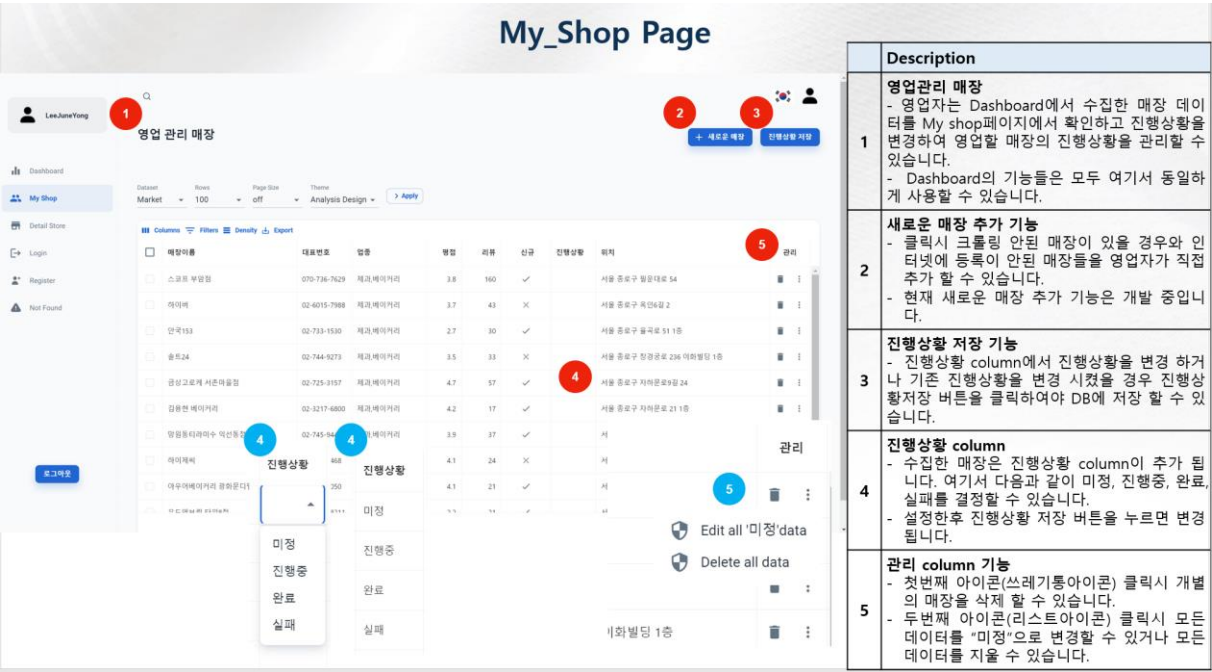


그림 13. My Shop 페이지의 구성 및 설명

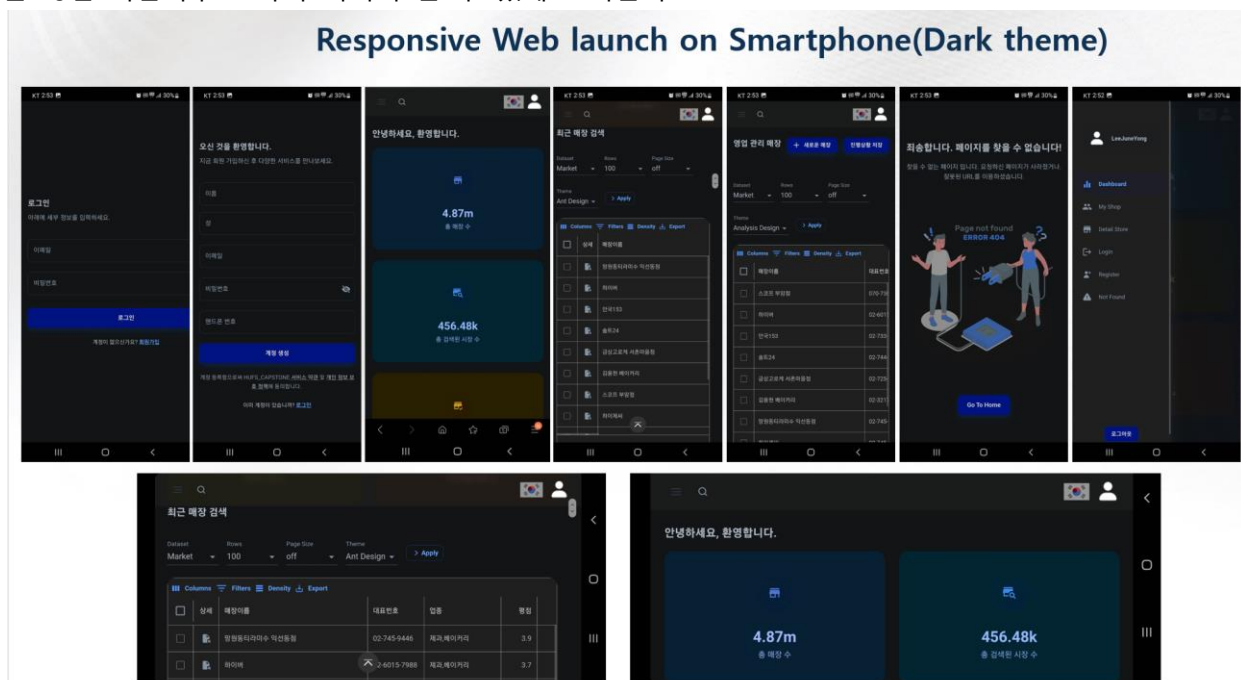
검색한 데이터들을 저장한다면 그 데이터들은 마이페이지에서 볼 수 있게 된다. 사용자가 원하는 데이터들만 모아서 확인할 수 있고 진행상황 옵션을 통해 영업 관리를 도와준다.





그림 14. 404 페이지 구성 및 설명

반응형 웹은 웹과 유저의 상호작용이 중요하므로 매순간의 서버와의 통신이 중요하다. 통신이 끊기게 되면 데이터를 도중에 잃을 수 있기 때문이다. 이는 재빠르게 유저에게 404 페이지를 보여줌으로써 하던 작업을 끝낸 시점이후로 다시 이어서 할 수 있게 도와준다.



**그림 15.** 스마트폰 환경에서 작동 및 dark theme 기능

반응형 웹은 스마트폰 환경에서도 사용할 수 있어야 한다. 이는 안드로이드(Android), 맥(MAC)환경에서 모두 가능하다. 본인의 스마트폰에 적용된 밝은 테마와 어두운 테마에 맞게 적용되고 이후에 변경도 가능해야 하므로 위와 같이 어두운 테마가 적용된 모습이다.

## 5. 연구 결과 분석 및 시사점

자동화된 크롤링 알고리즘을 구현 후 크롤링 적용 후 크롤링 시 데이터 라벨링 전처리를 통한 자동화를 구현하였다. 이를 통해 더 질 높은 정보를 제공될 것이다. MySQL 데이터베이스 구축과 Express.js API 서버를 구축함으로써 라벨링된 데이터를 더 빠르게 사용자에게 전송될 수 있을 것이다. 출력된 매장데이터에서 원하는 데이터만 획득 가능하도록 하여 여러 필터(filter)로 검색할 수 있는 기능을 개발하여 사용자의 요구사항에 맞는 데이터가 전달이 될 것이다. 최신화된 반응형 UI 와 사용자에게 제공해 줄 수 있는 직관적이고 편리한 기능을 개발하여 제공하였다. 다양한 매장 통계 데이터를 획득 가능하게 하여 영업의 활성화에 도움이 될 것이다.

## 6. 결론 및 토론

카카오 맵과 네이버 지도에서 웹 크롤링 서버를 통해 데이터 수집 및 라벨링 과정을 거쳐 영업에 필요한 라벨링된 데이터를 영업 DB 에 저장하고 이렇게 저장된 데이터를 사용자에게 원하는 정보를 제공해줄 수 있는 리액트 기반 반응형 웹을 구축을 제안하였다. 반응형 웹의 비동기식 개발은 계속 연구되고 있는 분야이고, 여러 언어들과 프레임워크의 복합적으로 연결된 수많은 코드를 디버깅하는 부분에서 많은 시간적 어려움이 연구 개발에 제한이 있었다. 연구의 핵심 목표는 달성했지만 보안 소켓 계층(SSL)과 전송 계층 보안 (TLS) 인증서로 보호되는 하이퍼 텍스트 전송 프로토콜 보안(HTTPS)를 구현해보는 웹 보안적인 구현 부분과 더 다양한 페이지구성과 기능을 구현하지 못한 것에 한계가 있었다.

이로서 기대효과는 사용자에게 영업 진행시 영업을 진행해야 할 매장에 대한 정보를 제공하고, 각 매장의 업태, 업종을 정확하게 파악할 수 있다. 영업 현황에 대한 정보를 제공하여 동선 및 계획을 효과적으로 관리할 수 있다. 영세 업체가 겪던 소상공인에 대한 DB 구축시 어려움에 도움을 줄 수 있다. 결과적으로 영업 진행자와 영업을 받는 사람 모두에게 편의성과 효율성을 제공할 것이 기대된다.

## 참고문헌 - References

1. 김기식, 박선나. 2020 대한민국 자영업 보고서, pp14~32, 2020 [인터넷자료]
2. 박정태. 파이썬으로 배우는 웹 크롤러. 정보문화사, p. 394 [책]
3. 김민준. 리액트 네이티브를 다루는 기술+리엑트를 다루는 기술 (전 2 권) 세트 (주)도서출판 길벗 [책]
4. React-redux, <https://react-redux.js.org/> [인터넷자료]

---

<sup>i</sup> 본 논문의 활용가능한 코드는 [https://github.com/yousirong/CAPSTONE\\_AWS](https://github.com/yousirong/CAPSTONE_AWS) 에서 참고 가능하다.