



ICEWAY

FRESH DELIVERY SHOP

CONTENT



01

주제

02

기획의도

03

ERD

04

기능명세서

05

Flow Chart

06

Tech Stack

07

담당업무

08

프로젝트 시연

09

개발 후기

10

Q&A

아이스크림 배송 추적 서비스

9,900₩ ~ 13,900₩

녹지 않는 신선함을 경험하세요 물론 녹기전에 도착하는 ICEWAY
고객님의 문 앞까지 신속하게 배달됩니다. 시원한 맛 놓치지 마세요!

Flavor

20

SIZE

ONE



Green tea



Strawberry



Chocolate



Vanilla



CookieAndCream

+15

Try Our Service !

제품이 녹기 전에 신속하게 배달합니다!
만약 제품이 녹았을 경우, 관리자가 직접 확인하여
제품 회수 후 새로운 제품을 바로 보내드립니다!

SHOP NOW

Try Our Service !



*keep
cold!*

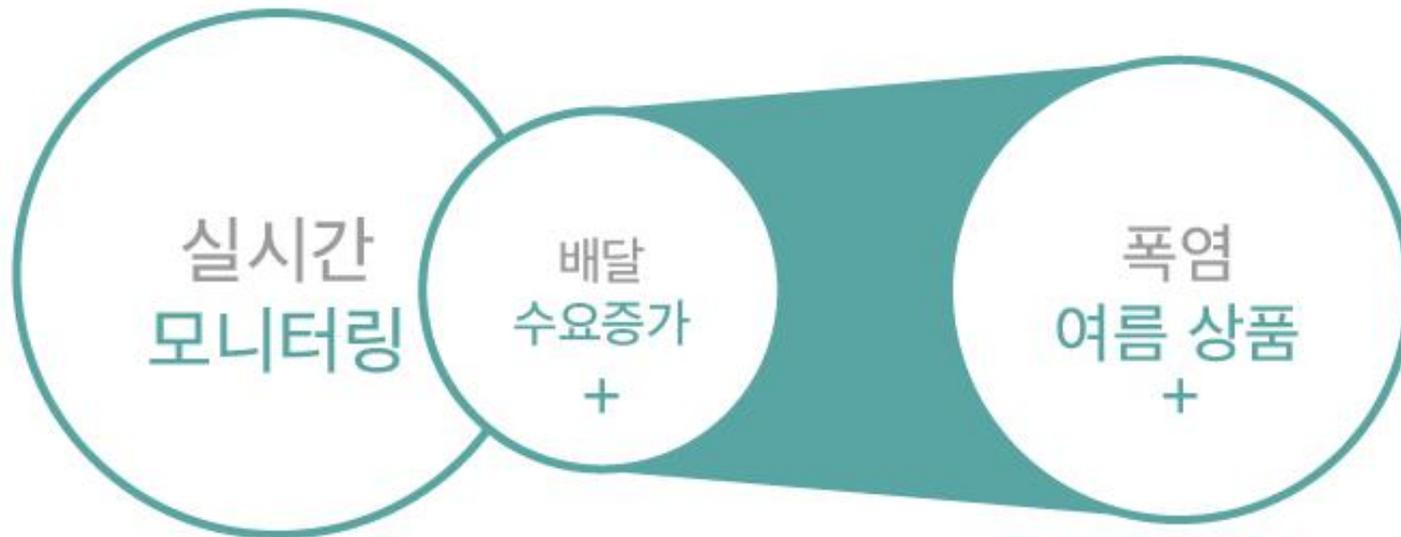
Read more



기획의도

코로나로 인한 배달 수요의 증가
폭염으로 인한 아이스크림 소비 증가

전보다 독하고 길어진 폭염에 유통가는 “폭염효과”



Entity: 배달 (delivery)

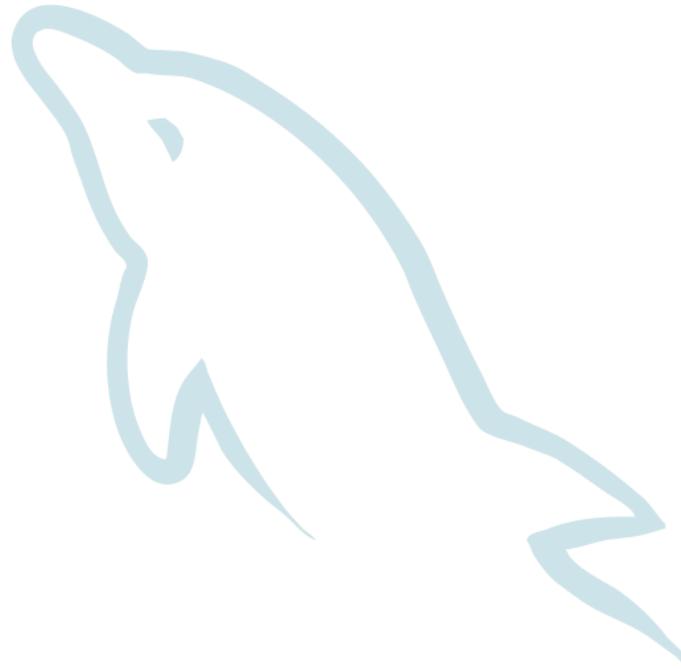
property:

- id:** 배달의 고유 식별자
- latitude, longitude:** 배달 위치의 위도와 경도
- temperature:** 배달 온도
- delivery_status:** 배달 상태
- recall_status:** 리콜 상태
- delivery_time:** 배달 시간
- arrival_time:** 도착 시간
- recall_time:** 리콜 시간

Entity: 제품 (products)

property:

- id:** 제품의 고유 식별자
- product_data:** 제품에 관한 정보를 담고 있는 JSON 객체
 - **name:** 제품의 이름
 - **price:** 제품의 가격
 - **temperature:** 제품의 온도



Entity: 고객 (customers)

property:

- id:** 고객의 고유 식별자
- customer_data:** 고객에 관한 정보를 담고 있는 JSON 객체
 - **name:** 고객의 이름
 - **latitude, longitude:** 고객의 위도와 경도

Entity: 회수 (recall)

property:

- id:** 회수의 고유 식별자
- name:** 회수를 요청한 사람의 이름
- delivery_id:** 관련 배달의 고유 식별자

Entity: 관리자 (admin)

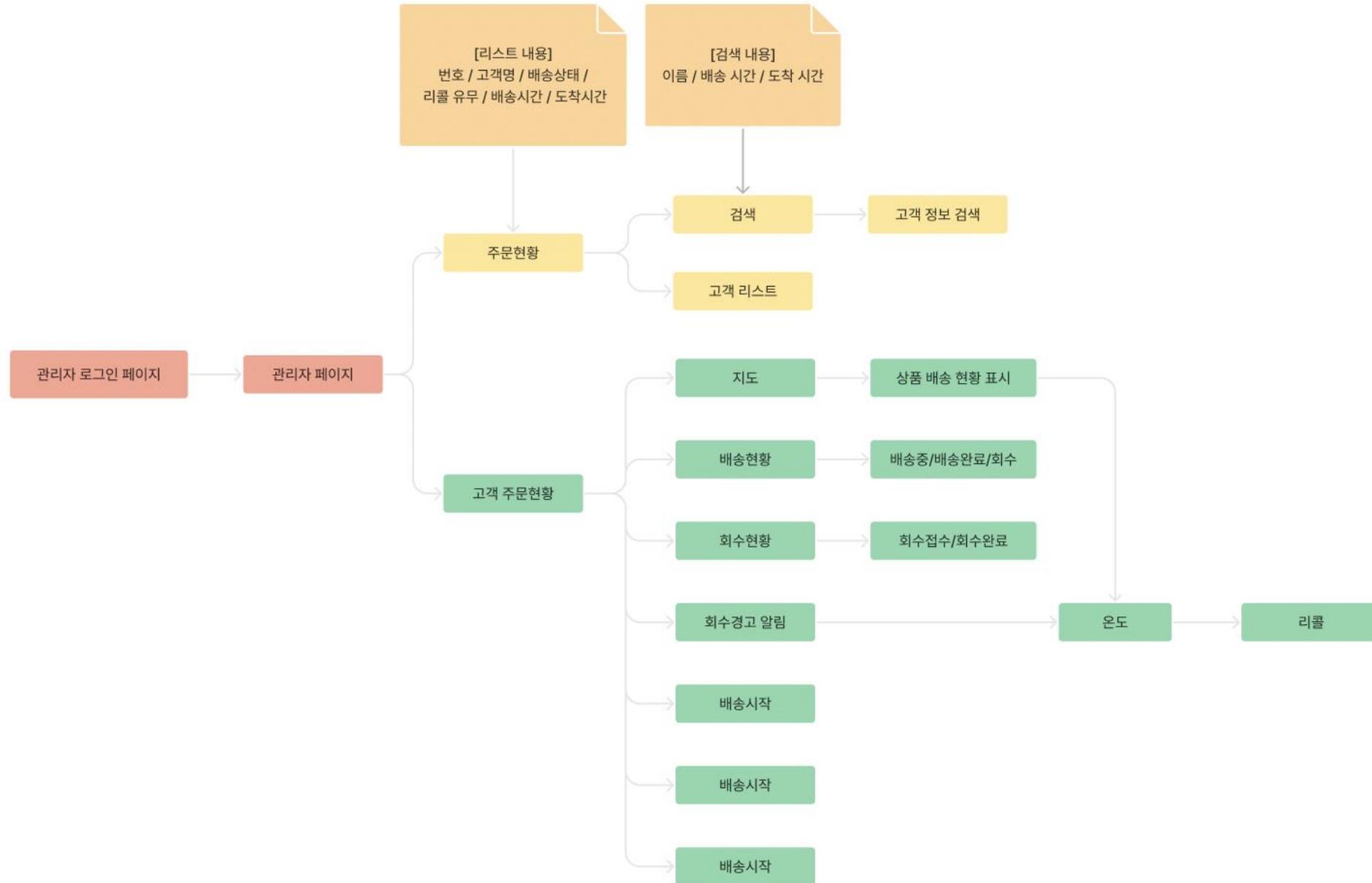
property:

- id:** 관리자의 고유 식별자
- user_name:** 관리자의 사용자 이름
- user_pwd:** 관리자의 비밀번호
- user_role:** 관리자의 역할

기능명세서

구분	No	Page	메뉴			기능		
			1Depth	2Depth	3Depth	기능명	기능 설명	구현방법
로그인	1	로그인 (0.0.1)	로그인 폼	로그인		1) 로그인 폼 2) 로그인	1) 하기 폼 요소 노출 1-1) 아이디, 비밀번호 2) ID, PW 유효성 검사, Spring Boot Security 3) 관리자 페이지 이동	1) 로그인 폼 구현 2) 아이디, 패스워드 유효성 검사 구현 및 Spring Boot Security 3) 관리자 페이지 이동
관리자 대시보드	3	배송 추적 대시보드 (0.0.1)	지도	지도 이벤트		1) 지도 2) 지도 이벤트 (축소, 확대, 키 이벤트)	1) 지도 영역 노출 2) 지도의 축소, 확대 및 키 이벤트 상하좌우, +, -, Enter) 눌렀을때 동작	1) 카카오 맵 API 를 이용하여 구현 - 초기 지도 레벨은 6으로 설정 2) 키보드 이벤트 관련 EventListener 등록 카카오 Maps API 의 지도 줌 컨트롤 관련 자료 확인하여 구현 - '-' : 지도 레벨을 1 증가시켜 지도를 축소함 - '+' : 지도 레벨을 1 감소시켜 지도를 확대함 - '+-' : 0.001 값 만큼 경도 좌표 증가 - '-+' : 0.001 값 만큼 경도 좌표 감소 - '- ' : 0.001 값 만큼 위도 좌표 증가 - ' -' : 0.001 값 만큼 위도 좌표 감소 - Enter : 지도레벨, 위도, 경도 값을 초기 디폴트 값으로 바꾸게 구현
			마커	마커 클릭 이벤트	마커 색상	1) 마커 2) 마커 클릭 이벤트 3) 마커 색상	1) 상품 주문 번호 별 지도 위 마커 표시 2) 마커 클릭 시 상품 정보 하단 사이드바에 표시 3) 마커 배송상태에 따른 색상 변경	1) DB에서 값 변경이 될때마다 위도 경도 Json 데이터 Websocket을 사용하여 받아옴 - 카카오 Maps API의 마커 관련 자료 확인하여 마커 구현 - Websocket에서 받아온 위도 경도 데이터 바탕으로 지도에 실시간으로 마커 업데이트 2) 상품 배송 정보 데이터 Websocket을 사용하여 DB에서 실시간으로 데이터 호출 - 마커 클릭 시 해당 상품 하단 사이드바에 호출 데이터 표시 2-1) 호출 데이터 목록 - 이름 - 상품명 - 온도 3) 마커의 색상을 Delivery테이블의 delivery_status 기준으로 색변경 - true : 파란색 - false : 검은색
			알림 리스트	리콜 경고 알림	알림 클릭 이벤트	1) 알림 리스트 2) 알림 클릭 이벤트 3) 회수 클릭 이벤트	1) 우측 사이드바 상단에 -10도 이상시 리스트에 누적 표시 - ex) xxx님 상품 온도 상승 알림 2) 알림 클릭 시 해당 상품 정보 하단 사이드바에 표시 3) 알림 리스트에서 삭제되며 회수 완료 카운트 증가	1) Websocket 사용하여 DB에서 온도 Json 데이터 호출하여 기준치 이하 발생시 알림 추가 2) 알림 클릭 시 클릭한 상품의 배송 정보 하단 사이드바에 표시 3) 회수 클릭 시 알림 리스트에서 삭제 및 배송 현황 회수 완료 카운트 증가 3-1) DB 처리 로직 - Recall 테이블의 Delivery_ID에 해당하는 Recall ID 레코드를 삭제한다. - Delivery 테이블의 recall_status 값을 true로 update.
			상품 정보			1) 상품 정보	1) 알림 클릭 시 지도 하단에 마커, 알림에 해당하는 상품 정보 표시 - 고객명, 온도, 상품명, 회수 버튼	1) Websocket 으로 DB에서 받아온 데이터로 각 고유 ID에 해당하는 정보를 노출 - 표시된 데이터는 DB에 값이 변경되면 실시간으로 바뀜
			배송 현황			1) 배송 현황	1) 우측 사이드바 하단에 하기 정보 노출 - 배송중 - 배송 완료 - 회수 경고 - 회수 완료	1) 배송 현황 Json 데이터 Websocket을 사용하여 DB에서 실시간으로 데이터 호출 - 우측 알림 리스트 하단에 구현 - 실시간으로 DB에 있는 데이터를 기준으로 배송 현황 데이터 변경 1-1) 각 상태는 DB에 Delivery 테이블에 저장된 다음과 같은 칼럼으로 판별 - 배송중 : delivery_status 값이 true - 배송 완료 : delivery_status 값이 false - 회수 경고 : temperature 값이 -10 이상 - 회수 완료 : recall_status 값이 true
			배송 컨트롤러			1) 배송시작 2) 배송중지 3) 배송초기화		
			고객 주문 내역	4	주문 내역 (0.0.1)	주문 내역 리스트	페이지네이션	검색

Flow Chart



Language

- Java
- Javascript
- SQL

Web FE

- JSP
- CSS
- WebSoket
- Kakao Map API
- BootStrap

Web BE

- Spring Boot
- Spring Security
- MyBatis
- Kakao Route API
- Maven
- AMQP

Infra & DB

- EC2
- RDS
- Amazon MQ
- Rabbit MQ
- MySQL

Tool

- VSCode
- Eclipse
- Intelij
- Git
- Source Tree

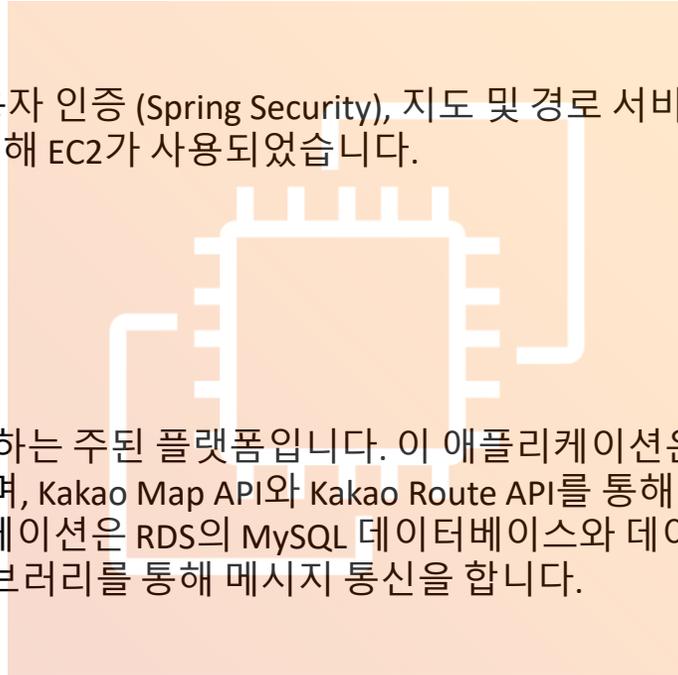
EC2 (Elastic Compute Cloud)

why?

웹 애플리케이션을 호스팅하고, 백엔드 로직, 사용자 인증 (Spring Security), 지도 및 경로 서비스 (Kakao Map API, Kakao Route API) 등을 실행하기 위해 EC2가 사용되었습니다.

how?

EC2 인스턴스는 Spring Boot 애플리케이션을 실행하는 주된 플랫폼입니다. 이 애플리케이션은 Spring Security를 사용하여 사용자 인증을 처리하며, Kakao Map API와 Kakao Route API를 통해 지도와 경로 정보를 제공합니다. 또한, 이 애플리케이션은 RDS의 MySQL 데이터베이스와 데이터를 주고 받고, AmazonMQ의 RabbitMQ와 AMQP 라이브러리를 통해 메시지 통신을 합니다.



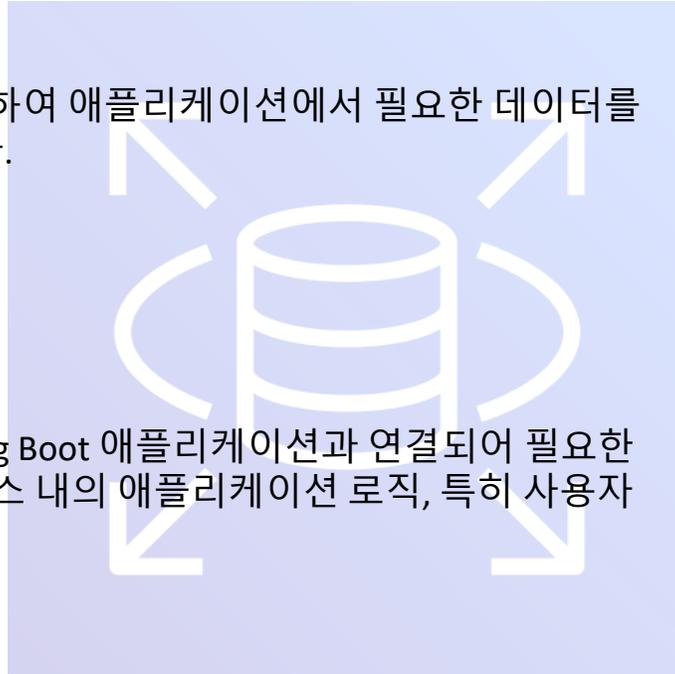
RDS (Relational Database Service) with MySQL

why?

데이터베이스 관리를 단순화하고, MySQL을 사용하여 애플리케이션에서 필요한 데이터를 안정적으로 제공하기 위해 RDS가 사용되었습니다.

how?

RDS의 MySQL 인스턴스는 EC2에서 실행되는 Spring Boot 애플리케이션과 연결되어 필요한 데이터를 주고 받습니다. 이 데이터는 EC2 인스턴스 내의 애플리케이션 로직, 특히 사용자 인증과 지도 정보 서비스에서 사용됩니다.



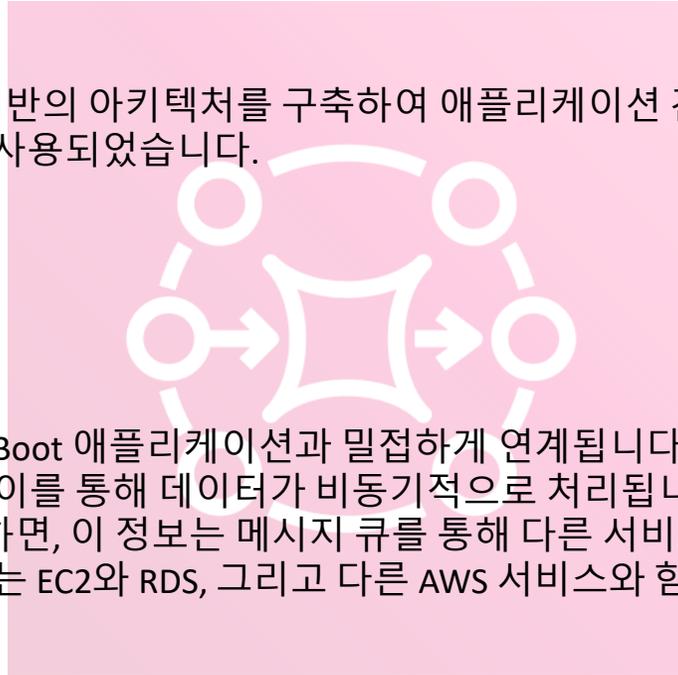
AmazonMQ with RabbitMQ

why?

RabbitMQ의 설정과 관리를 간소화하고, 메시지 기반의 아키텍처를 구축하여 애플리케이션 간의 비동기 통신을 가능하게 하기 위해 AmazonMQ가 사용되었습니다.

how?

AmazonMQ의 RabbitMQ 인스턴스는 EC2의 Spring Boot 애플리케이션과 밀접하게 연계됩니다. AMQP 라이브러리를 통해 메시지 큐와 통신하며, 이를 통해 데이터가 비동기적으로 처리됩니다. 예를 들어, 사용자가 지도 상의 특정 위치를 클릭하면, 이 정보는 메시지 큐를 통해 다른 서비스나 로직으로 전달될 수 있습니다. 이렇게 AmazonMQ는 EC2와 RDS, 그리고 다른 AWS 서비스와 함께 작동하여 복잡한 작업을 효율적으로 처리합니다.





조국*

Kakao 지도 API 연동
지도마커 생성 및 클릭이벤트
WebSocket 서버구축
기능명세서 작성



김민*

ERD 작성 및 DB 설계
Kakao 길찾기 API 연동
데이터 생성기 API 개발
AMQP 연동



주연*

회수알림 리스트
회수알림 클릭 리스트
회수 클릭 리스트
Flow Chart 작성



김선*

주문내역 리스트 구현
페이지네이션 구현
검색기능 구현
메뉴구조도 작성



남지*

상품정보 표시 구현
배송현황 구현
화면 명세서 작성



이우*

회원가입 및 로그인 폼 구현
Security 설정 후 관리자
UI디자인 설계

01. 로그인 폼

서버실행 시 관리자 계정 디폴트 생성

ICEWAY

사용자명

비밀번호

로그인

- STATUS
- HISTORY
- START
- STOP
- RESET



주문자명 상품명 온도 회수버튼

Recall Warning Notification

배송중	배송완료
0	0
회수경고	회수 완료
0	0

02. Status

자동으로 주문 등록시 마커 생성
 마커 클릭시 제품 상세 정보
 배송 중 온도 -10도 이상 회수 경고
 알람 클릭시 해당 마커 이동
 회수 버튼 클릭시 자동 리콜
 배송 상태 표시

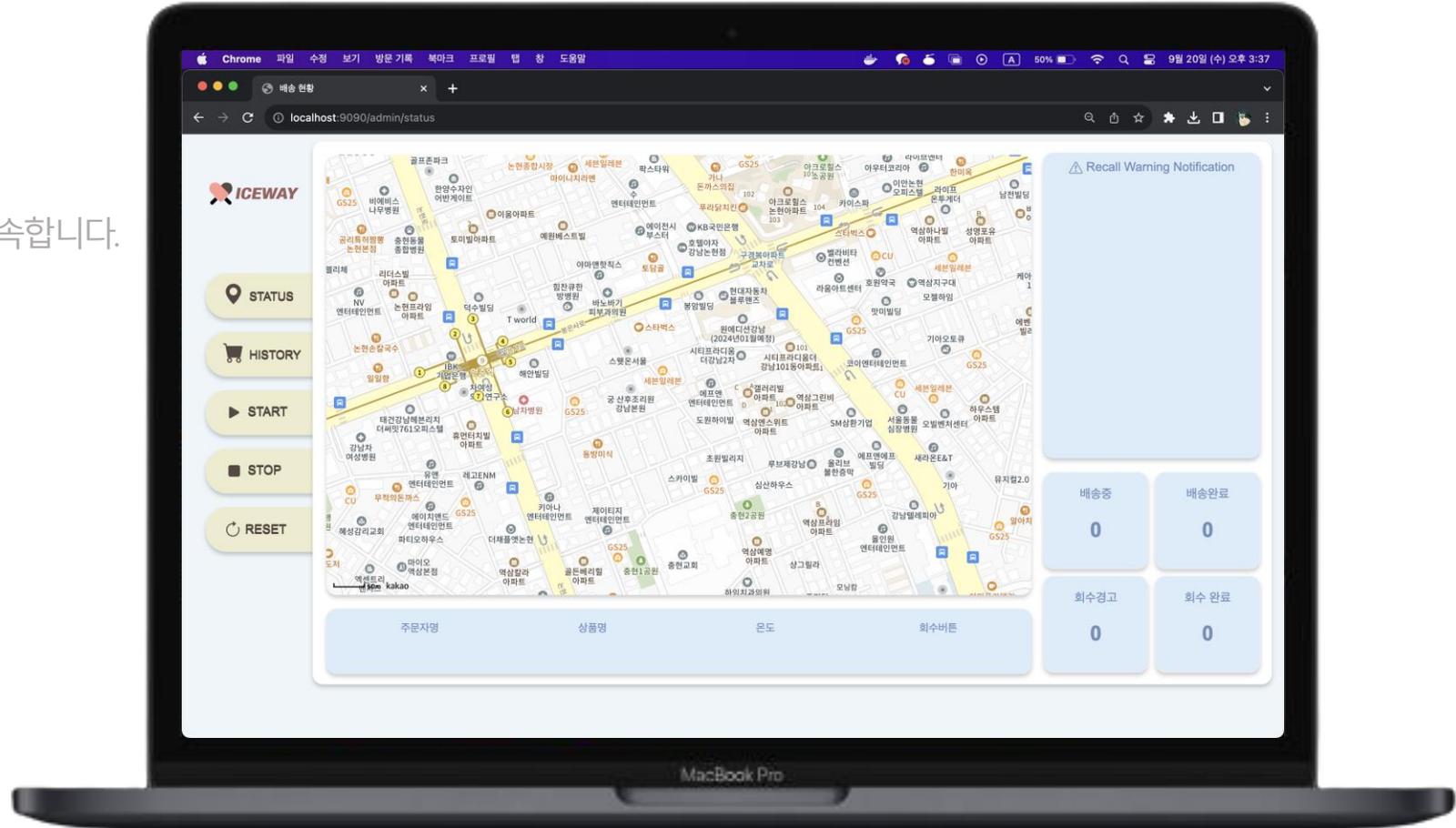
고객명	배송 상태	리콜 유무	배송 시간	도착 시간	
이승현	배송중	-	2023-09-21 17:30:28.0	2023-09-21 17:39:41.0	
송미란	배송중	-	2023-09-21 17:30:28.0	2023-09-21 17:41:58.0	
한지원	배송중	-	2023-09-21 17:30:28.0	2023-09-21 17:43:29.0	
임준호	배송중	-	2023-09-21 17:30:28.0	2023-09-21 17:54:41.0	
5	정태희	배송중	-	2023-09-21 17:30:28.0	2023-09-21 17:47:01.0
4	김소연	배송중	-	2023-09-21 17:30:28.0	2023-09-21 17:44:55.0
3	최준영	배송완료	-	2023-09-21 17:50:29.0	2023-09-21 17:50:29.0
2	박수진	배송완료	-	2023-09-21 17:46:53.0	2023-09-21 17:46:53.0
1	이동욱	배송완료	-	2023-09-21 17:46:12.0	2023-09-21 17:46:12.0

03. HISTORY

주문 등록 시 주문자, 상품, 온도 정보 누적
 배송 시간, 도착시간 정보 표시
 검색 필터로 주문 정보 검색

실시간 모니터링 서비스

ICEWAY 서비스는 실시간 모니터링을 통해서 고객에게 녹지 않는 신선함을 보여줍니다. 만약 배송 중인 제품이 녹았을 경우 관리자가 제품 회수 후 새 제품을 바로 보내드릴 것을 약속합니다.



조국

프로젝트를 진행하며 수업 시간에 배우지 않은 처음 본 다양한 기술을 접하면서 스스로 공부하고 발전하는 시간이 되었고, Websocket 으로 서버를 구축하고 Javascript 사용하여 메인 페이지를 동작하는 기능을 설계하고 개발하여 의도대로 동작하지 않는 기능을 고쳐가면서 뿌듯함을 느끼고 많은 발전을 했습니다. 다만 중간에 여러 가지 사정으로 인해 주제와 기획이 많이 바뀌어서 더 많은 페이지를 개발하기에는 시간이 많이 부족한 점이 아쉬웠습니다.

주제를 오랫동안 정하지 못하고, Docker, RabbitMQ 등등 몰랐던 기술을 적용하려고 시간을 소모해서 완성도에 아쉬움이 있었고, 개발중 오류 해결하다가 어느 곳이 문제인지 모르는 오류가 나와서 기존에 작성한 코드를 전부 갈아엎으면서 어려움, 부족함을 느꼈습니다.

김선

김민

이번 프로젝트를 통해 많은 것을 배우고 경험했습니다. 특히, 실제 환경과 가상 환경에서의 데이터 차이에 대해 깊은 인사이트를 얻을 수 있었습니다. 초기 설계에서는 GPS와 온도 측정기를 이용하여 실시간으로 데이터를 수집하는 것을 목표로 하였으나, 여러 상황으로 인해 이를 구현하지 못한 것은 아쉬운 부분입니다. 이 경험을 바탕으로 다음 프로젝트에는 실제 장비를 도입하여 더욱 정확하고 실용적인 결과를 도출할 계획입니다.

프로젝트를 진행하면서 팀원들과 협업해 나가는 과정을 배우게 되었습니다. 혼자라면 어려웠을 것도 서로 도와가며 완성할 수 있었습니다. 또한 수업을 통해 배웠던 기술에 새로운 기술을 적용해 프로젝트를 완성한 것에 보람을 느꼈습니다.

남지

주연

팀 프로젝트를 진행하면서, 학원에서 배웠던 기술 이외에 API, AMQP 서비스 등 구현해 내기 어려운 순간들이 있었지만, 조원들과 같이 공부하고 해결하면서 끝까지 마무리 할 수 있게 되었습니다. 팀워크의 중요성에 대해서 다시 한번 깨닫게 되었고, 다음 팀 프로젝트를 하게 된다면 개인의 역량을 키우며, 끊임없는 자기 계발과 함께 모두가 협력하는 자세로 이끌어 나갈 수 있도록 함께 노력하겠습니다.

동료들과 협업을 통해 다양한 아이디어와 의견을 주고 받으며, 프로젝트를 향상하는 과정에서 소통과 협력의 중요성을 깨달았습니다. 팀원들과 협업 덕분에 프로젝트가 더욱 효율적으로 진행 되었습니다. 기술적으로 미숙함을 극복하고, 더 나은 프로젝트를 구현하기 위해 끊임없이 학습하고 노력할 것입니다.

이우

Q&A

THANK YOU

ICEWAY

FRESH DELIVERY SHOP