요구사항 정의서

en·core | PLAYDATA

한화시스템

팀명 : Woof

조원 : 강문혜 강지흔 이창훈 임연진

한화시스템 BEYOND CAMP 2기 - 1st project

목차

1. 프로젝트 개요

1.1 프로젝트 소개 ---------- 3p

1.2 프로젝트 배경 ---------- 4p

1.3 업무 배경도 (Context Diagram) ---------- 5p

2. 설계

2.1 ERD ---------- 6p

2.2 릴레이션 스키마 ---------- 7p

3. 구현

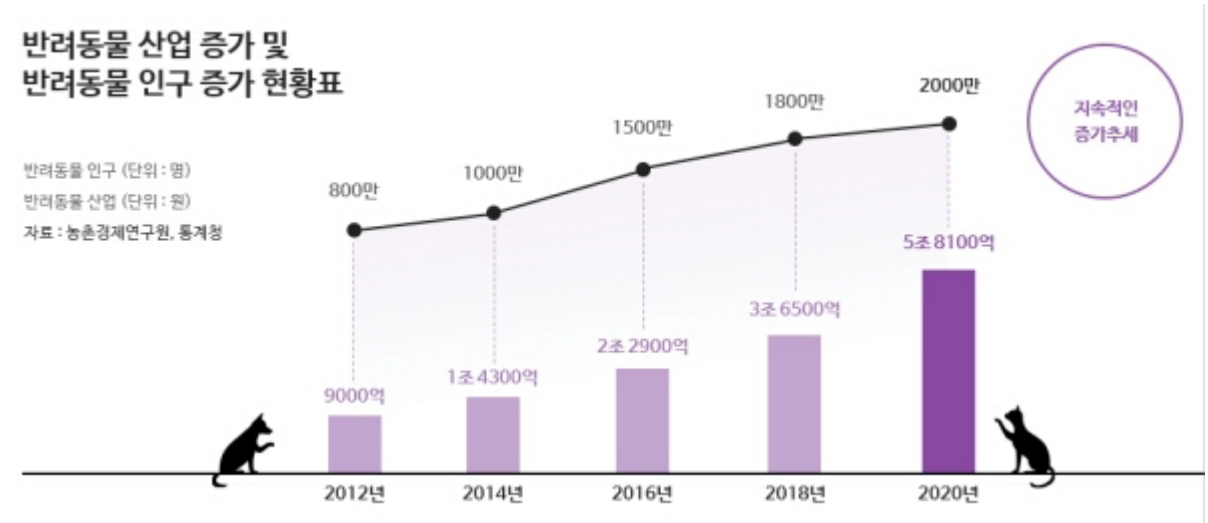
3.1 릴레이션 ---------- 8p

3.2 기능별 쿼리 ---------- 10p

3.3 시스템 아키텍처 ---------- 11p

1. 프로젝트 개요

1.1 프로젝트 소개

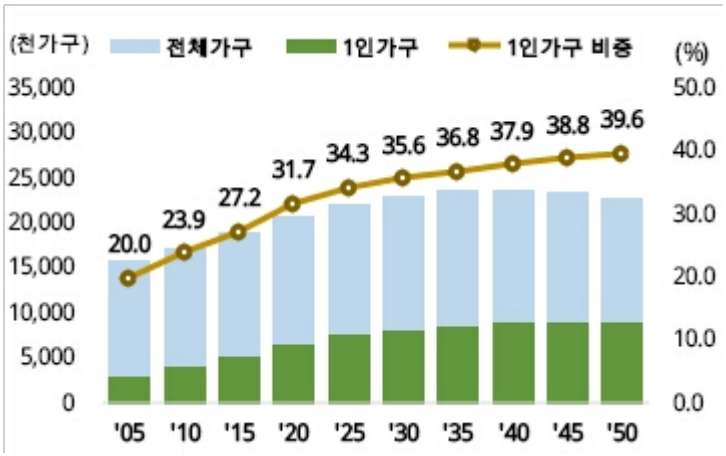


'Woof' 서비스는 반려동물에게 필요한 업체를 검색하고 방문하는데 어려움을 겪는 사용자들을 위해 만들어진 서비스이다.

최근 꾸준히 반려동물 산업과 반려동물 인구가 증가됨과 동시에 사회적으로 소규모가구 또한 증가하면서 집에 반려동물이 혼자 남아있는 시간이 많거나, 사회성이 부족한 반려동물들에 대한 우려가 생겨나고 있다. 반려동물 유치원이나 병원, 미용실 등에 방문을 해야 하는데 거대해진 산업에 비해 사람이 다니는 학원/병원/미용실보다 훨씬 검색, 예약 등이 어렵고 방문할 시간이 없다는 의견이 많이 존재한다.

이를 해결하기 위하여 'Woof' 에서는 사용자 주변에 있는 반려동물 업체의 정보, 사진, 리뷰를 모아서 보여주고, 바쁜 사회인들을 위하여 전문 자격증을 보유한 매니저가 보호자 대신 반려동물과 업체에 대신 방문해주는 서비스를 제공한다.

1.2 프로젝트 배경



자료 = 통계청 (인구주택총조사, 장래가구추계:2020~2050)

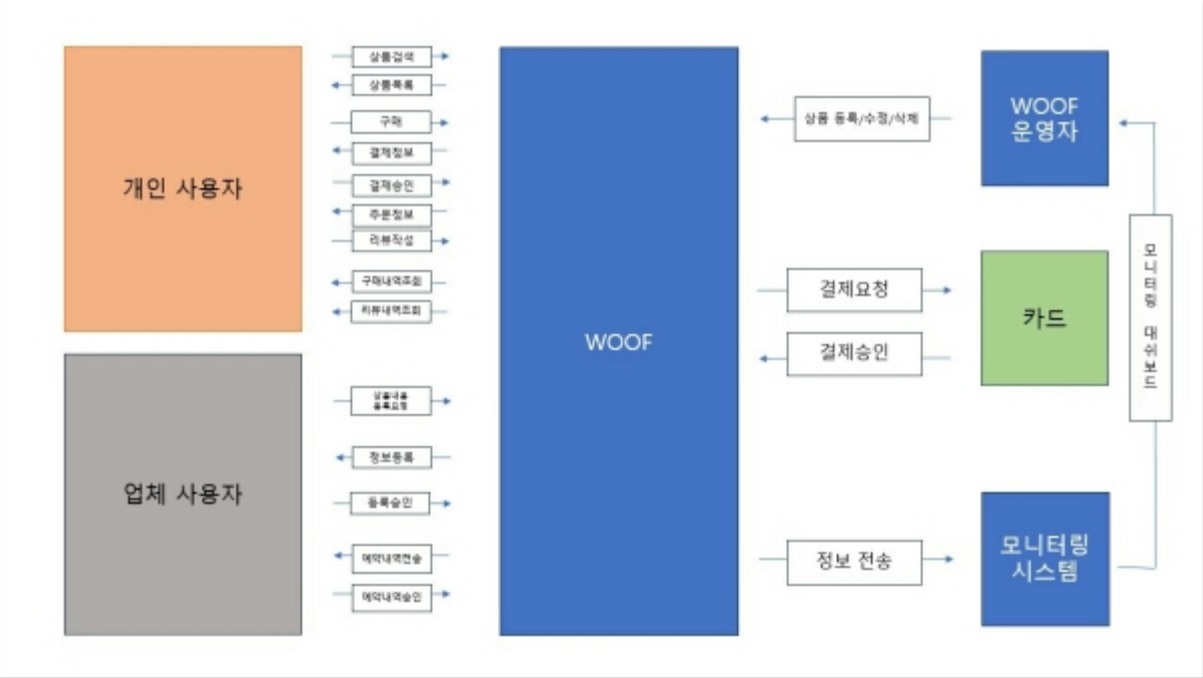
전체 가구 중 1인 가구의 비중은 2005년 20%였으나, 2030년 35.6%, 2050년엔 39.6%에 이를 것으로 전망된다. 반려동물 인구의 증가 역시 늘어나고 있는데 이러면 비교적으로 반려동물 한 마리당 보호자의 수가 줄어들고 있다고 봐야한다.

경제활동이나 개인사정으로 인해 반려동물이 집에서 홀로 보내는 시간이 많다보니 가족과 함께하지 않는 동안에도 안전하게 충분한 자극을 즐길 수 있는 반려동물 유치원의 수요 역시 늘어나고 있다.

'Woof' 서비스는 유치원에 반려동물을 데리고 등원시킬 수 있는 여유가 없거나, 다양한 유치원의 프로그램과 양질의 관리, 가격 등을 비교하는 시간이 부족하다던가 하는 등 바쁜 사회인들을 위해 해결방안을 주기 위해 만들어졌다.

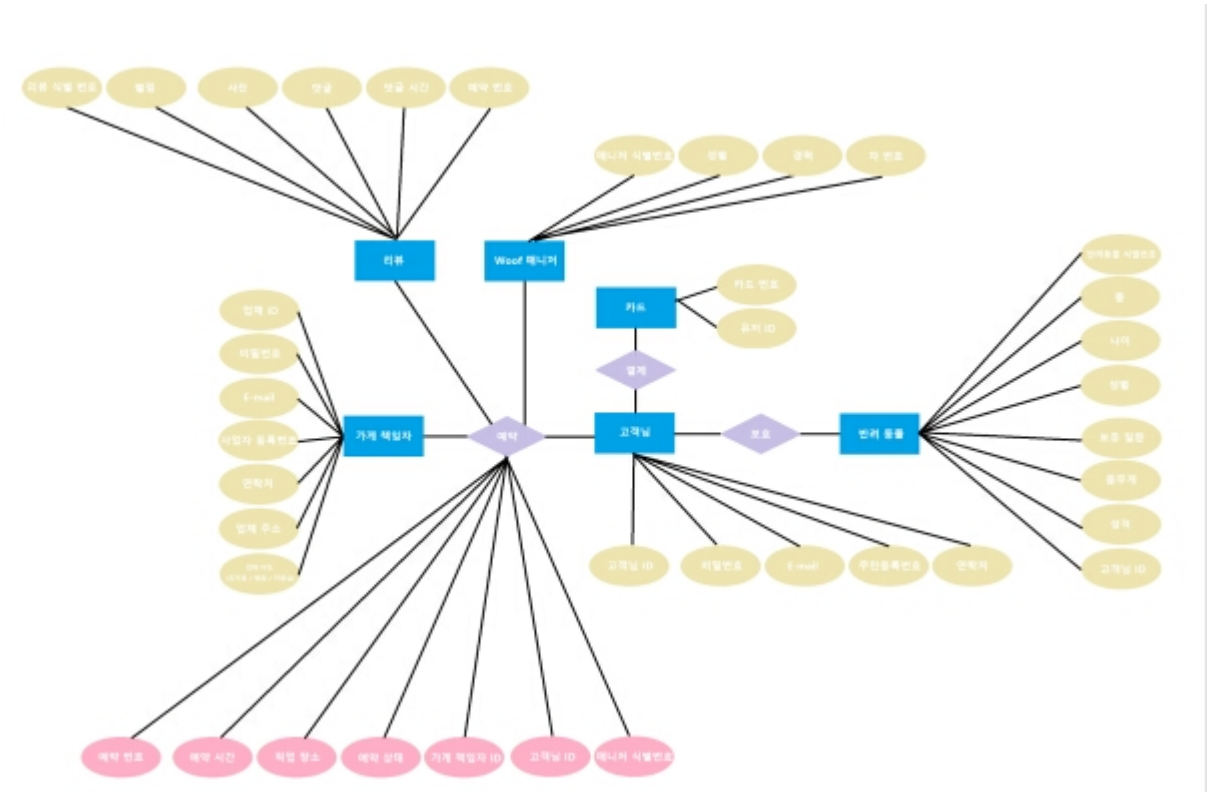
현재는 사용자 주변의 유치원, 병원, 미용실에 대한 정보를 제공하고, Woof 매니저를 통해 보호자 대신 원하는 업체로 반려동물을 픽업해주는 서비스를 제공하고 있고, 반려동물 사업이 확대됨에 따라 'Woof'의 서비스 역시 다양해질 것으로 기대되고 있다.

1.3 업무 배경도 (Context Diagram)



2. 설계

2.1 ERD



상세보기

테이블 - [가게 책임자]
속성 - 업체 ID/비밀번호/ E-mail/사업자등록번호/연락처/업체주소/업체타입

테이블 - [리뷰]
속성 - 리뷰 식별 번호 / 별점 / 사진 / 댓글 / 댓글 시간 / 예약 번호

테이블 - [Woof 매니저]
속성 - 매니저 식별 번호 / 성별 / 경력 / 차 번호

테이블 - [카드]
속성 - 카드 번호 / 유저ID

관계 - <결제>

관계 - <보호>

테이블 - [고객님]
속성 - 고객님ID / 비밀번호 / E-mail / 주민등록번호 / 연락처

테이블 - [반려동물]
속성 - 반려동물 식별번호 / 종 / 나이 / 성별 / 보유질환 / 몸무게 성격 / 고객님 ID

관계 - <예약>
속성 - 예약 번호 / 예약 시간 / 픽업 장소 / 예약 상태 / 가게 책임자ID / 고객님ID / 매니저 식별번호

2.2 릴레이션 스키마

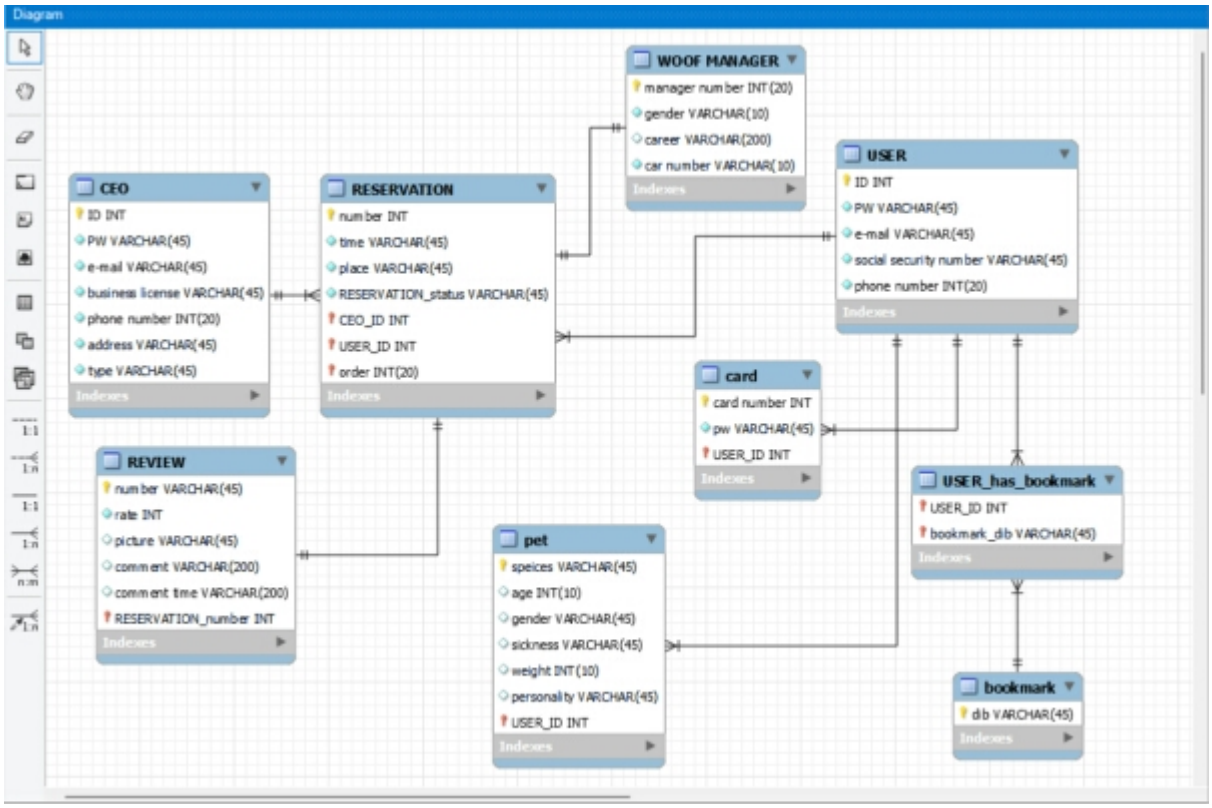


Diagram
WOOF MANAGER
manager number INT(20)
gender VARCHAR(10)
career VARCHAR(200)
car number VARCHAR(10)
Indexes
CEO
ID INT
PW VARCHAR(45)
e-mail VARCHAR(45)
business license VARCHAR(45)
phone number INT(20)
address VARCHAR(45)
type VARCHAR(45)
Indexes
RESERVATION
number INT
time VARCHAR(45)
place VARCHAR(45)
RESERVATION_status VARCHAR(45)
CEO_ID INT
USER_ID INT
order INT(20)
Indexes
USER
ID INT
PW VARCHAR(45)
e-mail VARCHAR(45)
social security number VARCHAR(45)
phone number INT(20)
Indexes
card
card number INT
pw VARCHAR(45)
USER_ID INT
Indexes
REVIEW
number VARCHAR(45)
rate INT
picture VARCHAR(45)
comment VARCHAR(200)
comment time VARCHAR(200)
RESERVATION_number INT
Indexes
USER_has_bookmark
USER_ID INT
bookmark_dib VARCHAR(45)
Indexes
pet
speices VARCHAR(45)
age INT(10)
gender VARCHAR(45)
sickness VARCHAR(45)
weight INT(10)
personality VARCHAR(45)
USER_ID INT
Indexes
bookmark
dib VARCHAR(45)
Indexes

3. 구현

3.1 릴레이션

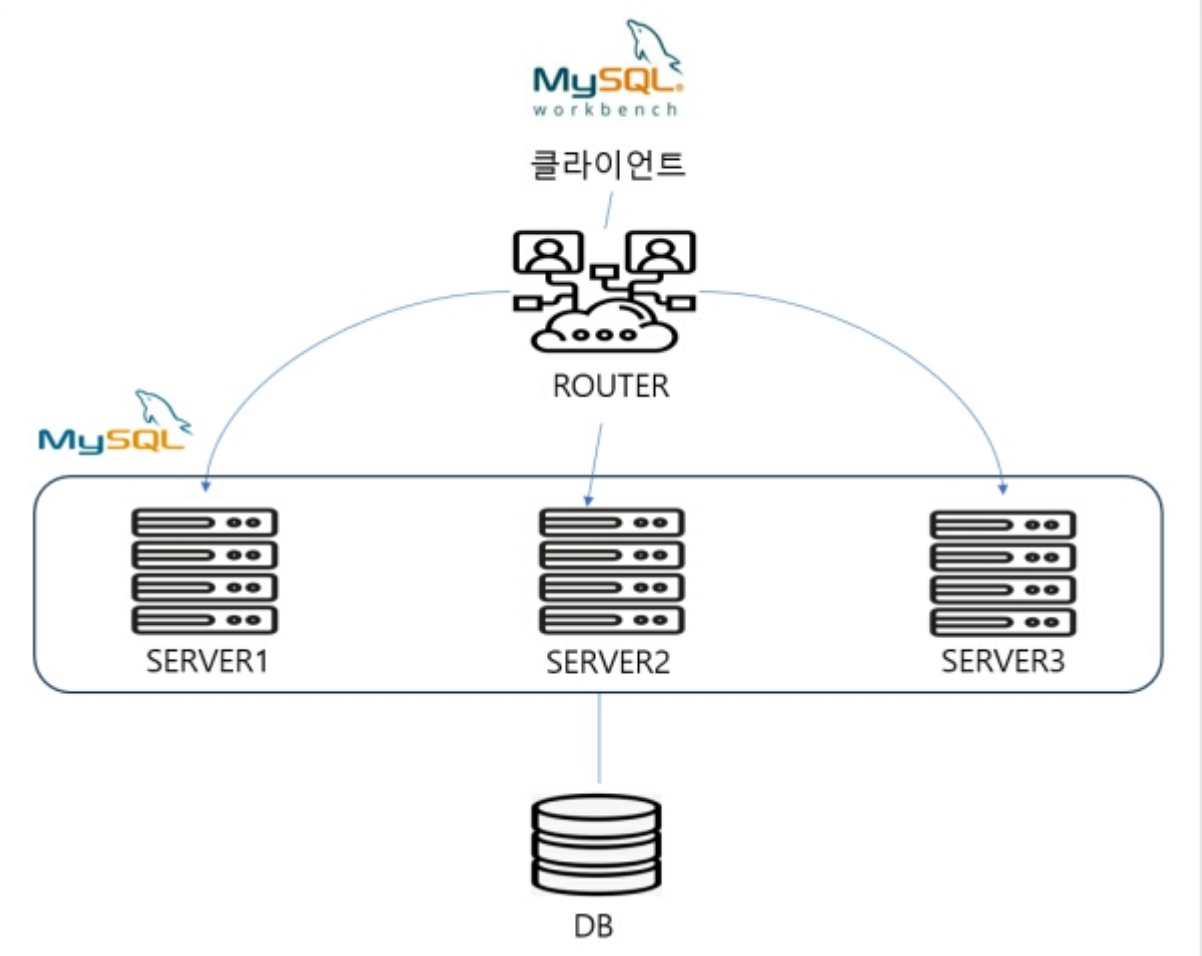
업체 테이블(가게 책임)	CREATE TABLE web. CEO (`ID` INT NOT NULL, `PW` VARCHAR(45) NOT NULL, `e-mail` VARCHAR(45) NOT NULL, `business license` VARCHAR(45) NOT NULL, `phone number` INT NOT NULL, `address` VARCHAR(45) NOT NULL, `type` VARCHAR(45) NOT NULL, PRIMARY KEY (`ID`)) ENGINE = InnoDB
사용자 테이블	CREATE TABLE web. USER (`ID` INT NOT NULL, `PW` VARCHAR(45) NOT NULL, `e-mail` VARCHAR(45) NOT NULL, `social security number` VARCHAR(45) NOT NULL, `phone number` INT NOT NULL, PRIMARY KEY (`ID`)) ENGINE = InnoDB
WOOF 매니저 테이블	CREATE TABLE web.WOOF_MANAGER (`manager number` INT NOT NULL, `gender` VARCHAR(10) NOT NULL, `career` VARCHAR(200) NULL, `car number` VARCHAR(10) NOT NULL, PRIMARY KEY (`manager number`)) ENGINE = InnoDB
예약 테이블	CREATE TABLE web. RESERVATION (`number` INT NOT NULL, `time` VARCHAR(45) NOT NULL, `place` VARCHAR(45) NOT NULL, `RESERVATION_status` VARCHAR(45) NOT NULL, `CEO_ID` INT NOT NULL, `USER_ID` INT NOT NULL, `order` INT NOT NULL, PRIMARY KEY (`number`), INDEX `fk_RESERVATION_CEO1_idx` (`CEO_ID` ASC) VISIBLE,

	INDEX `fk_RESERVATION_USER1_idx` (`USER_ID` ASC) VISIBLE, INDEX `fk_RESERVATION_order1_idx` (`order` ASC) VISIBLE, CONSTRAINT `fk_RESERVATION_CEO1` FOREIGN KEY (`CEO_ID`) REFERENCES `web`.`CEO` (`ID`) ON DELETE NO ACTION ON UPDATE NO ACTION, CONSTRAINT `fk_RESERVATION_USER1` FOREIGN KEY (`USER_ID`) REFERENCES `web`.`USER` (`ID`) ON DELETE NO ACTION ON UPDATE NO ACTION, CONSTRAINT `fk_RESERVATION_ORDER1` FOREIGN KEY (`order`) REFERENCES `web`.`WOOF_MANAGER` (`manager number`) ON DELETE NO ACTION ON UPDATE NO ACTION) ENGINE = InnoDB
리뷰 테이블	CREATE TABLE web.REVIEW (`number` VARCHAR(45) NOT NULL, `rate` INT NOT NULL, `picture` VARCHAR(45) NULL, `comment` VARCHAR(200) NULL, `comment time` VARCHAR(200) NULL, `RESERVATION_number` INT NOT NULL, PRIMARY KEY (`number`), INDEX `fk_REVIEW_RESERVATION1_idx` (`RESERVATION_number` ASC) VISIBLE, CONSTRAINT `fk_REVIEW_RESERVATION1` FOREIGN KEY (`RESERVATION_number`) REFERENCES `web`.`RESERVATION` (`number`) ON DELETE NO ACTION ON UPDATE NO ACTION) ENGINE = InnoDB
반려동물 테이블	CREATE TABLE web. PET (`number` VARCHAR(45) NOT NULL, `speices` VARCHAR(45) NOT NULL, `age` INT NOT NULL, `gender` VARCHAR(45) NOT NULL, `sickness` VARCHAR(45) NOT NULL, `weight` INT NOT NULL,

	`personality` VARCHAR(45) NOT NULL, `USER_ID` INT NOT NULL, PRIMARY KEY (`number`), CONSTRAINT `USERfk_pet_USER1` FOREIGN KEY (`USER_ID`) REFERENCES `web`.`USER` (`ID`) ON DELETE NO ACTION ON UPDATE NO ACTION) ENGINE = InnoDB
카드 테이블	CREATE TABLE web. CARD (`card number` INT NOT NULL, `USER_ID` INT NOT NULL, PRIMARY KEY (`card number`), INDEX `fk_card_USER1_idx` (`USER_ID` ASC) VISIBLE, CONSTRAINT `fk_card_USER1` FOREIGN KEY (`USER_ID`) REFERENCES `web`.`USER` (`ID`) ON DELETE NO ACTION ON UPDATE NO ACTION) ENGINE = InnoDB

3.2 기능별 쿼리

3.3 시스템 아키텍처



CLUSTER 방식으로 세 개의 서버를 구성하여 활동하던 서버가 다운돼도 다른 서버로 이동해서 서비스를 끊김 없이 제공할 수 있도록 하였다.

작업 비중 중 쓰기 작업의 비중이 높을 것으로 예상되어 읽기와 쓰기 모두 가능한 6446 포트를 세 개 서버에 전부 사용하였다.