

프로젝트 보고서

Waiting for JMT

데 베 겔 즈

2 0 1 9 2 1 0 8 5

2 0 1 9 2 1 1 0 1

2 0 1 9 2 1 0 5 1

곽 수 정

박 지 영

정 가 희

프로젝트 개요

개발하는 서비스에 대한 간단한 설명

온라인에서 맛집과 평점/리뷰를 제공해 주는 서비스를 토대로 오프라인에서 맛집을 직접 방문했을 때 웨이팅이 너무 많아 방문하지 못한 경험은 누구나 한 번씩 있을 것입니다.

따라서 온라인으로 음식점의 실시간 웨이팅 정보와 시간뿐만 아니라 동시에 음식점 위치, 평점, 리뷰 정보 등을 함께 제공해 주는 서비스를 개발하고자 하였습니다.

이 서비스를 통해 사용자들이 직접 맛집에 가지 않아도 실시간 웨이팅 정보를 알 수 있어서 매장 입구에 줄을 서서 기다리는 시간은 아끼고 평점 높고 대기 시간이 짧은 맛집은 발 빠르게 찾을 수 있을 것입니다.

제공 기능

서비스 제공 기능(Functionality) 설명



평점 3 이상

평점 3 이상의 음식
점들과 음식점들의
위치 제공 서비스



위치 정보

해당 음식점의 위치
제공 서비스



리뷰 및 평점

해당 음식점의 맛, 위생,
응접 등의 리뷰 점수와
평점 제공 서비스



웨이팅 정보

해당 음식점의 웨이팅
정보 제공 서비스

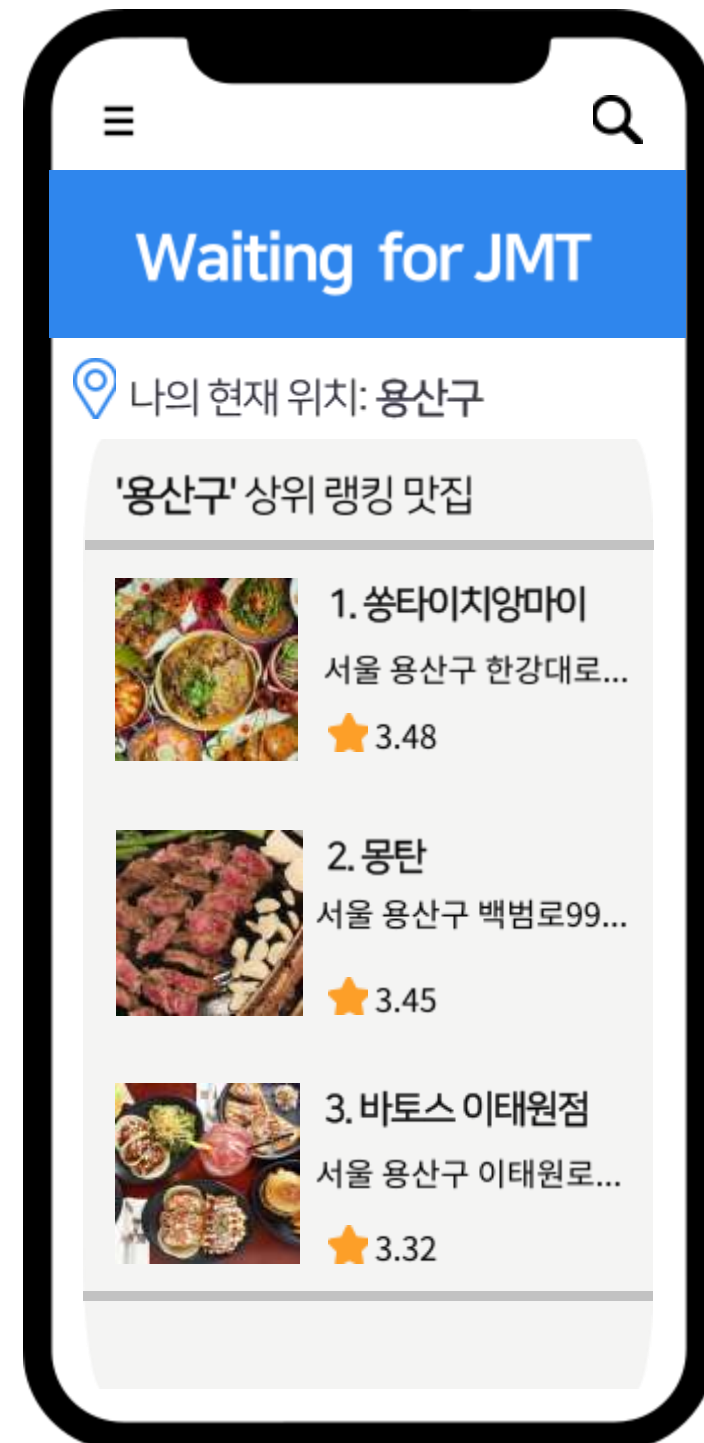


예약 서비스

해당 음식점에 예약 서
비스

사용자 인터랙션 시나리오

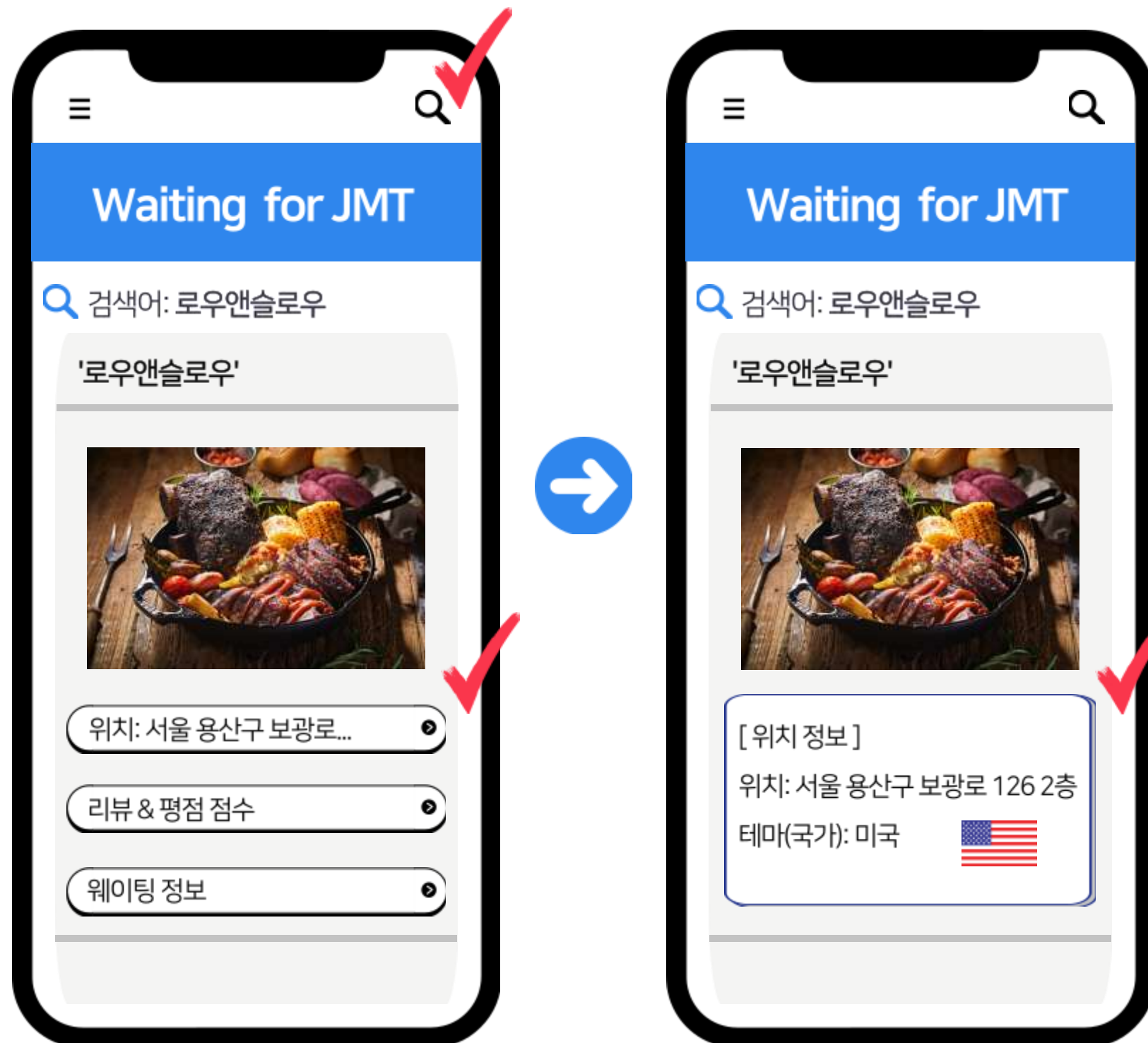
사용자 인터페이스 화면 1



1. 먼저 사용자가 어플을 실행했을 때,
평점 3점 이상의 높은 평가를 받은 음식점과 그 위치를 보여줍니다.

사용자 인터랙션 시나리오

사용자 인터페이스 화면 2



2. 사용자가 검색 기능을 통해 특정 음식점을 검색하면
위치, 리뷰+평점, 웨이팅 정보를 제공합니다.

위치를 선택하면 해당 음식점의 위치를 보여줍니다.

사용자 인터랙션 시나리오

사용자 인터페이스 화면 3



3. 리뷰+평점 정보를 통해 맛, 위생, 응접 점수와
전체 평점 점수를 알 수 있습니다.

4. 웨이팅 정보는 현재 몇 개의 팀이 기다리고 있는지와
예상 대기 시간, 오늘의 누적 방문객 수를 제공합니다.

사용자 인터랙션 시나리오

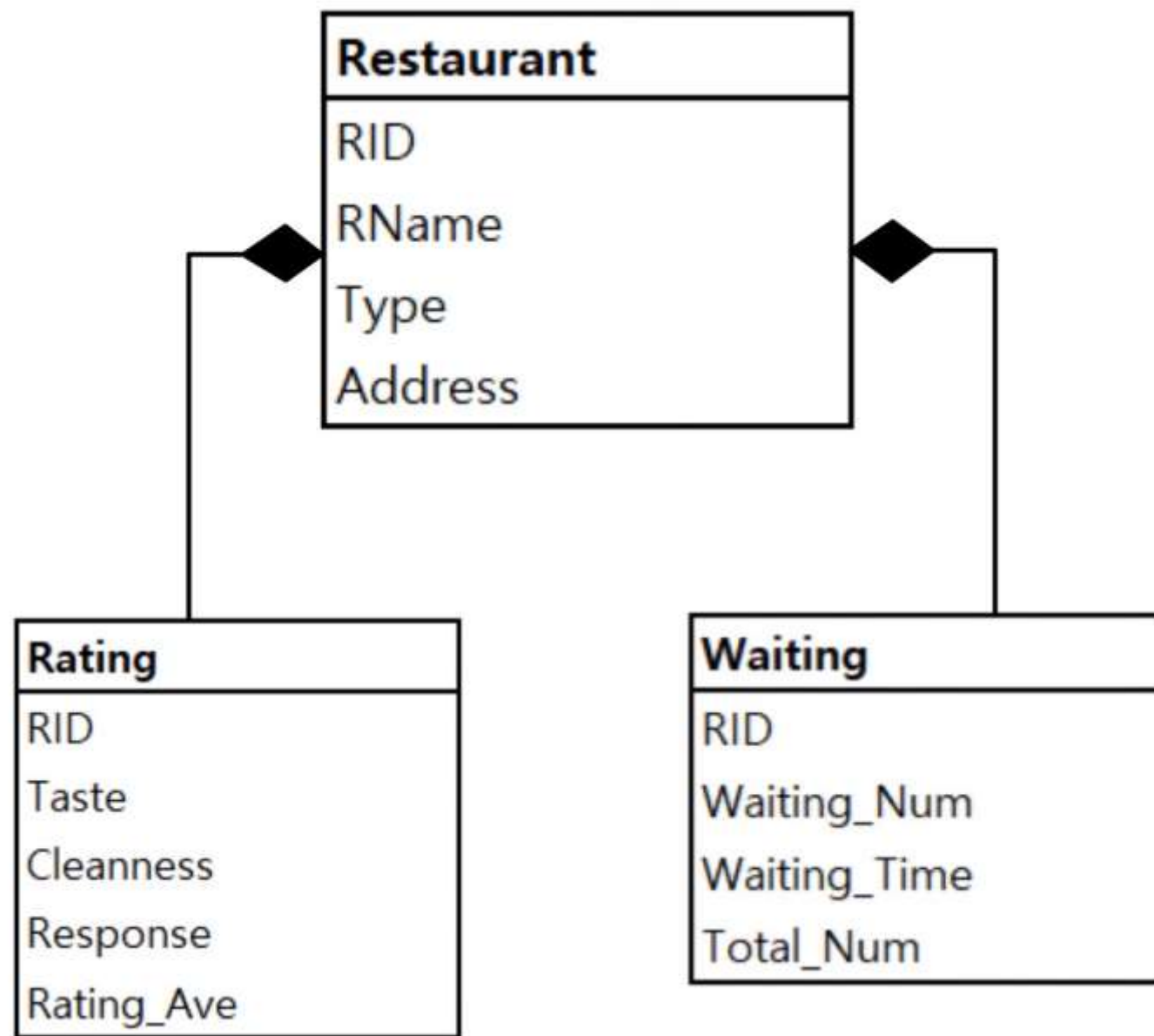
사용자 인터페이스 화면 4



5. 웨이팅 인터페이스 화면의 밑 쪽에 예약하기 버튼을 누르면 해당 음식점에 예약이 가능합니다.

6. 방금 예약건에 따라 달라진 대기 팀과 예상 대기 시간에 대한 웨이팅 정보가 보여집니다.

테러베이스 스키마



Relational 스키마

Restaurant (RID, RName, Type, Addr1, Addr2)

Waiting (RID, Wating_Num, Waiting_Time, Total_Num)

Rating (RID, Taste, Cleanness, Response, Rating_Ave)

SQL문

데이터베이스 스키마 생성하는 SQL문

// Restaurant 테이블 생성 후, 공공데이터포털의 용산구 음식점 데이터 csv 를 이용해 데이터 삽입

```
Create table Restaurant(RID int, Type varchar(20), RName varchar(40), Addr1 varchar(40), Addr2 varchar(40));
```

```
COPY Restaurant FROM 'C:\Users\Wjghga\youngsangu.csv' WITH CSV Encoding 'euc-kr';
```

// Rating, Waiting 테이블 생성

```
Create table Rating(RID int, Taste int, Cleanness int, Response int, Rating_Ave float);
```

```
Create table Waiting(RID int, Waiting_Num int, Waiting_Time int, Total_Num int);
```

//tuple들을 Rating, Waiting table에 삽입

```
insert into Rating values(" ", " ", " ", " ");
```

```
insert into Waiting values(" ", " ", " ", " ");
```

SQL문

서비스가 제공하는 각 기능과 연계되는 Query를 나타내는 SQL문

//평점 3이상인 음식점과 위치 뽑아내는 쿼리문

```
select Rname, Addr1, Addr2, rating_ave from Restaurant natural join Rating where Rating_Ave >= 3;
```

//음식점 이름만 모아서 view 생성

```
create view Name as select RName from Restaurant;
```

//사용자가 입력한 해당 음식점의 이름이 존재하는지 검토하는 쿼리문

```
select * from Name;
```

// 해당 음식점의 위치를 찾는 쿼리문

```
select Rname, Addr1, Addr2 from Restaurant where Rname = '' + namelist + '' ;
```

SQL문

서비스가 제공하는 각 기능과 연계되는 Query를 나타내는 SQL문

// 해당 음식점의 맛, 위생, 응접 등의 리뷰 점수와 평점을 뽑아내는 쿼리문

```
select Rname, Taste, Cleanness, Response, Rating_Ave from Restaurant natural join Rating where RName = "" + namelist + "" ;
```

// 해당 음식점의 웨이팅 정보를 뽑아내는 쿼리문

```
select RID, Rname, Waiting_Num, Waiting_Time, Total_Num from Restaurant natural join Waiting where RName = "" + namelist + "";
```

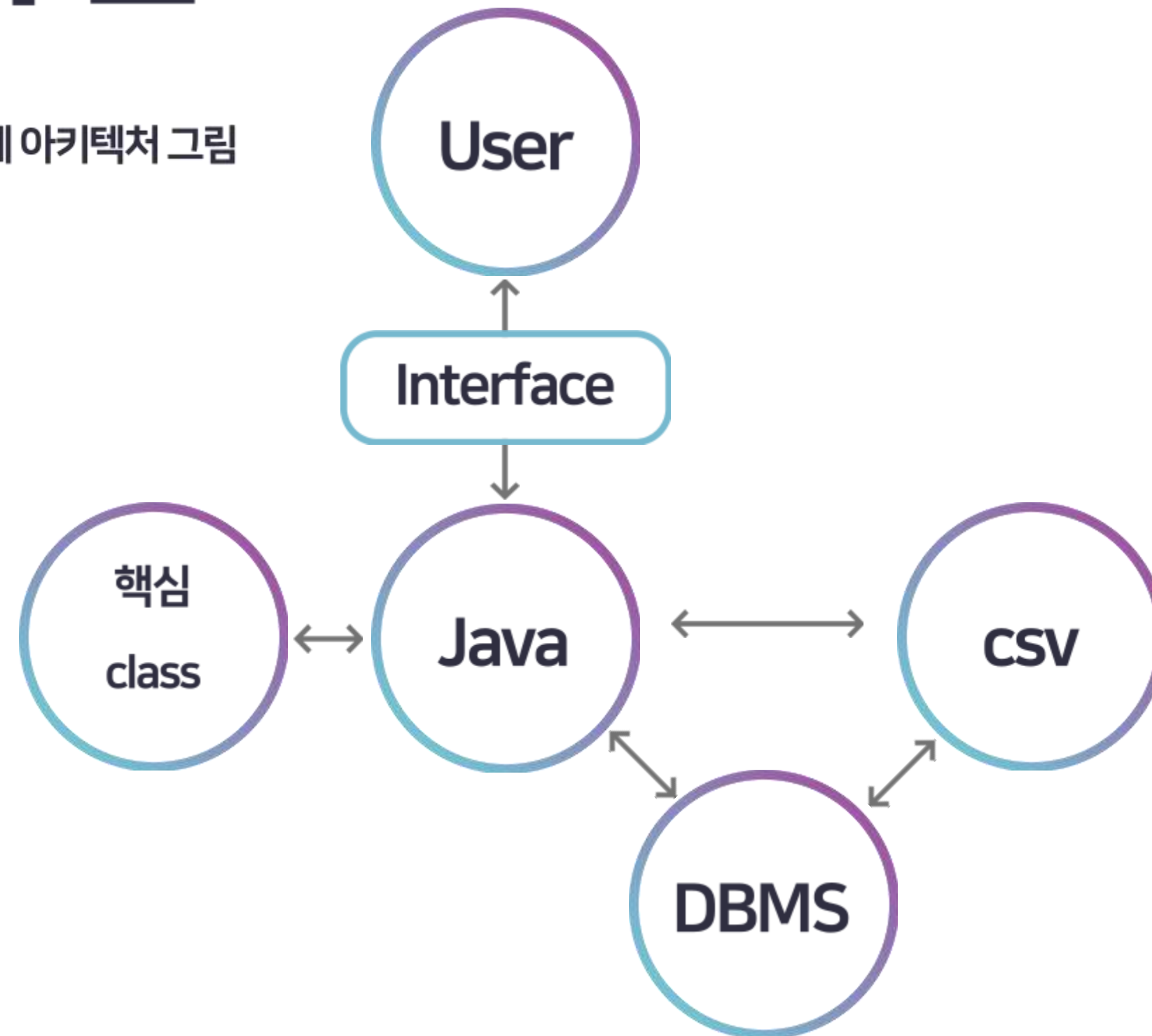
//사용자가 해당 음식점에 예약하면 현재 대기 수와 대기 시간, 누적 방문 팀 수를 업데이트하는 쿼리문

```
Waiting_Num += 1; Waiting_Time += 10; Total_Num += 1;
```

```
UPDATE Waiting SET Waiting_Num = ""+ Waiting_Num + "", Waiting_time = ""+ Waiting_Time + "", Total_Num = "" + Total_Num + "" W  
HERE RID = "" + RID + "";
```

시스템 구조

서비스 구성 컴포넌트 - 전체 아키텍처 그림



개발 환경

DBMS: PostgreSQL

사용 언어: Java

공공 데이터: 서울특별시 용산구_맛집

<https://www.data.go.kr/data/15066516/fileData.do>

공공데이터포털의 데이터를 CSV로 다운 받은 후, 아래와 같은 방법으로 postgresql의 DB로 불러와 해당 Restaurant 테이블에 insert하여 개발 환경을 구축하였다.

1. postgresql에 csv와 동일한 자료형을 가지는 table을 생성
2. COPY FROM 구문을 활용하여 csv에 있는 데이터들을 새로 생성한 table에 삽입

DB project plan

THANK YOU