Databases Programming Project: **Proposal Report**

1. **프로젝트 주제 (Project topic)**

|  |
| --- |
| 붕어빵 정보 시스템 (Fish-shaped Bun Information System). 붕어빵을 판매하는 가게들은 지도 앱에 나타나지 않아 찾아가기가 힘들고, 그로 인해 정보가 등록되지 않아 소비자들의 리뷰를 얻기 힘들다. 이를 해결하기 위해 붕어빵 가게에 대한 영업 시간, 리뷰, 위치 정보를 등록하고 관리하는 시스템 개발을 목표로 하고 있다. |

1. **사용자 (역할) (Users / Roles)**

|  |
| --- |
| 붕어빵 정보 시스템을 이용하는 사용자로는 소비자(Consumer), 가게 주인(Owner), 관리자(Admin)가 있다.  소비자는 붕어빵을 먹고 리뷰를 등록한다. 리뷰에는 별점, 한줄평, 작성 시간이 포함된다. 소비자는 다른 소비자가 작성한 붕어빵 가게에 대한 리뷰를 조회할 수 있다.  가게 주인은 붕어빵 가게 이름, 판매하는 붕어빵 종류, 가격, 영업 시간, 가게 위치 정보를 등록한다. 가게 주인은 자신의 가게에 등록된 리뷰를 조회할 수 있다.  관리자는 악의적인 리뷰, 악성 소비자 정보, 잘못 입력된 가게 정보들을 수정, 삭제할 수 있다. |

1. **기능 (Functions)**

|  |
| --- |
| 공통 사항   * 모든 시스템 사용자는 시스템을 사용하기 위해 회원가입 후 로그인 상태여야 한다. 회원가입과 로그인 과정은 트랜잭션으로 처리된다.   소비자 (Consumer)   * 붕어빵 가게 조회: 인근 붕어빵 가게를 조회한다. * 붕어빵 가게를 거리순, 가격순으로 정렬하여 조회할 수 있다. * 리뷰: 붕어빵에 대한 리뷰를 등록한다. * 별점과 한줄평을 등록한다. 이때 작성 시간이 함께 자동으로 리뷰에 포함된다. * 자신이 작성한 리뷰를 조회할 수 있다. * 자신이 작성한 리뷰를 수정할 수 있다. * 다른 소비자가 작성한 리뷰를 조회할 수 있다. * 리뷰를 조회할 때에는 가게별 별점순, 시간순으로 조회할 수 있다.   가게 주인(Owner)   * 가게 정보 등록: 판매하는 붕어빵 종류, 가격, 영업 시간, 가게 위치 정보를 등록한다. * 가게 정보를 작성한다. * 작성한 가게 정보를 수정할 수 있다. * 가게 리뷰 조회 * 가게에 등록된 리뷰를 시간순, 별점순으로 정렬하여 조회할 수 있다.   관리자(Admin)   * 악의적인 리뷰 삭제: 악의적인 리뷰라고 판단될 경우, 관리자는 리뷰를 삭제할 수 있다 * 악성 소비자 정보 삭제: 소비자가 지속적으로 악의적인 리뷰를 작성할 경우, 소비자의 회원가입 정보를 삭제한다. * 잘못 입력된 가게 정보 수정 및 삭제: 잘못된 가게 정보일 경우, 관리자는 직접 잘못된 정보를 수정 및 삭제할 수 있다. |

1. **데이터베이스 스키마 (Database schema)**

|  |
| --- |
| 1. Table member { 2. id integer [primary key] 3. password varchar(20) 4. phone\_number integer(20) 5. } 6. Table consumer { 7. consumer\_id integer 8. nickname varchar(8) 9. } 10. Table review { 11. review\_id integer [primary key] 12. consumer\_id integer 13. shop\_id integer 14. rating numeric(1,0) 15. comment tinytext(100) 16. review\_time timestamp 17. } 18. Table owner { 19. owner\_id integer 20. shop\_id integer 21. } 22. Table shop { 23. shop\_id integer [primary key] 24. shop\_name varchar(10) 25. owner\_id integer 26. bun\_id integer 27. latitude real 28. longitude real 29. } 30. Table bun { 31. bun\_id integer [primary key] 32. taste varchar(10) 33. price numeric(5,0) 34. } 35. Table admin { 36. admin\_id integer 37. } 38. Ref: consumer.consumer\_id - member.id 39. Ref: owner.owner\_id - member.id 40. Ref: admin.admin\_id - member.id 41. Ref: review.consumer\_id - consumer.consumer\_id 42. Ref: review.shop\_id - shop.shop\_id 43. Ref: owner.shop\_id < shop.shop\_id 44. Ref: shop.bun\_id < bun.bun\_id |