**요구사항 정의서**

**목차**

1. 프로젝트 개요
   1. 프로젝트 개요
      1. 프로젝트 소개
      2. 프로젝트 배경
      3. 프로젝트 시나리오
2. 설계
   1. ERD
   2. 릴레이션 스키마
3. 구현
   1. 릴레이션
   2. 기능 별 쿼리
      1. 회원 관련 기능
      2. 로그인 관련 기능
      3. 상품 관련 기능
      4. 장바구니 기능
      5. 주문 관련 기능
      6. 판매 관련 기능
   3. 시스템 아키텍처
      1. Cluster 구성 내용
      2. Cluster 구성을 사용한 이유
      3. MySQL Cluster 구성 방법

1. 프로젝트 개요

1.1. 프로젝트 개요

1. 1. 1. 프로젝트 소개

최근 소비자들 간 협력을 통해 다양한 제품을 합리적인 가격으로 구매할 수 있는 효과적인 방법 중 하나로 공동구매가 부상하고 있습니다. 특히, 토스와 같은 대기업이 시장에 참여함으로 공동구매 시장이 점차 성장하고 있습니다.

단체 구매력이 강화되어 저희 플랫폼의 가격 경쟁력이 향상되면 더 많은 소비자 참여를 모을 수 있습니다.

저희 공동구매 플랫폼 PamPam은 다양한 품목들을 제공하여 여러 사람들이 참여해 보다 합리적인 가격으로 원하는 상품을 구매할 수 있습니다. 소비자들은 그룹을 형성하여 상품을 함께 구매함으로 그룹 내에서 혜택을 공유할 수 있습니다.

* 1. 2. 프로젝트 배경

텍스트, 스크린샷, 도표, 그래프이(가) 표시된 사진

자동 생성된 설명

최근 국내 근원물가의 증가로 인해 소비자들에게 경제적인 부담을 가져오고 있습니다. 근원물가가 상승할수록 소비자가 필수 소비제를 구매하는 데 더 많은 돈을 지불해야 합니다.

또한, 현재 근원물가는 최근 여러 전쟁과 그로 인한 유가 상승, 에너지 값 상승하여 국내 근원물가 상승세가 지속적으로 상승하고 있습니다.

언제 끝날지 모르는 이러한 물가 상승 배경은 저희 공동구매 서비스가 소비자의 요구사항을 충족시켜줄 하나의 대안이 될 수 있습니다.

1. 1. 3. 프로젝트 시나리오

2. 설계

2. 1. 개념 설계 – ERD

도표, 그림, 스케치, 종이접기이(가) 표시된 사진

자동 생성된 설명

2. 2. 논리 설계 – 릴레이션 스키마

텍스트, 스크린샷, 도표, 폰트이(가) 표시된 사진

자동 생성된 설명

3. 구현

* 1. 릴레이션

|  |  |
| --- | --- |
| **구매자 회원가입 테이블** | CREATE TABLE Consumer (  Consumer\_ID VARCHAR(45) PRIMARY KEY NOT NULL UNIQUE,  Consumer\_PW VARCHAR(30),  Consumer\_Name VARCHAR(20) NOT NULL,  Consumer\_Addr VARCHAR(50) NOT NULL,  Consumer\_pNum VARCHAR(15) NOT NULL,  SocialLogin VARCHAR(30) NULL DEFAULT '소셜 로그인 사용 안함'  ); |
| **판매자 회원가입 테이블** | CREATE TABLE Seller (  Seller\_ID VARCHAR(45) PRIMARY KEY NOT NULL,  Seller \_PW VARCHAR(30) NOT NULL,  Seller \_Name VARCHAR(20) NOT NULL,  Seller \_Addr VARCHAR(50) NOT NULL,  Seller \_pNum VARCHAR(15) NOT NULL,  Seller \_Number VARCHAR(30) NOT NULL  ); |

|  |  |
| --- | --- |
| **상품 테이블** | CREATE TABLE Product (  Product\_ID INT AUTO\_INCREMENT PRIMARY KEY NOT NULL UNIQUE,  Image VARCHAR(100),  Name VARCHAR(20) NOT NULL,  Information TEXT(2000),    Seller\_ID VARCHAR(45),  Category\_ID INT,  FOREIGN KEY (Seller\_ID) REFERENCES Seller(Seller\_ID),  FOREIGN KEY (Category\_ID) REFERENCES Category(Category\_ID)    ); |
| **가격 테이블** | CREATE TABLE Price (  Price\_ID INT AUTO\_INCREMENT PRIMARY KEY NOT NULL,  Price INT NOT NULL,  Number\_of\_People INT NOT NULL,    Product\_ID INT,  FOREIGN KEY (Product\_ID) REFERENCES Product(Product\_ID)  ); |
| **카테고리 테이블** | CREATE TABLE Category (  Category\_ID INT AUTO\_INCREMENT PRIMARY KEY,  Category\_Name VARCHAR(40)  ); |
| **마감일 테이블** | CREATE TABLE DeadLine (  DeadLine\_ID INT AUTO\_INCREMENT PRIMARY KEY NOT NULL,  Deadline DATE,  FOREIGN KEY (Product\_ID) REFERENCES Product(Product\_ID)  ); |
| **좋아요 테이블** | CREATE TABLE Heart (  Consumer\_ID VARCHAR(45),  Product\_ID INT,  FOREIGN KEY (Consumer\_ID) REFERENCES Consumer(Consumer\_ID),  FOREIGN KEY (Product\_ID) REFERENCES Product(Product\_ID)  ); |
| **주문 테이블** | CREATE TABLE Order (  Order\_idx INT AUTO\_INCREMENT PRIMARY KEY NOT NULL,  Order\_Date DATE,    Product\_ID INT,  Consumer\_ID VARCHAR(45),  FOREIGN KEY (Product\_ID) REFERENCES Product(Product\_ID),  FOREIGN KEY (Consumer\_ID) REFERENCES Consumer(Consumer\_ID)  ); |
| **장바구니 테이블** | CREATE TABLE Shopping\_Cart (  Product\_ID INT,  Consumer\_ID VARCHAR(45),  FOREIGN KEY (Product\_ID) REFERENCES Product(Product\_ID),  FOREIGN KEY (Consumer\_ID) REFERENCES Consumer(Consumer\_ID)  ); |

* 1. 기능 별 쿼리
     1. 회원 관련 기능

|  |  |
| --- | --- |
| **회원가입(일반회원)** | INSERT INTO Consumer(`Consumer\_pNum`, `Consumer\_Addr`,`Consumer\_Name`,`Consumer\_PW`,`Consumer\_ID`)  values('전화번호', '주소, '이름', '비밀번호', '아이디'); |
| **회원가입(소셜 로그인)** | INSERT INTO Consumer(Consumer\_pNum, Consumer\_Addr, Consumer\_Name, Consumer\_ID, SocialLogin)  values('전화번호', '주소', '이름', '아이디', '소셜로그인'); |
| **회원가입(판매자)** | INSERT INTO Seller(Business\_Number, Seller\_pNum, Seller\_Addr, Seller\_Name, Seller\_PW, Seller\_ID) values('사업자번호', '전화번호', '주소', '이름', '비밀번호', '아이디'); |
| **회원정보를 수정** | UPDATE Consumer SET Consumer\_PW ='비밀번호' WHERE Consumer\_ID='아이디'; |

* + 1. 로그인 관련 기능

|  |  |
| --- | --- |
| **일반 로그인** | select \* from Consumer where Consumer\_ID = '아이디' , Consumer\_PW = '비밀번호'; |
| **소셜 계정으로 로그인** | Select \* from Consumer where SocialLogin='소셜계정'; |

* + 1. 상품관련 기능

|  |  |
| --- | --- |
| **판매자 상품 등록** | INSERT INTO Product(Image, Name, Information) values('상품\_이미지', '상품\_이름', '상품\_정보'); |
| **상품 검색** | SELECT \* FROM Product  WHERE Name LIKE '상품\_이름'; |
| **카테고리 선택** | SELECT \* FROM Product  INNER JOIN Category  ON Product.Category\_ID = Category.Category\_ID  WHERE Product.Category\_ID = 카테고리\_번호; |
| **판매자 상품 삭제** | DELETE FROM Product  Where Product\_ID = 상품\_아이디; |

3.2.4 장바구니 기능

|  |  |
| --- | --- |
| **장바구니에 상품 추가** | INSERT INTO Shopping\_Cart(Consumer\_ID) values('구매자\_아이디'); |
| **관심 상품 목록을 조회** | SELECT \*  FROM Shopping\_Cart  INNER JOIN Product  ON Product.Product\_ID = Shopping\_Cart.Product\_ID  WHERE Consumer\_ID= "구매자\_아이디"; |

* + 1. 주문관련 기능

|  |  |
| --- | --- |
| **구매자 주문 내역 조회** | SELECT \*  FROM Buy  INNER JOIN Product  ON Buy.Product\_ID = Product.Product\_ID  WHERE Consumer\_ID= "구매자\_아이디"; |
| **공동구매 인원수별 가격 조회** | Select pricePerPerson  from Price  where Product\_ID=1 and Number\_Of\_People = 총\_인원수; |

3.2.5 판매관련 기능

|  |  |
| --- | --- |
| **판매자 판매 내역 조회** | SELECT Consumer\_ID, Product.Name  FROM Buy  LEFT JOIN Product  ON Buy.Product\_ID = Product.Product\_ID  WHERE Product.Seller\_ID = 'chlghtls'; |

* 1. 시스템 아키텍처
     1. Cluster 구성 내용

텍스트, 도표, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

* DB1 컴퓨터에 Primary 설정
* DB2 컴퓨터에 Primary 설정
* DB3 컴퓨터에 Primary 설정
  + 1. Cluster 구성한 이유

저희 공동구매 서비스는 특정 이벤트 시간에 많은 구매 행위가 발생합니다. 따라서 Multi-Primary-Mode로 구성하여 하나의 DB 서버에 구매 행위가 몰려서 발생하는 병목현상을 줄일 수 있습니다.

Replication 구성과 비교했을 때 Cluster 구성은 DB 서버 증설이 비교적 간단하고, 유연한 대처가 가능합니다. 하지만, Replication은 DB서버와 스토리지를 같이 증설해야 되기 때문에 특정 시간대에 끊김 없는 서비스를 제공하는 것은 Cluster 구성이 더 유리합니다.