

소프트웨어공학 프로젝트 보고서

<과제 3 : Requirement Analysis & Design - 의류 쇼핑 사이트>

4분반 6조

B511187 정지환

B611128 유창조

B774012 유민지

1. 역할 분담

팀원 전체 참여: **requirement list** 작성, **use case diagram** 작성, **Design class diagram** 결합 작성

정지환 : **use case 5, 6, 7, 8**에 대해 **use case description, communication diagram** 작성, **class diagram** 판매자 관련된 부분 작성 및 구현

유창조: **use case 1,2,3,4**에 대해 **use case description, communication diagram** 작성, **class diagram** 회원 관련된 부분 작성 및 구현

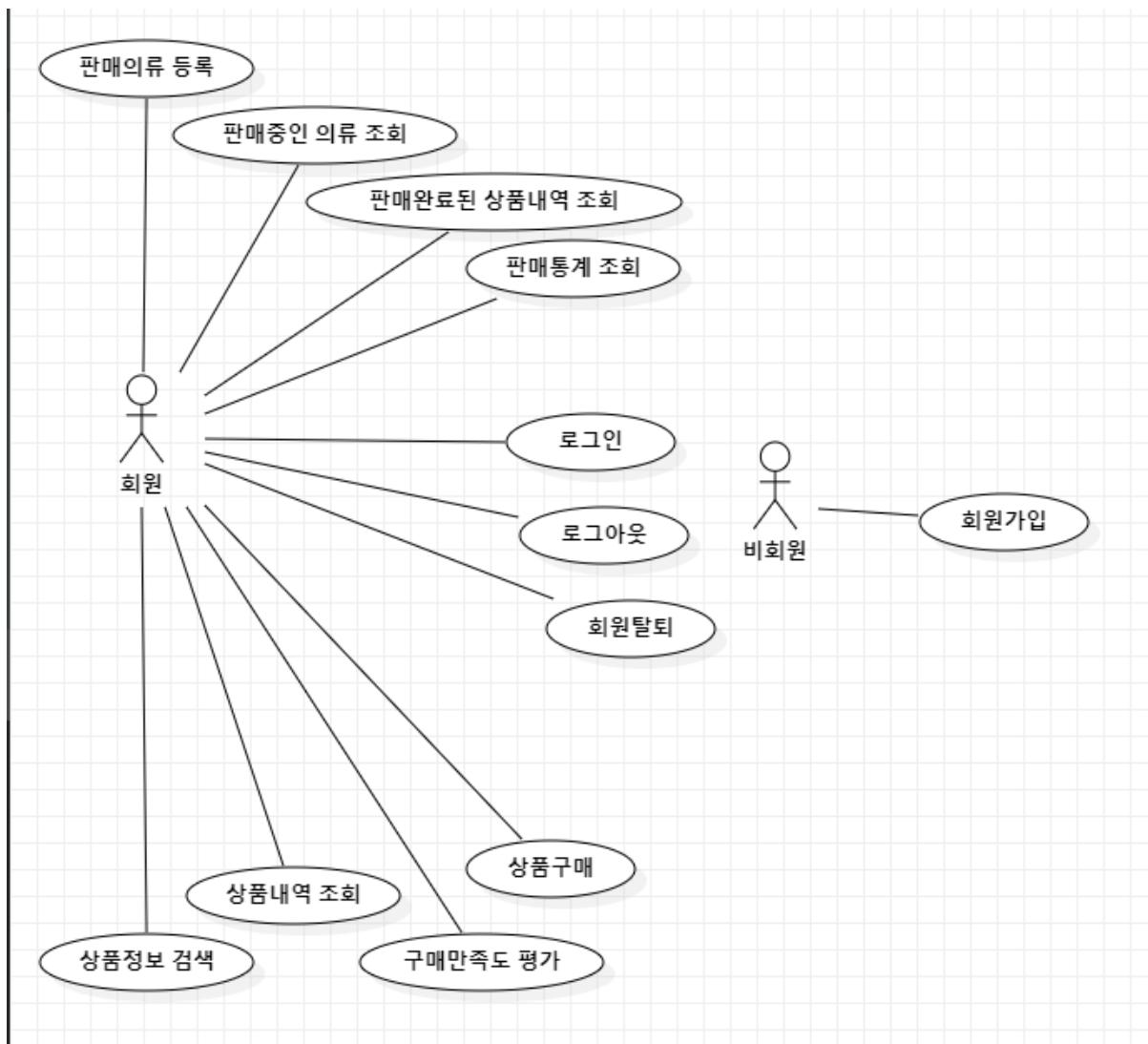
유민지: **use case 9,10,11,12 communication diagram** 작성, **class diagram** 구매자(**purchaser**)관련 부분 작성 및 구현

2. Requirement List

No.	Requirements	Use case
1	사용자는 의류 쇼핑 사이트 사용 권한을 얻기 위해서 회원 가입을 해야 한다. 자신의 기본 정보(이름, 주민번호)과 ID/Password를 입력해야 한다.	회원 가입
2	회원은 이 사이트에서 탈퇴할 수 있다. 회원 탈퇴와 동시에 회원의 시스템의 사용 권한은 소멸되어야 한다.	회원 탈퇴

3	회원은 등록한 ID와 password로 사이트에 로그인 할 수 있다.	로그인
4	회원은 쇼핑 사이트에서 로그아웃할 수 있다.	로그아웃
5	판매자는 판매하려는 의류를 등록할 수 있다. 등록 시 상품명, 제작회사명, 가격, 수량 정보를 입력해야 한다.	판매 의류 등록
6	판매자는 자신이 등록한 판매 중인 의류 상품 리스트를 조회할 수 있다.	등록 상품 조회
7	판매자는 자신이 판매 완료한 의류 상품을 조회할 수 있다. 이때 상품명의 오름차순으로 정렬해서 출력하며 각 상품에 대해 상품명, 제작회사명, 가격, 판매된 수량, 평균 구매만족도를 출력한다	판매 완료 상품 조회
8	회원이 현재까지 판매한 상품에 대한 리스트를 출력한다 각 항목은 상품명, 상품 판매 총액, 평균 구매만족도를 출력한다.	판매 상품 통계
9	회원이 상품명을 입력하여 상품을 검색할 수 있으며 이에 해당하는 상세한 정보(판매자ID, 상품명, 제작회사명, 가격, 남은 수량)가 출력된다.	상품 정보 검색
10	회원은 출력된 상품을 즉시 구매 할 수 있다.	상품 구매
11	회원은 상품명, 제작회사명, 판매자, 가격, 평균 구매 만족도 정보가 있는 구매내역을 조회할 수 있다. 이때 상품명의 오름차순으로 정렬해서 출력한다.	상품 구매 내역 조회
12	구매자는 조회된 구매 내역 중에서 하나의 상품을 선택해서 구매만족도를 평가할 수 있다. 구매 만족도는 1-5 사이의 정수 값을 입력한다.	구매만족도 평가

3. usecase diagram

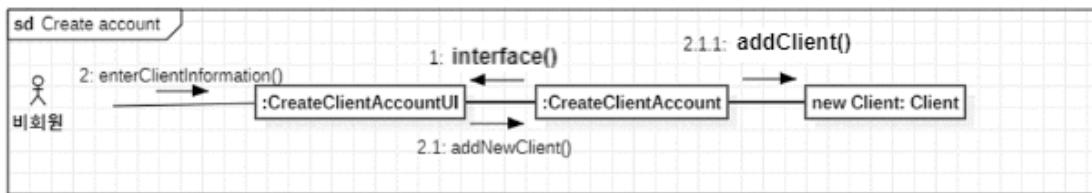


4. Use case descriptions & communication diagram

Use case 1: 회원가입

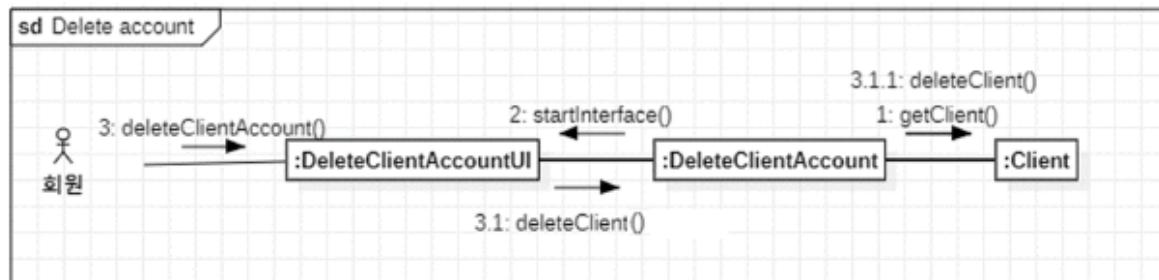
Actor Action	System Response
--------------	-----------------

1. 회원가입 항목과 이름, 주민번호, ID, Password 를 입력한다.	2. 회원가입이 완료되면 이름, 주민번호, ID, Password 가 출력된다.
---	---



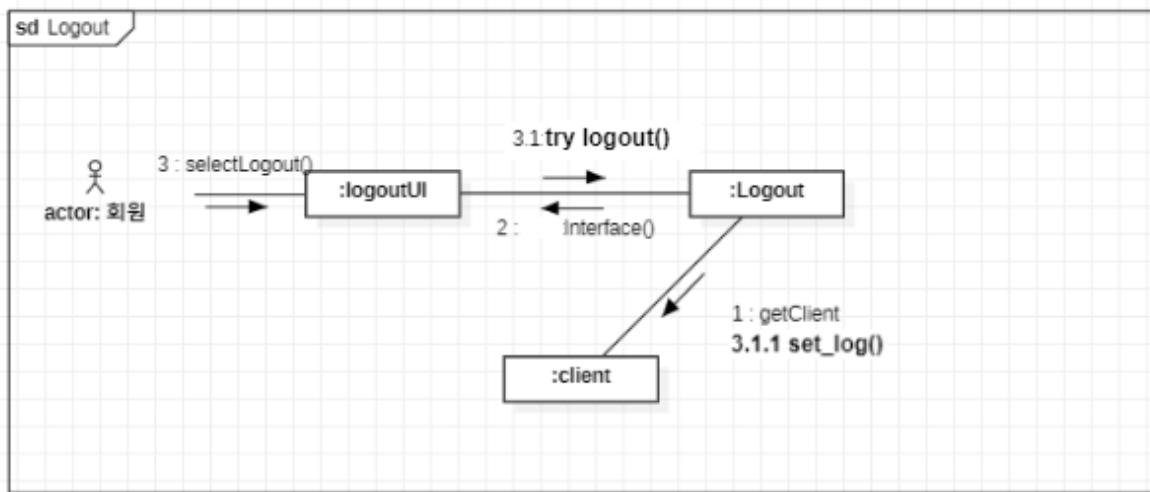
Use case 2: 회원 탈퇴

Actor Action	System Response
1. 회원 탈퇴 항목을 입력한다.	2. 회원 탈퇴 기능을 수행하고 ID 를 출력한다.



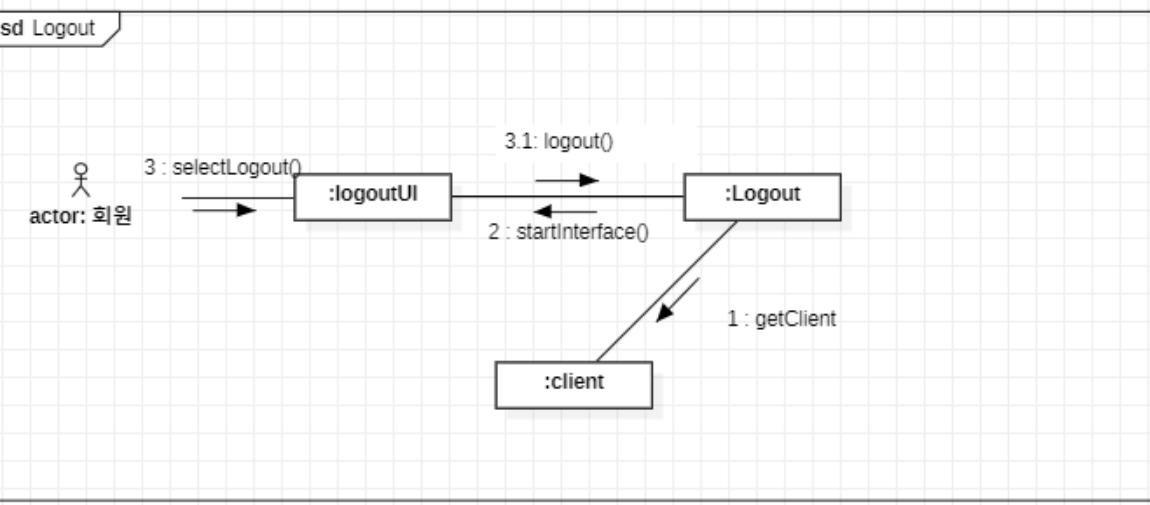
Use case 3 description: 로그인

Actor Action	System Response
1. 로그인 항목과 ID, Password 를 입력한다.	2. 로그인을 완료하고 로그인한 ID 와 Password 를 출력한다.



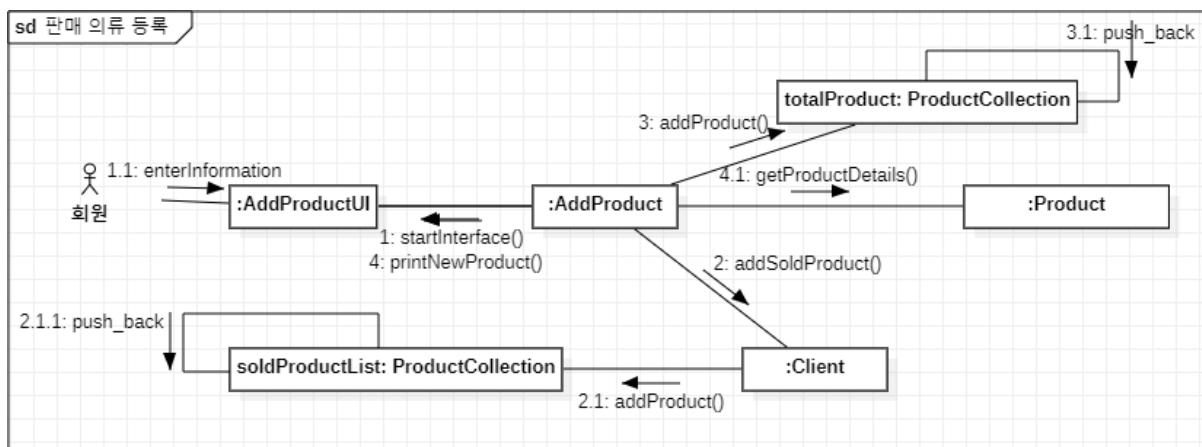
Use case 4 description: 로그아웃,

Actor Action	System Response
1. 로그아웃 항목을 선택한다.	2. 로그아웃을 완료하고 로그아웃한 ID를 출력한다.



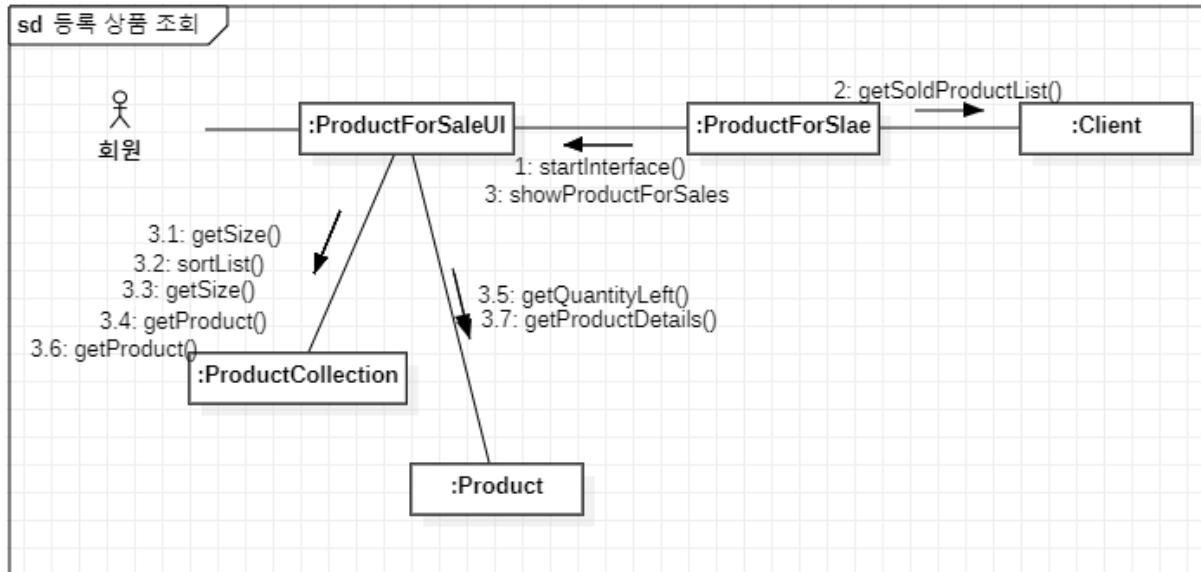
Use case 5: 판매 의류 등록

Actor Action	System Response
<p>1. 판매 의류 등록 메뉴(3 1)을 선택하고, 상품명, 제작회사명, 가격, 수량을 입력한다.</p>	<p>2. “3.1. 판매 의류 등록”을 출력하고 입력받은 상품명, 제작회사명, 가격, 수량 정보를 출력한다.</p>



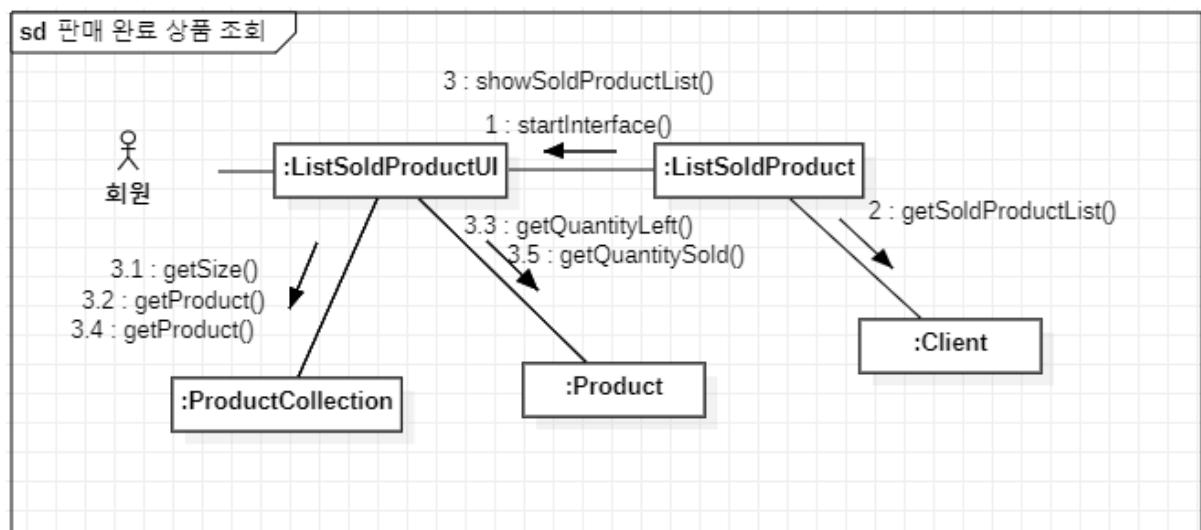
Use case 6: 등록 상품 조회

Actor Action	System Response
<p>1. 등록 상품 조회 메뉴(3 2)를 선택한다.</p>	<p>2. “등록 상품 조회”를 출력하고 회원이 등록한 판매 의류의 상품명, 제작회사명, 가격, 수량 정보의 리스트를 출력한다.</p>



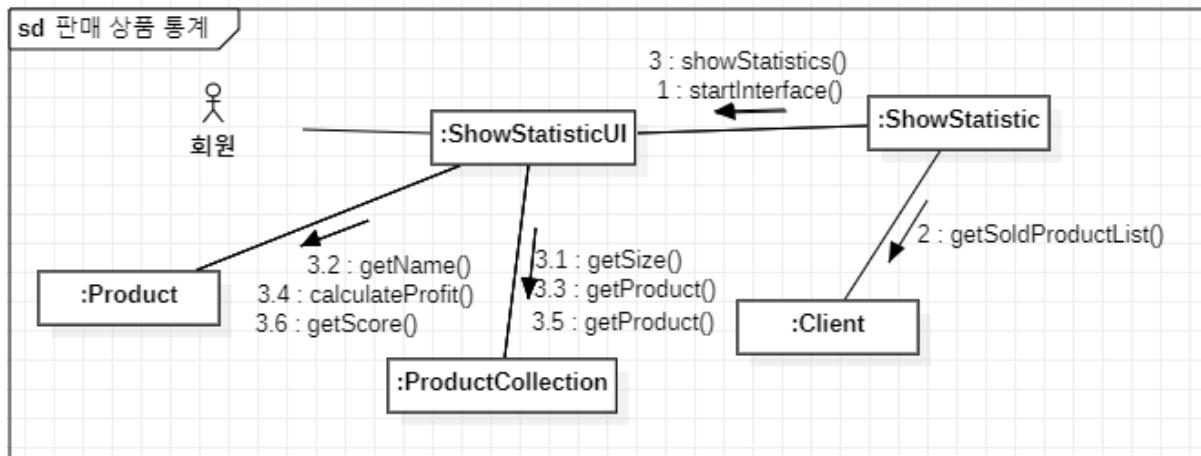
Use case 7: 판매 완료 상품 조회

Actor Action	System Response
1. 등록 상품 조회 메뉴(3 3)를 선택한다.	2. “판매 완료 상품 조회”를 출력하고 회원이 등록한 상품중 판매가 완료된 상품의 상품명, 제작회사명, 가격, 판매된 수량, 평균 구매만족도 정보의 리스트를 출력한다.



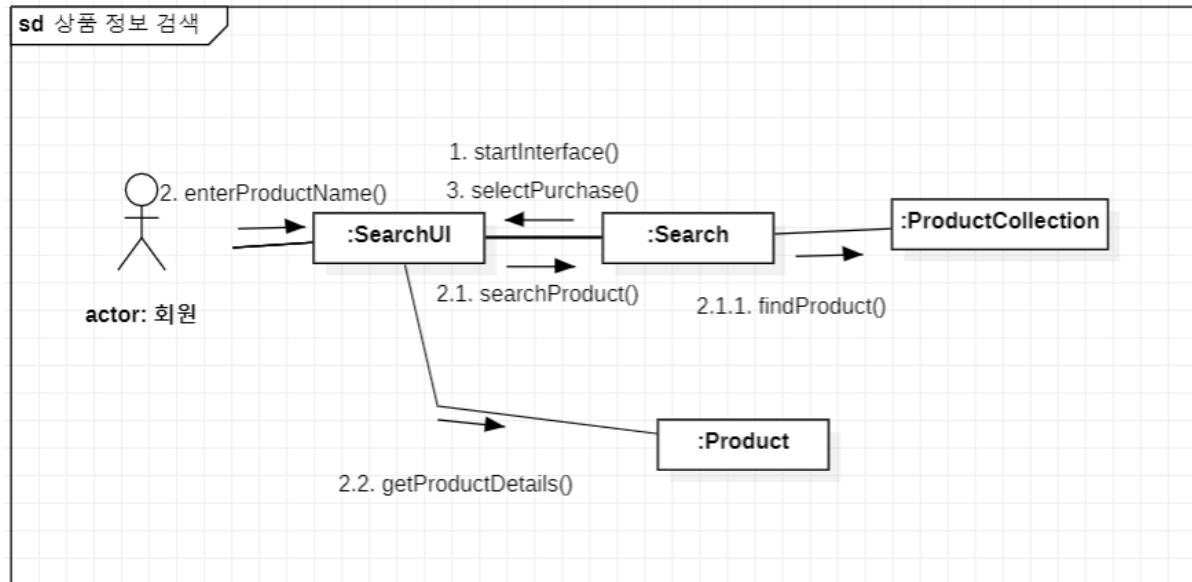
Use case 8: 판매 상품 통계

Actor Action	System Response
<p>1. 판매 상품 통계(5 1) 메뉴를 선택한다.</p>	<p>2. “판매 상품 통계”를 출력하고 회원이 등록한 상품중 판매된 상품의 상품명, 상품 판매 총액, 평균 구매만족도 정보의 리스트를 출력한다.</p>



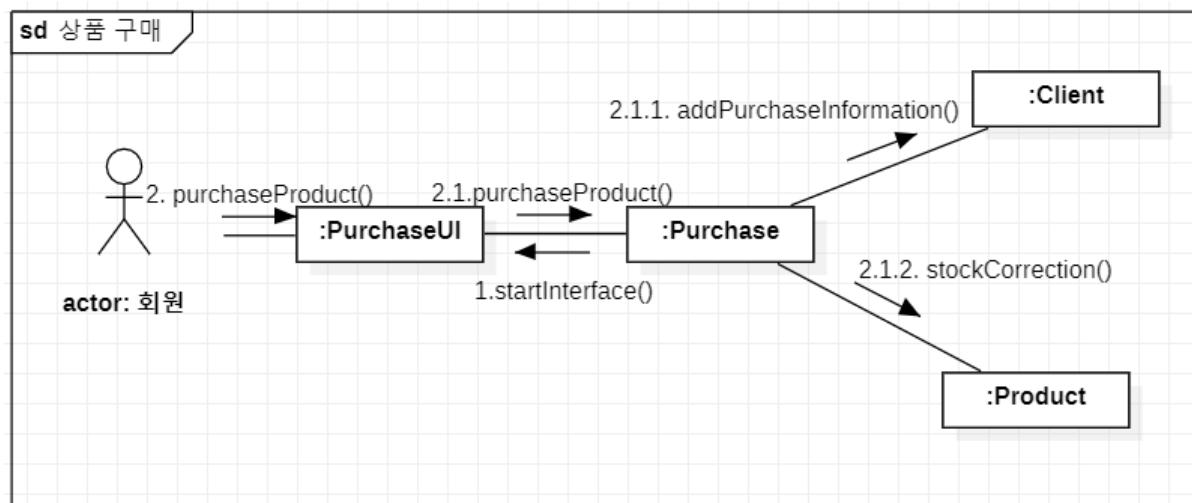
Use case 9: 상품 정보 검색

Actor Action	System Response
<p>1. 상품 정보 검색(4 1) 메뉴를 선택한다.</p>	<p>2. “4.1. 상품 정보 검색”을 출력한다.</p>
<p>3. 상품명을 입력한다.</p>	<p>4. 입력된 상품의 정보를 [판매자ID] [상품명] [제작회사명] [가격] [남은수량] [평균 구매만족도] 순으로 출력한다.</p>



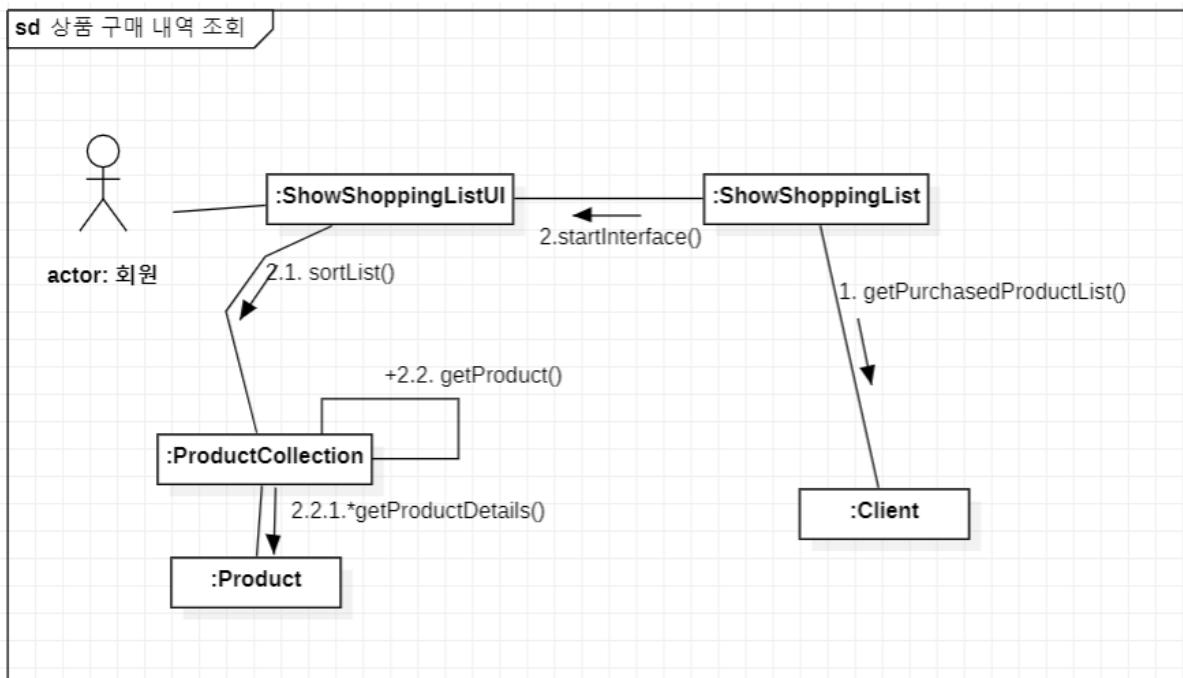
Use case 10 : 상품 구매

Actor Action	System Response
1. 상품 구매(4 2) 메뉴를 선택한다.	2. “4.2. 상품 구매”를 출력한다. 검색됐던 상품의 구매가 완료된다.



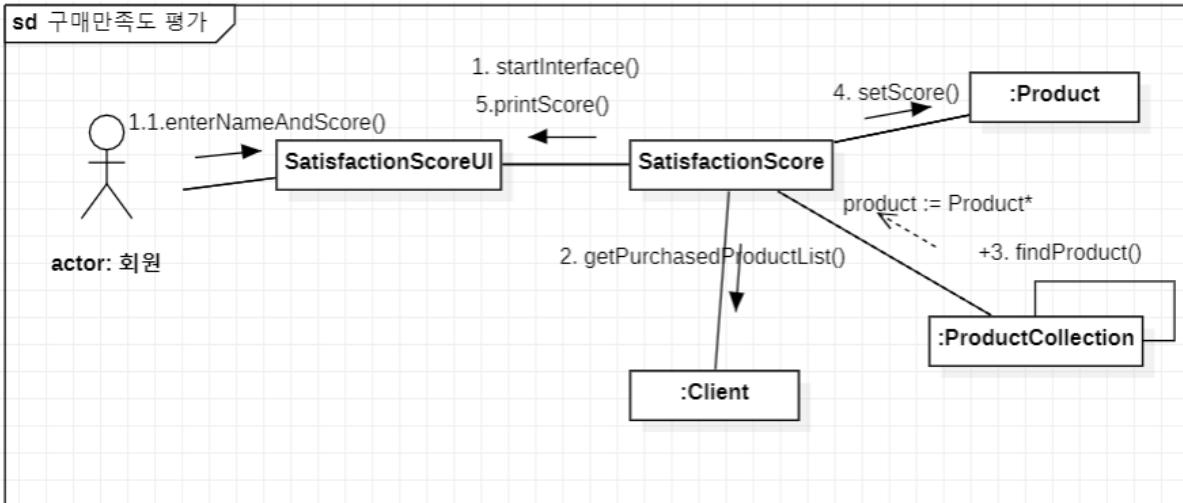
Use case 11: 상품 구매 내역 조회

Actor Action	System Response
1. 상품 구매 내역 조회(4 3) 메뉴를 선택한다.	2. “4.3. 상품 구매 내역 조회”와 함께 회원이 구입한 모든 상품의 정보를 【판매자ID】 【상품명】 【제작회사명】 【가격】 【남은수량】 【평균 구매만족도】 순으로 출력한다.

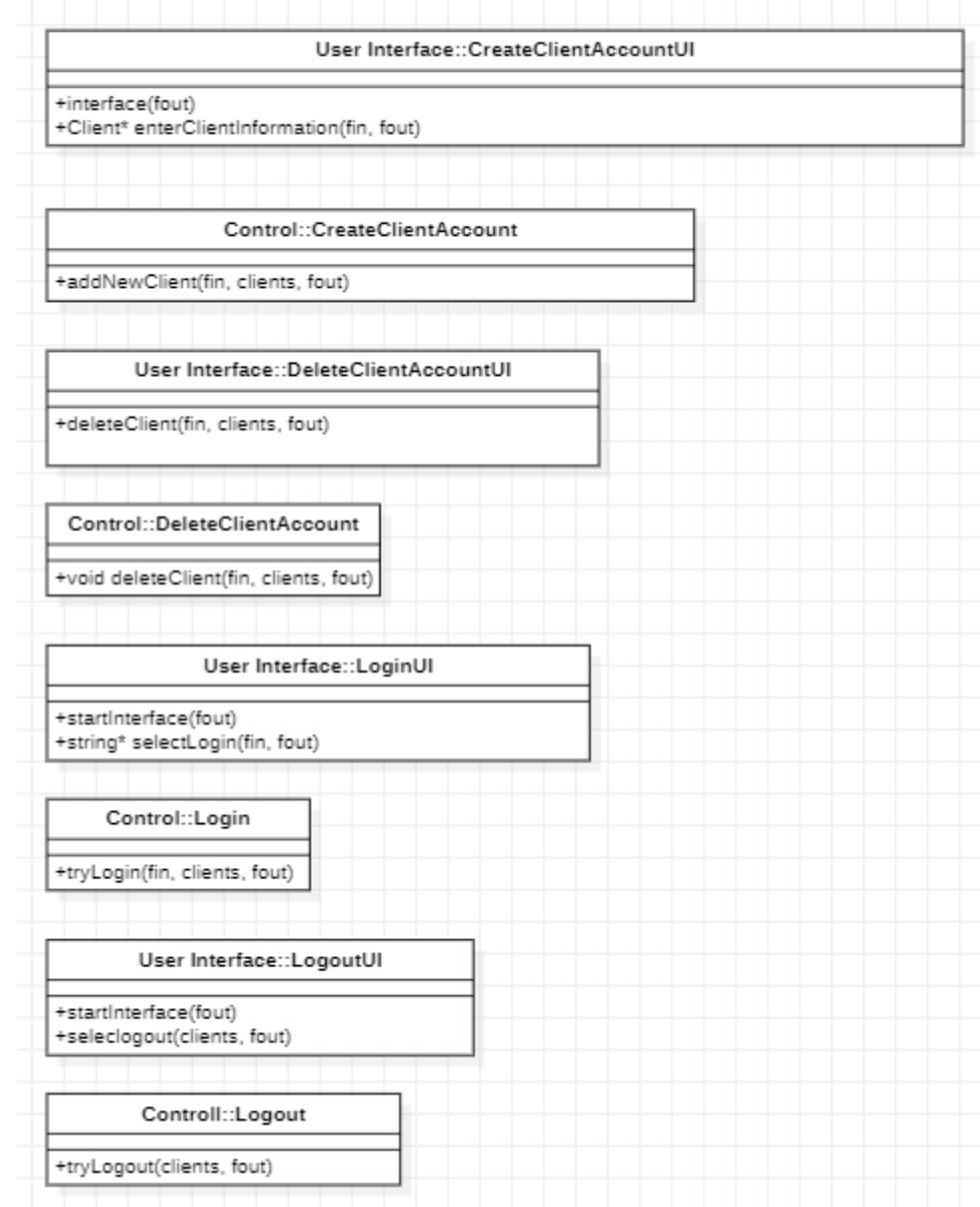


Use case 12: 구매 만족도 평가

Actor Action	System Response
1. 상품 구매 만족도 평가(4 4) 메뉴를 선택한다.	2. “4.2. 상품 구매 만족도 평가”를 출력한다.
3. 상품명과 구매만족도를 입력한다.	4. 평가가 완료된 상품의 【판매자ID】 【상품명】 【구매만족도】를 출력한다.



5.Design class diagram



User Interface::SearchUI

```
+startInterface(fin, fout)
+enterProductName(search, products, fout)
+Product* selectPurchase()
```

Control::Search

```
+Product* run(fin, products, selected, fout)
+Product* searchProduct(productName, products)
```

User Interface::PurchaseUI

```
+startInterface(product, actor, fout, purchase)
+purchaseProduct()
```

Control::Purchase

```
+Client* run(product, actor, fout)
+purchaseProduct()
```

User Interface::ShowShoppingList

```
+startInterface(list, fout)
```

Control::ShowShopping

```
+run(actor, fout)
```

User Interface::SatisfactionScoreUI

```
+startInterface(fin, productName, score, fout)
+enterNameAndScore(fin, productName, score)
+printScore(sellerID, productName, score, fout)
```

Control::SatisfactionScore

```
+run(fin, actor, fout)
```

User Interface::AddProductUI

```
+startInterface(fin, productName: String, companyName: String, cost: Int, quantityLeft: Int, fout)  
+printNewProduct(fout, product: Product)  
+enterInformation(fin, productName: String, companyName: String, cost: Int, quantityLeft: Int)
```

Control::AddProduct

```
+run(fin, actor: Client, totalProduct: ProductCollection, fout)
```

User Interface::ProductForSaleUI

```
+startInterface(fout)  
+showProductForSales(products: ProductCollection, fout)
```

Control::ProductForSale

```
+run(actor: Client, fout)
```

User Interface::ListSoldProductUI

```
+startInterface(fout)  
+showSoldProductList(products ProductCollection, fout)
```

Control::ListSoldProduct

```
+run(actor: Client, fout)
```

User Interface::ShowStatisticsUI

```
+startInterface(fout)  
+showStatistics(list: ProductCollection, fout)
```

Control::ShowStatistics

```
+run(actor: Client, fout)
```

