

객체지향 프로그래밍 과제 3 - 영화 예매 프로그램
<요구사항 리스트>

번호	요구사항	세부적 사항
1.	로그인	이미 회원 가입이 되어있는 사용자 계정으로 로그인한다. 로그인이 되어있지 않은 상태에서 다른 기능은 수행할 수 없다.
2.	영화 전체 조회 및 예매	로그인이 되었을 경우에만 실행 가능하다. 현재 상영하는 전체 영화 정보를 띄우고, 번호 선택을 통해 영화를 예매한다. 예매된 영화는 예매내역에 추가된다.
3.	영화 검색 및 예매	로그인이 되었을 경우에만 실행 가능하다. 직접 영화 제목을 검색, 해당되는 영화가 있으면 번호 선택을 통해 영화를 예매한다. 예매된 영화는 예매내역에 추가된다.
4.	예매된 영화 조회	로그인이 되었을 경우에만 실행 가능하다. 사용자가 예매한 영화 내역을 출력한다.
5.	종료하기	프로그램 종료 메시지를 출력한 후 프로그램을 종료한다.

객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Step-by-step description >

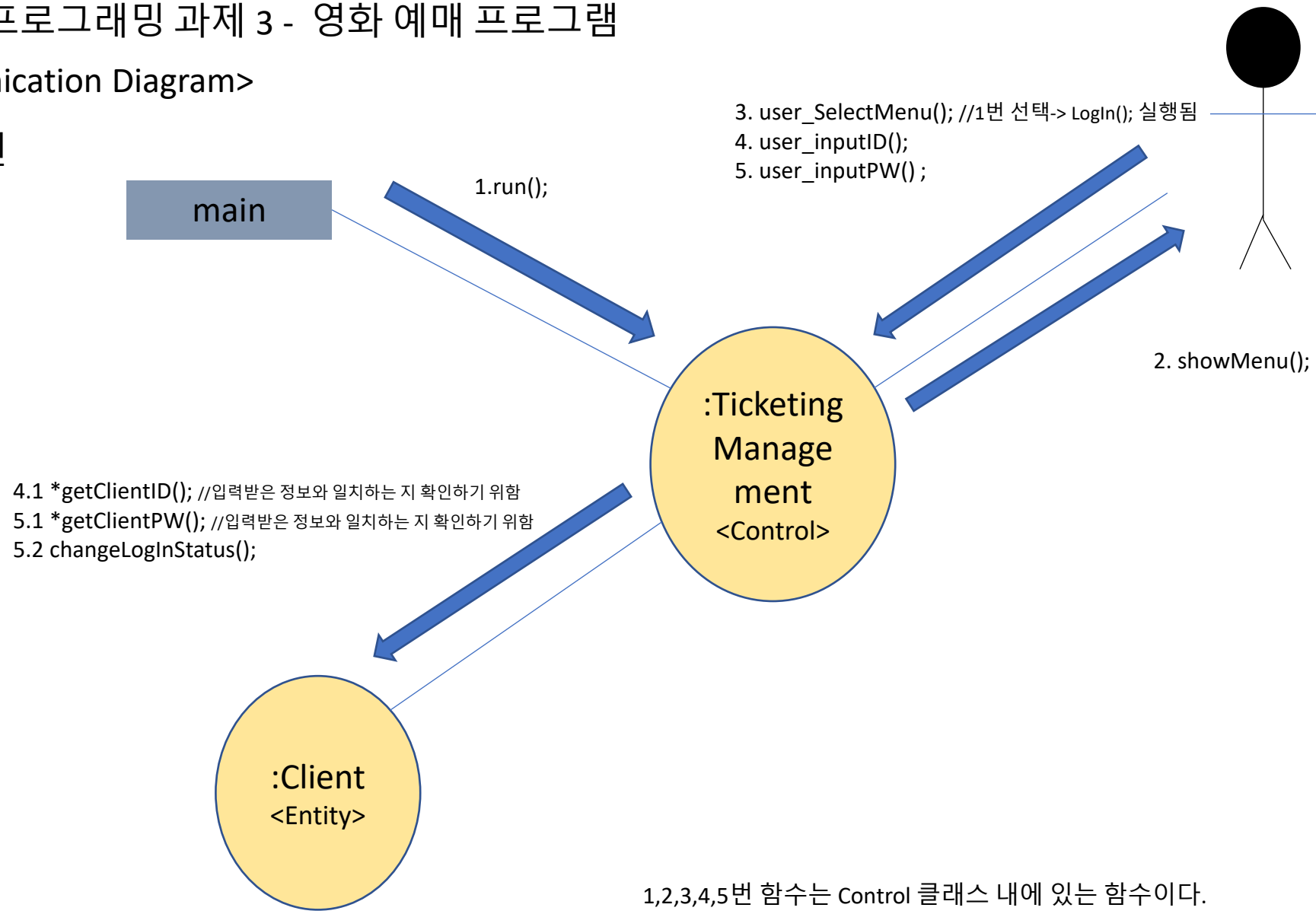
1. 로그인

번호	User Action	System Response
1.		프로그램을 실행한다.
2.		메뉴 선택 필드를 console 창에 출력한다.
3.	Console창에 1번 입력한다.	
4.		로그인 기능 수행 시작한다.
5.		프로그램 상 이미 로그인이 되어있는 지 확인한 후, 로그인이 되어있지 않다면 ID, 패스워드 입력 필드를 console창에 출력한다.
		<Alternatives> 프로그램 상 이미 로그인이 되어있는 경우, 로그인이 이미 되어있다는 메시지를 console창에 출력한다.
6.	ID, 패스워드를 console창에 입력한다.	
7.		입력 받은 ID,패스워드 정보가 적합한 경우 해당 계정을 로그인 승인하고, 로그인 승인 메시지를 console창에 출력한다.
		<Alternatives> 6단계에서 입력 받은 로그인 정보가 적합하지 않은 경우, 로그인 실패 메시지를 console창에 출력한다. 그리고 다시 ID, 패스워드를 입력 받는다.
8.		로그인이 되었으므로, 사용자가 또 로그인 시도 시 로그인이 이미 되었다는 메시지를 console창에 출력한다.

객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Communication Diagram>

1. 로그인



객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Step-by-step description >

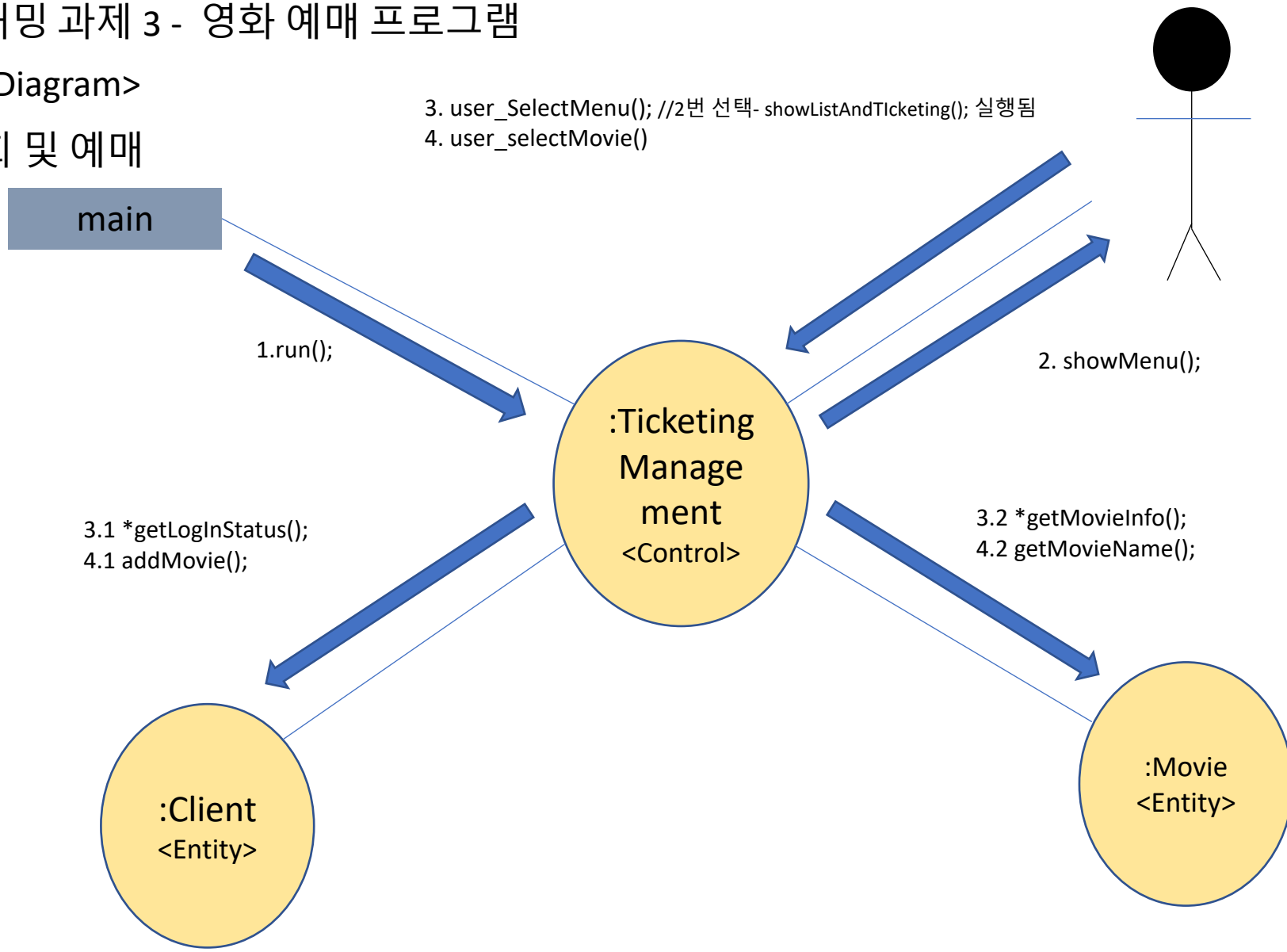
2. 영화 전체 조회 및 예매

번호	User Action	System Response
1.		프로그램을 실행한다.
2.		메뉴 선택 필드를 console 창에 출력한다.
3.	Console창에 2번 입력한다.	
4.		영화 전체 조회 및 예매 기능 수행 시작한다.
5.		프로그램 상 이미 로그인이 되어있는 지 확인한 후, 로그인이 되어있지 않다면, 실행 불가능한 영역임을 console창에 출력한다.
		<Alternatives> 로그인이 되어 있다면, 해당 계정이 어떤 계정인 지 확인한 후 영화 전체 리스트를 console창에 출력한다.
6.	Console 창에 예매할 영화번호를 입력한다.	
7.		입력 받은 영화 번호가 적합한 경우, 선택된 영화의 정보를 해당 계정의 영화 예매 내역에 추가하고, 영화이름을 포함한 예매완료 메시지를 console창에 출력한다.
		<Alternatives> 6단계에서 입력 받은 영화번호가 적합하지 않은 경우, 다시 영화 번호를 입력하라는 메시지를 console창에 출력한다.

객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Communication Diagram>

2. 영화 전체 조회 및 예매



1,2,3,4번 함수는 Control 클래스 내에 있는 함수이다.

객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Step-by-step description >

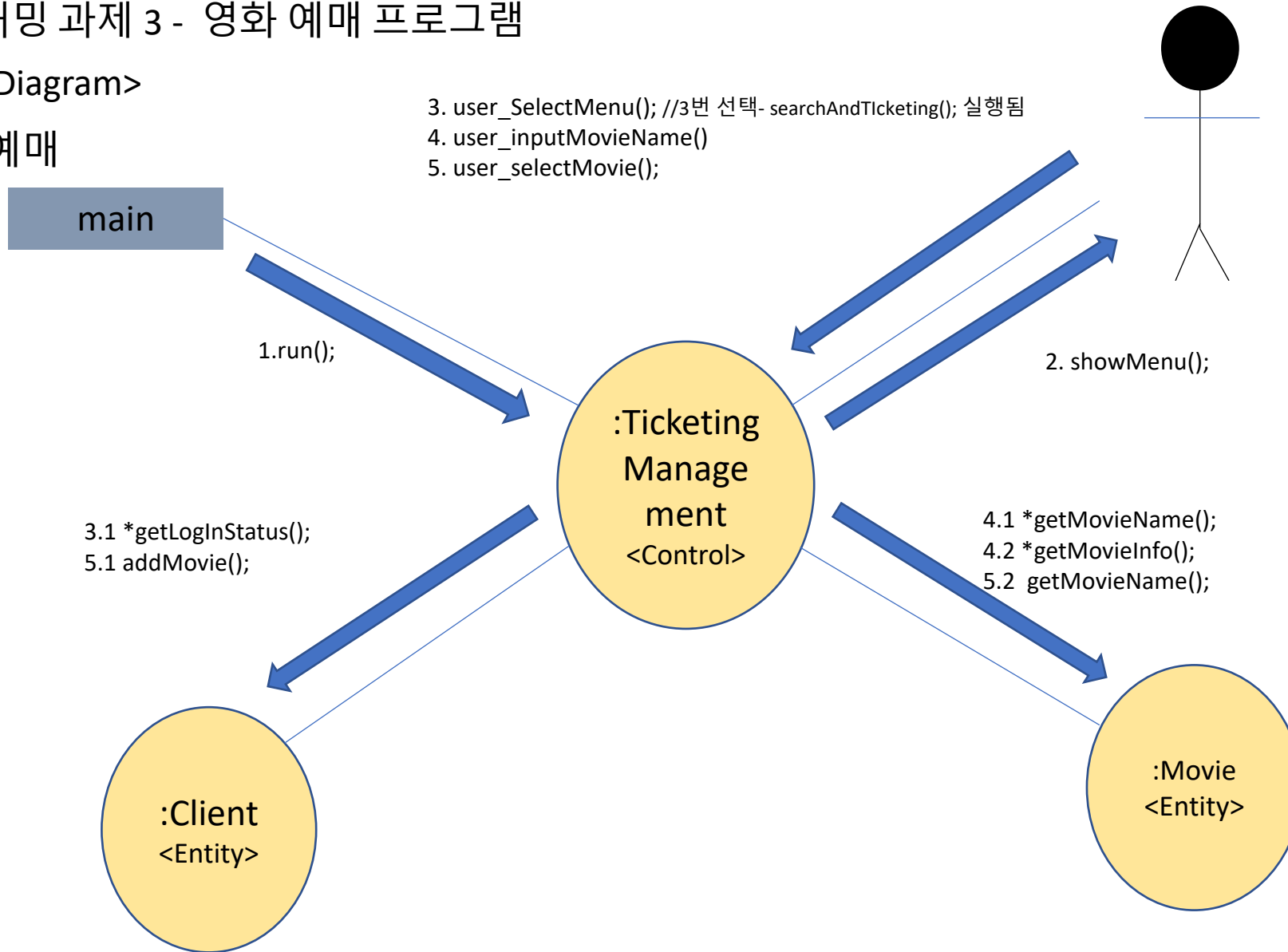
3. 영화 검색 및 예매

번호	User Action	System Response
1.		프로그램을 실행한다.
2.		메뉴 선택 필드를 console 창에 출력한다.
3.	Console창에 3번 입력한다.	
4.		영화 검색 및 예매 기능 수행 시작한다.
5.		프로그램 상 이미 로그인이 되어있는 지 확인한 후, 로그인이 되어있지 않다면, 실행 불가능한 영역임을 console창에 출력한다.
		<Alternatives> 로그인이 되어 있다면, 해당 계정이 어떤 계정인지 확인한 후, 영화 제목을 입력 받을 수 있는 필드를 출력한다.
6.	Console 창에 예매할 영화 제목을 입력한다.	
7.		입력 받은 영화 제목이 현재 상영 중인 영화인지 확인한 후 만약 일치하는 영화가 있다면 해당 영화 정보를 console창에 출력한다.
		<Alternatives> 6단계에서 입력 받은 영화제목과 일치하는 영화가 없는 경우, 해당 영화가 없다는 메시지를 console창에 출력한다.
8.	검색 결과로 나온 영화들 중 예매할 영화의 번호를 입력한다.	
9.		입력 받은 영화 번호가 적합한 경우, 선택된 영화의 정보를 해당 계정의 영화 예매 내역에 추가하고, 영화이름을 포함한 예매완료 메시지를 console창에 출력한다.
		<Alternatives> 8단계에서 입력 받은 영화번호가 적합하지 않은 경우, 다시 영화 번호를 입력하라는 메시지를 console창에 출력한다.

객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Communication Diagram>

3. 영화 검색 및 예매



1,2,3,4,5번 함수는 Control 클래스 내에 있는 함수이다.

객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Step-by-step description >

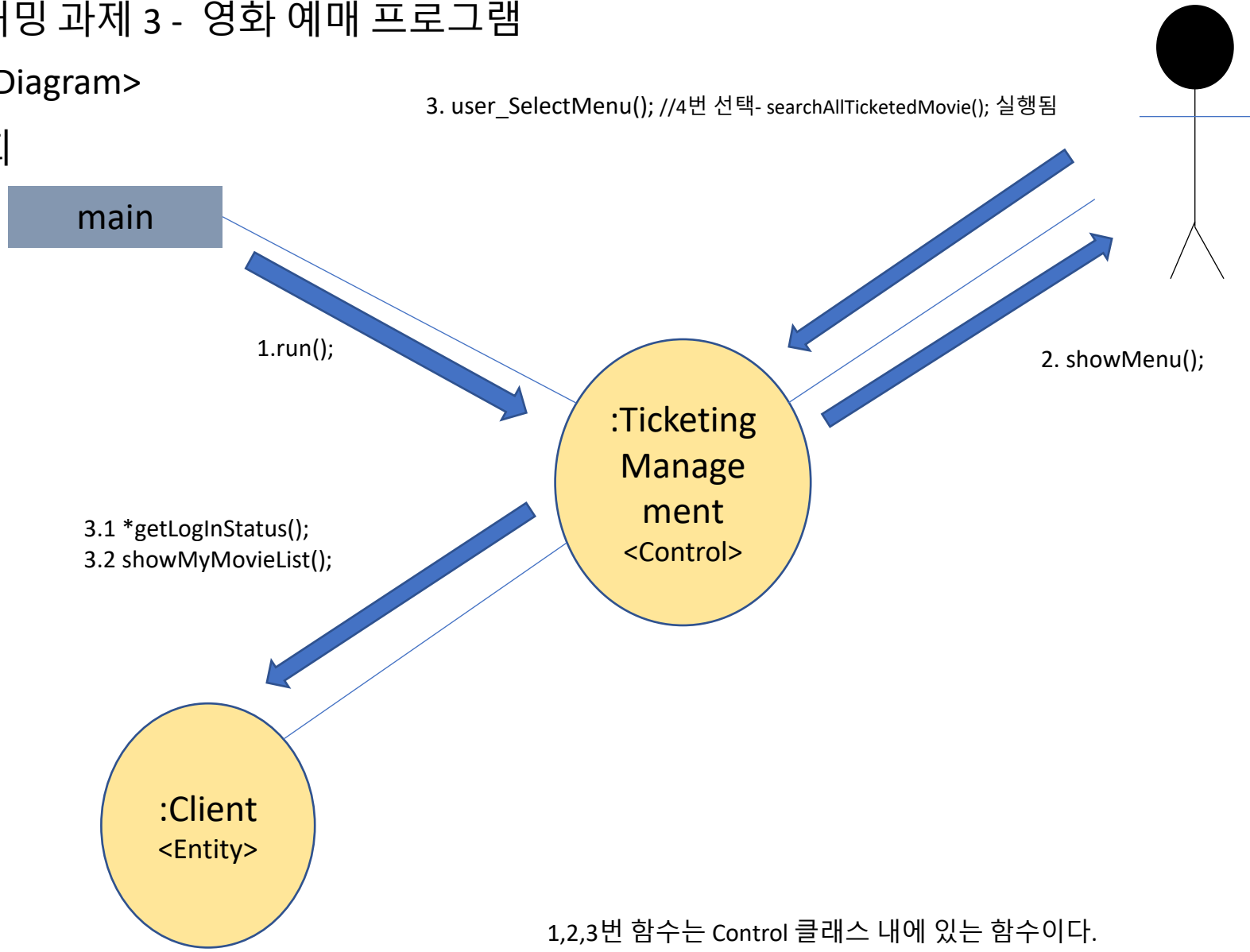
4. 예매 영화 조회

번호	User Action	System Response
1.		프로그램을 실행한다.
2.		메뉴 선택 필드를 console 창에 출력한다.
3.	Console창에 4번 입력한다.	
4.		예매 영화 조회 기능 수행 시작한다.
5.		프로그램 상 이미 로그인이 되어있는 지 확인한 후, 로그인 되어있지 않다면, 실행 불가능한 영역임을 console창에 출력한다.
		<Alternatives> 로그인이 되어 있다면, 해당 계정이 어떤 계정인지 확인한 후 해당 계정의 예매 내역에 들어있는 영화 정보들을 console창에 출력한다.

객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Communication Diagram>

4. 예매 영화 조회



객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Step-by-step description >

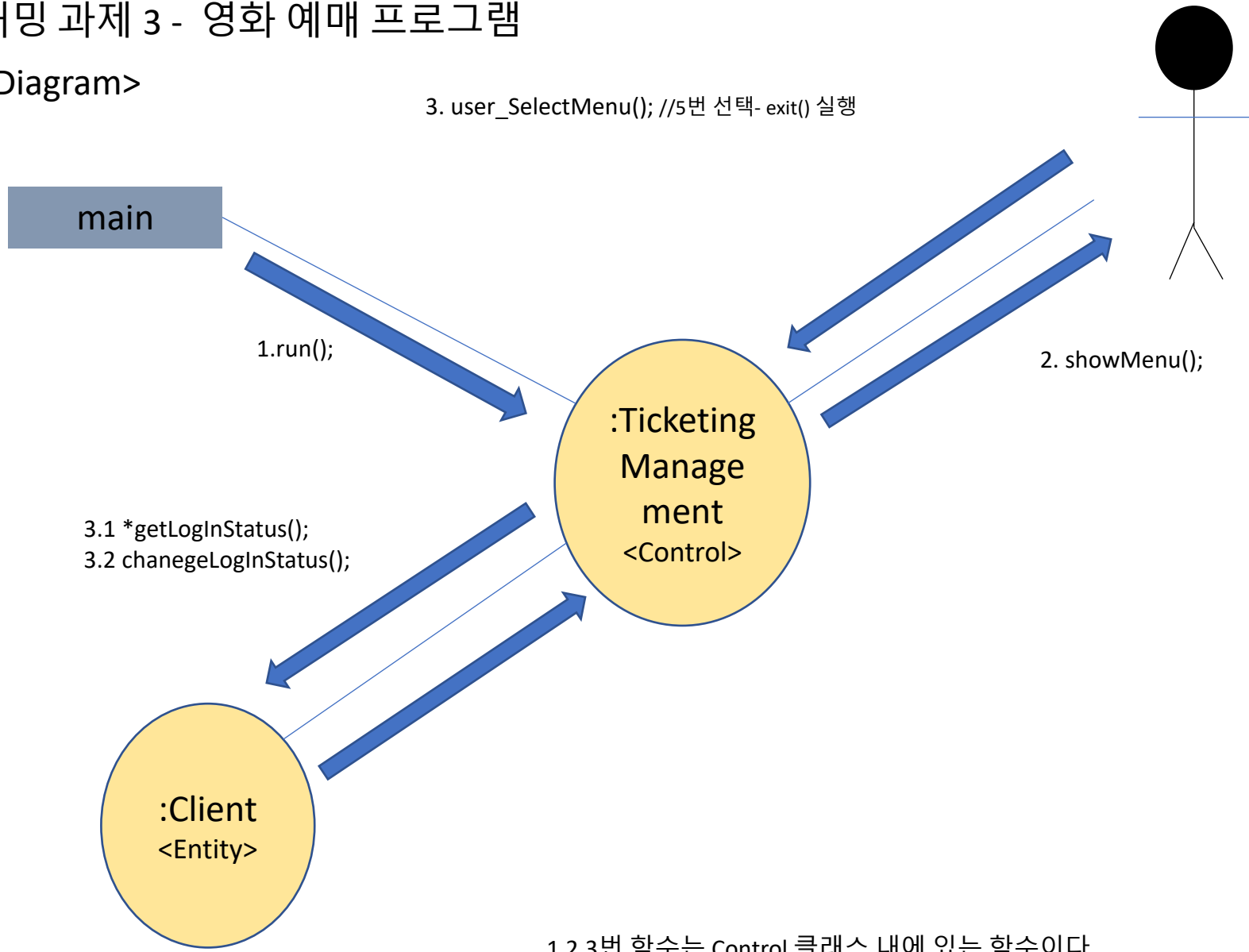
5. 종료하기

번호	User Action	System Response
1.		프로그램을 실행한다.
2.		메뉴 선택 필드를 console 창에 출력한다.
3.	Console창에 5번 입력한다.	
4.		로그인이 되어 있는 계정이 어떤 계정인 지 확인한 후 로그인 상태를 로그아웃 상태로 바꿔준다. 종료 메시지를 console창에 출력한 후 프로그램을 종료한다

객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Communication Diagram>

5. 종료하기



객체지향 프로그래밍 과제 3 - 영화 예매 프로그램

<Class 정의> - *각 클래스의 생성자 및 소멸자는 생략하였음*

class Client

{

public:

char* getClientID();

char* getClientPW();

bool getLoginStatus();

void changeLoginStatus();

void addMovie();

void showMyMovieList();

};

class Movie

{

public:

char* getMovieName();

char* getMovieInfo();

};

Class TicketingManagement

{

public:

void run();

};