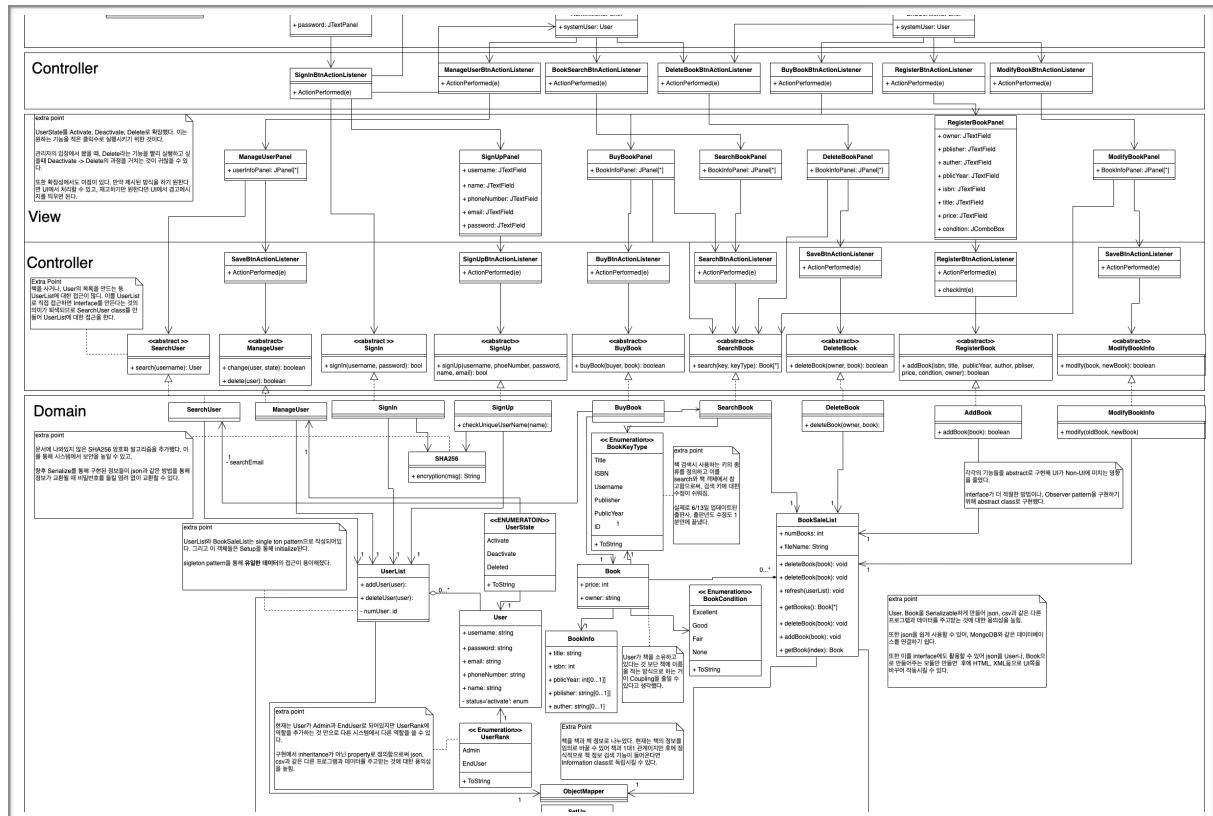
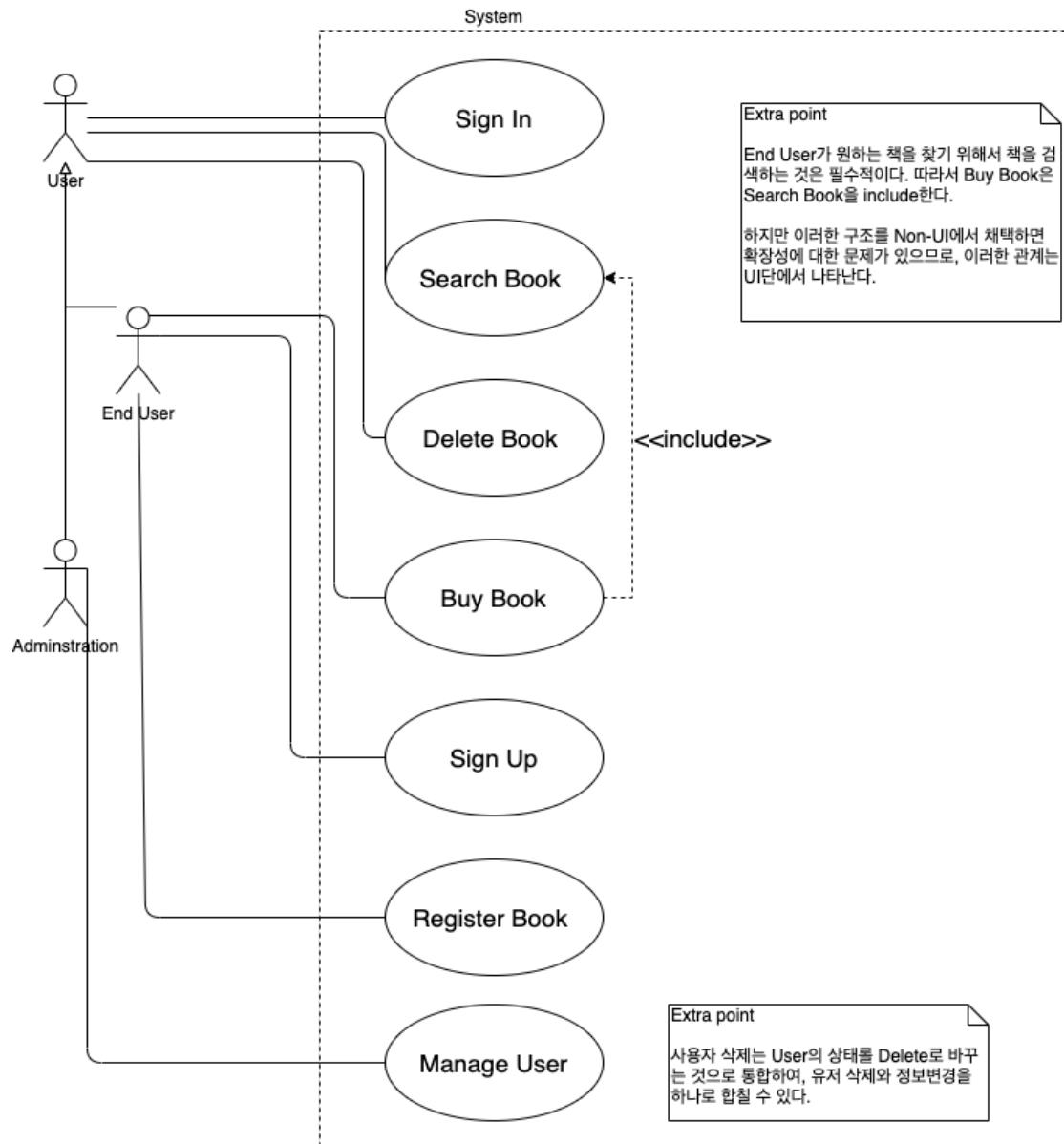


## 헌책 장터 시스템



20182592 장원범

# Use-case Diagram



## End-user Use-case 명세서

### End-user Use case UC1: Buy Book

#### Main Success Scenario

1. 사용자가 책 검색화면으로 이동한다.
2. 사용자가 제목, ISBN 번호, 저자, 판매자 id 중 하나를 입력한다.
3. 검색에 해당하는 책의 제목, 출판사, 저자 정보, 출판년도, 가격과 상태가 출력된다.
4. 구매하고자 하는 책을 선택한다.
5. 구매하고자 하는 책의 주인에게 메일을 보낸다.

사용자는 구매하고 싶은 책을 모두 구매할때까지 2~4를 반복한다.

6. 사용자가 구매를 끝낸다.

#### Extensions

\*a. 시스템이 실패할 경우

1. 시스템을 재시작 한다.

2a. 키워드를 입력 안한 채 검색을 하는 경우

1. 키워드를 입력하지 않았다는 메시지를 출력한다.
2. 책 검색화면으로 넘어간다.

3a. 검색한 키워드에 해당하는 책이 없는 경우

1. 사용자에게 책이 없다는 메시지를 출력한다.
2. 책을 검색하는 화면으로 돌아간다.

4a. 구매하고자 하는 책의 주인의 상태가 deactivate일 경우

1. 해당 사용자는 비활성화중이라는 메시지를 출력한다.
2. 해당 사용자를 제외한 책 목록을 다시 출력한다.

### End-user Use case UC2: Register Book

#### Main Success Scenario

1. 사용자가 책 등록 화면으로 이동한다.
2. 책의 제목, 출판사, 저자 정보, 출판년도, 가격 및 상태를 입력한다.
3. 책이 시스템에 등록된다.

사용자의 모든 책이 등록될때까지 2~3을 반복한다.

4. 사용자가 책 등록을 끝낸다.

#### Extensions

\*a. 시스템이 실패할 경우

1. 시스템을 재시작 한다.

2. 책의 제목이 입력이 안되어있을 경우
  1. 책 제목이 입력이 안되어있다는 메시지를 출력한다.
  2. 책 등록 화면으로 돌아간다.

## Administration Use-case 명세서

### End-user Use case UC1:Delete User

#### Main Success Scenario

1. 사용자 관리 화면으로 이동한다.
2. 모든 사용자의 목록이 출력된다.
3. 사용자를 삭제한다.
4. 변경된 사용자 목록이 출력된다.

원하는 모든 사용자가 삭제될때 까지 2~4번을 반복한다.

5. 사용자 삭제를 끝낸다.

#### Extensions

\*a. 시스템이 실패할 경우

1. 시스템을 재시작 한다.

3a. 사용자의 상태가 activate인 경우

1. 사용자의 상태가 activate라는 메시지를 출력한다.
2. 사용자 2로 돌아간다.

### End-user Use case UC2: Delete Book

#### Main Success Scenario

1. 책 관리 화면으로 이동한다.
2. 접근할 수 있는 모든 책의 목록을 출력한다.
3. 책의 정보를 입력한 후 삭제 버튼을 누른다
4. 변경된 책의 목록을 출력한다.

원하는 모든 책이 삭제될때 까지 3~4를 반복한다.

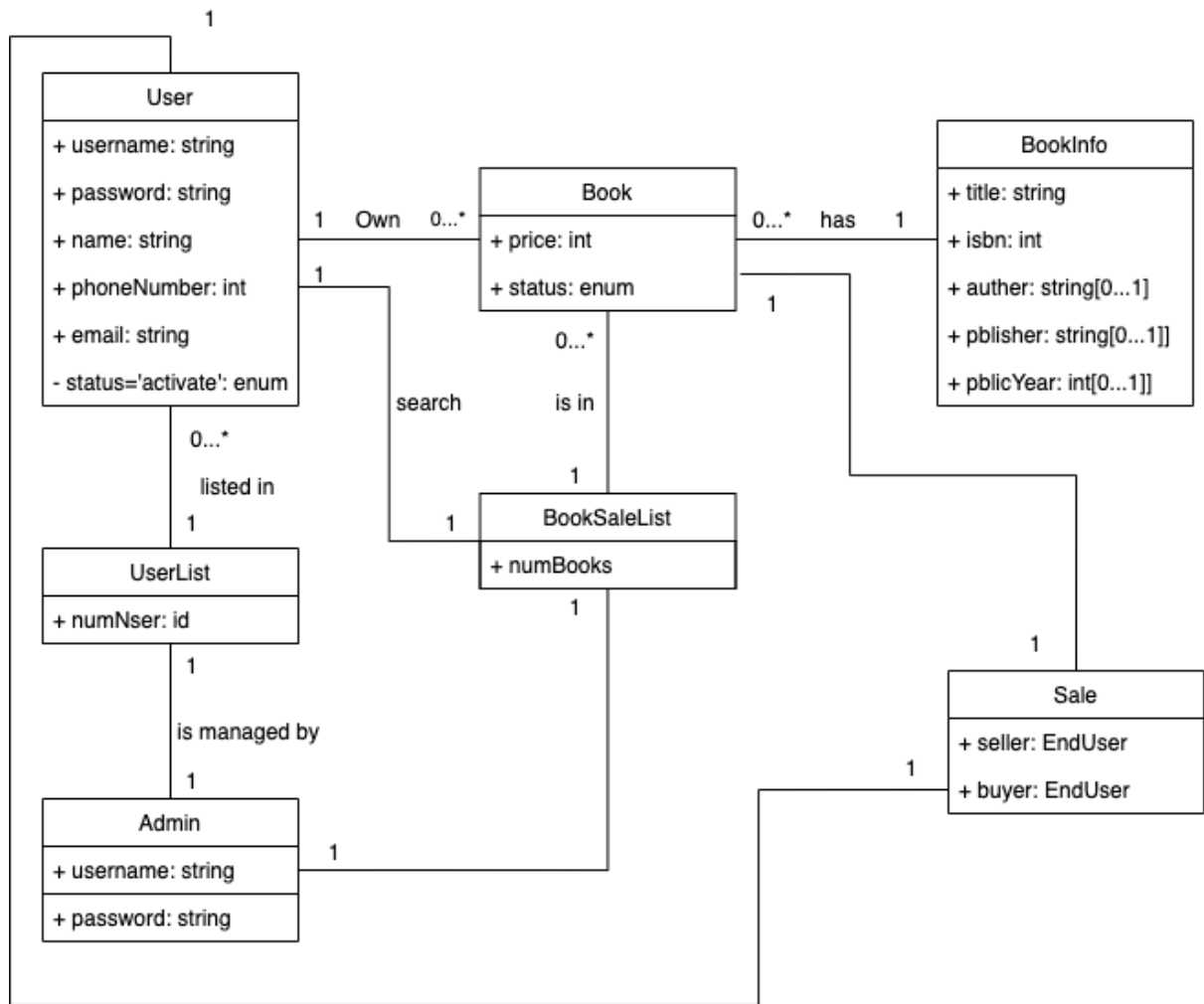
5. 책 삭제를 종료한다.

#### Extensions

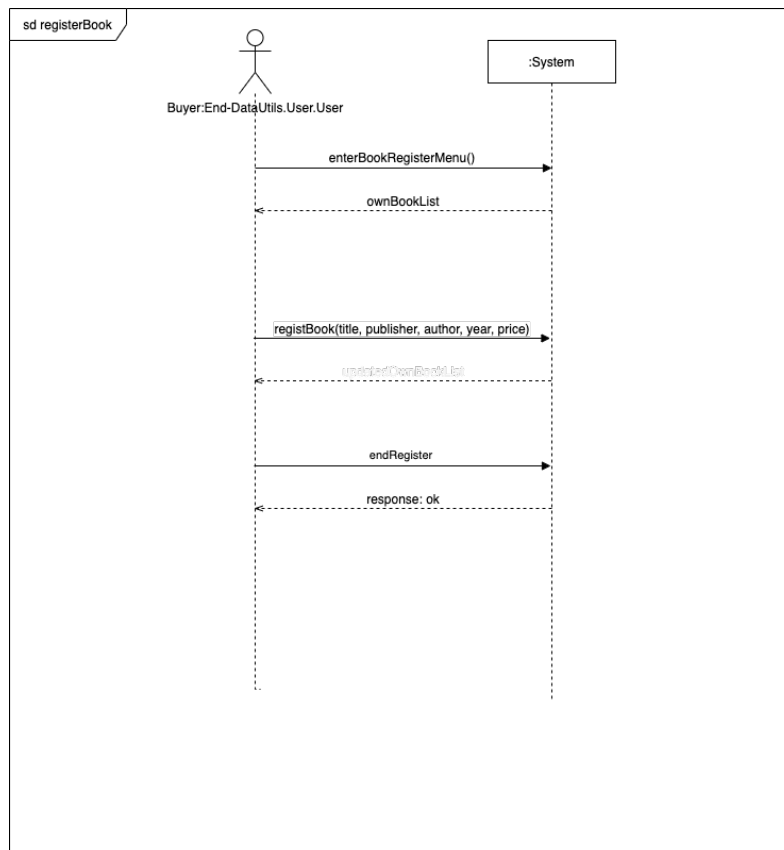
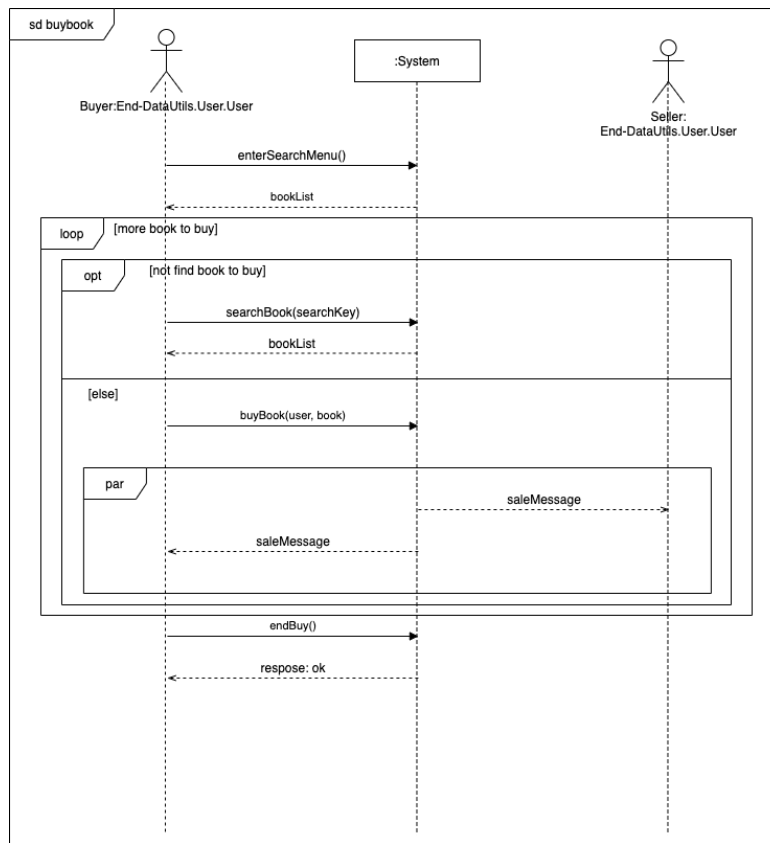
\*a. 시스템이 실패할 경우

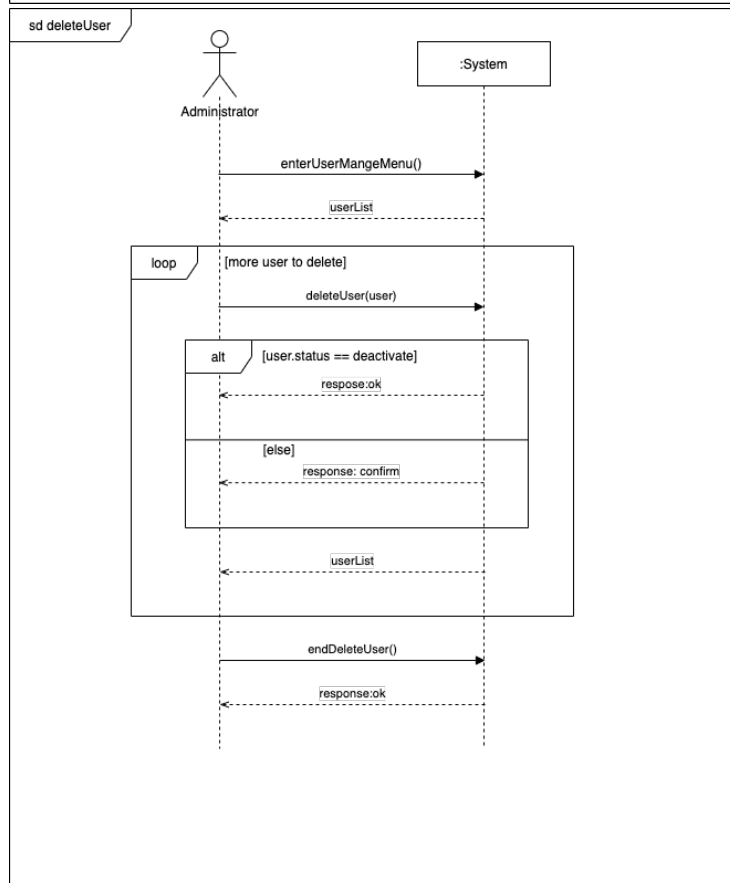
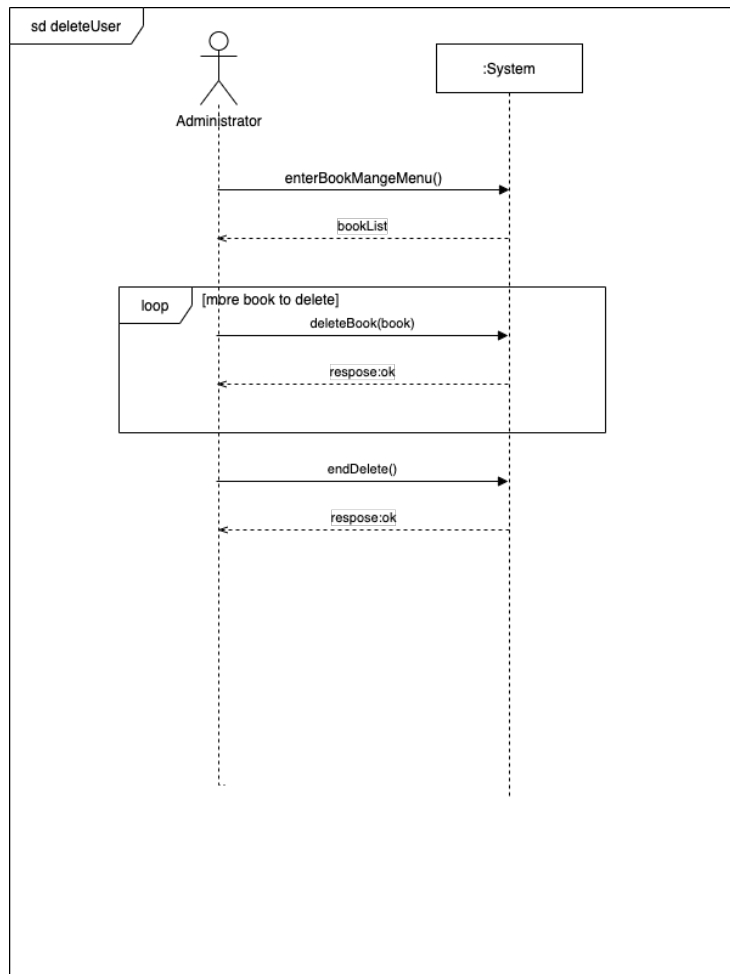
1. 시스템을 재시작 한다

# Domain Model



# System Sequence Diagram





## Operation Contract

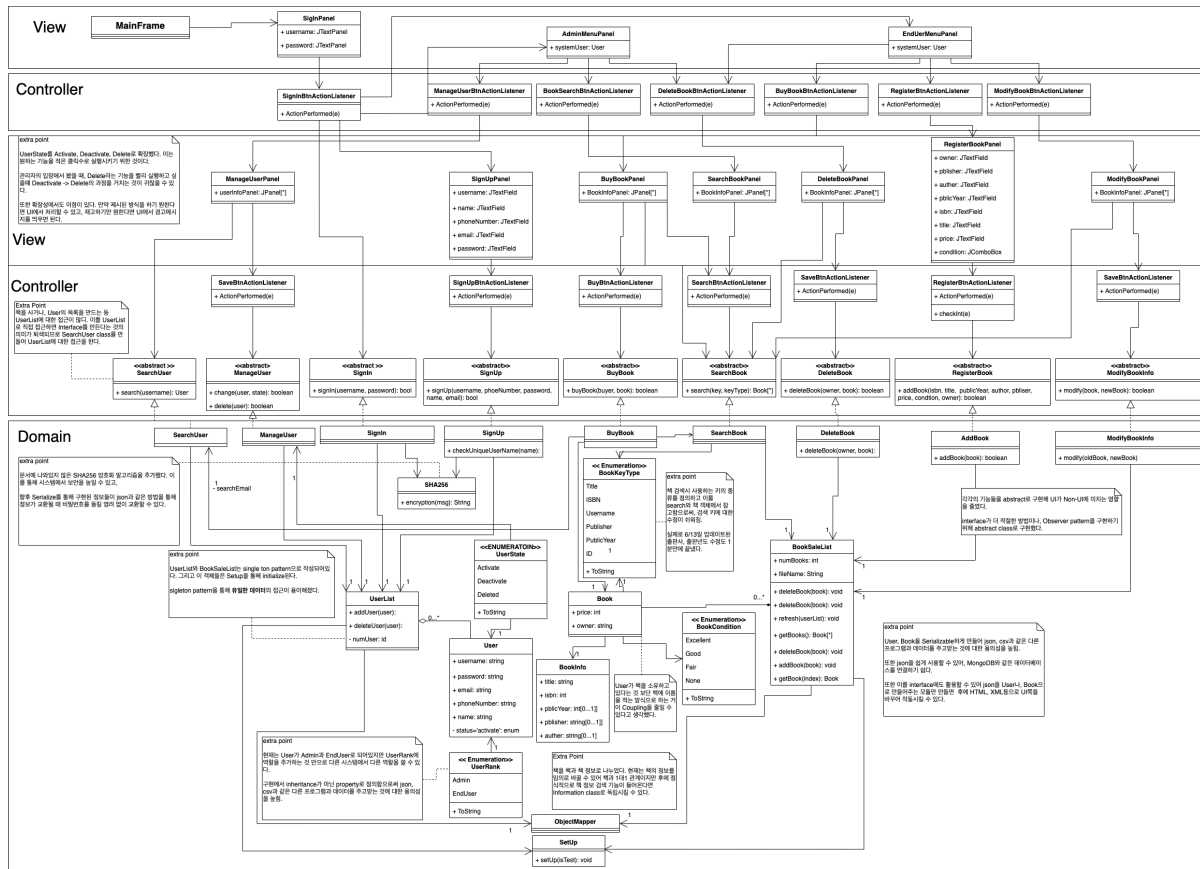
Operation	registerBook(title, publisher, author, year, price)
Cross References	End-user Use case UC2: Register Book
Preconditions	BookRegisterMenu에 있어야함.
Postconditions	<ul style="list-style-type: none"> <li>- BookOnSale객체가 생성되고 받은 parameter로 초기화가 됨.</li> <li>- BookOnSale객체가 BookSaleList에 연결됨.</li> <li>- 만약에 책 정보가 없다면 BookInfo객체가 생성되고 주어진 parameter로 초기화 되며, BookList에 연결됨.</li> </ul>

Operation	buyBook()
Cross References	End-user Use case UC2: Register Book
Preconditions	Sale객체가 생성되오 있어야함.
Postconditions	<ul style="list-style-type: none"> <li>- Sale객체에 사는 사람, 파는 사람, 거래되는 책에 정보가 저장됨.</li> <li>- Sale에 저장된 정보를 바탕으로 BookSaleList의 정보의 수정이 일어남</li> </ul>



## Class Diagram



# Sequence Diagram

