Software Engineering Project

**시스템 설계 명세서(System Design Specification)**

**장애인을 위한 호텔 예약 플랫폼 BFT ver 1.2  
(Best Friends’ Tour / Barrier Free Tour)**

12팀

201900724 김민정  
201800802 김소연  
201701924 신세인  
201702838 이지은  
201702803 이준혁

목차

**1. 개요**

1.1 이 보고서(시스템 설계 명세서)의 목적

1.2 보고서 구성 및 내용

1.3 보고서 버전 및 관리 내용

1.4 설계에 사용된 툴과 설명

1.5 참고

**2. Architectural Design**

2.1 System Architecture

2.2 Data-flow Model

**3. Entire Model**

3.1 Actors

3.2 Use-Cases

3.3 Package

**4. 모듈(객체 모델) 명세서**

4.1 UML 다이어그램

4.1.1 Use-Case Diagram

4.1.2 Class Diagram

4.1.3 Sequence Diagram

**5. 입출력 설계**

5.1 회원가입

5.2 상품 설계

**6. Data 설계**

6.1 ERD (E-R Diagram)

6.2 DB Table Description

6.3 Method

6.4 Sequence Diagram

**7. Project Planning**

7.1 Tasks

7.2 CPM Network

7.3 Task Allocation

**8. User Interface**

8.1 비로그인 사용자

8.2 로그인 된 회원

8.3 호스트

**9. 프로그램 작성 지침**

**\*첨부**

**1. 개요**

1.1 이 보고서(시스템 설계 명세서)의 목적

여행업이 크게 발전하고 있음에도 불구하고, 장애인의 여행에 대한 지원과 관심은 아직까지 높지 않다. 일부 지자체와 공공기관에서 이러한 정보를 제공하고 있지만, 이들은 정보만 제공할 뿐, 그러한 시설과의 연결을 해주지는 않는다. 그렇기 때문에, 이번 프로젝트 BFT(Best Friends’ Travel)은 장애인들이 보다 편리하게 여행할 수 있도록, 그들을 지원하는 숙박업소와 그들을 중개해주는 역할을 할 것이다.

이번 시스템 설계 명세서는 이번 프로젝트를 구현하기 위한 기술적인 점들에 대해서 정리하고, 구체적인 작업을 정의하고, 분배할 것이다. 또한, 그에따라 일정을 확정할 것이다. 마지막으로 시스템을 설계하고 구현함에 있어서 신경 써서 보아야 할 점에 대해서 정리할 것이다.

1.2 보고서 구성 및 내용

본 설계 명세서는 프로그램을 구현하기 위한 기술적 설계도를 완성하는 것을 목표로 한다. 아키텍쳐 설계를 통해서 본 프로젝트 기술적 개요를 살펴본다. 다음으로, 요구명세서에서 본 유스케이스 명세를 다시금 살펴볼 예정이다. 요구명세서 작성 이후, 변경된 내용을 포함해서 액터와 유스케이스 간의 상관관계를 다시금 살펴볼 것이다.

전체 모델을 정리한 후, 각 모델 별로 다이어그램을 통해 객체 모델의 상세사양을 기술하고, 코드와 입출력 설계 내용에 대해서 설명할 것이다. 코드설계를 살펴보는 과정에서 기능 키에 대한 정의도 할 예정이다. 이에 더해, 데이터 관리에 대한 상세 설명도 진행할 것이다.

마지막으로 전체 프로젝트를 완성하기 위한 작업을 정의하고, 작업의 순서 계획 및 개인별 작업 분담을 통해 원활한 진행을 도모한다.

1.3 보고서 버전 및 관리 내용

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **버전** |  | **일자** | **주요 내용** | **비고** |
| v1.0 |  | 2021.11.22 | 초판 작성 | 2021.11.22 회의를 통한 피드백 |
| v1.1 |  | 2021.11.25 | 1차 피드백 반영 수정  코드 설계 추가 | 2021.11.25  회의를 통한 피드백 |
| v1.2 |  | 2021.11.27 | 2차 피드백 반영 수정  프로그램 설계 지침 추가 |  |

1.4 설계에 사용된 툴과 설명

|  |  |  |
| --- | --- | --- |
| **도구명** | **용도** | **비고** |
| Figma | UI 디자인 |  |
| StarUML | 각종 다이어그램 작성 |  |
| Github | 작성 코드 버전 관리 |  |
| Google Drive | 문서 통합 관리 |  |

1.5 참고

* https://docs.djangoproject.com/ko/3.2/glossary/

**2. Architectural Design**

2.1 system architecture

architecture style: 본 프로젝트는 MTV (Model-Template-View) 아키텍쳐 설계 패턴을 따른다. MTV 아키텍쳐 설계 패턴은 소프트웨어의 패턴으로서 MVC(Model-View-Controller) 방식과 유사하지만, 그보다 장고의 방식을 더 잘 설명하는 아키텍쳐 설계 패턴이다.

Model: DataBase와 상호작용 기능

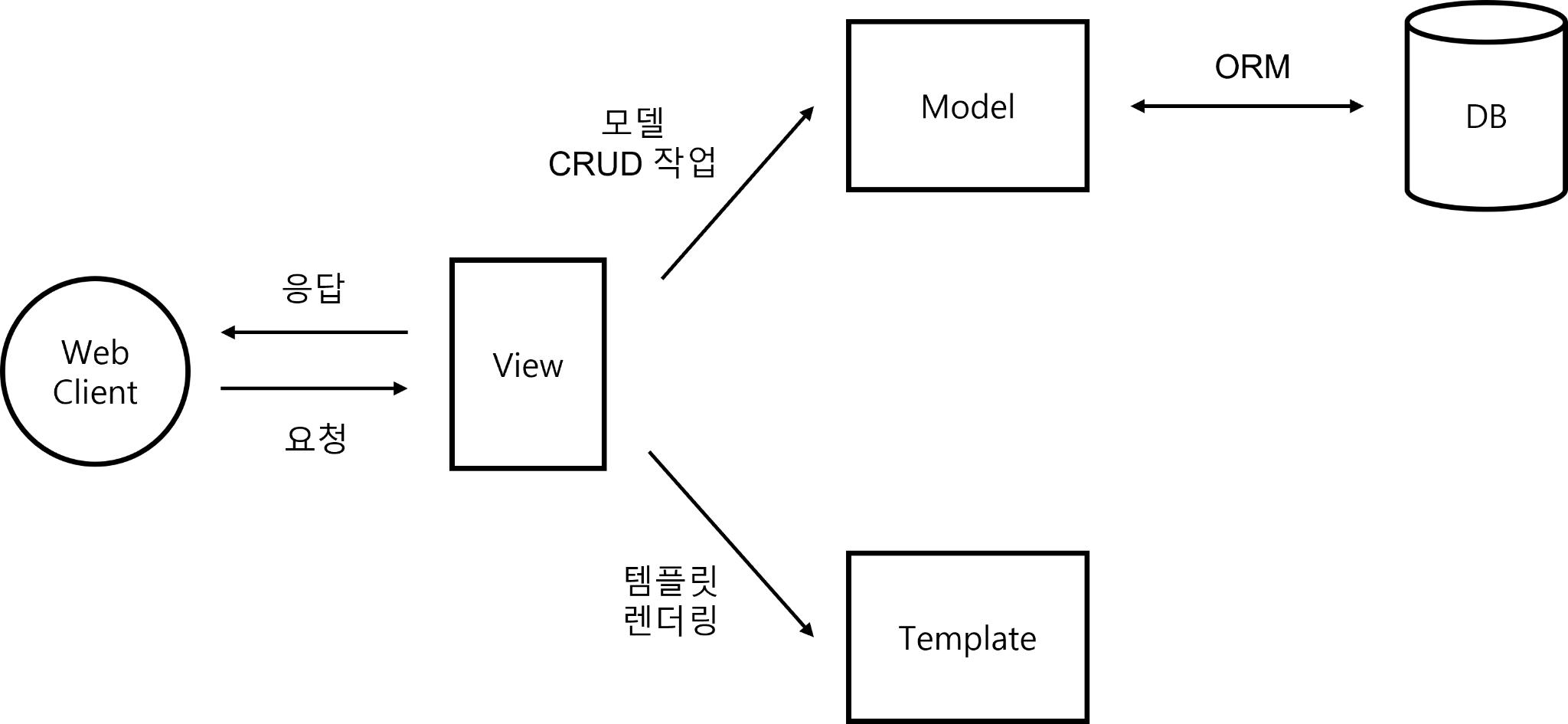
* User (회원관리 테이블)
* Review (리뷰관리 테이블)
* Hotel (상품관리-호텔 테이블)
* Room (상품관리-객실 테이블)
* Reservation (상품 예약관리 테이블)
* Pay (상품 결제 테이블)

Template: 사용자 인터페이스 기능

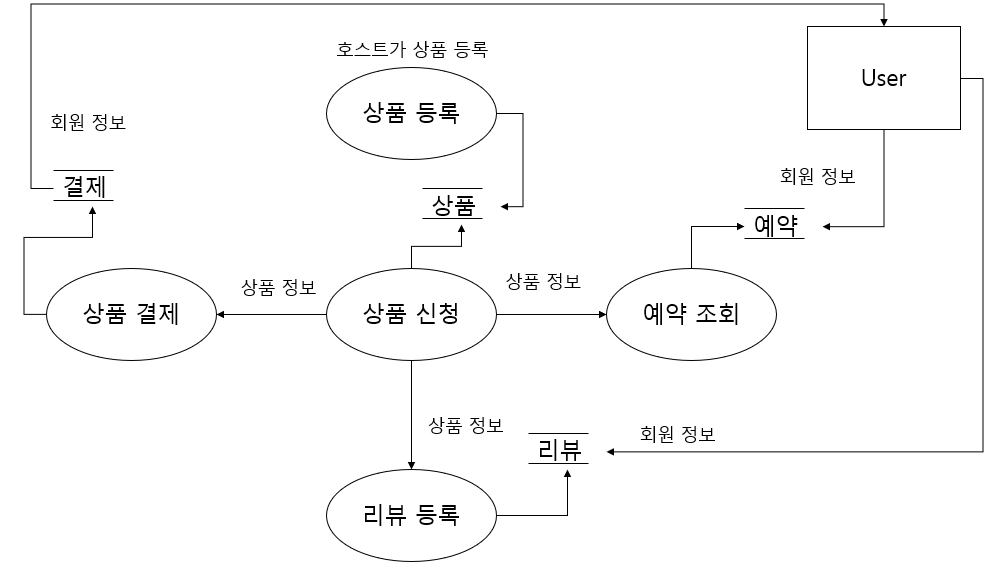
* 8장 User Interface 에서 설명.

View: 웹서비스 내부 동작의 논리 기능

* User 관련
* Register(): 회원가입
* Login(): 로그인
* Logout(): 로그아웃
* UserInfo(): 개인정보 조회
* UpdateUser(): 개인정보 수정
* Withdrawal(): 회원 탈퇴
* Hotels 관련
  + MakeHotel(): 호텔상품 등록
  + UpdateHotel(): 호텔상품 수정
  + DeleteHotel(): 호텔상품 삭제
  + SearchHotel(): 호텔상품 검색
  + ViewHotel(): 호텔상품 조회
* Rooms 관련
  + MakeRoom(): 객실상품 등록
  + UpdateRoom(): 객실상품 수정
  + DeleteRoom(): 객실상품 삭제
  + SearchRoom(): 객실상품 검색
* Reservations 관련
  + MakeReservation(): 상품 예약
  + UpdateReservation(): 예약 수정
  + DeleteReservation(): 예약 삭제
  + ViewReservation(): 예약 조회
* Reviews관련
  + WriteReview(): 리뷰 작성
  + UpdateReview(): 리뷰 수정
  + DeleteReview(): 리뷰 삭제
  + GetReview(): 리뷰 조회
* Payments
  + ConfirmPayment(): 결제
  + ViewPayment(): 결제 조회



2.2 Data-flow Model



3.

**3. Entire Model**

3.1 Actors

1. 고객: 서비스를 이용하는 사람으로서 상품 검색, 상품 예약을 할 수 있다.
2. 호스트: 서비스를 제공하는 사람으로서 상품 등록을 할 수 있다.

3.2 Use-Cases

3.2.1 회원관리 관련 기능요구

가. 회원 등록: 고객 또는 호스트의 개인정보를 입력 받아 저장한다. 이로써 고객은 장바구니 기능을, 호스트는 상품 등록 및 관리 기능을 이용할 수 있게 한다.

|  |  |
| --- | --- |
| 유스케이스 이름 | 회원 등록 |
| 액터 | 고객, 호스트 |
| 시작 조건 | 고객 또는 호스트가 시스템에 접속해야 한다. |
| 기본 흐름 | 1. 고객 또는 호스트가 ‘회원 가입’을 누른다.  2. 시스템이 이용약관을 표시한다.  3. 고객 또는 호스트가 약관에 동의하면 동의를 누르고 다음으로 넘어간다.  4. 시스템이 회원 가입 양식을 표시한다.  5. 고객 또는 호스트가 양식에 맞게 정보를 입력한다.  (회원 아이디, 비밀번호, 이름, 이메일, 연락처, 고객 및 호스트 구분)  6. 시스템이 입력된 내용을 확인하는 양식을 표시한다.  7. 고객 또는 호스트가 내용을 확인하고 ‘확인’ 버튼을 누른다.  8. 고객 또는 호스트의 정보를 시스템에 저장한다. |
| 대안 흐름 | 3A  1. 고객 또는 호스트가 약관의 동의하지 않으면 동의하지 않음을 누른다  2. 시스템이 오류 메시지를 표시한다.  3. 고객 또는 호스트가 메시지를 확인하고 유스케이스를 종료한다.  5A  1. 고객 또는 호스트가 내용을 확인하고 ‘수정’ 버튼을 누른다.  2. 기본 흐름 3으로 돌아간다. |
| 종료 조건 | 고객 또는 호스트의 정보가 시스템에 저장된다. |

나. 로그인: 사용자 정보를 식별하여 고객 또는 호스트에게 개인화된 정보를 제공한다.

|  |  |
| --- | --- |
| 유스케이스 이름 | 로그인 |
| 액터 | 고객, 호스트 |
| 시작 조건 | 고객 또는 호스트가 회원등록이 되어있어야 한다. |
| 기본 흐름 | 1. 고객 또는 호스트가 ‘로그인’을 누른다.  2. 시스템이 로그인 양식을 표시한다.  3. 고객 또는 호스트가 양식에 맞게 아이디와 비밀번호를 입력한다.  4. 시스템이 입력된 내용을 유효성 확인하고 결과를 표시한다.  5. 정보가 유효하면, 시스템이 로그인한 사용자 이름을 표시한다. |
| 대안 흐름 | 5A  1. 정보가 유효하지 않으면, 시스템이 오류 메시지를 표시한다.  2. 기본흐름 2로 돌아간다. |
| 종료 조건 | 고객 또는 호스트의 사용자 이름이 표시된다. |

다. 회원 정보 조회: 고객 또는 호스트가 자신의 회원 정보를 조회할 수 있도록 한다.

|  |  |
| --- | --- |
| 유스케이스 이름 | 개인정보 조회 |
| 액터 | 고객, 호스트 |
| 시작 조건 | 고객 또는 호스트가 로그인이 되어있어야 한다. |
| 기본 흐름 | 1. 고객 또는 호스트가 ‘회원 정보 조회’를 누른다.  2. 시스템이 회원 정보를 표시한다. |
| 종료 조건 | 고객 또는 호스트의 회원 정보를 표시한다 |

라. 회원 정보 수정: 고객 또는 호스트가 필요 및 요구에 맞게 정보를 수정할 수 있게 한다.

|  |  |
| --- | --- |
| 유스케이스 이름 | 개인정보 수정 |
| 액터 | 고객, 호스트 |
| 시작 조건 | 고객 또는 호스트가 로그인이 되어있어야 한다. |
| 기본 흐름 | 1. 고객 또는 호스트가 ‘회원 정보 수정’을 누른다.  2. 시스템이 회원 정보 수정 양식을 표시한다.  3. 고객 또는 호스트가 수정을 원하는 정보를 양식에 맞게 입력한다.  4. 시스템은 고객이 작성한 정보를 확인하는 양식을 표시한다.  5. 고객 또는 호스트가 내용을 확인하고 ‘확인’을 누른다.  6. 변경된 정보를 시스템에 저장한다. |
| 대안 흐름 | 5A  1. 고객 또는 호스트가 내용을 확인하고 ‘수정’ 버튼을 누른다.  2. 기본 흐름 3으로 돌아간다. |
| 종료 조건 | 변경된 개인정보가 시스템에 저장된다. |

마. 회원 탈퇴: 고객 또는 호스트가 더 이상 서비스가 필요하지 않을 때, 서비스 탈퇴를 하게 한다

|  |  |
| --- | --- |
| 유스케이스 이름 | 회원 탈퇴 |
| 액터 | 고객, 호스트 |
| 시작 조건 | 고객 또는 호스트가 로그인이 되어있어야 한다. |
| 기본 흐름 | 1. 고객 또는 호스트가 ‘회원 탈퇴’을 누른다.  2. 시스템이 회원 탈퇴 여부를 묻는 양식을 표시한다.  3. 고객 또는 호스트가 양식을 확인하고 ‘회원 탈퇴’를 누른다.  4. 시스템이 탈퇴 완료를 표시한다. |
| 종료 조건 | 고객 또는 호스트가 서비스에서 탈퇴한다. |

3.2.2. 상품 등록 관련 기능요구

가. 상품 등록: 호스트가 본인이 판매하고 싶은 상품에 대한 정보를 등록하는 도구를 제공한다. 호스트는 본 기능을 통해 상품의 위치, 장소, 가격, 특징, 지원가능한 장애의 종류, 사진 등을 등록하여 충분한 정보를 소비자에게 제공할 수 있도록 한다.

|  |  |
| --- | --- |
| 유스케이스 이름 | 상품 등록 |
| 액터 | 호스트 |
| 시작 조건 | 호스트로 등록된 사용자이고 로그인 되어야 한다. |
| 기본 흐름 | 1. 호스트가 ‘상품 등록’을 선택한다.  2. 시스템이 상품 등록 양식을 표시한다.  3. 호스트가 양식에 맞게 상품에 대한 정보를 작성한다.  (상품명, 소유자, 상품 정보, 가격 정보)  4. 시스템이 작성된 내용을 확인하는 양식을 표시한다.  5. 호스트가 내용을 확인하고 ‘확인’ 버튼을 누른다.  6. 상품 정보를 시스템에 저장한다. |
| 대안 흐름 | 5A  1. 호스트가 내용을 확인하고 ‘수정’ 버튼을 누른다.  2. 기본 흐름 3으로 돌아간다. |
| 종료 조건 | 호스트가 입력한 상품 정보가 시스템에 저장된다. |

나. 상품 관리: 호스트가 본인이 판매중인 상품에 변경이 있거나 추가하고 싶은 내용이 있을 경우, 이를 지원하는 기능을 한다.

|  |  |
| --- | --- |
| 유스케이스 이름 | 상품 관리 |
| 액터 | 호스트 |
| 시작 조건 | 호스트가 로그인 되어야 하고 상품이 등록되어 있어야 한다. |
| 기본 흐름 | 1. 호스트가 ‘상품 관리’를 선택한다.  2. 시스템은 호스트가 등록한 상품을 표시한다.  3. 관리를 원하는 상품을 선택한다.  4. 시스템은 선택된 상품의 등록 정보와 수정가능한 서식 이에 관한 예약 정보를 표시한다.  5. 호스트가 정보를 수정한 후, ‘확인’을 누른다.  6. 호스트가 수정한 정보가 시스템에 저장된다. |
| 종료 조건 | 호스트가 수정한 상품 정보가 시스템에 저장된다. |

3.2.3. 상품 검색 관련 기능요구

가. 상품 검색: 고객이 원하는 날짜와 장소를 검색하면 조건에 맞는 상품을 검색하여 그에 맞는 결과를 표시해준다. 또한 고객은 원하는 상세 조건(지원 가능한 장애 요건, 특정 시설의 유무 등)을 원하는 결과를 추출할 수 있다.

|  |  |
| --- | --- |
| 유스케이스 이름 | 상품 검색 |
| 액터 | 고객 |
| 시작 조건 | 고객이 예약 페이지에 접속해야 한다. |
| 기본 흐름 | 1. 고객이 검색창에 희망 날짜와 지역, 지원하는 장애 유형을 입력 후, ‘검색’버튼을 누른다.  2. 시스템은 고객이 검색한 내용에 맞는 결과를 표시한다.  3. 고객은 원하는 상품을 선택하여 상세 페이지로 넘어간다.  4. 시스템은 고객이 선택한 상품의 상세 정보를 표시한다. |
| 대안 흐름 | 2A  1. 고객의 검색에 일치하는 결과가 없으면, ‘결과 없음’ 표시한다.  2. 고객이 결과 없음을 인식하고 유스케이스를 마친다. |
| 종료 조건 | 고객이 선택한 상품의 정보가 표시된다. |

3.2.4. 상품 예약 관련 기능요구

가. 상품 예약: 고객이 상품을 결정하면 예약에 필요한 정보를 입력 받아서 예약과 결제를 진행한다. 이때 요구사항 등을 입력하여 호스트 측에 전달될 수 있도록 한다. 또한 예약이 완료되었음을 고객과 호스트에게 통보한다.

|  |  |
| --- | --- |
| 유스케이스 이름 | 상품 예약 |
| 액터 | 고객 |
| 시작 조건 | 고객이 예약을 원하는 상품을 선택해야 한다. |
| 기본 흐름 | 1. 고객이 선택한 상품의 상세 정보를 확인하고 ‘예약’ 버튼을 누른다.  2. 시스템은 예약 정보를 입력하는 서식을 표시한다.  3. 고객은 예약의 필요한 정보를 서식에 입력하고 ‘확인’을 누른다.  (개인정보, 요구사항)  4. 시스템은 고객에게 입력한 정보를 확인시키는 서식을 표시한다.  5. 고객은 표시된 정보를 확인하고 ‘결제’ 버튼을 누른다.  6. 시스템은 결제정보를 입력하는 창을 표시한다.  7. 결제정보를 선택하고 ‘다음’ 버튼을 누른다.  8. 시스템은 결제정보의 유효성을 판단하고 결과를 표시한다.  9. 유효성에 문제가 없다면 결제를 완료하고 예약 내용을 시스템에 저장한 후, 고객과 호스트에게 예약 내용을 통보한다.  10. 시스템은 결제 화면을 표시한다. |
| 대안 흐름 | 1A  1. 고객이 예약 의사가 없으면, 페이지를 나가고 유스케이스를 마친다.  5A  1. 고객이 내용을 확인하고 ‘수정’ 버튼을 누른다.  2. 기본 흐름 3으로 돌아간다.  9A  1. 유효성에 문제가 있다면 오류 메시지를 띄우고 기본 흐름 3으로 돌아간다. |
| 종료 조건 | 예약에 대한 결제 화면이 표시된다. |

3.2.5. 마이페이지 관련 기능 요구

가. 예약 조회: 고객이 자신이 한 예약 정보를 확인할 수 있도록 지원한다. 또한 호스트가 자신의 상품에 대한 예약을 확인하여 이를 대비할 수 있게 한다

|  |  |
| --- | --- |
| 유스케이스 이름 | 예약 조회 |
| 액터 | 고객 |
| 시작 조건 | 고객이 로그인 되어있거나 예약번호가 있어야 한다. |
| 기본 흐름 | 1. 고객이 ‘예약 조회’ 버튼을 누른다.  2. 시스템이 고객의 로그인 정보나 예약번호를 입력하는 양식을 표시한다.  3. 고객이 로그인 정보나 예약번호를 양식에 맞게 입력한다.  4. 시스템이 위의 정보에 맞는 예약 목록을 표시한다.  5. 고객이 확인을 원하는 예약을 선택한다.  6. 시스템이 고객이 선택한 예약을 표시한다. |
| 종료 조건 | 고객의 예약 정보를 표시한다. |

|  |  |
| --- | --- |
| 유스케이스 이름 | 예약 조회 |
| 액터 | 호스트 |
| 시작 조건 | 호스트가 로그인 되어있거나 예약번호가 있어야 한다. |
| 기본 흐름 | 1. 호스트가 ‘예약 조회’ 버튼을 누른다.  2. 시스템이 호스트의 로그인 정보를 입력하는 양식을 표시한다.  3. 호스트가 로그인 정보를 양식에 맞게 입력한다.  4. 시스템이 위의 정보에 맞는 상품 목록을 표시한다.  5. 호스트가 확인을 원하는 상품을 선택한다.  6. 시스템이 고객이 선택한 상품의 예약을 표시한다. |
| 종료 조건 | 호스트의 상품에 대한 예약 정보를 표시한다. |

나. 리뷰 작성: 고객이 사용한 상품에 대한 리뷰를 작성하여 사용자 경험을 공유한다. 이때, 허위 리뷰를 방지하기 위해 예약정보를 확인한다.

|  |  |
| --- | --- |
| 유스케이스 이름 | 리뷰 작성 |
| 액터 | 고객 |
| 시작 조건 | 고객이 해당 숙소에 대한 예약 기록이 있어야 한다. |
| 기본 흐름 | 1. 고객이 예약을 조회한다.  2. 시스템이 고객의 예약 정보를 표시한다.  3. 고객이 원하는 예약을 표시한다.  4. 시스템이 위의 정보에 맞는 예약을 표시한다.  5. 고객이 ‘리뷰 작성’ 칸에 리뷰를 작성한다.  6. 시스템이 고객이 작성한 리뷰를 시스템에 저장하고 상품 상세 페이지에 표시한다. |
| 종료 조건 | 고객이 작성한 리뷰가 시스템에 저장되고 상품 상세 페이지에 표시된다. |

3.3 Package

1. Users
   1. 정의: 사용자와 관련된 클래스 파일들이 있는 패키지
   2. 기능: 사용자와 관련된 데이터베이스 모델 정의, 논리 동작 등을 기능
      1. 회원가입
      2. 로그인
      3. 로그아웃
      4. 개인정보 조회
      5. 개인정보 수정
      6. 회원탈퇴
2. Hotels
   1. 정의: 호텔상품과 관련된 클래스 파일들이 있는 패키지
   2. 기능: 호텔상품과 관련된 데이터베이스 모델 정의, 논리 동작 등을 기능
      1. 호텔 상품 등록
      2. 호텔 상품 수정
      3. 호텔 상품 삭제
      4. 호텔 상품 검색
3. Rooms
   1. 정의: 객실상품과 관련된 클래스 파일들이 있는 패키지
   2. 기능: 객실상품과 관련된 데이터베이스 모델 정의, 논리 동작 등을 기능
      1. 객실 상품 등록
      2. 객실 상품 수정
      3. 객실 상품 삭제
      4. 객실 상품 검색
4. Reservations
   1. 정의: 예약과 관련된 클래스 파일들이 있는 패키지
   2. 기능: 예약과 관련된 데이터베이스 모델 정의, 논리 동작 등을 기능
      1. 상품 예약
      2. 예약 조회
      3. 예약 수정
      4. 예약 삭제
5. Reviews
   1. 정의: 리뷰와 관련된 클래스 파일들이 있는 패키지
   2. 기능: 리뷰와 관련된 데이터베이스 모델 정의, 논리 동작 등을 기능
      1. 리뷰 조회
      2. 리뷰 작성
      3. 리뷰 수정
      4. 리뷰 삭제
6. Payments
   1. 정의: 결제와 관련된 클래스 파일들이 있는 패키지
   2. 기능: 결제와 관련된 데이터베이스 모델 정의, 논리 동작 등을 기능
      1. 결제
      2. 결제 조회

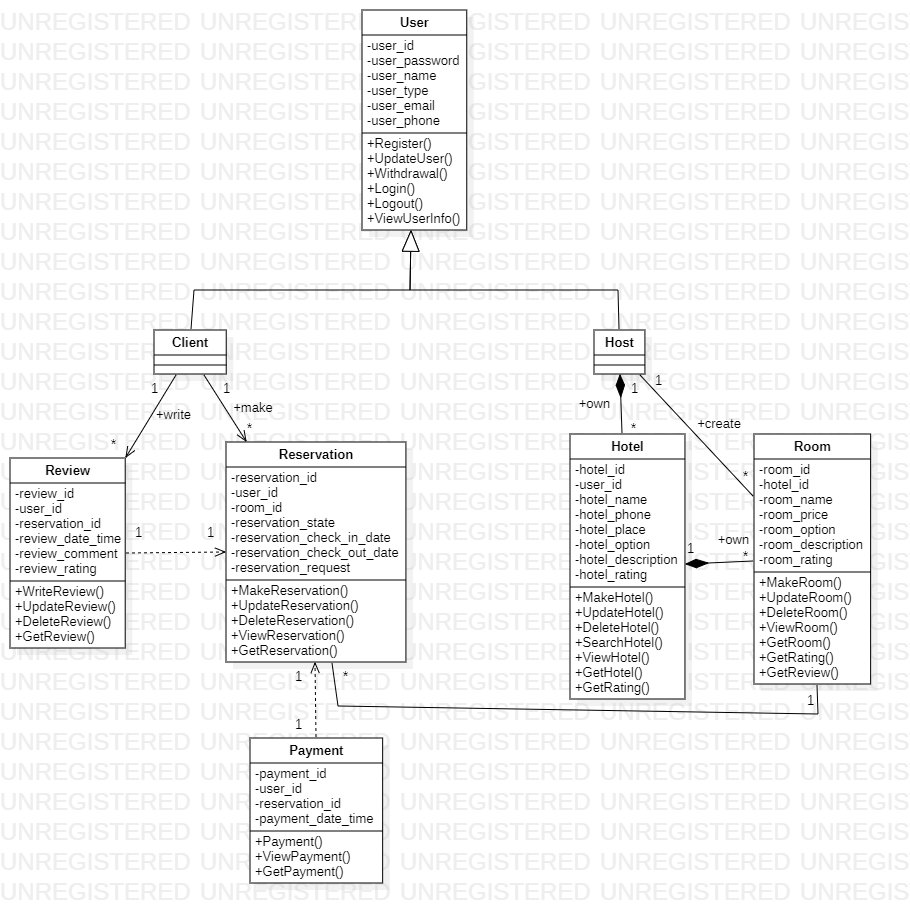
**4. 모듈(객체 모델) 명세서**

4.1 UML 다이어그램

4.1.1. Use-case diagram



4.1.2. class diagram,



**User class**

→ attribute

- user\_id : USER을 식별해주는 값

- user\_password : USER가 계정에 로그인할 때 사용하는 값

- user\_name : USER 이름

- user\_type : 해당 USER의 타입을 분별해주는 값

- user\_email : USER의 이메일

- user\_phone : USER의 핸드폰 번호

→ method

- Register() : USER객체를 생성하여 DB에 저장하는 Method

- UpdateUser() : DB에 있는 USER정보를 수정하는 Method

- Withdrawal() : DB에 있는 USER정보를 삭제하는 Method

- Login() : USER가 Server에 로그인하는 Method

- Logout() : USER가 Server에서 로그아웃하는 Method

- ViewUserInfo() : USER가 자신의 정보를 확인하는 Method

**Hotel class**

→ attribute

- hotel\_id : Hotel을 식별해주는 값

- user\_id : Hotel을 생성한 USER의 기본 키

- hotel\_name : Hotel 이름

- hotel\_phone : Hotel의 전화번호

- hotel\_place : Hotel이 위치한 장소

- hotel\_option : Hotel이 가지고 있는 장애인 편의시설 리스트

- hotel\_description : Hotel 설명

- hotel\_rating : Hotel의 별점

→ method

- MakeHotel() : : Hotel 객체를 생성하여 DB에 저장하는 Method

- UpdateHotel() : DB에 있는 Hotel 정보를 수정하는 Method

- DeleteHotel() : DB에 있는 Hotel 정보를 삭제하는 Method

- SearchHotel() : querysets을 이용하여 DB에 있는 Hotel을 가져오는 Method

- ViewHotel(): DB에 있는 Hotel 정보를 확인하는 Method

- GetHotel() : DB에 저장된 Reservation List를 가져오는 Method

- GetRating() : DB에 저장되어 있는 Review에서 Rating을 가져오는 Method

**Room class**

→ attribute

- room\_id : Room을 식별해주는 값

- hotel\_id : Room이 속해있는 Hotel의 기본 키

- room\_name : Room의 이름

- room\_price : Room의 가격

- room\_option : Room 이 가지고 있는 장애인 편의시설 리스트

- room\_description : Room 설명

- room\_rating : Room 평점

→ method

- MakeRoom() : Room 객체를 생성하여 DB에 저장하는 Method

- UpdateRoom() : DB에 있는 Room 정보를 수정하는 Method

- DeleteRoom() : DB에 있는 Room 정보를 삭제하는 Method

- ViewRoom() : DB에 있는 Room 정보를 확인하는 Method

- GetRoom() : DB에 저장된 Room List를 가져오는 Method

- GetRating() : DB에 저장되어 있는 Review에서 Rating을 가져오는 Method

- GetReview() : DB에 저장되어 있는 Review에서 Review을 가져오는 Method

**Reservation class**

→ attribute

- reservation\_id : Reservation을 식별해주는 값

- user\_id : Reservation을 생성하는 USER의 기본 키

- room\_id : Reservation이 생성되는 Room의 기본 키

- reservation\_state : 현재 Reservation 의 상태

- reservation\_check\_in\_date : USER가 요청한 Reservation의 체크인 날짜

- reservation\_check\_out\_date : USER가 요청한 Reservation의 체크아웃 날짜

- reservation\_request : USER의 요청 메세지

→ method

- MakeReservation() : Reservation 객체를 생성하여 DB에 저장하는 Method

- UpdateReservation(): DB에 있는 Reservation 정보를 수정하는 Method

- DeleteReservation() : DB에 있는 Reservation 정보를 삭제하는 Method

- ViewReservation() : DB에 있는 Reservation 정보를 확인하는 Method

- GetReservation() : DB에 저장된 Reservation List를 가져오는 Method

**Review class**

→ attribute

- review\_id : 생성되는 Review을 식별해주는 값

- user\_id : Review를 작성하는 USER의 기본 키

- reservation\_id : Review가 작성되는 Reservation의 기본 키

- review\_date\_time : Review가 작성되는 날짜 및 시간

- review\_comment : Review 내용

- review\_rating : 별점

→ method

- WriteReview() : Review 객체를 생성하여 DB에 저장하는 Method

- UpdateReview() : DB에 있는 Review를 수정하는 Method

- DeleteReview() : DB에 있는 Review를 삭제 Method

- GetReview() : DB에 저장된 Review를 가져오는 Method

**Payment class**

→ attribute

- payment\_id : Payment 을 식별해주는 값

- user\_id : Payment를 하는 USER의 기본 키

- reservation\_id : Payment가 이루어지는 Reservation의 기본 키

- payment\_date\_time : Payment가 생성되는 날짜 및 시간

→ method

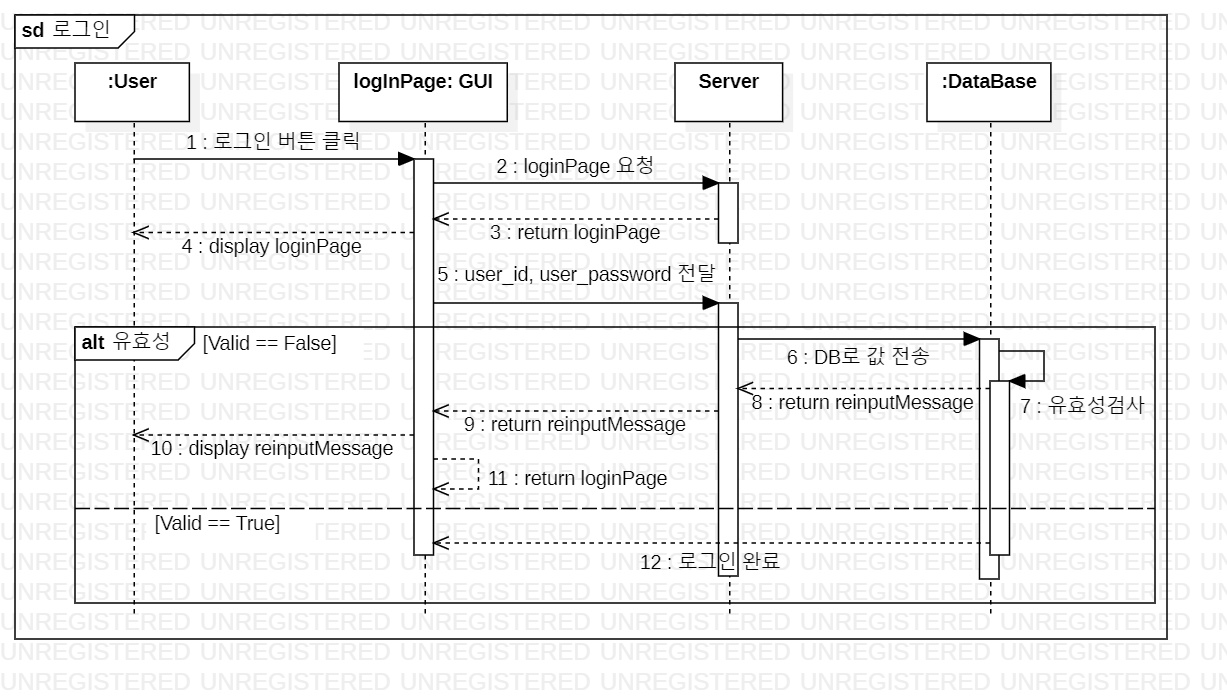
- ConfirmPayment() : Payment 객체를 생성하여 DB에 저장하는 Method

- ViewPayment() : DB에 저장된 Payment를 확인하는 Method

- GetPayment() : DB에 저장된 Payment를 가져오는 Method

4.1.3. Sequence Diagram

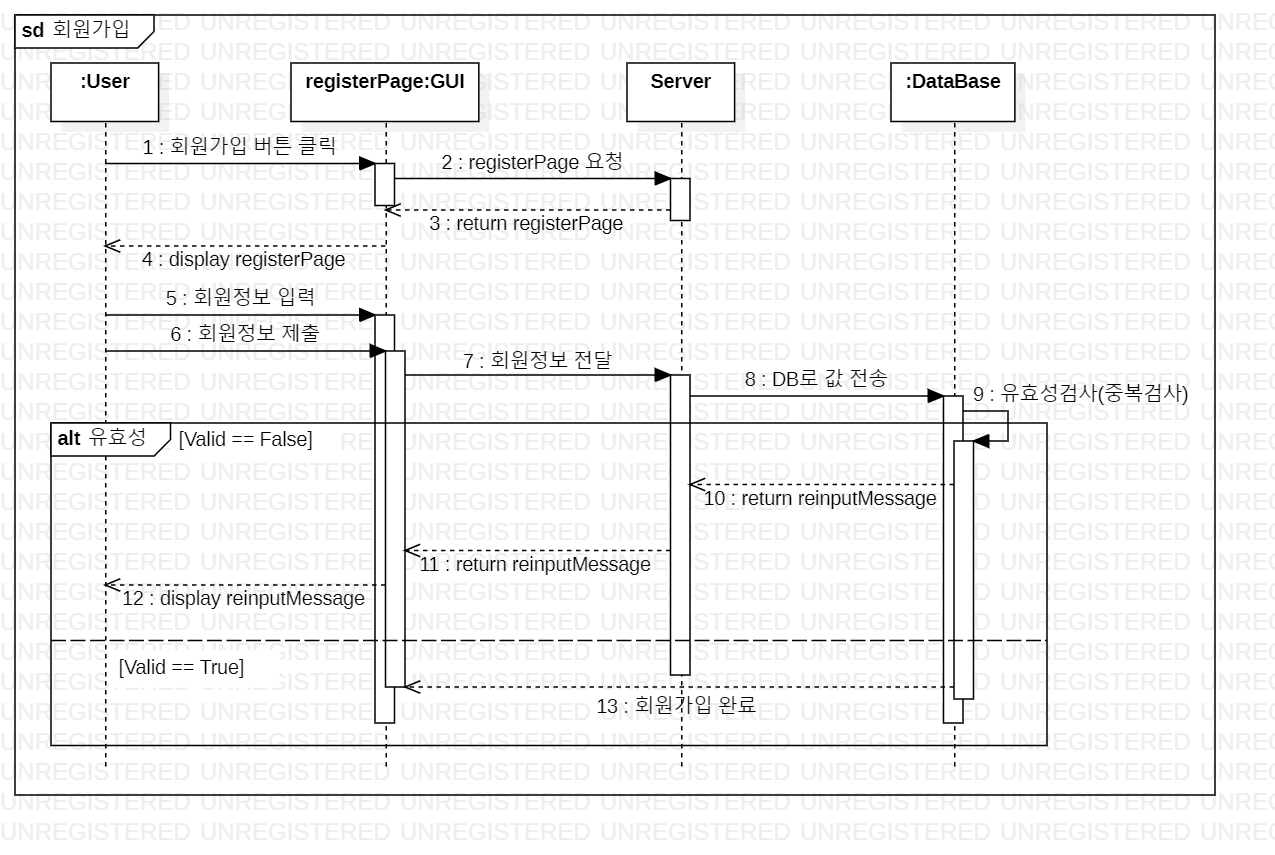
가. 로그인



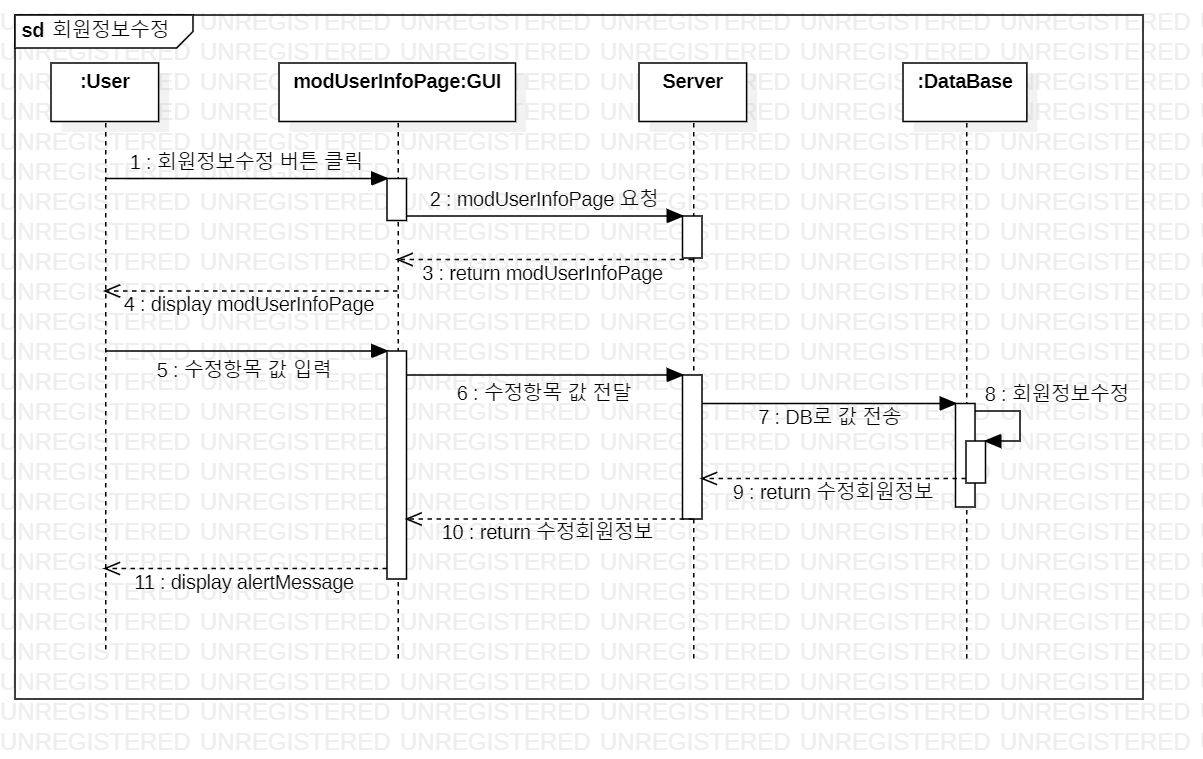
나. 로그아웃



다. 회원가입



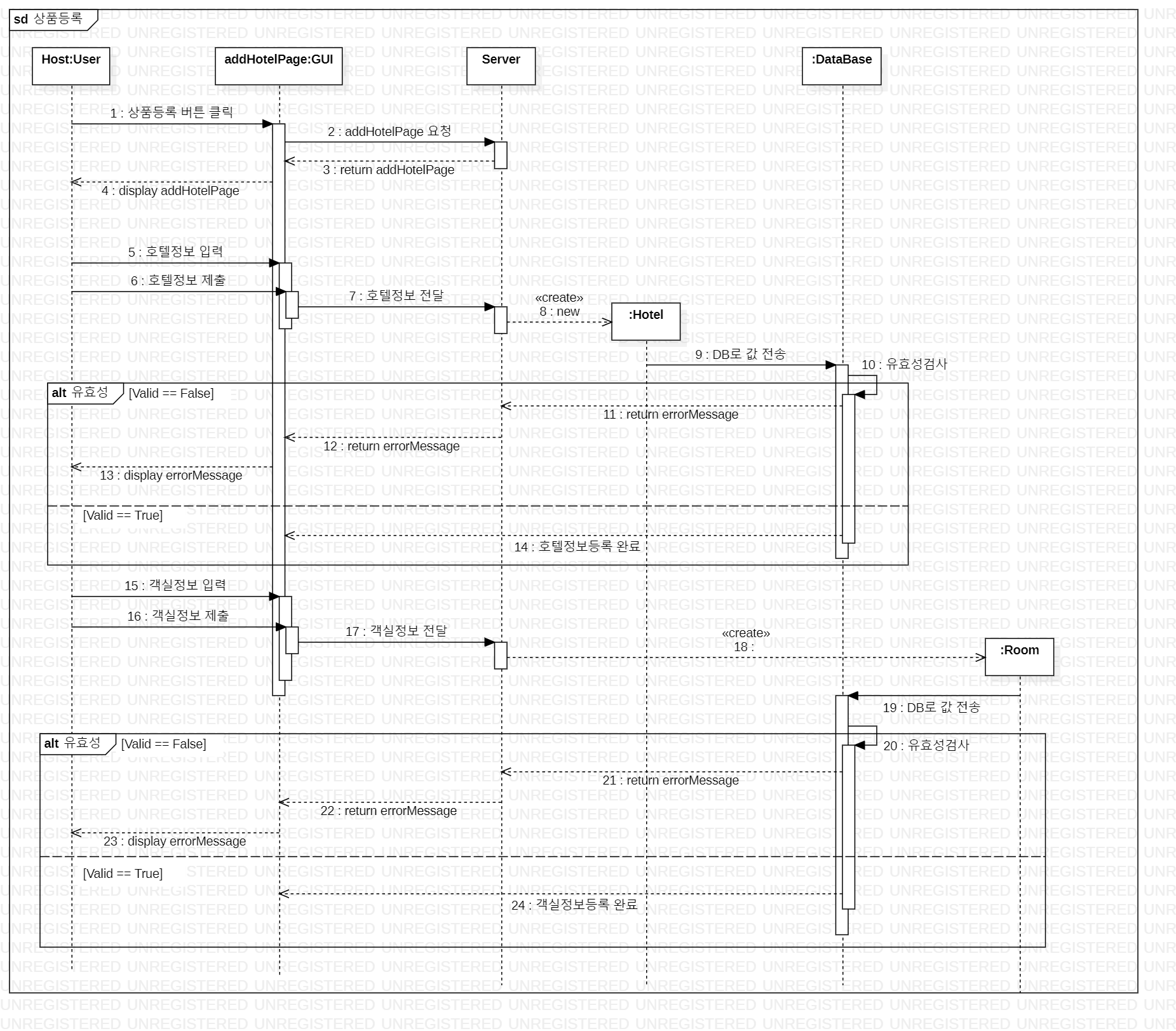
라. 회원정보수정



마. 회원탈퇴

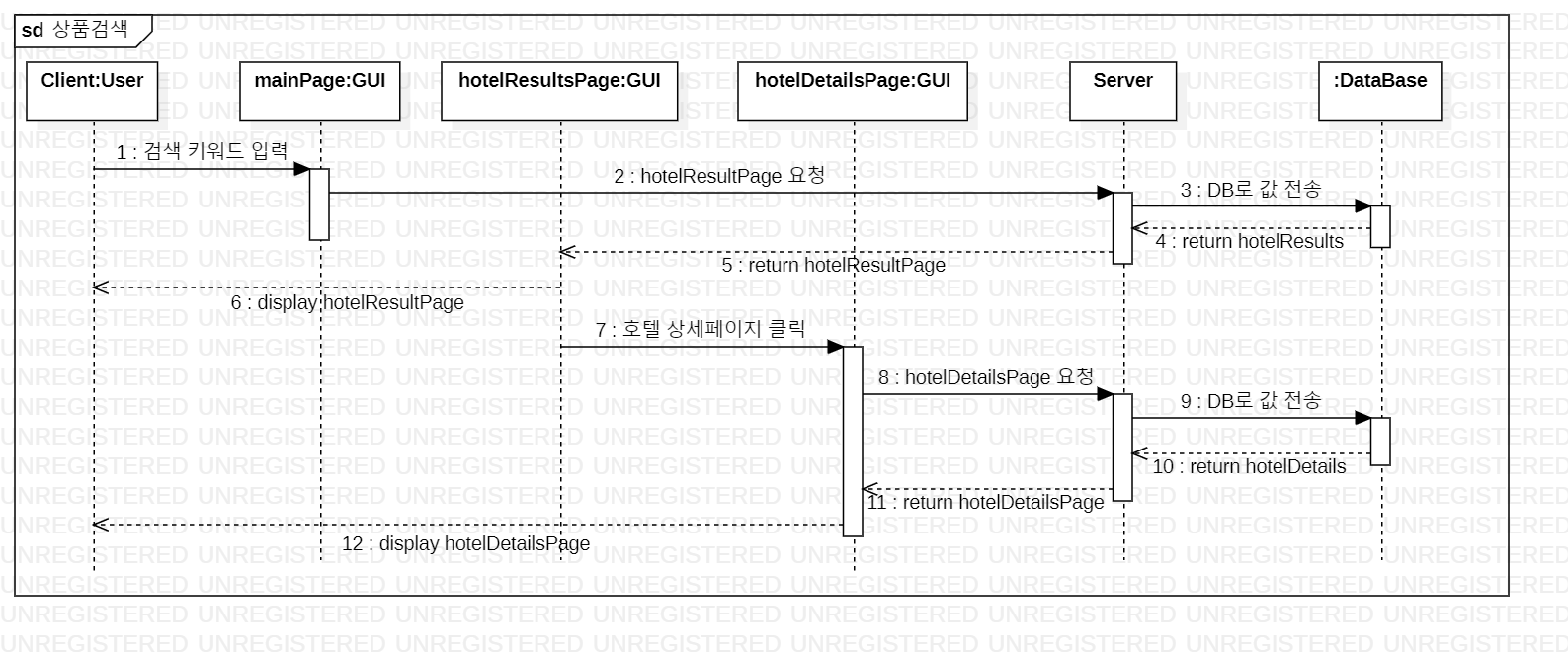


바. 상품등록

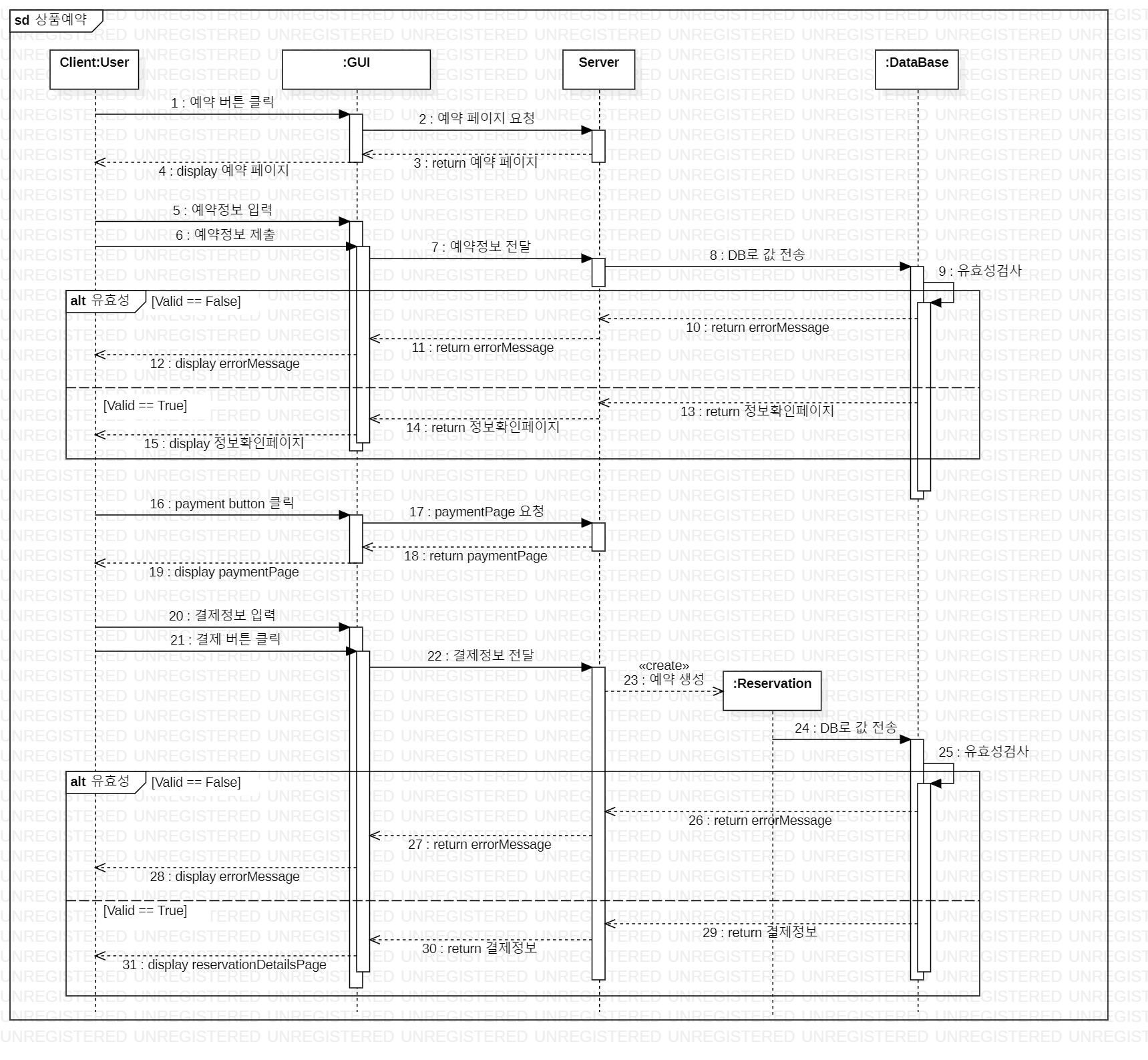


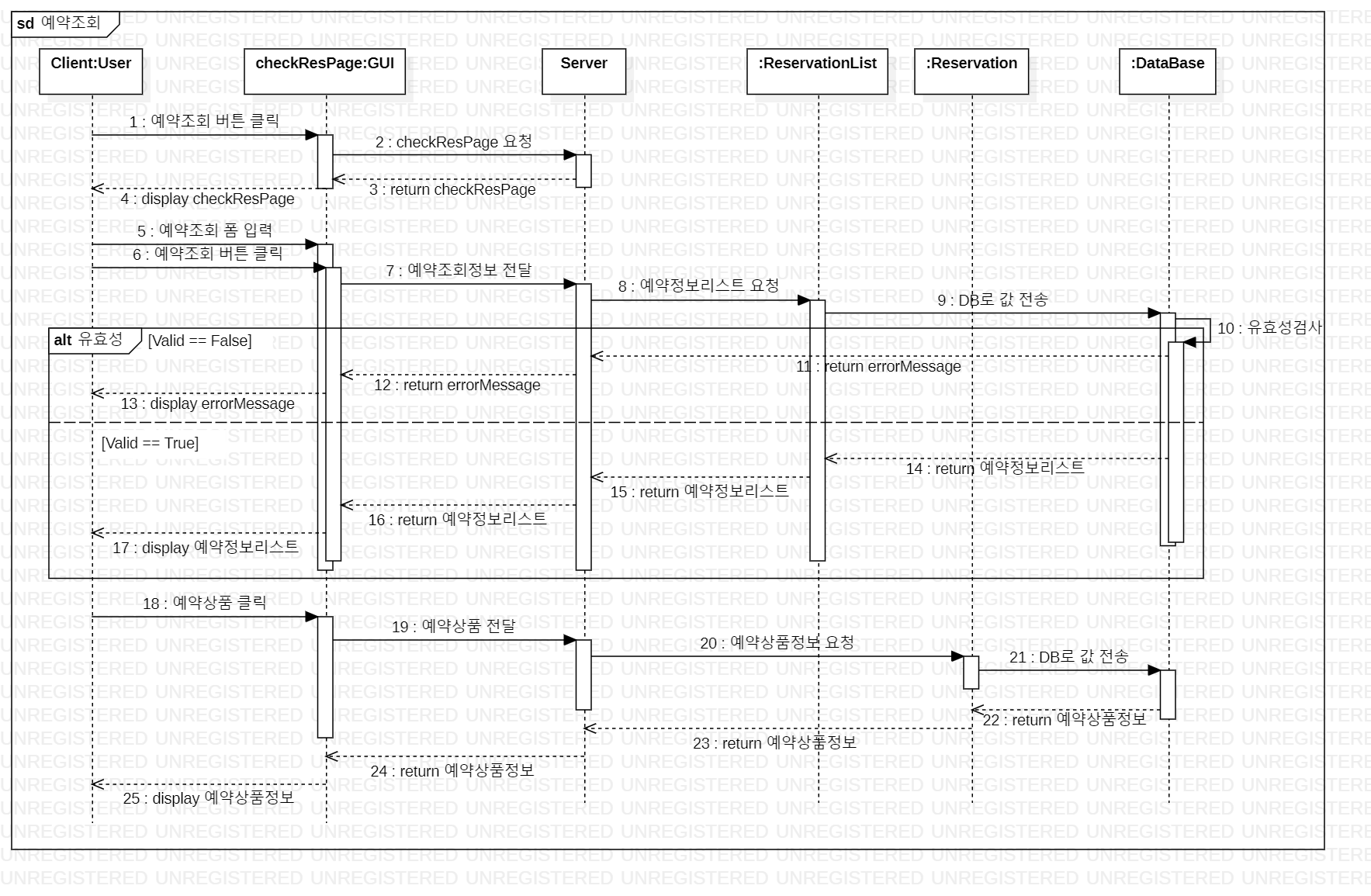
사. 상품관리



아. 상품검색

자. 상품예약



차. 예약조회 Architectural Design 

카. 리뷰작성



**5. 입출력 설계**

5.1 회원 관리

5.1.1 회원 가입

|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **입력** | **입력 제한** | **출력** |
| 약관 동의 | 약관 동의 체크 | 체크박스 | 회원정보  입력 화면 |
| 회원정보  입력 | 이름  아이디  비밀번호  이메일  전화번호 | 이름: 국문 25자 이내  아이디: 영문, 숫자, 특수문자 50자 이내  비밀번호: 영문, 숫자 특수문자 255자 이내  이메일: 영문, 숫자 특수문자 40자 이내  전화번호: 숫자 30자 이내 | 회원가입  완료 화면 |

5.1.2 로그인

|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **입력** | **입력 제한** | **출력** |
| 로그인 | 아이디  비밀번호 | 아이디: 영문, 숫자, 특수문자 50자 이내  비밀번호: 영문, 숫자 특수문자 255자 이내 | 로그인된  메인화면 |

5.1.3 회원정보 수정

|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **입력** | **입력 제한** | **출력** |
| 회원정보  수정 | 이름  아이디  비밀번호  이메일  전화번호 | 이름: 국문 25자 이내  아이디: 영문, 숫자, 특수문자 50자 이내  비밀번호: 영문, 숫자 특수문자 255자 이내  이메일: 영문, 숫자 특수문자 40자 이내  전화번호: 숫자 30자 이내 | 회원정보 수정  완료 화면 |

5.2 상품 등록

5.2.1 상품 등록 및 수정

|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **입력** | **입력 제한** | **출력** |
| 등록 | 상품명  주소  소개  가격  지원 종류 | 상품명: 없음  주소: 없음  소개: 없음  가격: 숫자  지원 종류: 없음 | 등록 결과  화면 |
| 수정 | 위와 동일 | 위와 동일 | 수정 결과  화면 |

5.3 상품 예약

5.3.1 상품 검색

|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **입력** | **입력 제한** | **출력** |
| 검색 | 인원  방 개수  검색어  이용 일시  희망 시설 | 인원: 체크박스  방 개수: 체크박스  검색어: 국문, 영문 50자 이내  이용 일시: 체크박스  희망 시설: 체크박스 | 검색 결과  화면 |

5.3.2 상품 예약

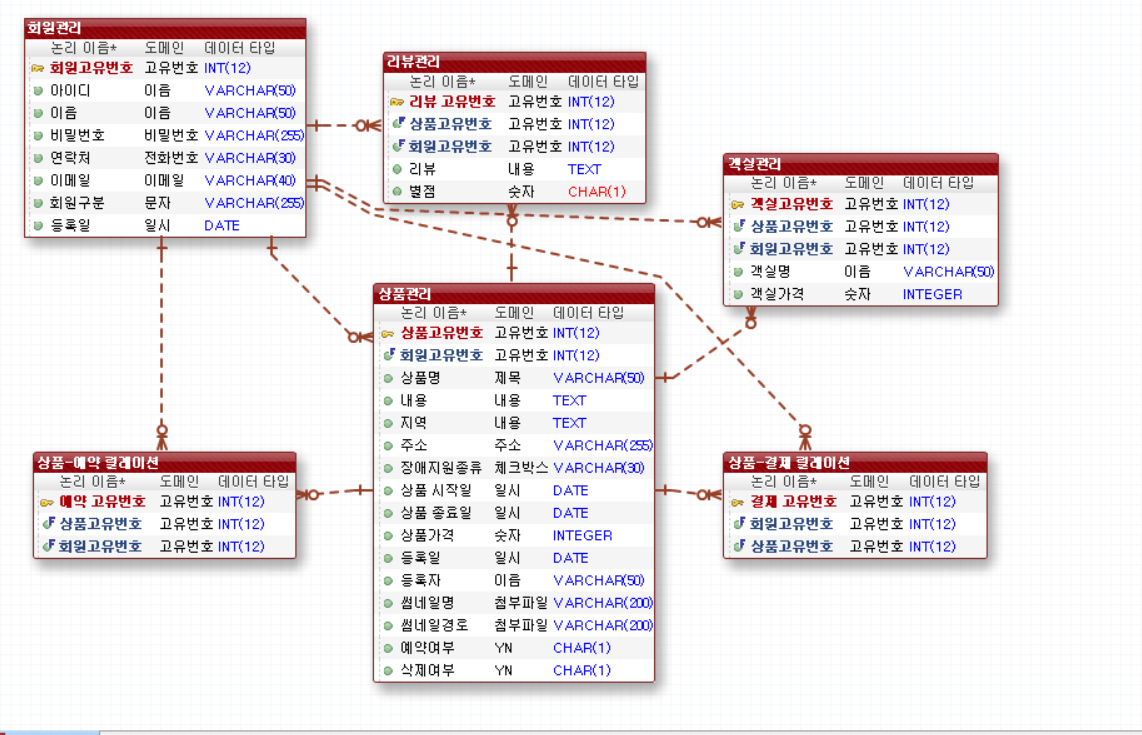
|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **입력** | **입력 제한** | **출력** |
| 예약 | 이름  이메일  전화번호  결제수단 | 이름: 국문 25자 이내  이메일: 영문, 숫자 특수문자 40자 이내  전화번호: 숫자 30자 이내  결제수단: 체크박스 | 예약 결과  화면 |

5.3.3 리뷰 작성

|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **입력** | **입력 제한** | **출력** |
| 리뷰 작성 | 총점  리뷰 | 총점: 체크박스  리뷰: 없음 | 예약 결과  화면 |

**6. Data 설계**

6.1 ERD (E-R Diagram)



6.2 DB Table Description:

1. 회원관리 테이블 (TB\_MEMBER)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **NULL** | **Key** | **Default** | **Description** |
| MEM\_SEQ | INT(12) |  | PK | AUTO\_INCREMENT | 회원고유번호 |
| MEM\_ID | VARCHAR(50) |  |  | NULL | 아이디 |
| MEM\_NM | VARCHAR(50) |  |  | NULL | 이름 |
| MEM\_PW | VARCHAR(255) |  |  | NULL | 비밀번호 |
| PHONE | VARCHAR(30) |  |  | NULL | 연락처 |
| EMAIL | VARCHAR(40) |  |  | NULL | 이메일 |
| MEM\_TYPE | VARCHAR(255) |  |  | NULL | 회원구분 |
| REG\_DT | DATE |  |  | NULL | 등록일 |

1. 리뷰관리 테이블(TB\_REVIEW)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **NULL** | **Key** | **Default** | **Description** |
| REV\_SEQ | INT(12) |  | PK | AUTO\_INCREMENT | 리뷰 고유번호 |
| PRO\_SEQ | INT(12) |  | FK |  | 상품 고유번호 |
| MEM\_SEQ | INT(12) |  | FK |  | 회원 고유번호 |
| REVIEW | TEXT | O |  | NULL | 리뷰 |
| RATING | CHAR(1) | O |  | NULL | 별점 |

1. 상품관리 테이블(TB\_PRODUCT)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **NULL** | **Key** | **Default** | **Description** |
| PRO\_SEQ | INT(12) |  | PK | AUTO\_INCREMENT | 상품고유번호 |
| MEM\_SEQ | INT(12) |  | FK |  | 회원고유번호 |
| TITLE | INT(12) |  |  | NULL | 상품명 |
| CONTENTS | TEXT |  |  | NULL | 내용 |
| PLACE | TEXT |  |  | NULL | 지역 |
| ADDRESS | VARCHAR(255) |  |  | NULL | 주소 |
| TYPE | VARCHAR(30) |  |  | NULL | 장애지원종류 |
| START\_DT | DATE |  |  | NULL | 상품 시작일 |
| END\_DT | DATE |  |  | NULL | 상품 종료일 |
| PRO\_PRICE | INTEGER |  |  | NULL | 상품가격 |
| REG\_DT | DATE |  |  | NULL | 등록일 |
| REG\_ID | VARCHAR(50) |  |  | NULL | 등록자 |
| THUMB\_NM | VARCHAR(200) |  |  | NULL | 썸네일명 |
| THUMB\_FILE | VARCHAR(200) |  |  | NULL | 썸네일경로 |
| RESERVE\_YN | CHAR(1) |  |  | NULL | 예약여부 |
| DEL\_YN | CHAR(1) |  |  | NULL | 삭제여부 |

1. 상품/예약 관리 (TB\_PRO\_RES\_REL)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **NULL** | **Key** | **Default** | **Description** |
| RES\_SEQ | INT(12) |  | PK | AUTO\_INCREMENT | 예약 고유번호 |
| PRO\_SEQ | INT(12) |  | FK |  | 상품고유번호 |
| MEM\_SEQ | INT(12) |  | FK |  | 회원고유번호 |

1. 상품/결제 관리 (TB\_PRO\_PAY\_REL)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **NULL** | **Key** | **Default** | **Description** |
| PAY\_SEQ | INT(12) |  | PK | AUTO\_INCREMENT | 결제 고유번호 |
| MEM\_SEQ | INT(12) |  | FK |  | 회원고유번호 |
| PRO\_SEQ | INT(12) |  | FK |  | 상품고유번호 |

1. 객실 관리 (TB\_ROOM)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **NULL** | **Key** | **Default** | **Description** |
| RO\_SEQ | INT(12) |  | PK | AUTO\_INCREMENT | 결제 고유번호 |
| MEM\_SEQ | INT(12) |  | FK |  | 회원고유번호 |
| PRO\_SEQ | INT(12) |  | FK |  | 상품고유번호 |
| ROOM\_NM | VARCHAR(50) |  |  |  | 객실명 |
| PRICE | INTEGER |  |  |  | 객실가격 |

6.3 Method

- DB 를 운용하는 Method

1. Create() 레코드 생성

- Register() : USER객체를 생성하여 DB에 저장하는 Method

- MakeHotel() : : Hotel 객체를 생성하여 DB에 저장하는 Method

- MakeReservation() : Reservation 객체를 생성하여 DB에 저장하는 Method

- WriteReview() : Review 객체를 생성하여 DB에 저장하는 Method

- ConfirmPayment() : Payment 객체를 생성하여 DB에 저장하는 Method

1. Update() 레코드 수정

- UpdateUser() : DB에 있는 USER정보를 수정하는 Method

- UpdateHotel() : DB에 있는 Hotel 정보를 수정하는 Method

- UpdateRoom() : DB에 있는 Room 정보를 수정하는 Method

- UpdateReservation(): DB에 있는 Reservation 정보를 수정하는 Method

- UpdateReview() : DB에 있는 Review를 수정하는 Method

1. Delete() 레코드 삭제

- Withdrawal() : DB에 있는 USER정보를 삭제하는 Method

- DeleteHotel() : DB에 있는 Hotel 정보를 삭제하는 Method

- MakeRoom() : Room 객체를 생성하여 DB에 저장하는 Method

- DeleteRoom() : DB에 있는 Room 정보를 삭제하는 Method

- DeleteReservation() : DB에 있는 Reservation 정보를 삭제하는 Method

- DeleteReview() : DB에 있는 Review를 삭제 Method

1. Select() 레코드 조회

- ViewUserInfo() : USER가 자신의 정보를 확인하는 Method

- SearchHotel() : querysets을 이용하여 DB에 있는 Hotel을 가져오는 Method

- GetHotel() : DB에 저장된 Reservation List를 가져오는 Method

- GetRating() : DB에 저장되어 있는 Review에서 Rating을 가져오는 Method

- ViewRoom() : DB에 있는 Room 정보를 확인하는 Method

- GetRoom() : DB에 저장된 Room List를 가져오는 Method

- GetRating() : DB에 저장되어 있는 Review에서 Rating을 가져오는 Method

- GetReview() : DB에 저장되어 있는 Review에서 Review을 가져오는 Method

- GetReservation() : DB에 저장된 Reservation List를 가져오는 Method

- GetReview() : DB에 저장된 Review를 가져오는 Method

- ViewPayment() : DB에 저장된 Payment를 확인하는 Method

- GetPayment() : DB에 저장된 Payment를 가져오는 Method

6.4 Sequence Diagram

- 레코드 생성

1. 회원 관리 테이블(TB\_MEMBER) : user가 회원 가입 시 회원 레코드 생성
2. 리뷰 관리 테이블(TB\_REVIEW) : user가 리뷰 등록 시 리뷰 레코드 생성
3. 상품 관리 테이블(TB\_PRODUCT) : 호스트가 상품 등록시 상품 레코드 생성
4. 상품 예약 관리(TB\_PRO\_RES\_REL) : user가 상품 예약 시 예약 레코드 생성
5. 상품 결제 관리(TB\_PRO\_PAY\_REL) : user가 상품 결제 시 결제 상품 레코드 생성

- 레코드 수정

1. 회원관리 테이블(TB\_MEMBER) : 개인정보 수정 시 회원 정보 레코드 수정
2. 리뷰 관리 테이블(TB\_REVIEW) : user가 리뷰 수정 시 레코드 수정
3. 상품관리 테이블(TB\_PRODUCT) : 호스트가 상품 수정 시 레코드 수정
4. 상품 예약 관리(TB\_PRO\_RES\_REL) : user가 예약 정보 수정 시 레코드 수정
5. 상품 결제 관리(TB\_PRO\_PAY\_REL) : 수정 기능 x

- 레코드 삭제

1. 회원관리 테이블(TB\_MEMBER) : 회원 탈퇴 시 레코드 삭제
2. 리뷰 관리 테이블(TB\_REVIEW) : user가 리뷰 삭제 시 레코드 삭제
3. 상품관리 테이블(TB\_PRODUCT) : 호스트가 상품 삭제 시 레코드 삭제
4. 상품 예약 관리(TB\_PRO\_RES\_REL) : user가 예약 취소 시 레코드 삭제
5. 상품 결제 관리(TB\_PRO\_PAY\_REL) : user가 결제 취소 시 레코드 삭제

- 레코드 조회

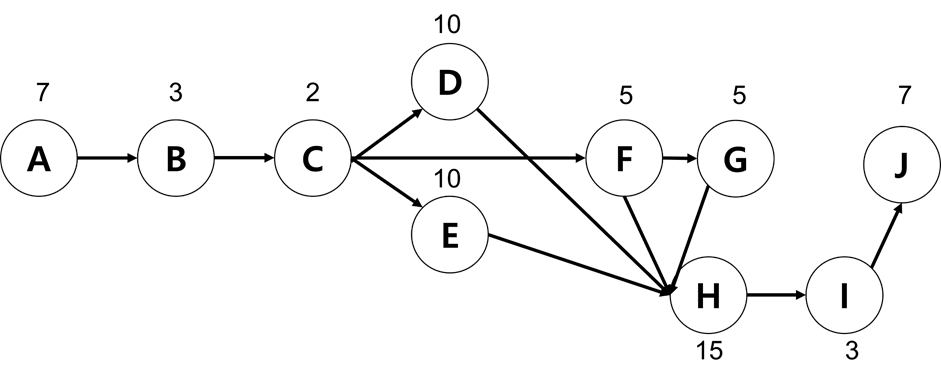
1. 회원관리 테이블(TB\_MEMBER) : 마이페이지 개인정보 조회 시 레코드 조회
2. 리뷰 관리 테이블(TB\_REVIEW) : user가 리뷰 조회 시 레코드 조회
3. 상품관리 테이블(TB\_PRODUCT) : 호스트가 상품 조회 및 수정 시 레코드 조회
4. 상품 예약 관리(TB\_PRO\_RES\_REL) : user가 마이페이지 예약 확인 시 레코드 조회
5. 상품 결제 관리(TB\_PRO\_PAY\_REL) : user가 마이페이지 결제 목록 확인 시 레코드 조회

**7. Project Planning**

7.1 Tasks

|  |  |  |
| --- | --- | --- |
| **작업** | **선행작업** | **소요기간** |
| A. 구조 설계  B. 스토리보드 작성  C. 페이지 목록 설계  D. 프로그램 설계  E. 디자인 시안 설계 | -  A  B  C  C | 7  3  2  10  10 |
| F. 페이지 디자인  G. 페이지 퍼블리싱  H. 프로그래밍 | C, E  F  D, E, F, G | 5  5  15 |
| I. 통합테스트  J. 테스트 결과 반영 | G  J | 3  7 |

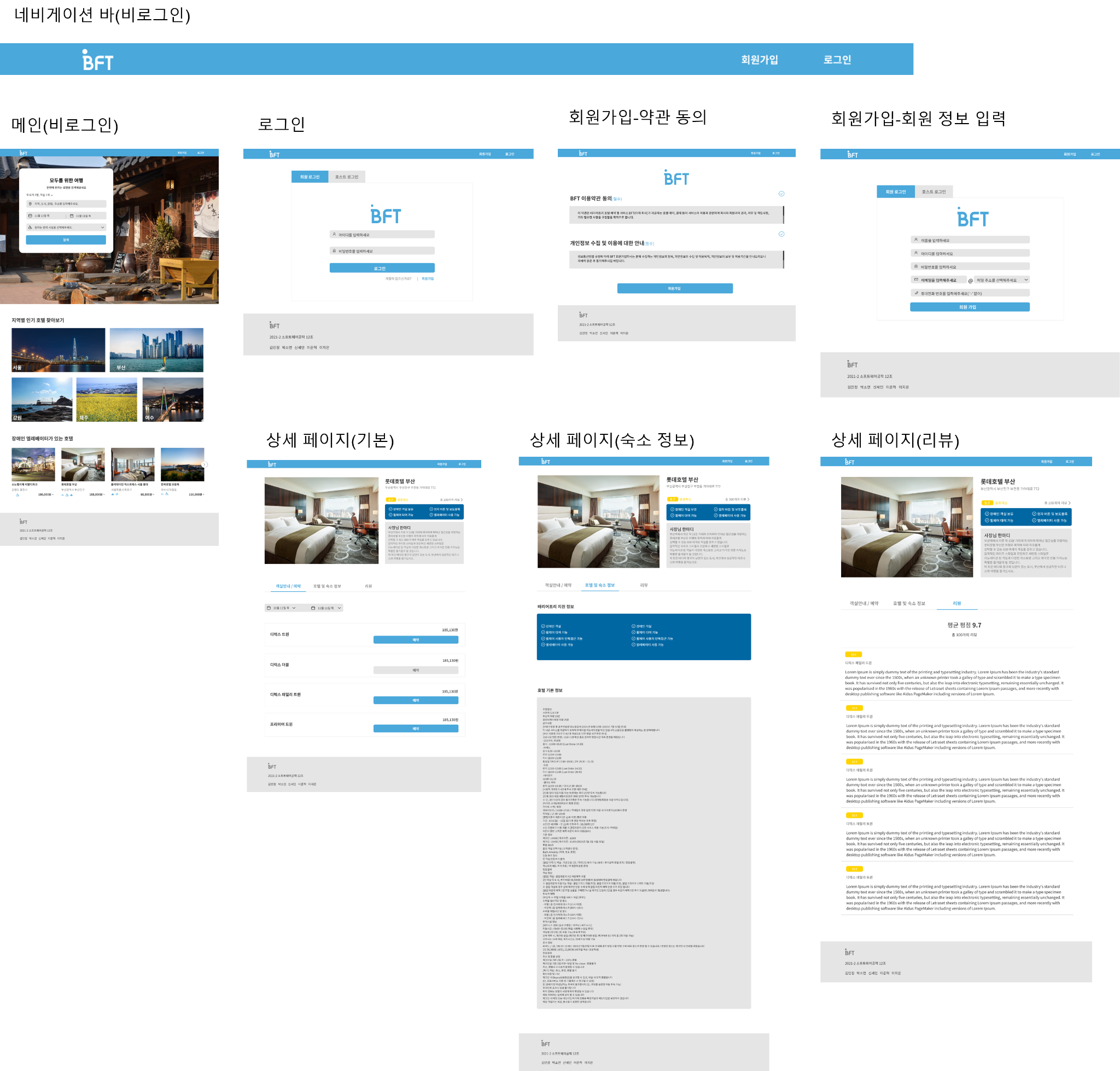
7.2 CPM Network

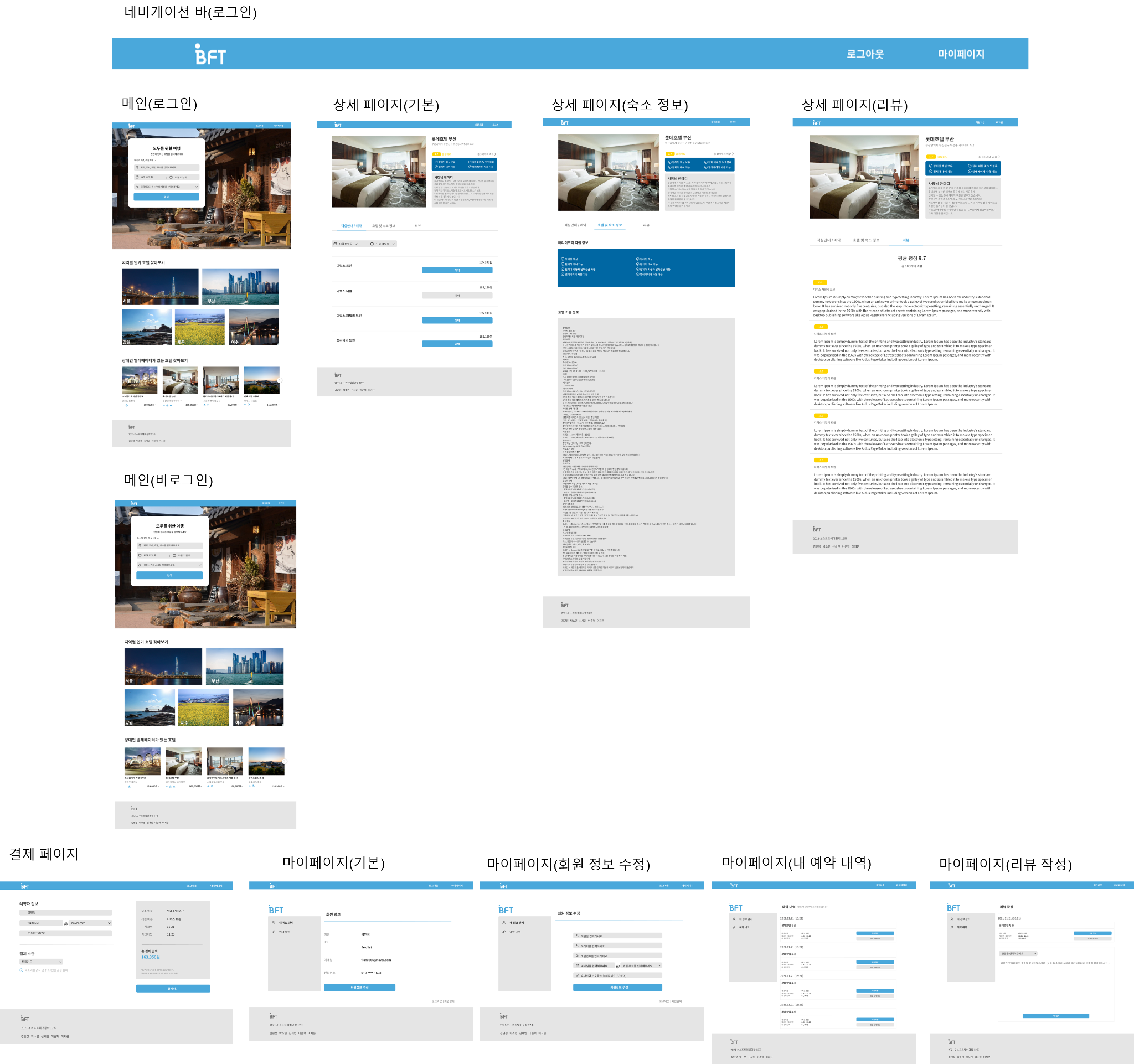


7.3 Task Allocation

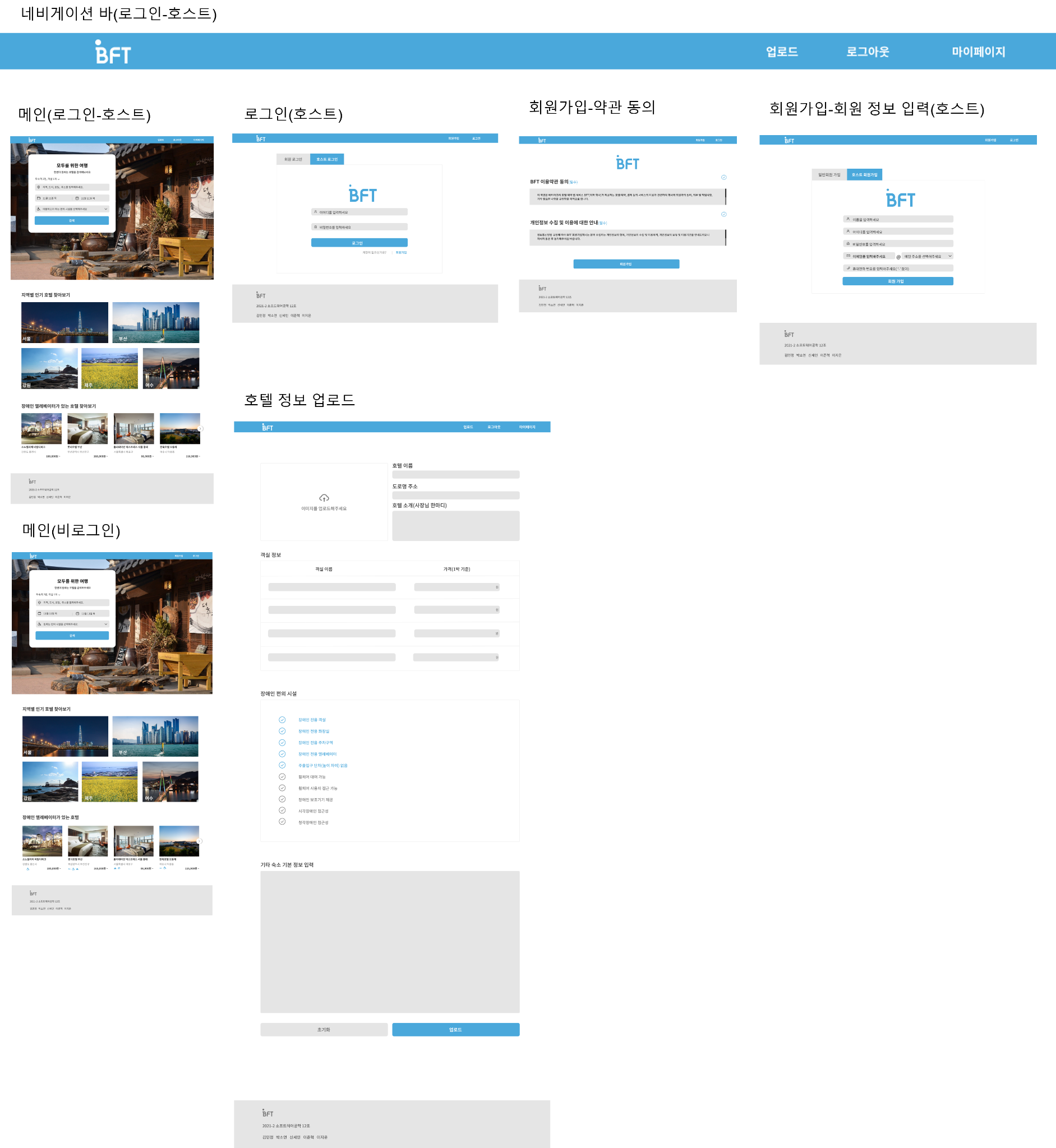
|  |  |
| --- | --- |
| 담당자 | 역할 |
| 김민정 | 웹페이지 목록 설계, 디자인 시안 설계 웹 퍼블리싱 |
| 김소연 | 구조 설계, 프로그램 설계 |
| 신세인 | 스토리보드 작성, 프로그램 설계 |
| 이지은 | 스토리보드 작성, 프로그램 설계 |
| 이준혁 | 구조 설계, 웹페이지 디자인 및 퍼블리싱 |
| 공통 | 프로그래밍, 통합테스트, 테스트 결과 반영 |

8. User Interface

8.1. 비로그인 사용자

8.2. 로그인된 회원

8.3. 호스트



9. 프로그램 작성 지침

1. 리엑트는 클래스형보다 함수형으로 작성한다
2. 클래스, 메소드 작성 시 이름을 명시적으로 사용한다.
3. 클래스 마다 명확한 목적을 부여하고, 복잡한 것은 간단하게 나눈다.
4. 서브 클래스에서 기본 기능을 확장하지 않는다.
5. 애트리뷰트 명은 소문자로 시작하고, ‘테이블명\_애트리뷰트 명’으로 작성한다
6. 변수는 카멜형으로 작성하도록 한다.
7. 중첩 반복문의 사용은 자제한다.
8. 코드는 가능한 함수로 묶어서 작성한다.
9. Python 코드 포멧터는 black을 사용하여 코드 형식을 표준화 한다.
10. 주석을 최대한으로 작성하여, 다른 팀원의 이해를 돕는다.
11. 커밋 메세지를 다른 팀원들이 이해할 수 있게 작성한다.

**\*첨부**

[1] 자료사전

1. 상품목록 = 상품번호+상품정보+등록일+상품 썸네일

1.1 상품번호 =상품 고유번호

1.2 상품정보 =상품명+객실명+객실가격+내용+지역+주소+장애지원종류+상품 기간

1.2.1 상품기간= 시작일+ 등록일 \*상품 유효 기간\*

1.2.2 장애지원종류= 객실 | 화장실 | 주차구역 | 엘리베이터 | 단차없음 | 휠체어 대여 | 휠체어 사용자 접근성 | 장애인 보조기기 제공 | 시각장애인 접근성 | 청각장애인 접근성

1.3 등록일= 년+’-‘+월+’-‘+일 \*상품등록일\*

1.4 상품 썸네일= 썸네일 명+파일경로

2. 리뷰목록 = 회원번호+상품번호+리뷰내용+별점

2.1 회원번호 = 회원 고유번호

2.2 상품번호 = 상품 고유번호

2.3 별점 = 1 | 2 | 3 | 4 | 5

3. 결제목록 = 회원번호+상품번호+결제일

3.1 회원번호 = 회원 고유번호

3.2 상품번호 = 상품 고유번호

3.3 결제일 =년+’-‘+월+’-‘+일

4. 예약목록 = 상품번호+ 회원번호+날짜

4.1. 상품번호 = 상품 고유번호

4.2 회원번호 = 회원 고유번호

4.3 날짜 = 년+’-‘+월+’-‘+일 \*예약날짜\*

5. 회원목록 = 회원번호+아이디+비밀번호+연락처+이메일+회원구분+등록일

5.1 회원번호 = 회원 고유번호

5.2 회원구분 = 관리자 | 일반회원

5.3 등록일 = 년+’-‘+월+’-‘+일 \*회원 가입 날짜\*