

경마 베팅 시스템

소프트웨어 설계 명세서

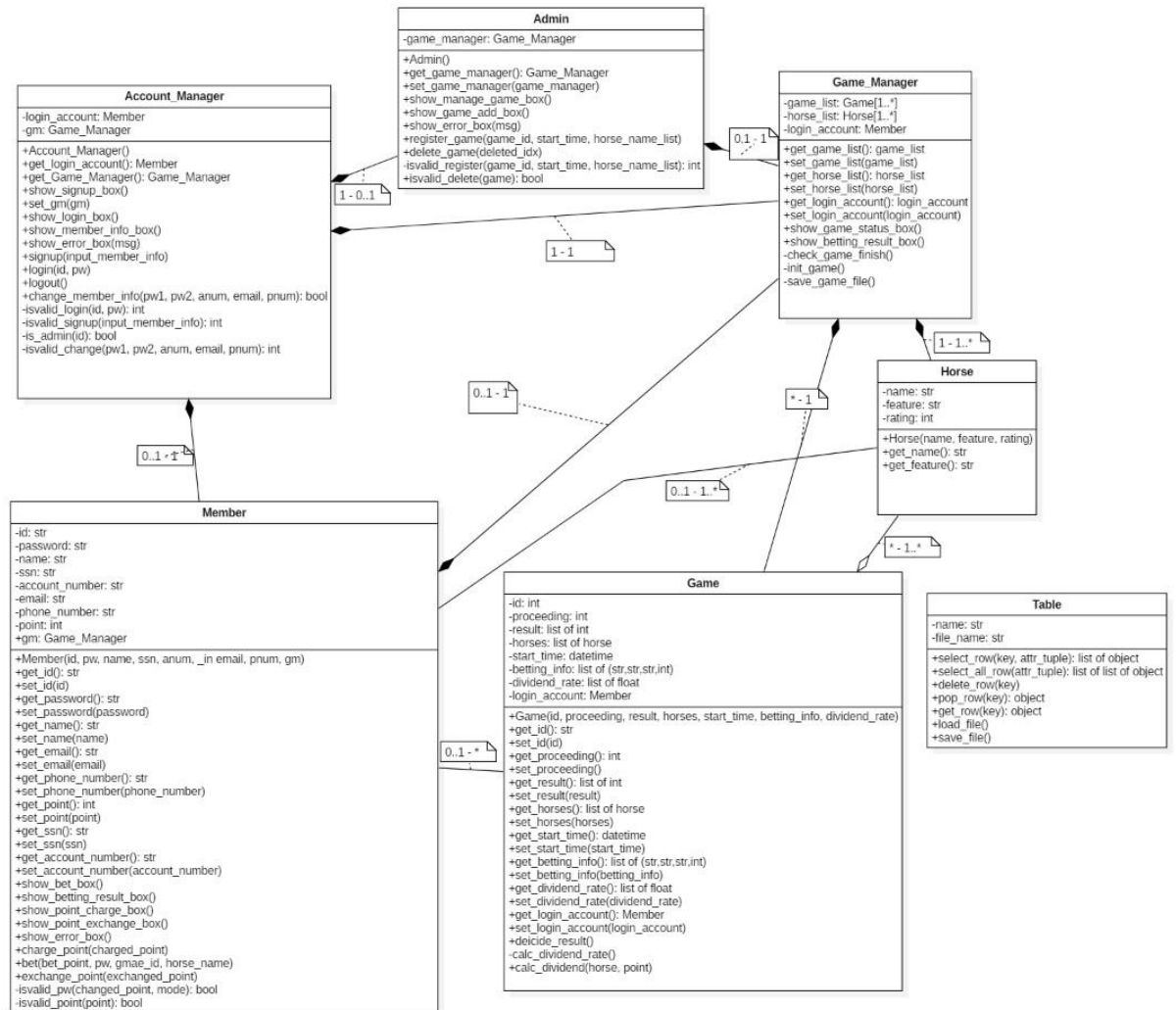
2013058668 서강휘

2014004675 손채준

2014005114 정근욱

1. 클래스 설계

- 클래스 다이어그램(UI 관련 클래스는 너무 복잡해서 생략. 시퀀스 다이어그램에는 존재)



- Account_Manager – Member

- ✓ 1 - 0..1 : Account_Manager 는 반드시 하나 존재하지만 Member 는 초기에는 존재하지 않다가 Member 사용자가 로그인 하면 하나가 생성된다.
- ✓ 포함 관계 : Account_Manager 는 login_account 속성으로 Member 객체를 가지고 있고, Account_Manager 없이 Member 가 존재할 수 없다.

- Account_Manager – Admin

- ✓ 1 - 0..1 : Account_Manager 는 반드시 하나 존재하지만 Admin 는 초기에는 존재하지 않다가 Admin 사용자가 로그인 하면 하나가 생성된다.
- ✓ 포함 관계 : Account_Manager 는 login_account 속성으로 Admin 객체를 가지고 있고, Account_Manager 없이 Admin 이 존재할 수 없다.

- Game_Manager – Game

- ✓ 1 - * : Game 은 없을 수도 있고 여러 개 있을 수도 있지만 Game_Manager 는 하나만 존재한다.
- ✓ 포함 관계 : Game_Manager 는 game_list 속성으로 Game 객체들을 가지고 있고, Game_Manager 없이 Game 이 존재할 수 없다.

- Game_Manager – Horse

- ✓ 1 - 1..* : Game_Manager 는 항상 하나 존재하고 Horse 는 항상 여러 개 존재한다.
- ✓ 포함 관계 : Game_Manager 는 horse_list 속성으로 Horse 객체들을 가지고 있고, Game_Manager 없이 Horse 가 존재할 수 없다.

- Game – Horse

- ✓ * - 1..* : game 은 없을 수도 있고 여러 개 있을 수도 있지만 Horse 는 항상 하나 이상 존재한다.
- ✓ 의존 관계 : Game 은 horses 속성으로 horse 객체들을 가지고 있다. 그리고 Game 없이 Horse 가 존재할 수 있다.

- Member – Game

- ✓ 0..1 - * : Member 는 없거나 1 개만 존재한다. Game 은 없을 수도 있고 여러 개 있을 수도 있다.
- ✓ 연관 관계 : Member 가 bet 오퍼레이션을 사용할 때 isvalid_game_info 같은 Game 의 오퍼레이션을 사용한다.

- Admin – Game_Manager

- ✓ 1 - 0..1 : Admin 는 초기에는 존재하지 않다가 Admin 사용자가 로그인 하면 하나가 생성되지만, Game_Manager 는 반드시 하나 존재한다.
- ✓ 포함 관계 : Admin 은 game_manager 속성으로 Game_Manager 객체를 가지고 있고, Game_Manager 없이 Admin 이 존재할 수 없다.

- Account_Manager – Game_Manager

- ✓ 1 - 1 : Account_Manager 와 Game_Manager 모두 하나씩 존재한다.
- ✓ 포함 관계 : Account_Manager 는 gm 속성으로 Game_Manager 객체를 가지고 있고, Game_Manager 없이 Account_Manager 가 존재할 수 없다.

- Game_Manager – Member

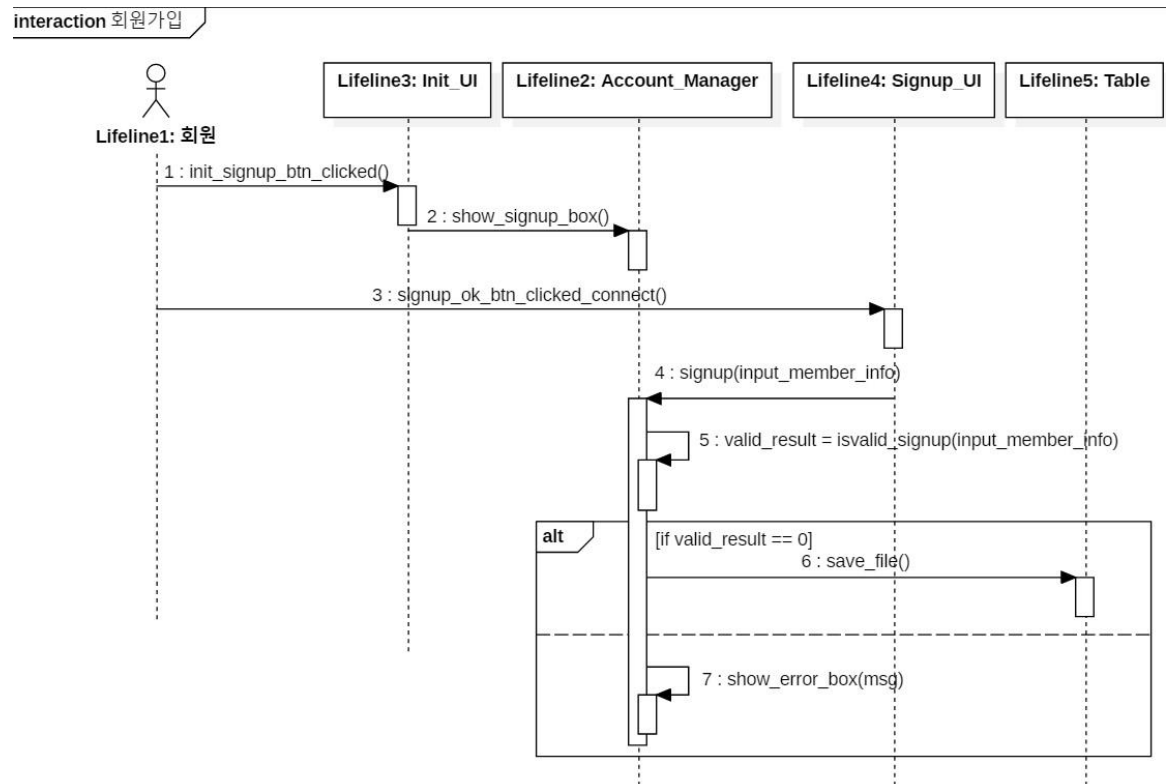
- ✓ 1 – 0..1 : Game_Manager 는 항상 하나 존재하지만 Member 객체는 Member 사용자가 로그인 할 때 하나 생긴다.
- ✓ 포함 관계 : Member 는 gm 속성으로 Game_Manager 객체를 가지고 있고, Game_Manager 없이 Member 가 존재할 수 없다.

- Member - Horse

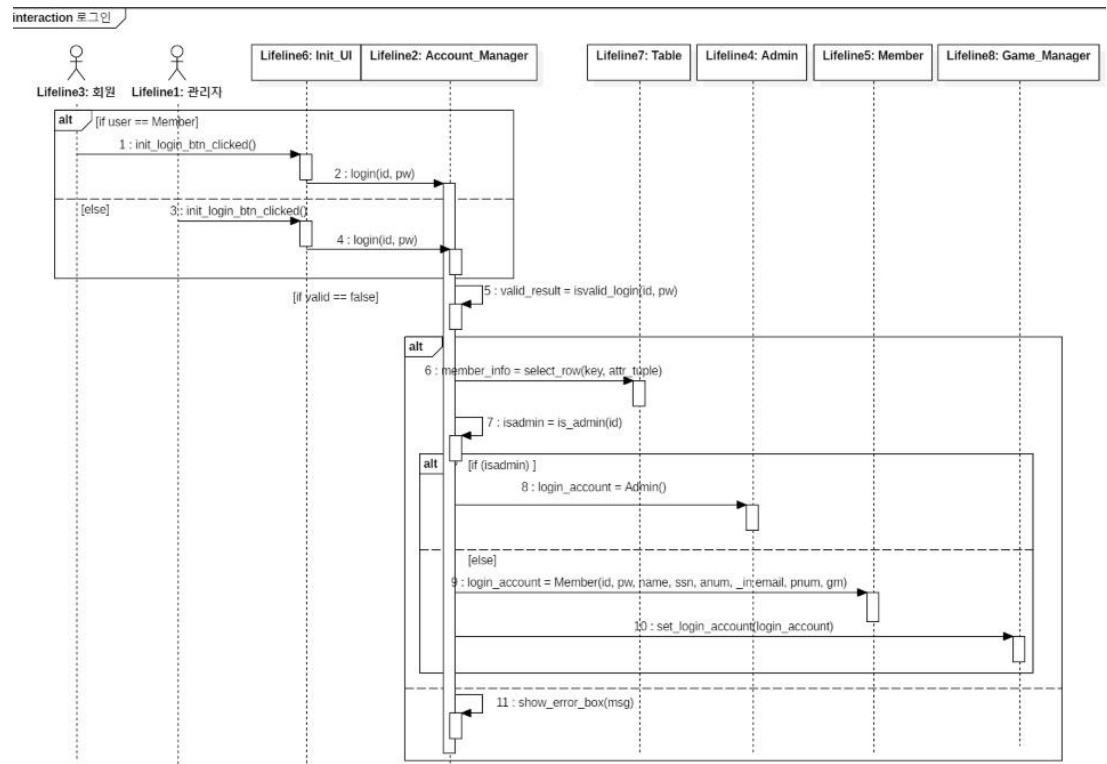
- ✓ 0..1 - * : Member 는 없거나 1 개만 존재한다. Horse 는 항상 여러 개 존재한다.
- ✓ 연관 관계 : Member 가 bet 오퍼레이션을 사용할 때 Horse 의 name, feature 속성을 가져오기 위해 get_name, get_feature 오퍼레이션을 사용한다.

2. 시퀀스 다이어그램

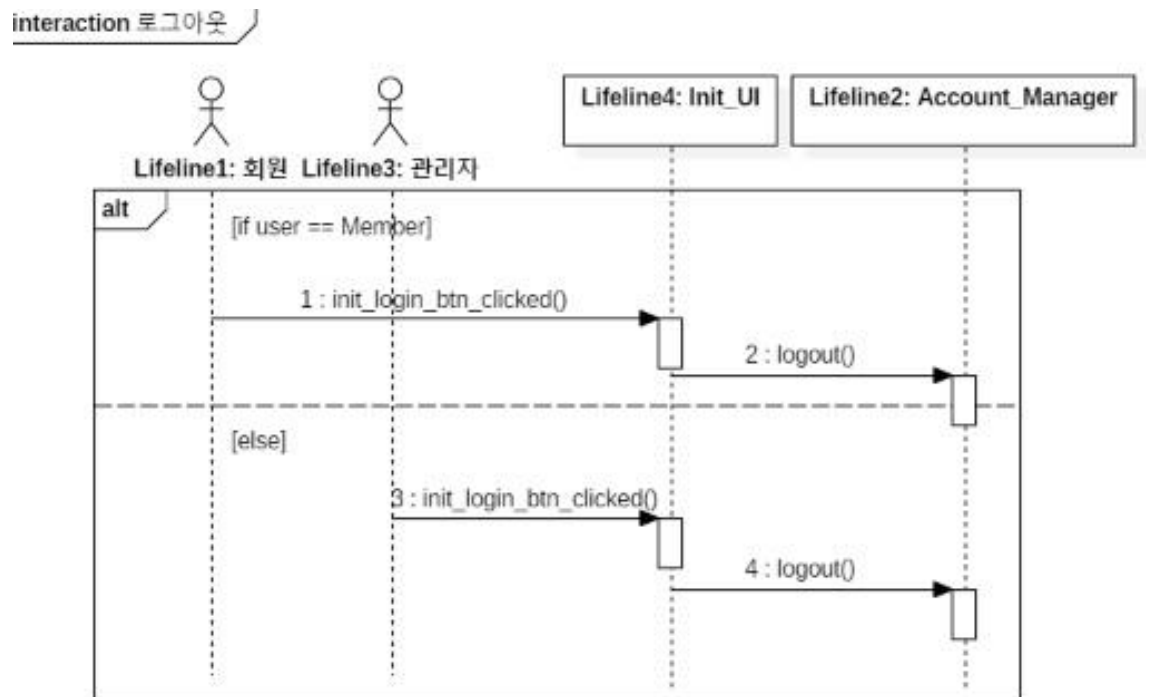
- 회원가입



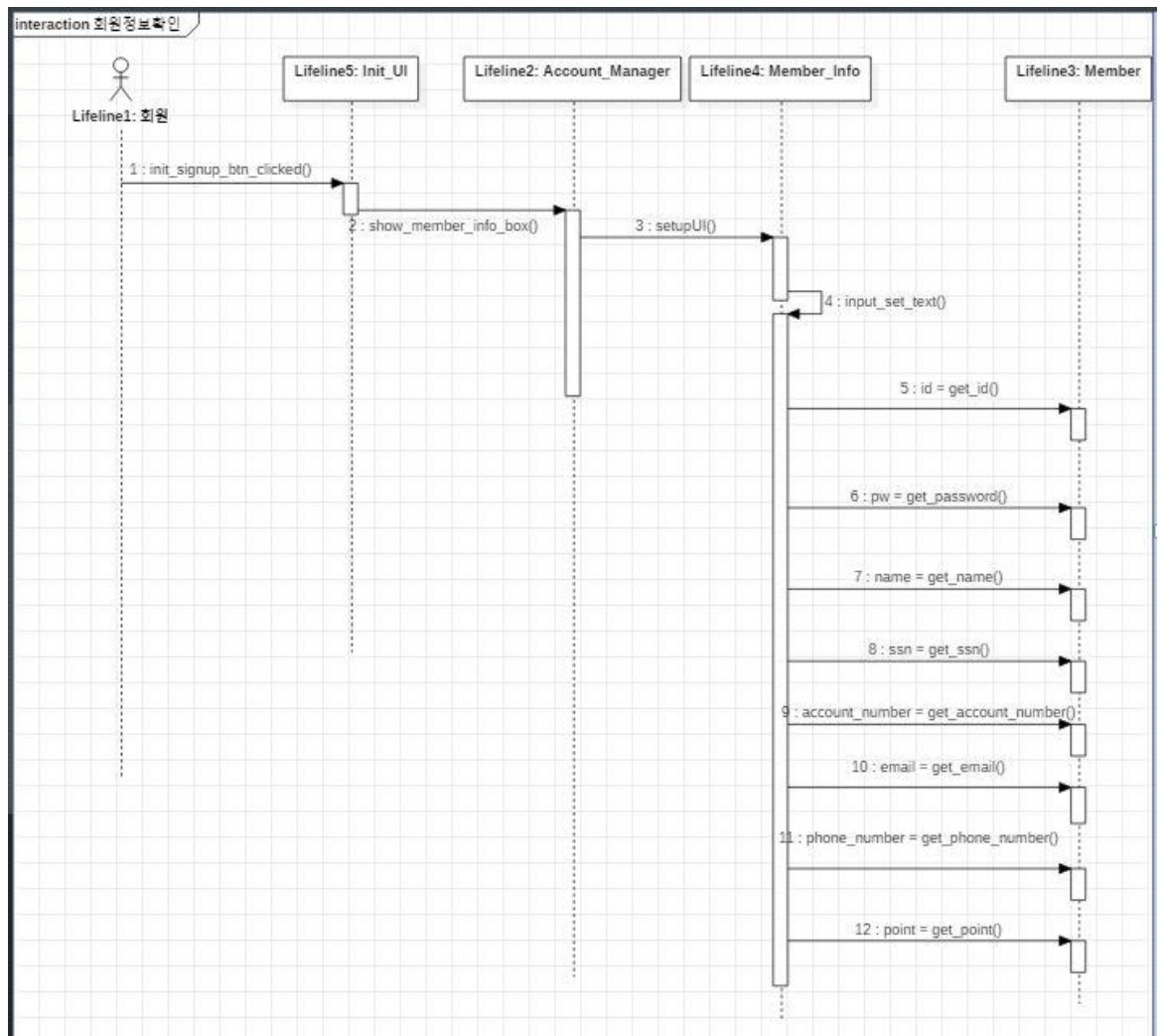
- 로그인



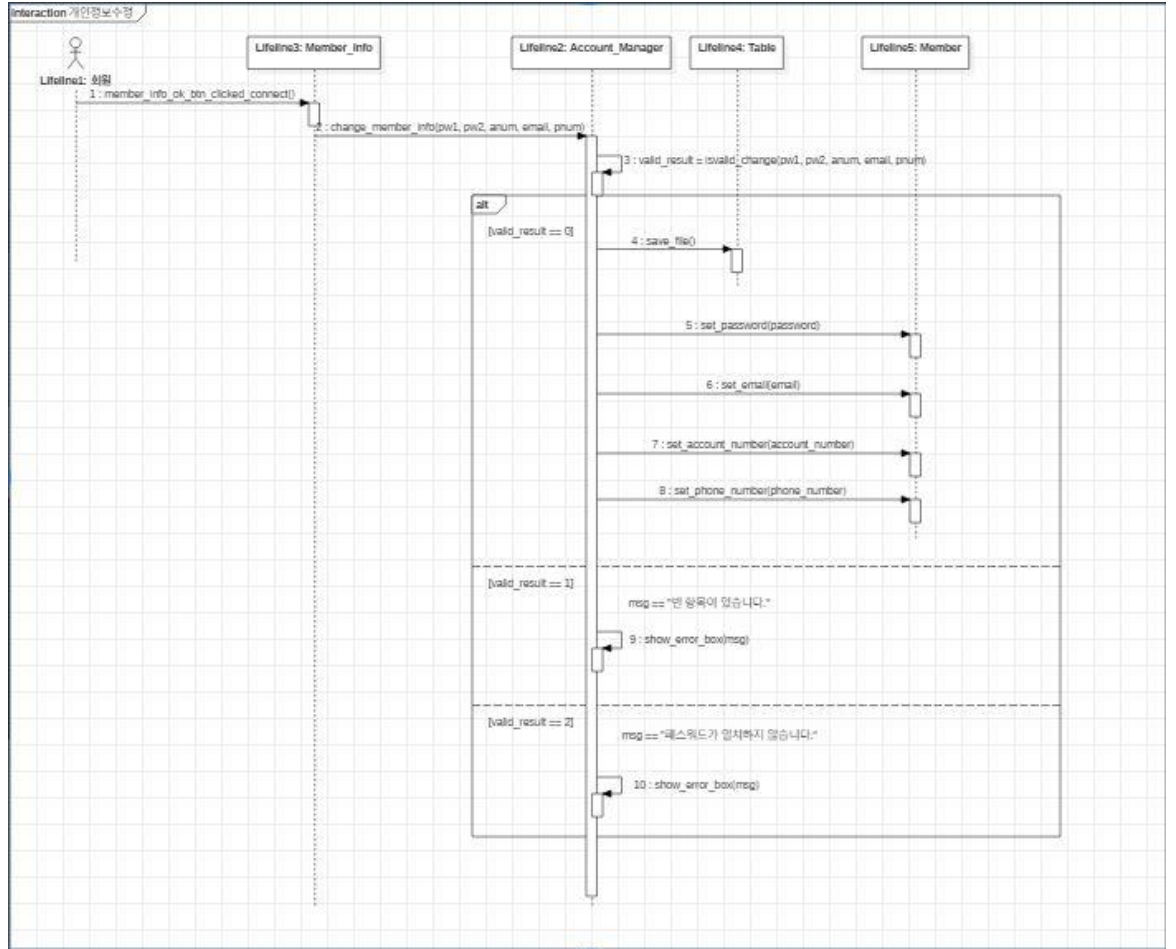
- 로그아웃



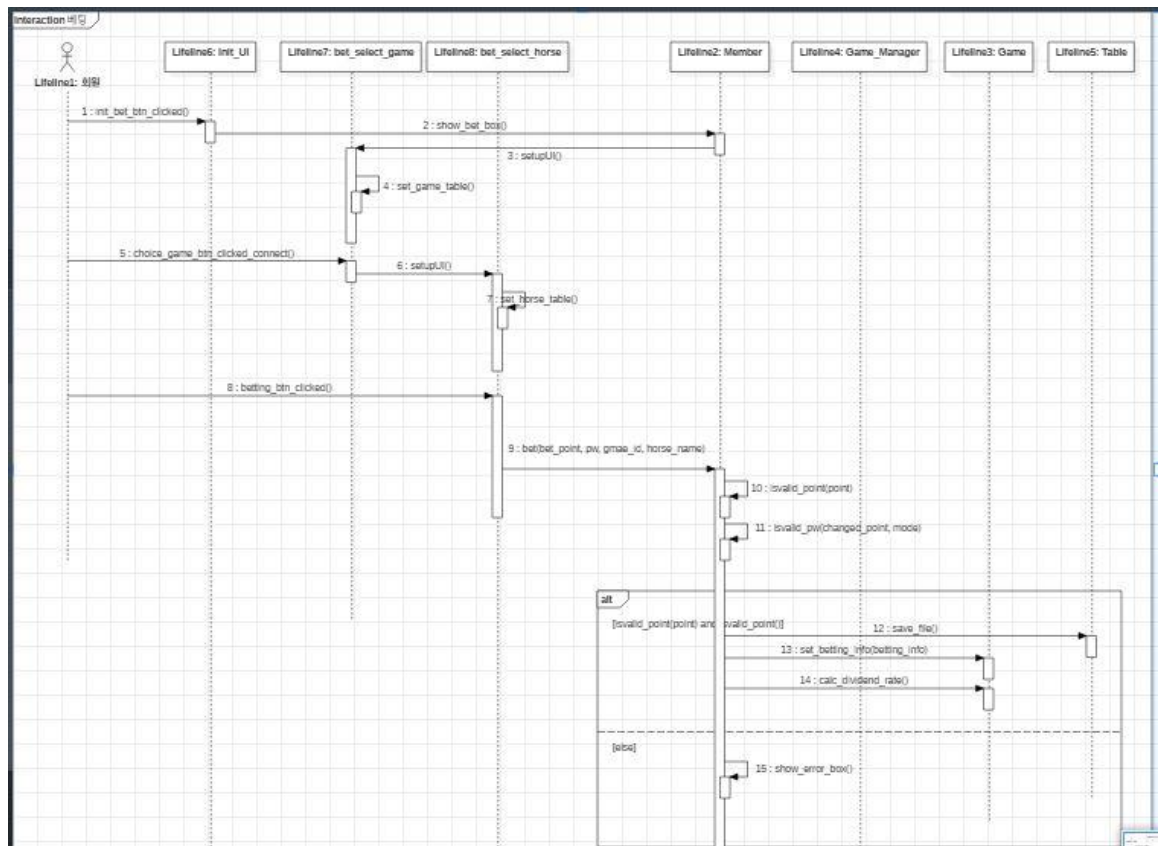
- 회원정보확인



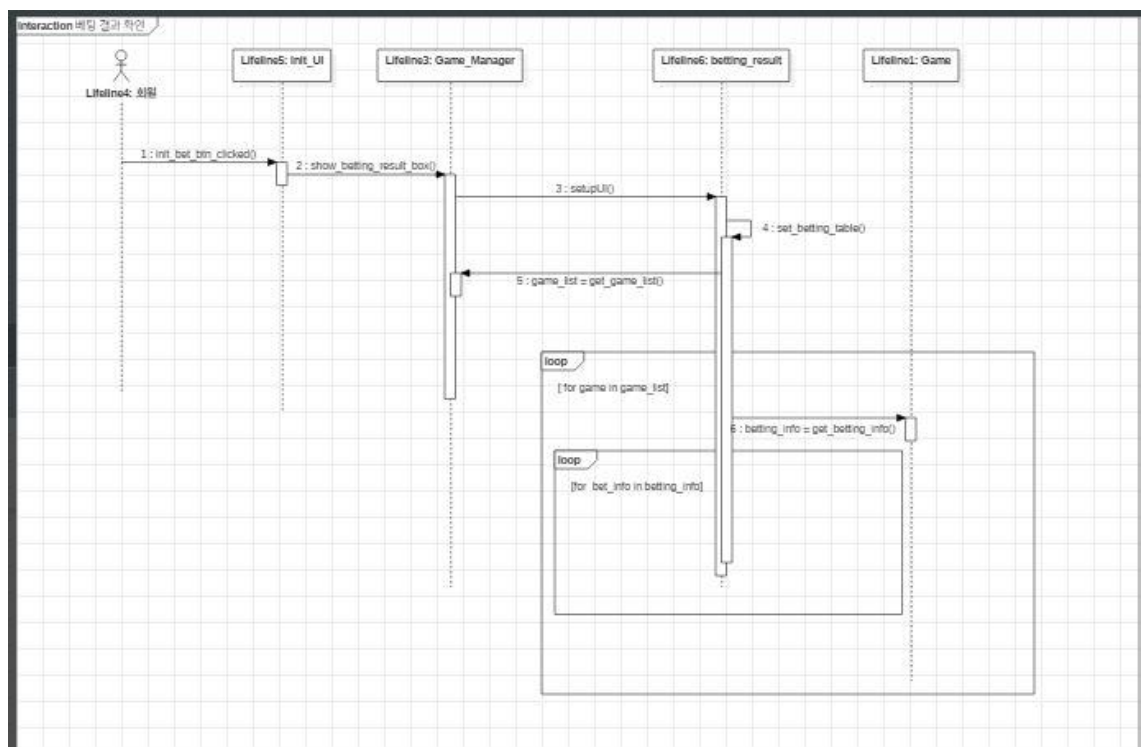
- 개인정보수정



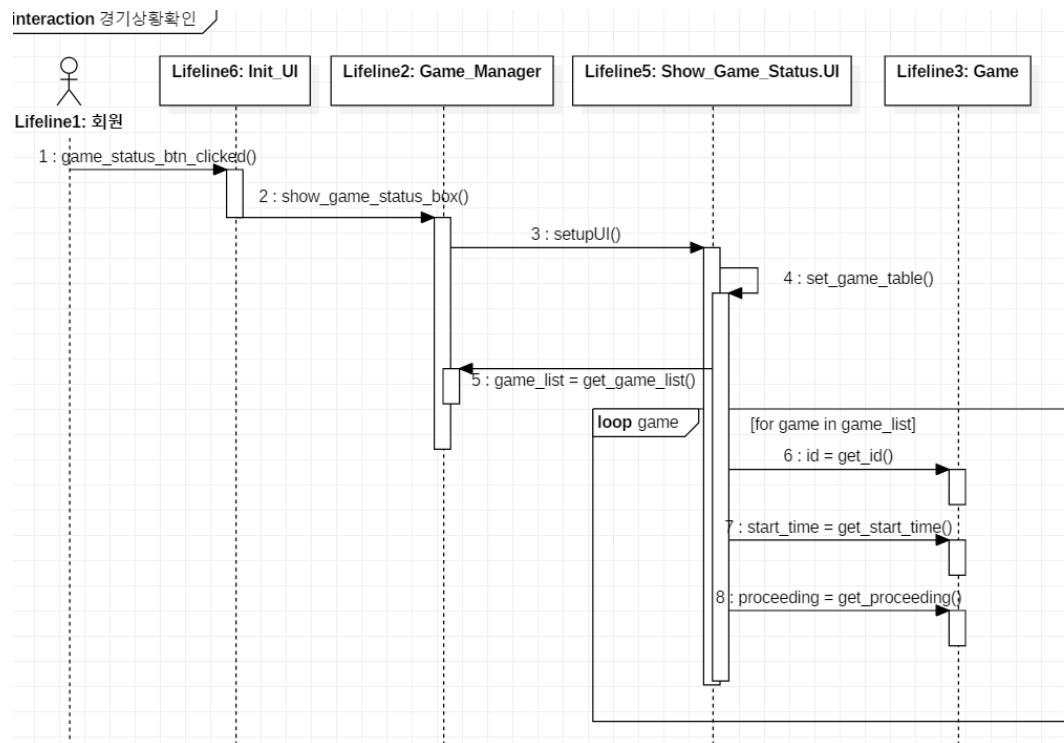
- 베팅



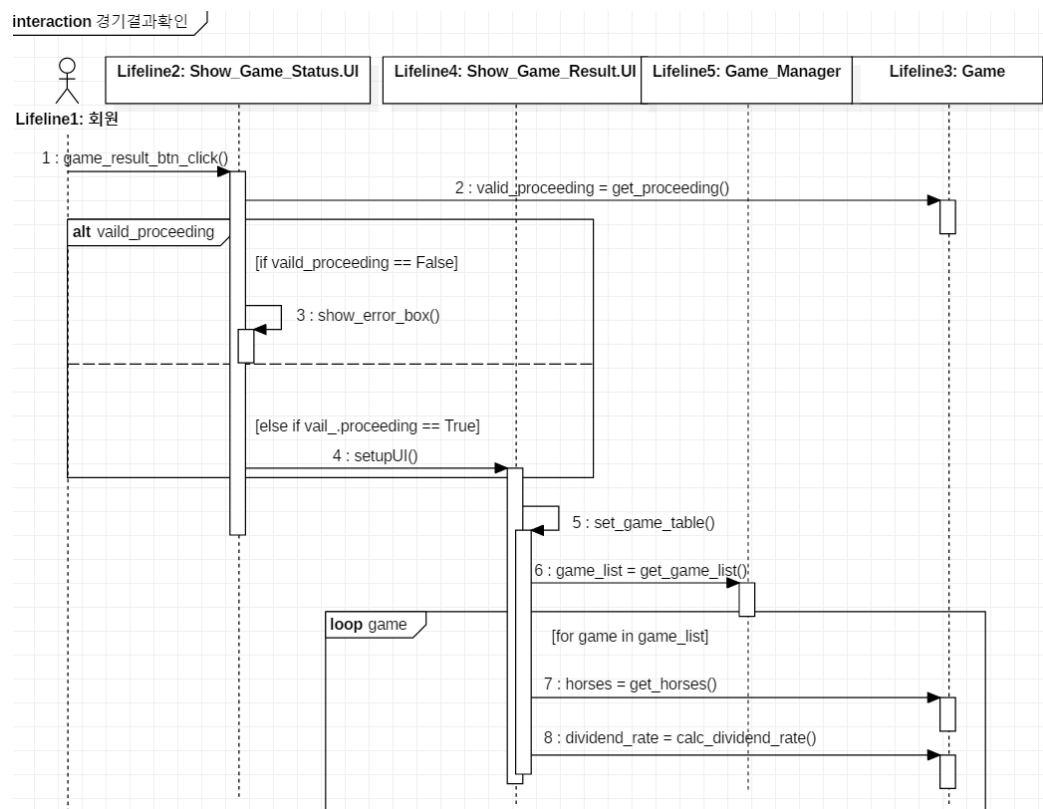
- 베팅 결과 확인



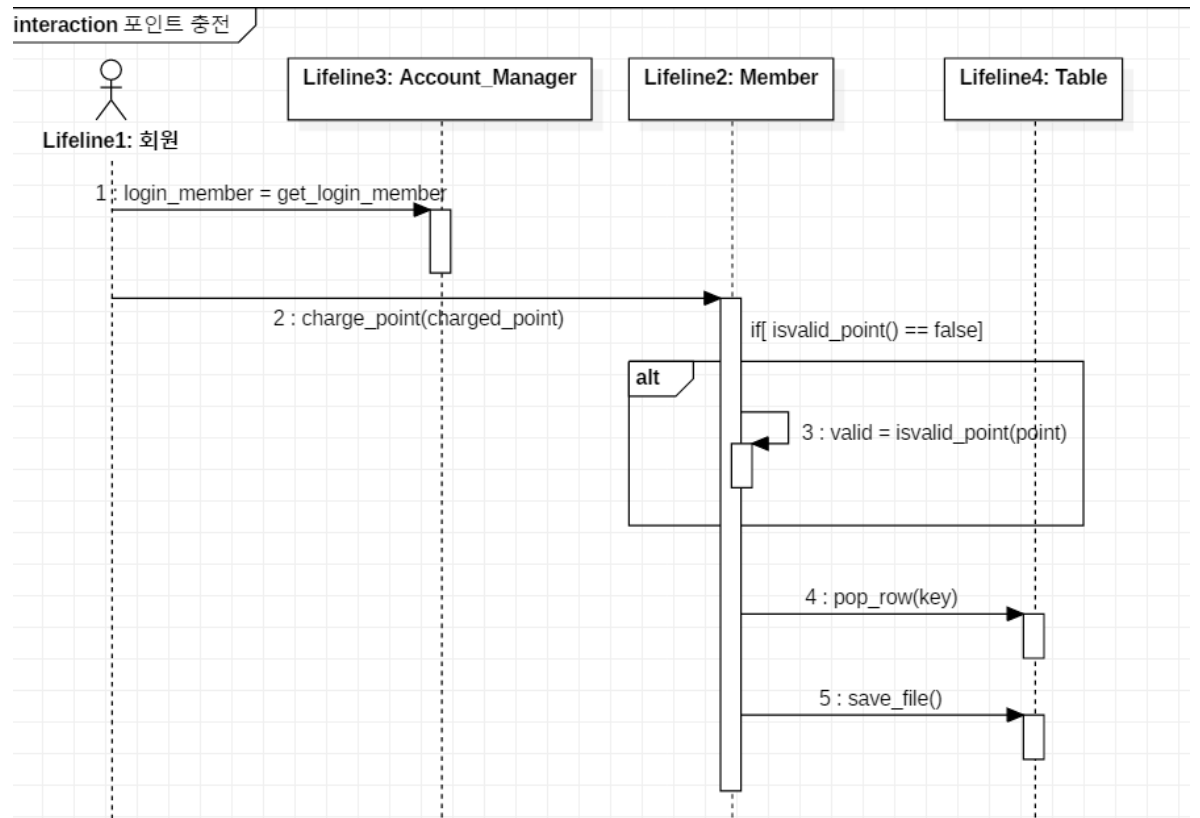
- 경기 상황 확인



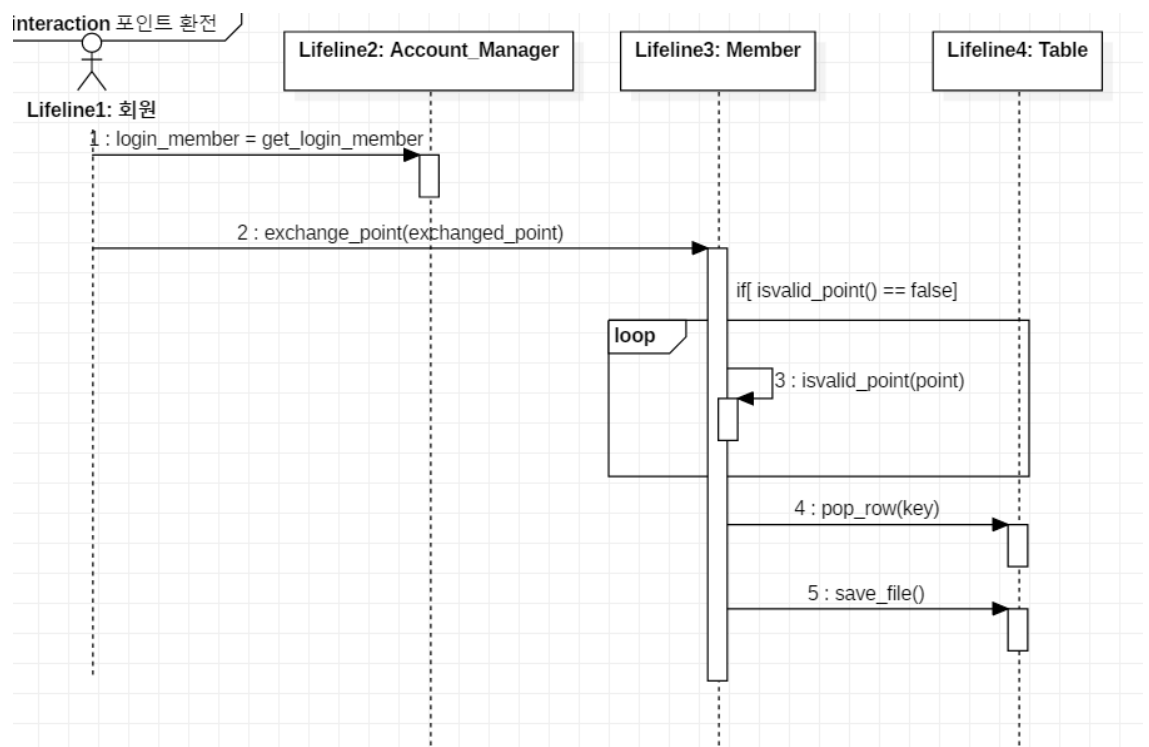
- 경기 결과 확인



- 포인트 충전

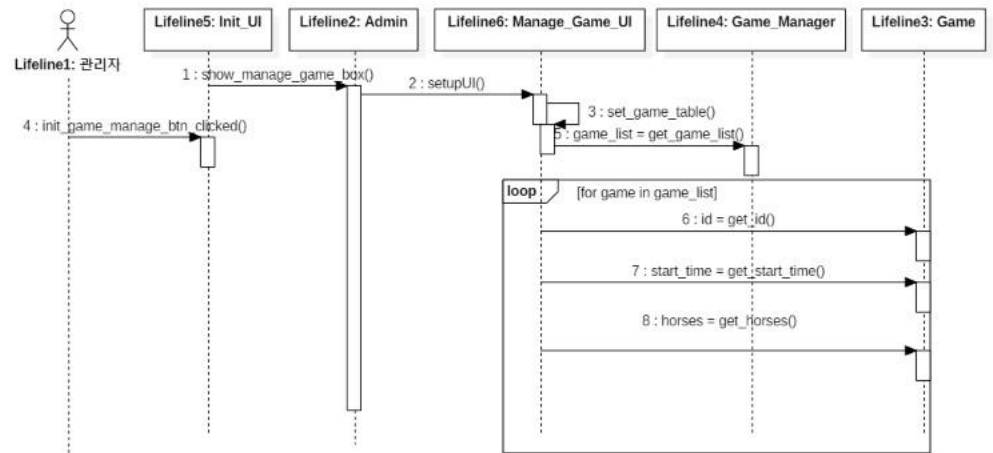


- 포인트 환전



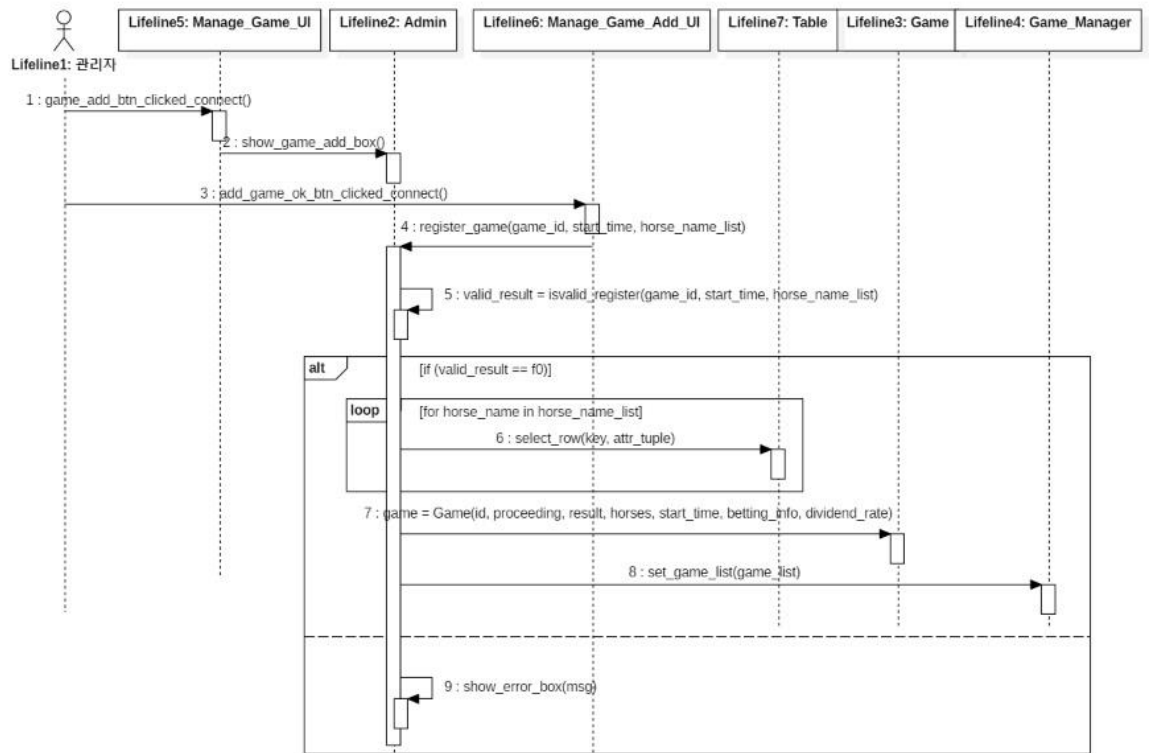
- 경기 확인

interaction 경기 확인

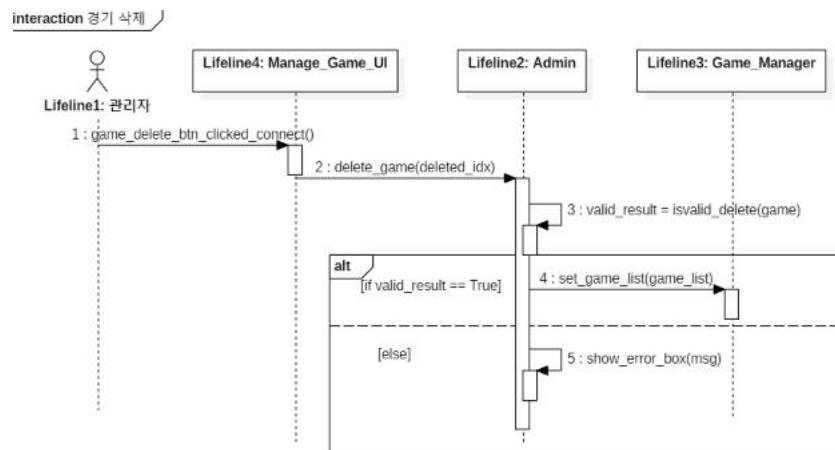


- 경기 추가

interaction 경기 추가

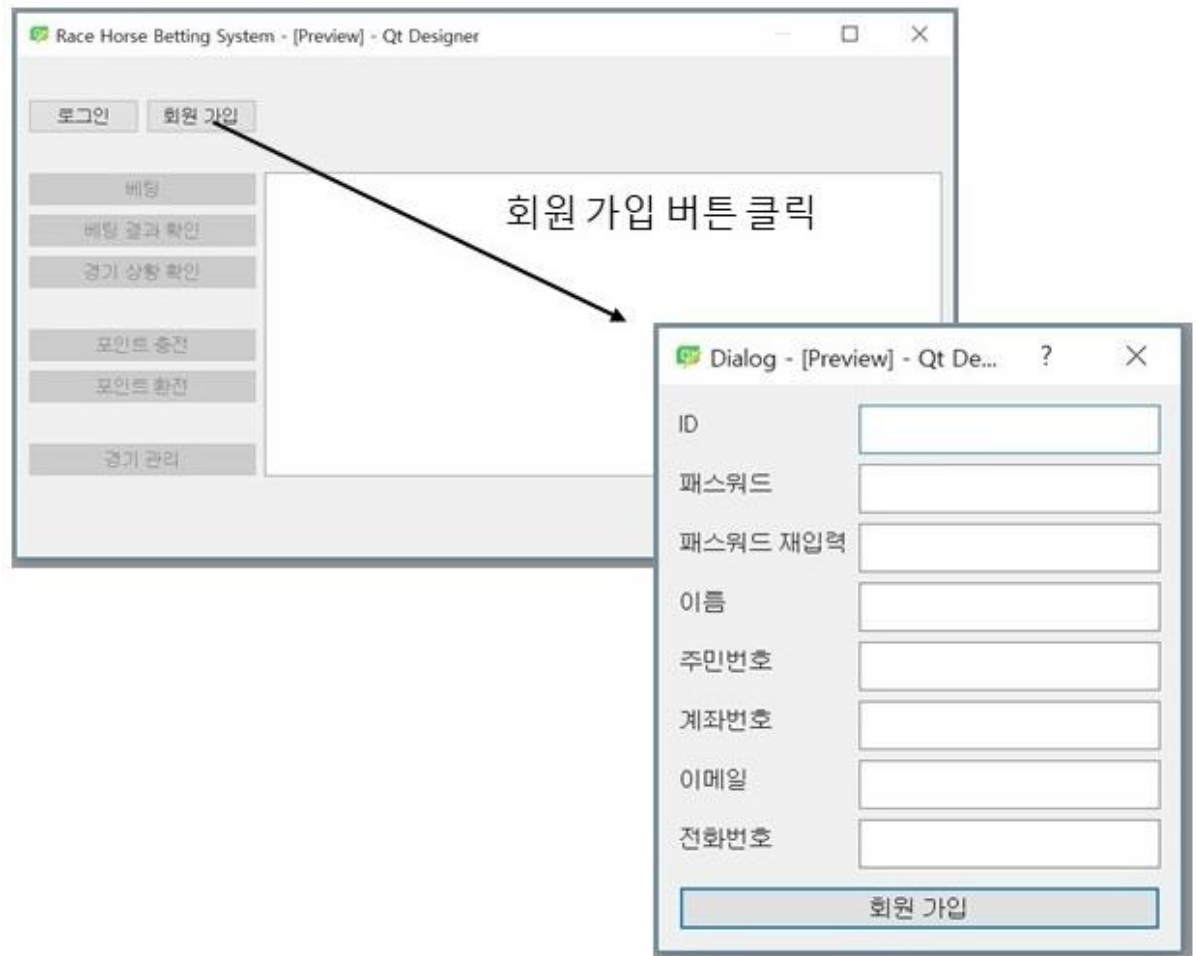


- 경기 삭제

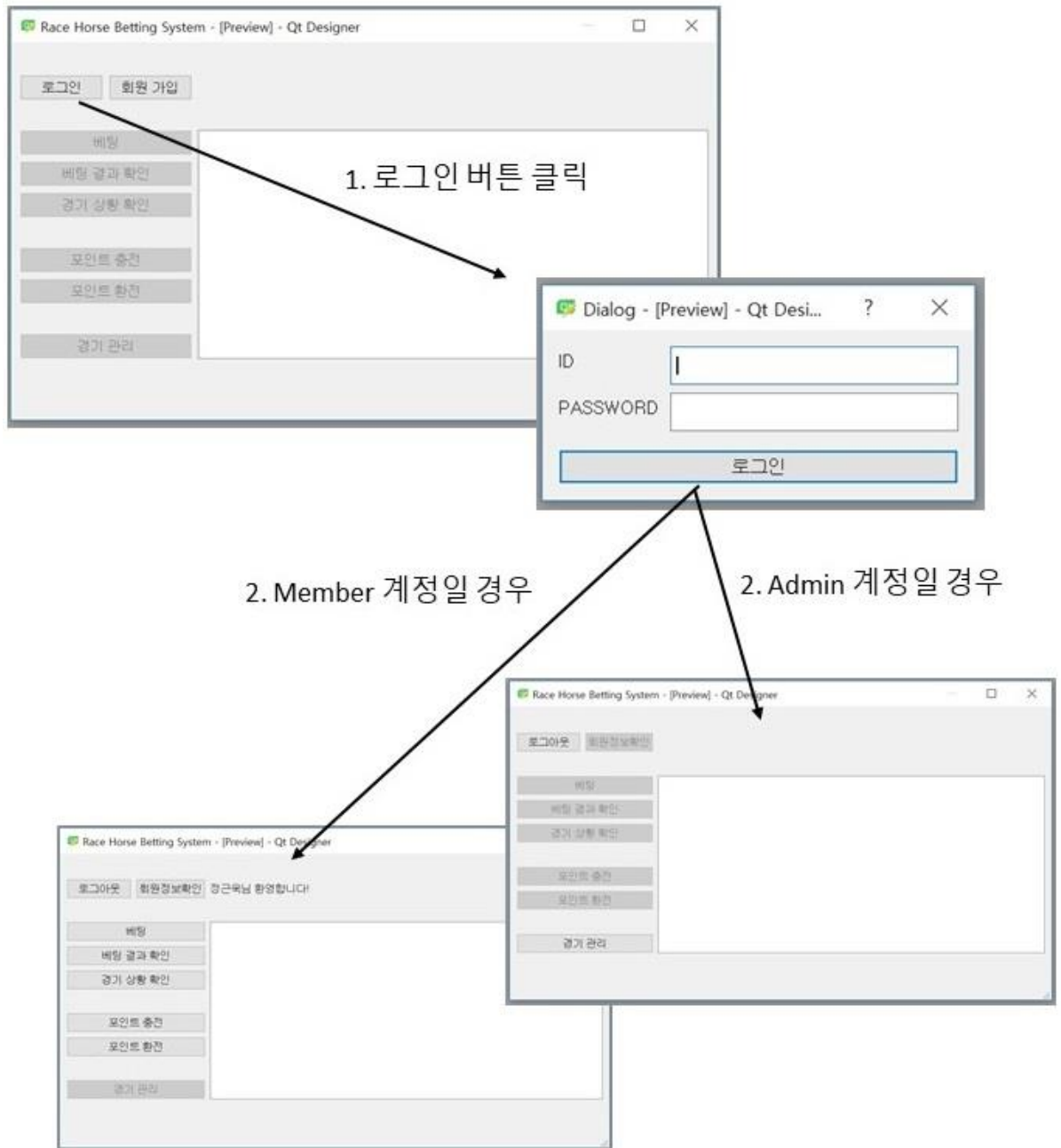


3. 인터페이스 설계

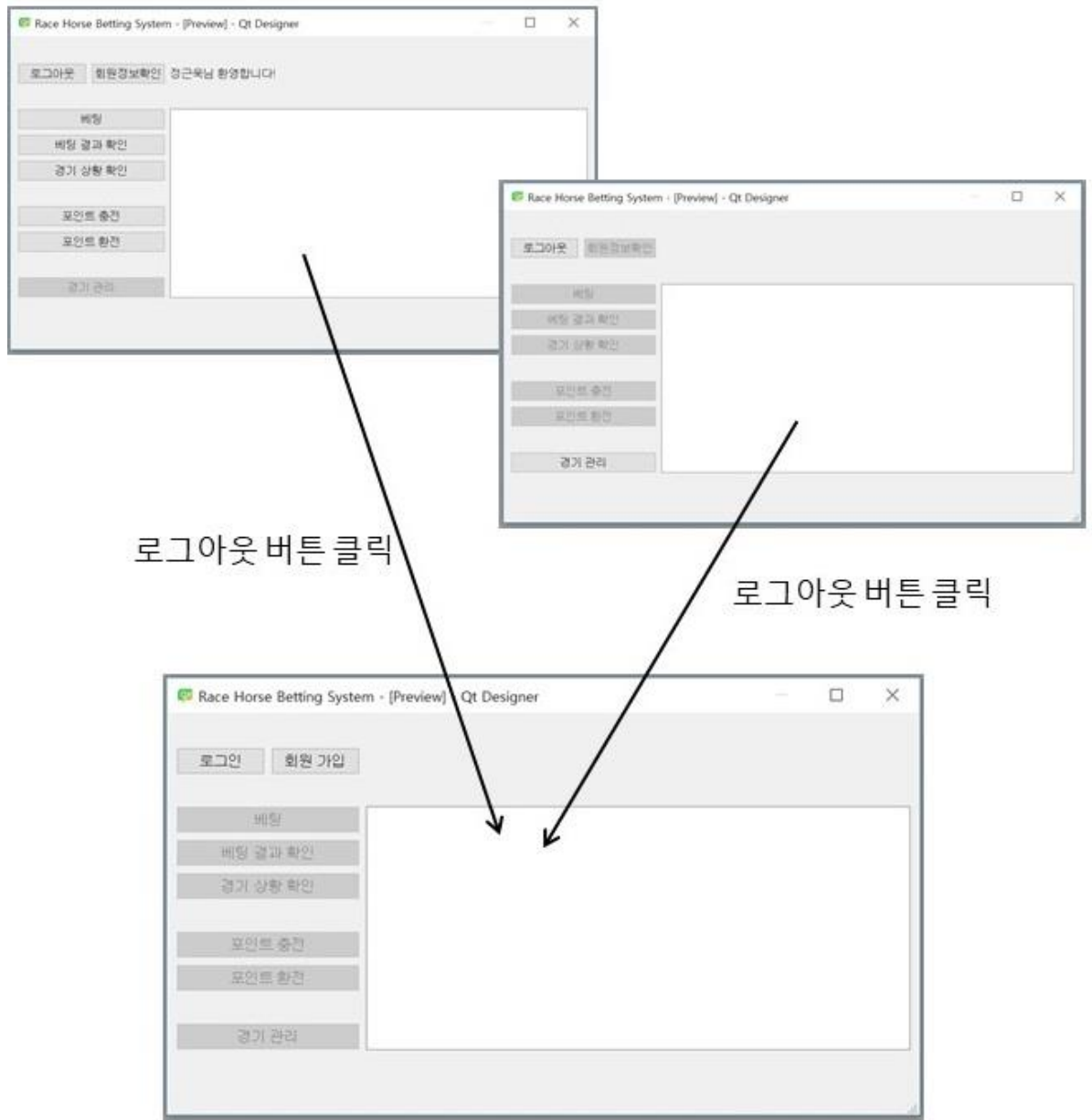
- 회원가입



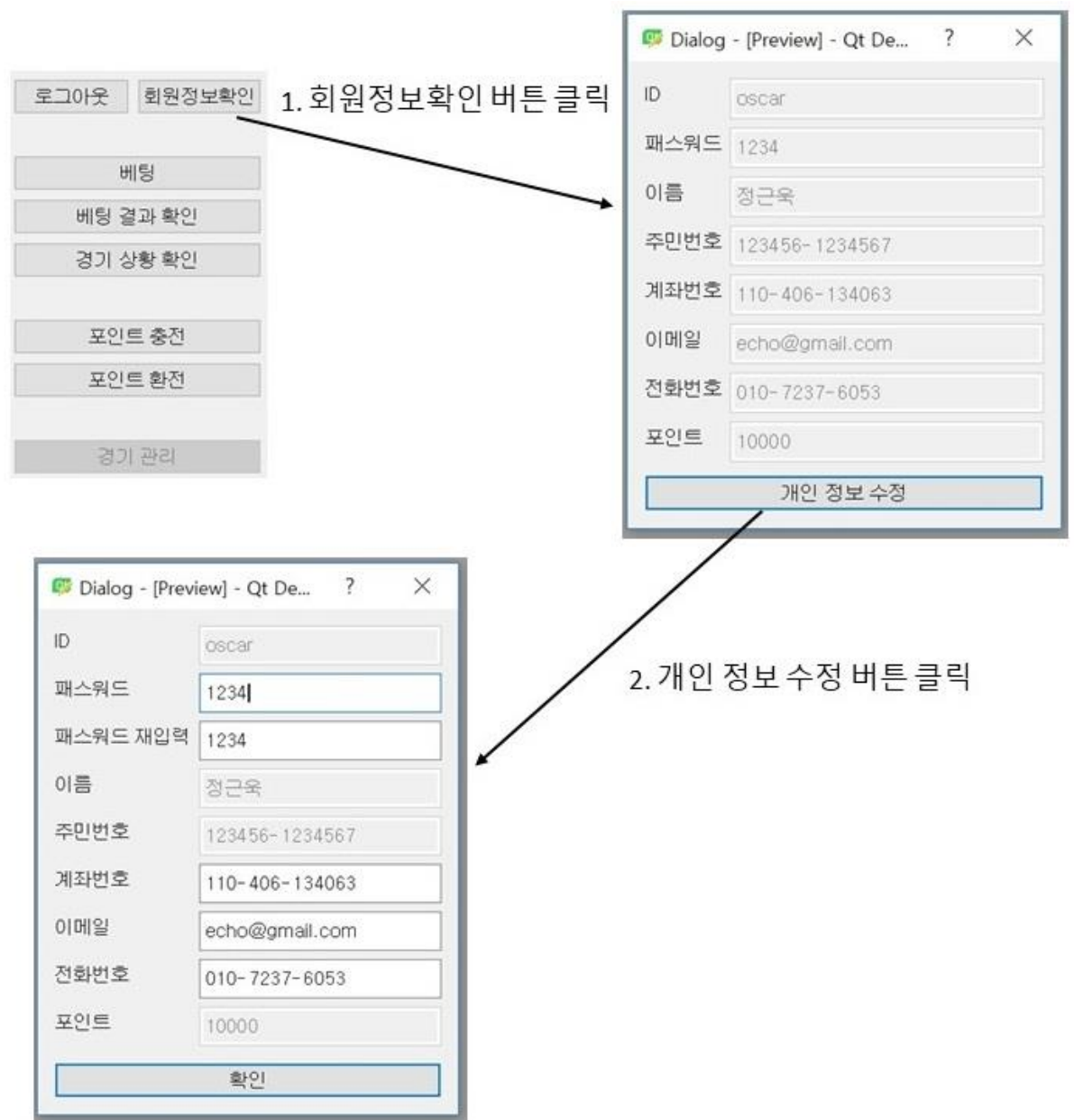
- 로그인



- 로그아웃



- 회원정보확인, 개인정보수정



- 베팅

1. 베팅 버튼 클릭

2. 경기 선택

3. 말, 베팅 금액, 비밀번호 입력

경기	시작 시간
1 경기1	2시
2 경기2	3시
3 경기3	4시

말 이름	소개	배당률
1 Apple	당근을 좋아함	1.3
2 Bravo	다리가 굵음	3.6
3 Charlie	얼굴이 잘생김	1.7

베팅 금액:

패스워드 재입력:

베팅

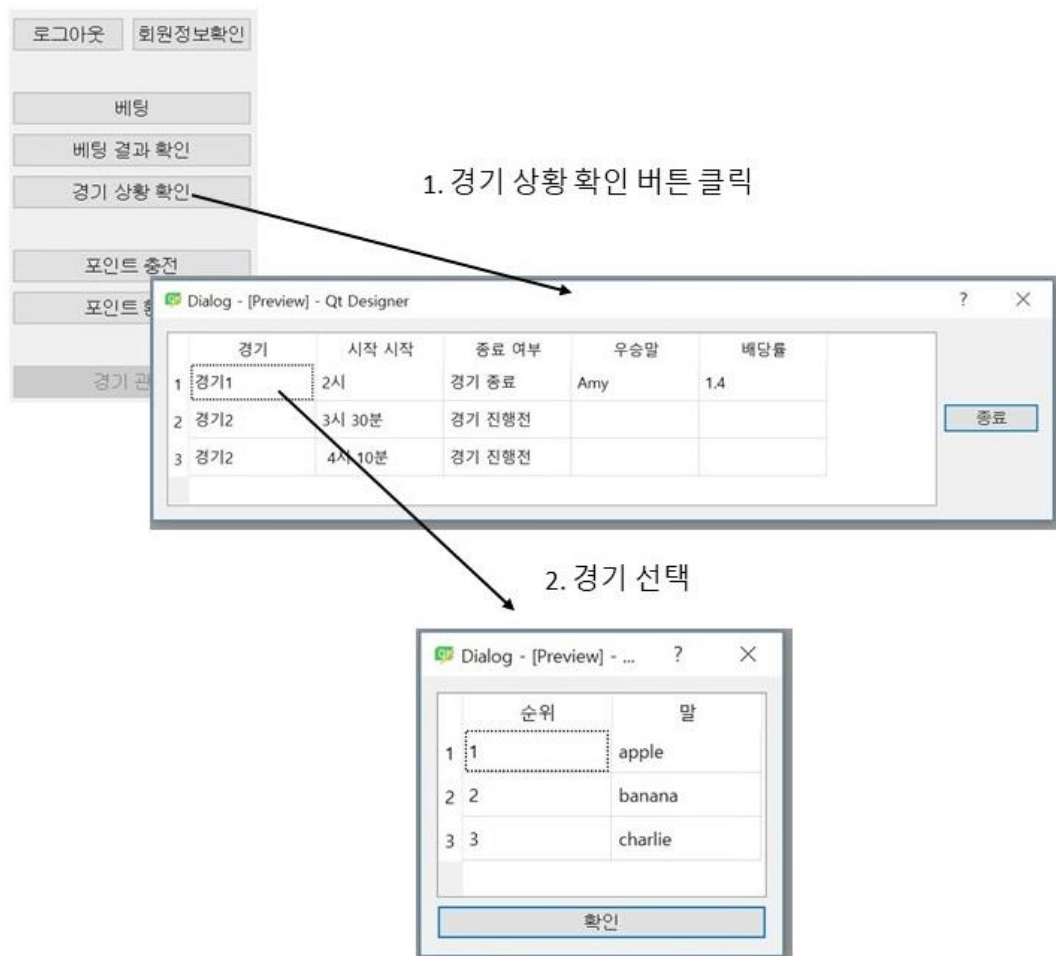
- 베팅 결과 확인

베팅 결과 확인 버튼 클릭

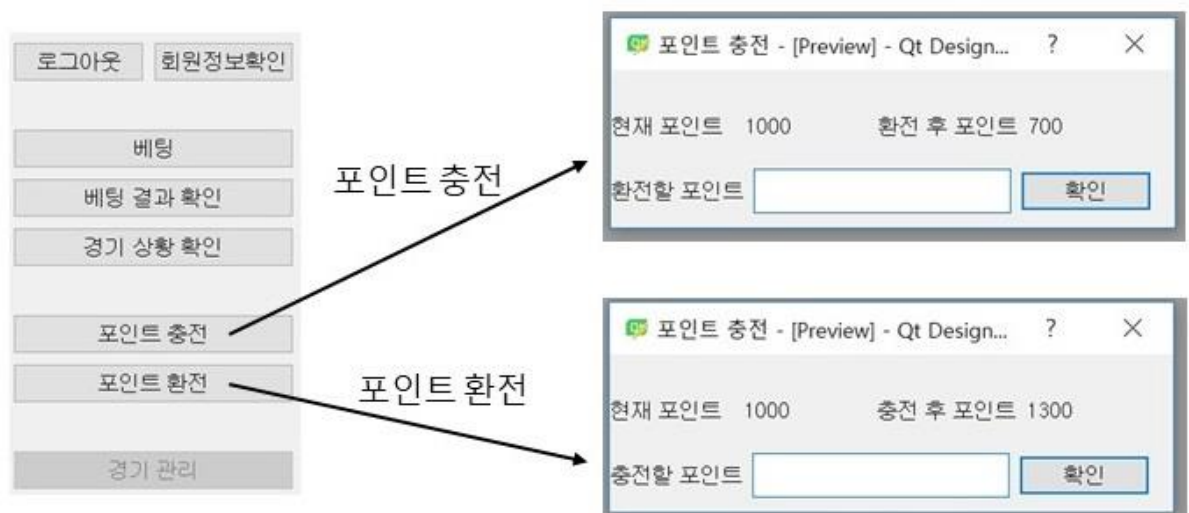
경기	말 이름	종료 여부	성적	배당률	베팅 금액	배당 금액
1 경기1	Apple	경기 전		1.34	100	0
2 경기2	Banna	경기 전		1.2	200	0
3 경기3	Charlie	경기 종료	1	1.6	3000	200
4 경기4	Delta	경기 종료	2	2.6	400	0

확인

- 경기 상황 확인, 경기 결과 확인



- 포인트 충전, 포인트 환전



- 경기 확인, 경기 추가, 경기 삭제



1. 경기 관리 버튼 클릭

2. 경기 추가 버튼 클릭



2. 경기 삭제 버튼 클릭

4. 데이터 설계

회원 정보	
member_id	str
password	str
name	str
ssn	str
account_number	str
email	str
phone_number	str
point	int

베팅 정보	
member_id	str
game_id	str
horse_name	str
bet_money	int

말 정보	
name	str
feature	str
rating	int

게임 정보	
game_id	int
start_time	datetime
result	lit of int
horses	list of Horse 객체
dividend_rate	list of float
proceeding	int

데이터는 파이썬의 pickle 라이브러리를 사용해 파일 형태로 저장