

Console  
Game

콘솔 프로젝트

# One Card

2025.10.10

윤종수

---

# 목 차

## 1. 게임 설명

기본 룰  
공격 카드  
특수 카드

## 2. 구성 요소

플로우 차트  
Play Manager  
Player  
Card Deck

## 3. 활용 기술

자료구조  
Etc...

# 게임 설명

1

모든 플레이어는 카드를 각각 7장 씩 나눠 가진다.

2

플레이어는 자신의 턴에 카드를 1장 내거나, 카드를 뽑을 수 있다.

단, 카드는 가장 최근 내려 놓은 카드와 문양 또는 숫자가 같은 카드만 낼 수 있다.

단, 플레이어는 손에 든 카드가 20장을 넘어가면 패배한다.

3

플레이어가 카드를 냈을 때 남은 카드가 한 장이라면, 원 카드를 선언할 수 있다. 이때, 다른 플레이어가 먼저 원카드를 외치면 원카드를 선언해야 하는 플레이어는 카드를 한 장 뽑는다.

---

# 공격 카드

게임 설명

*Card 2*

숫자 2가 적힌 카드를 낼 시  
에는 다음 턴의 사람이 2장을  
뽑는다.

*Card A*

A가 적힌 카드를 낼 시에는  
다음 턴의 사람이 카드 3장을  
뽑는다.  
단, 스페이드 A는 5장을 뽑는  
다.

*Joker*

Joker 카드를 낼 시에는 다  
음 턴의 사람이 카드를 뽑는  
다.  
흑백 Joker는 5장,  
컬러 Joker는 7장이다.

# 공격 카드 막기기

*Card 2*

숫자 2는 모든 공격 카드로  
막을 수 있다.

*Card A*

A 또는 Joker로 막을 수 있  
다.

단, 스페이드 A는 다른 A로  
막을 수 없다.

*Joker*

Joker는 Joker로만 막을 수  
있다.

단, 컬러 Joker는 어느 카드  
로도 막을 수 없다.

공격 카드는 공격 카드로 막을 수 있다.  
막은 공격 카드는 자신의 공격 카드와 함께 다음 턴 사람에게 페널티가 이전 된다.

---

# 특수 카드

게임 설명

Card Q

Q는 플레이어 턴을 반전 시킨다.

Card J

J는 다음 턴을 스킵한다.

Card K

K는 자신의 턴을 한번 더 수행한다.

Card 7

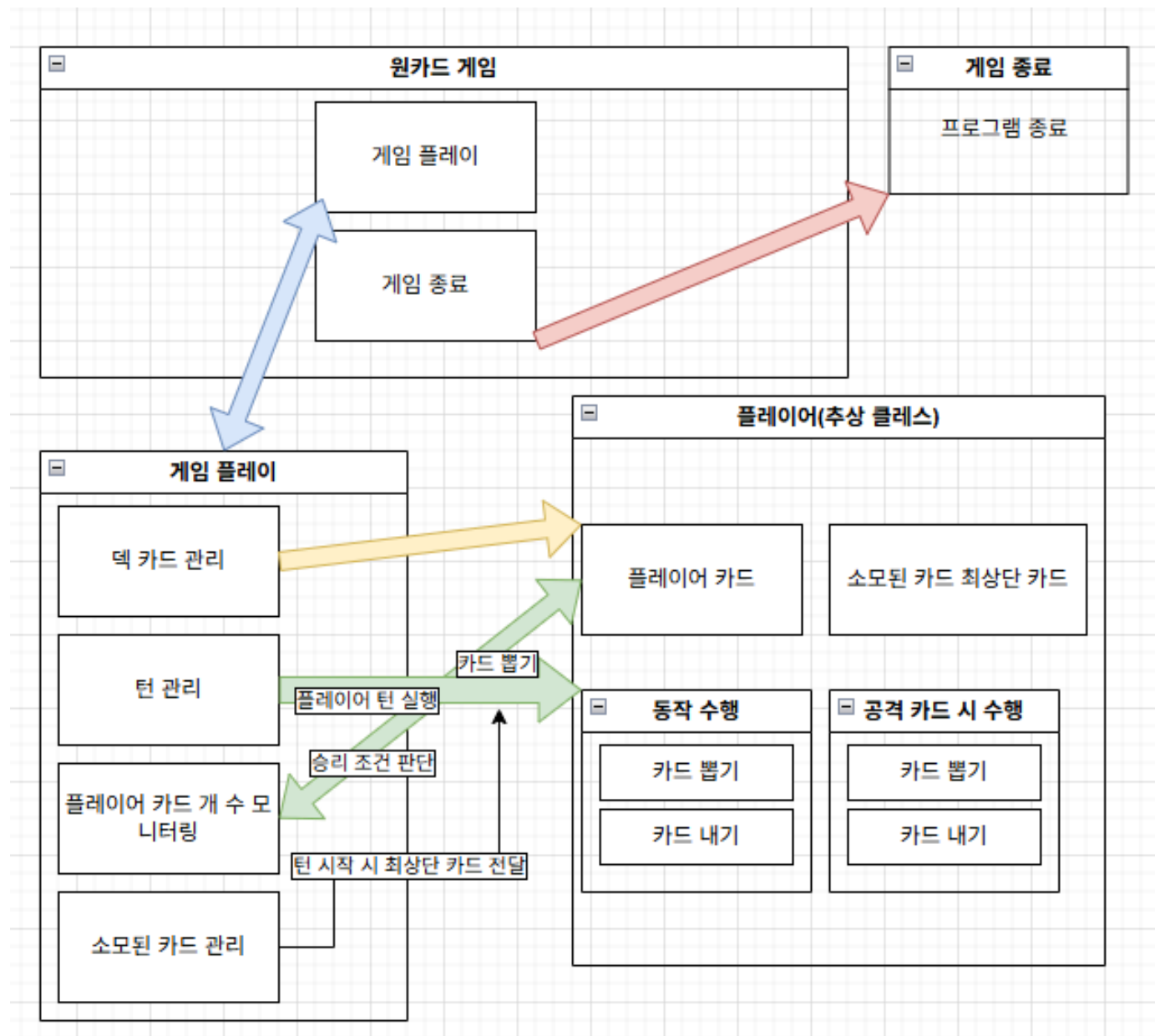
7은 카드의 문양을 자신이 원하는 문양으로 변경한다.

구성 요소



# 개요

구성 요소

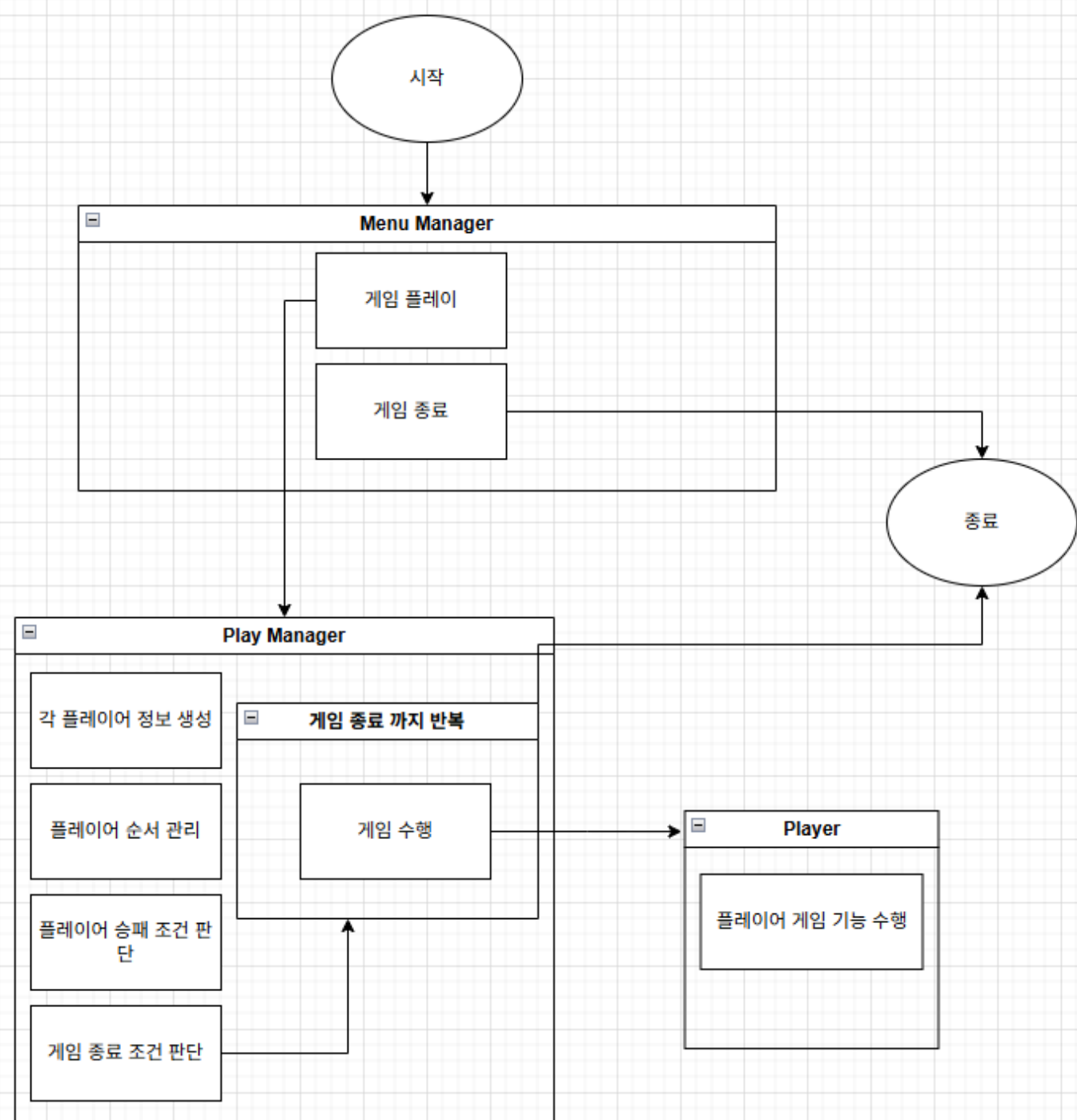


원카드 게임은 크게 3가지로 구성되어 있다.

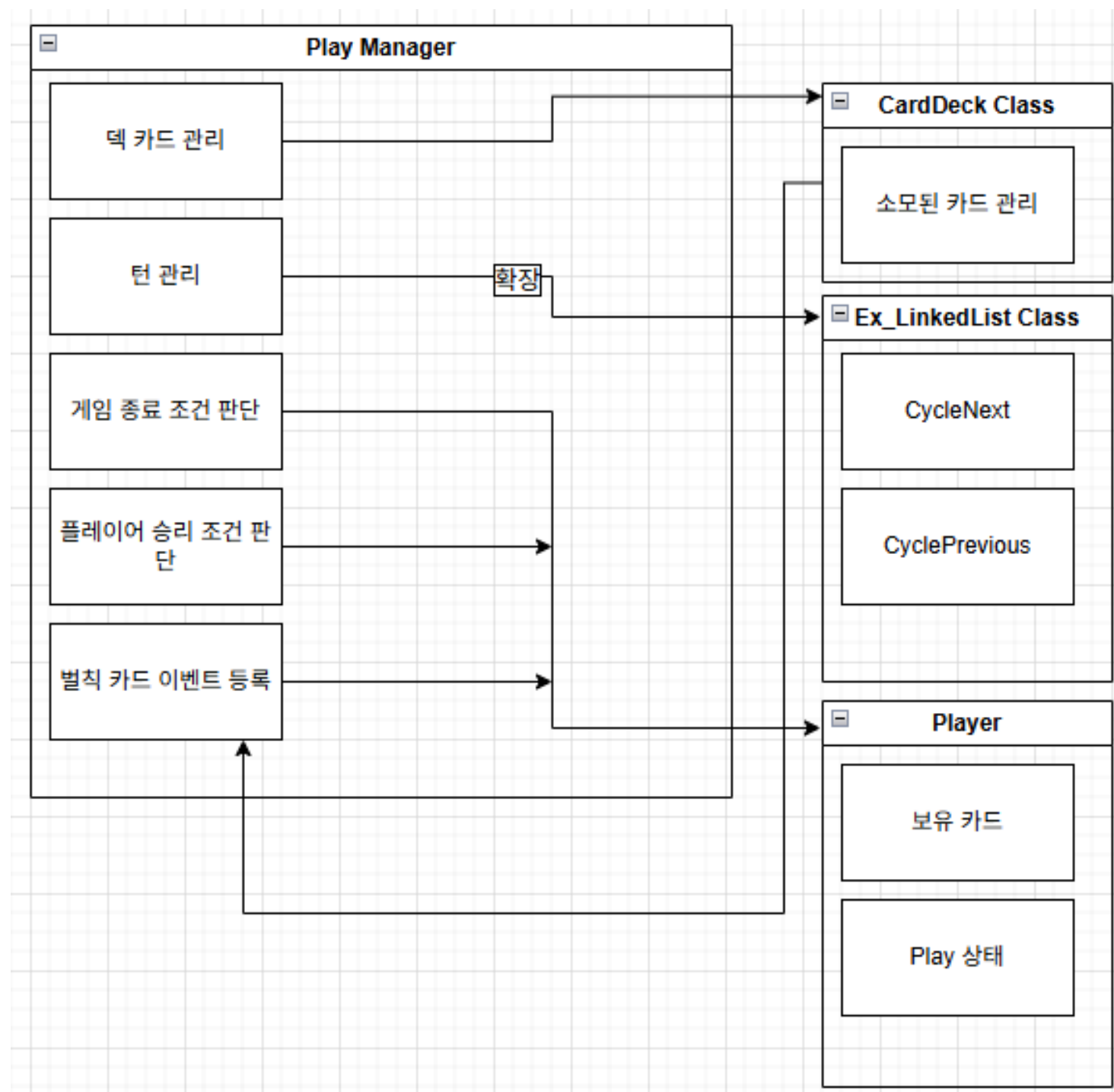
1. 메뉴 (Menu Manager)
2. 게임 플레이 관리자 (Play Manager)
3. 플레이어 (Player)

# 플로우 차트

## 구성 요소



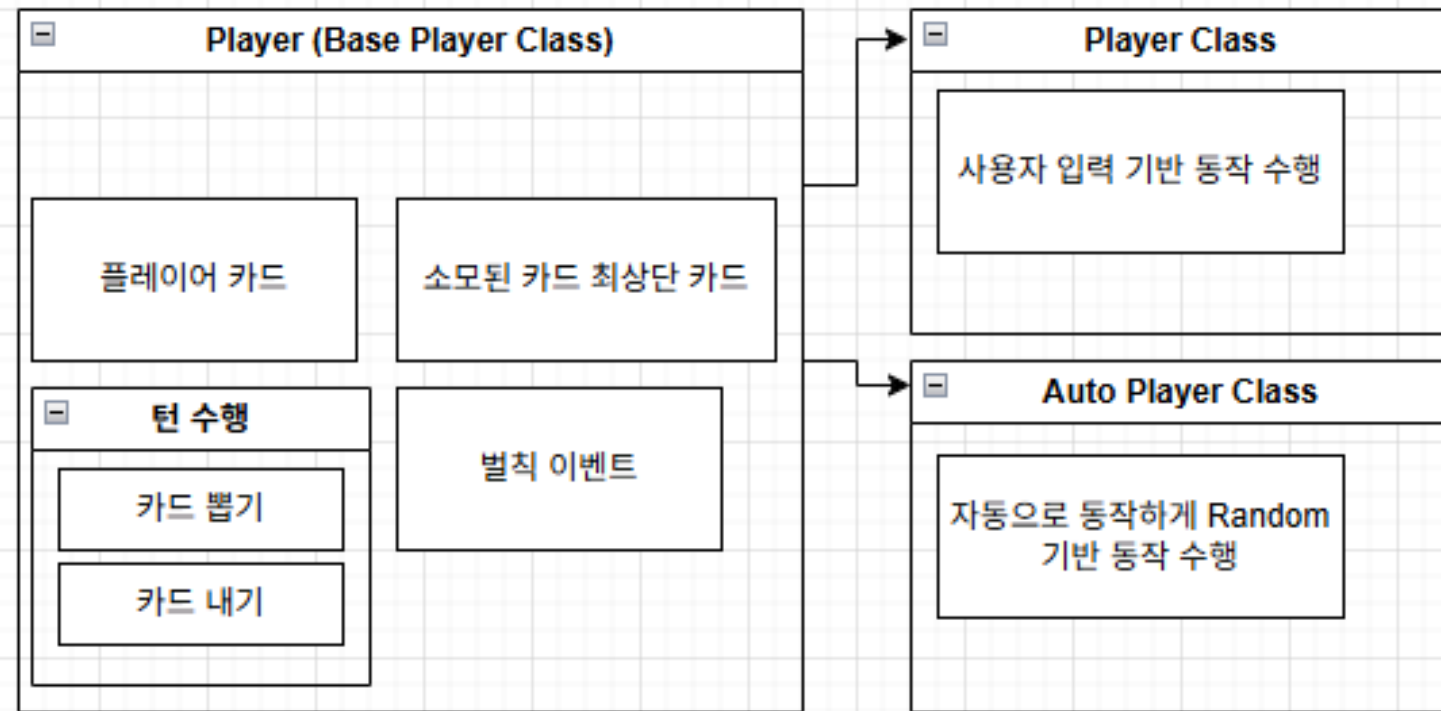
# Play Manager



Play Manager는 게임 플레이에 필요한 기능으로 묶여 있다.

1. 카드 관리 (CardDeck Class에 위임)
2. 턴 관리 (Circular Linked List)
3. 게임 종료 조건 판단
4. 플레이어 승리 조건 판단
5. 공격, 특수 카드 효과 처리

# Player

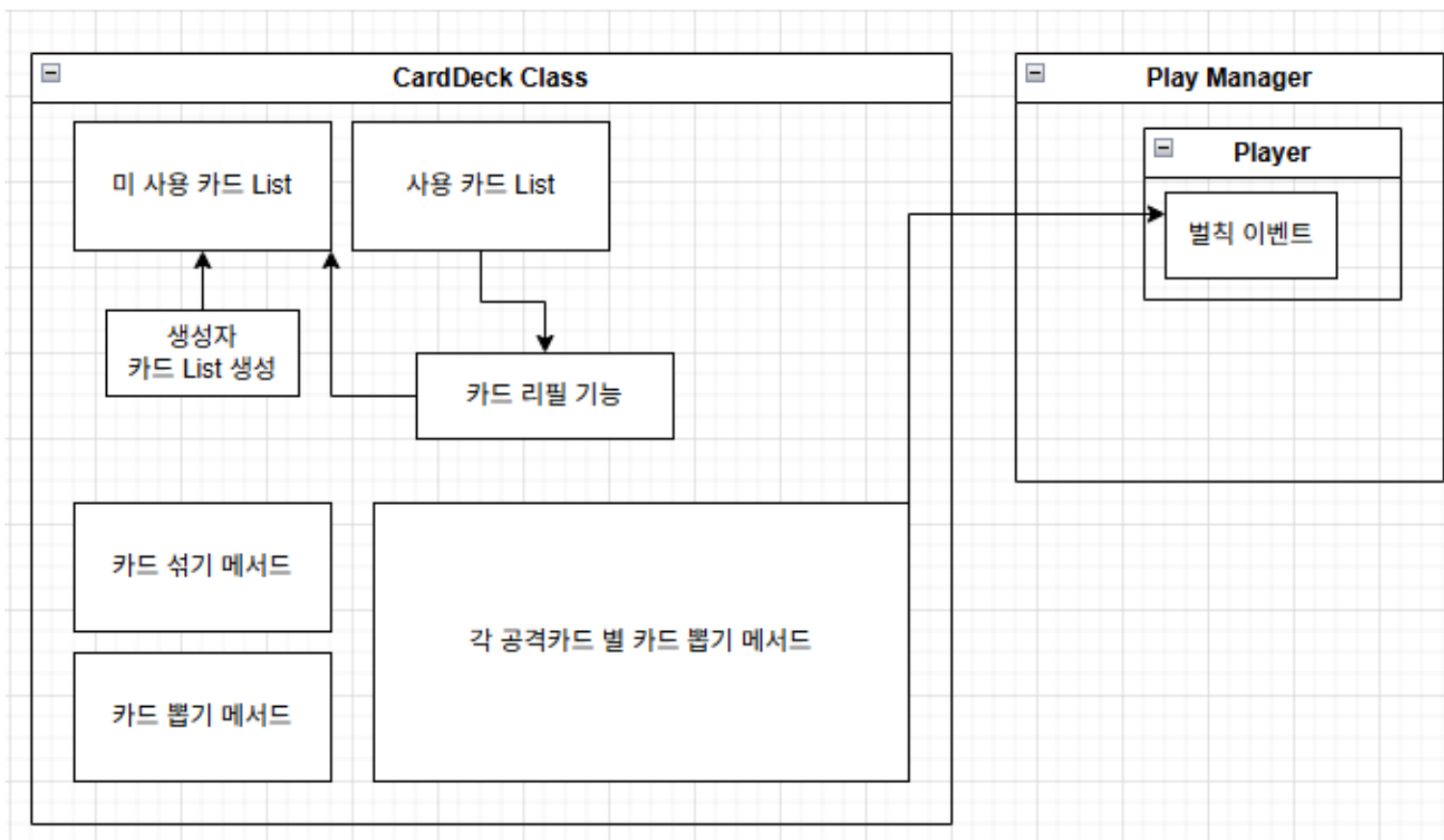


Player 는 각 플레이어 턴에 수행되어야 하는 기능들로 묶여 있다.

1. 턴 수행
2. 보유 카드 관리
3. 최근 사용된 카드 관리
4. 벌칙 수행 이벤트

Player는 게임을 직접 Play하는 사용자와, 가상의 플레이어로 나뉘기에 Base Player로 공통 기능을 묶어 각각 Player Class와 Auto Player Class로 파생되었다.

# Card Deck



Card Deck 은 원카드의 카드들을 관리하는 클래스이다.

- 1.카드 생성
- 2.미 사용 카드 (List)
- 3.사용 카드 (List)
- 4.카드 섞기
- 5.카드 뽑기

Card Deck에서는 각 공격카드 별 카드 뽑기 기능이 존재한다.  
해당 기능들은 공격카드가 사용 될 때, Player Class의 이벤트 변수  
에 등록된다.

물론 그 과정은 Play Manager가 주관한다.

# 활용 기술

1

Player 턴 순서가 꼬이지 않도록,  
Linked List로 player 객체 저장.  
확장 메서드로 원형 연결 리스트를 구현하여 순서가 무한히 반복 될 수 있게 구현.

2

Card 덱을 List로 구현.  
카드 셔플을 사용하여 인덱스로 쉽게 접근 편집이 가능하도록 List로 구현.

3

Queue.  
플레이어가 승리 시 Queue에 등록되어 있던 승리 상태를 뽑아감.  
기존 의도는 Queue가 비었을 때 게임 종료지만, 손패가 20장을 넘을 경우 Out 되는 것 조건 때문에 사실 의미는 없음

Etc...

1

Fisher Yates 셔플 알고리즘  
Card 를 셔플 할 때 사용된 알고리즘

2

Static Class/ Static Method  
객체를 구현할 필요 없는 Display  
manager 구현 시 사용

3

Static Field/Property  
모든 플레이어가 상태값을 공유하는  
LastCard에 사용.  
또한 이를 외부에서 알 수 있도록  
Static property로 구현

4

확장 Method  
Linked List를 원형 연결 구조로 변경  
하기 위해 사용.



*Thank you!*

whdtn218@gmail.com

010-2920-8991

[publish.obsidian.md/lattechicken](https://publish.obsidian.md/lattechicken)

---