Poker Game Requirement

Version 0.1.1

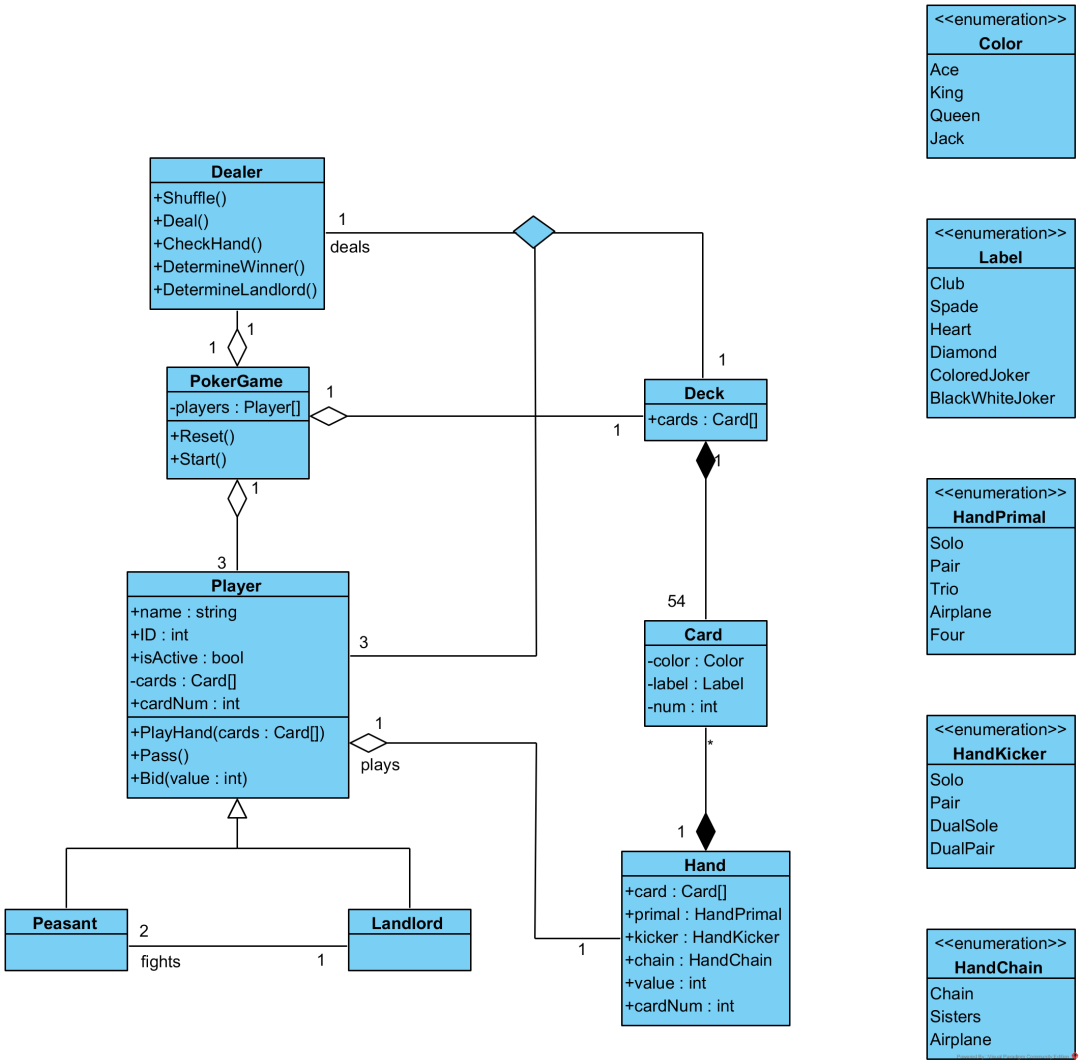
Log

Version	Reviser	Revision Date	Revision Content
0.1.0	Huang Chengyu	2018.4.16	Add the introduction Add the domain analysis UML Add the system composition UML Add the use case UML Add the requirement prioritization
0.1.1	Huang Chengyu	2018.4.21	Add the use case refinement UML Modify the domain analysis UML Modify the system composition UML Modify the use case UML Grade the requirements by stages

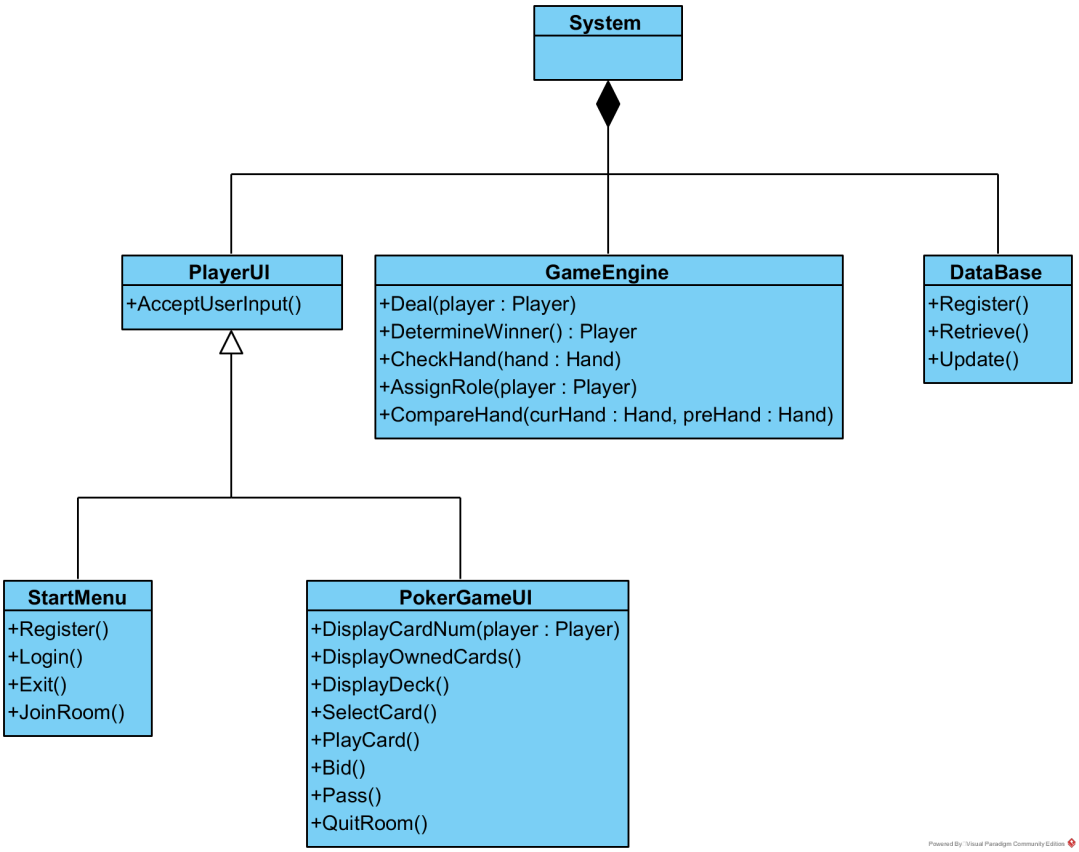
Introduction

Dou dizhu is played among three people with one pack of cards, including the two differentiated jokers. The game starts with players bidding for the "landlord" position. Those who lose the bid or don't bid enter the game as the "peasants" team competing against the landlord. The objective of the game is to be the first player to have no cards left.

Domain Analysis

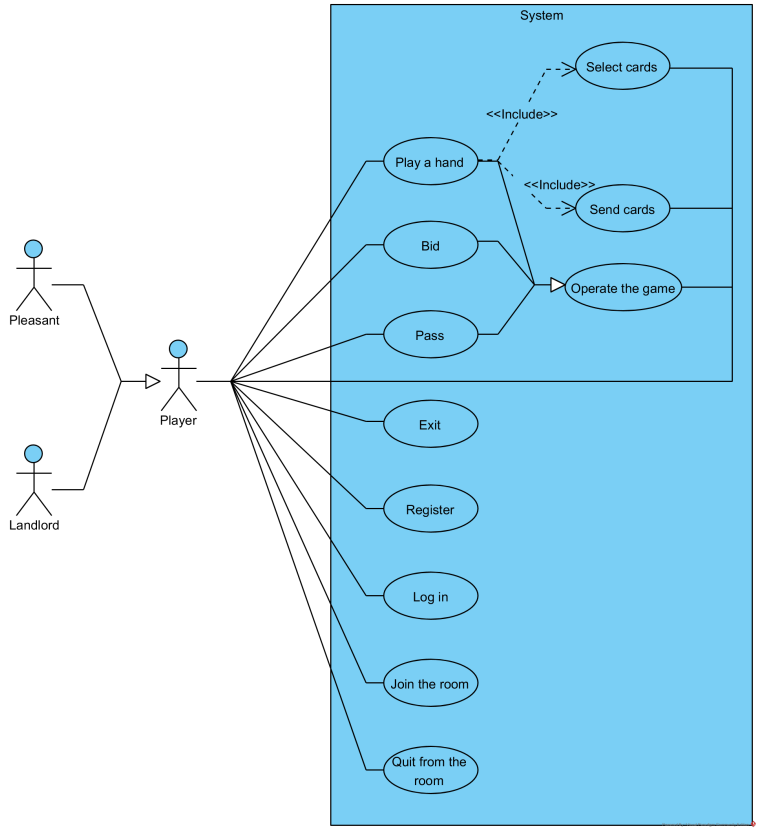


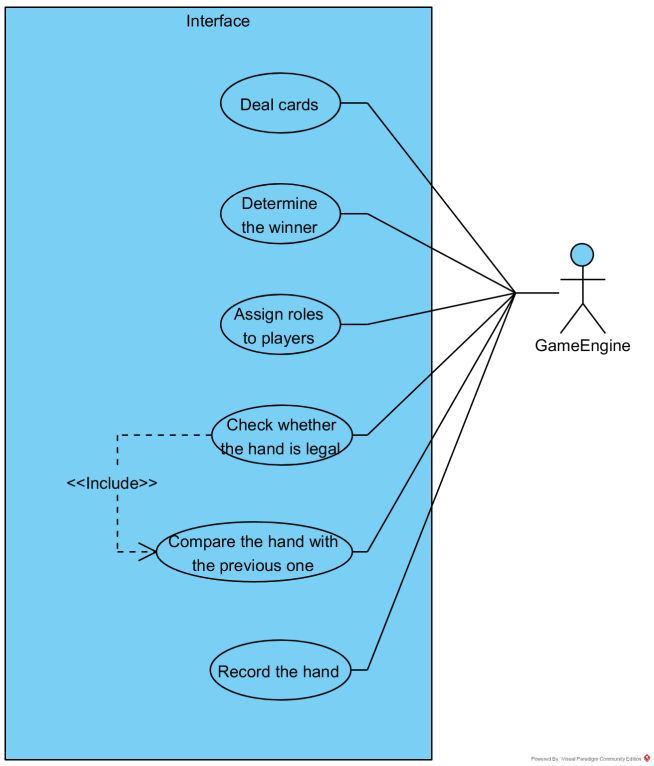
System Composition



- In the system, the game engine is the subsitute for the dealer in the domain analysis.

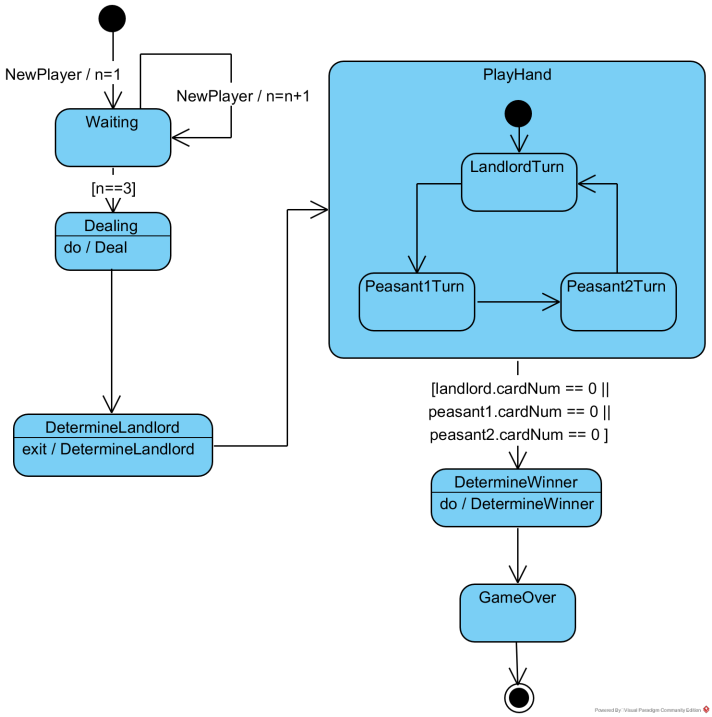
Use Case


System
Select cards
<<Include>>
Play a hand
<<Include>>
Send cards
Bid
Operate the game
Pass
Pleasant
Player
Landlord
Exit
Register
Log in
Join the room
Quit from the room


Interface
Deal cards
Determine the winner
Assign roles to players
Check whether the hand is legal
<<Include>>
Compare the hand with the previous one
Record the hand
GameEngine

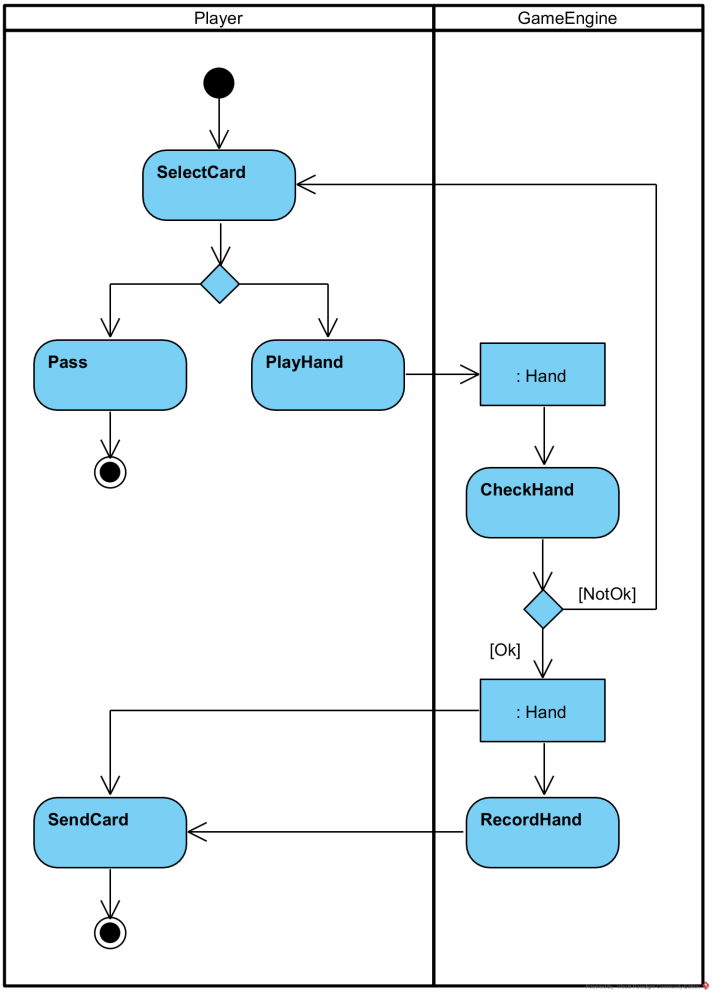
Use Case Refinement

Overall system



- In the DetermineLandlord State, the game engine randomly determines the landlord.
- In the PlayHand State, players take turns to operate in the corresponding substate.
- In the DetermineWinner State, the game engine determines the winner(s) according to the number of cards of each players. If the number of cards of the landlord is zero, the winner is the landlord. If the number of cards of some peasant is zero, the winners are the two peasants.

Operate the game



- The player can choose play a hand or skip the round in the corresponding *PlayHand* state.
- If the player skips the round, he or she should stop operating and the next player should operate.
- If the player plays a hand, the game engine check whether the hand is legal i.e. the hand matches one type of patterns and is greater than the previous one.
- If it is legal, the game engine should record the hand and the cards will be deducted from the player. If not, the player should operate again.

Requirement Prioritization

Stage 0

- Three players can play a poker game
- The game engine can determine the winner
- The game engine can randomly serve card
- The game engine can randomly determine the landlord
- A player can select cards
- A player can play a hand in the corresponding *PlayHand* state

- A player can skip the round in the corresponding *PlayHand* state

Stage 1

- The game engine can check the hand after a player plays a hand
- Players should take turns to operate the game
- Players should operate within certain time limits in the corresponding *PlayHand* state
- A player can bid in the corresponding *Bid* state
- A player can skip the round in the corresponding *Bid* state
- The game engine can determine the landlord according to the result of the bid

Stage 2

- A player can quit from the room when playing the poker game
- A player can register in the start menu
- A player can log in in the start menu
- A player can join a room in the start menu
- A player can exit the app in the start menu