**Лабораторная работа №2  
«Разработка UML-диаграмм с использованием PlantUML»**

**Цель лабораторной работы:** Освоить базовые навыки создания UML диаграмм с использованием языка PlantUML.

**Диаграммы последовательностей:**

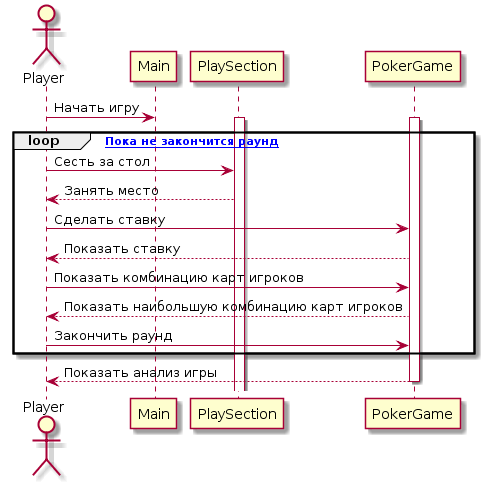
****

Рисунок 1 – Диаграмма последовательности для сценария использования «Незарегистрированный пользователь**»**

Код:

@startuml

actor Player

Player -> Main: Начать игру

activate PlaySection

activate PokerGame

loop [ Пока не закончится раунд]

Player -> PlaySection: Сесть за стол

PlaySection --> Player : Занять место

Player -> PokerGame: Сделать ставку

PokerGame --> Player: Показать ставку

Player -> PokerGame: Показать комбинацию карт игроков

PokerGame --> Player: Показать наибольшую комбинацию карт игроков

Player -> PokerGame: Закончить раунд

end

PokerGame --> Player: Показать анализ игры

deactivate PokerGame

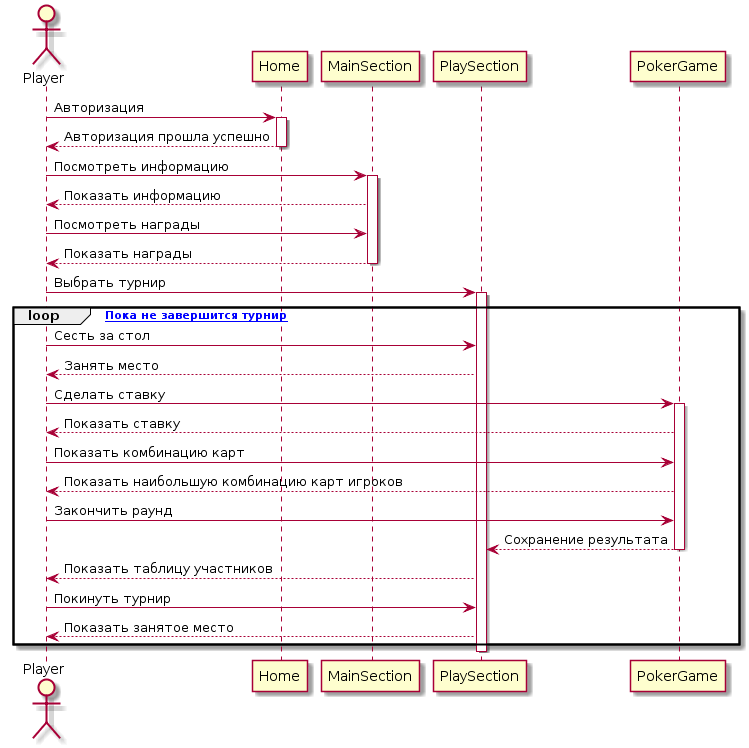


Рисунок 2 – Диаграмма последовательности для сценария использования «Зарегистрированный пользователь**»**

Код:

@startuml

actor Player

Player -> Home: Авторизация

activate Home

Home --> Player: Авторизация прошла успешно

deactivate Home

Player -> MainSection: Посмотреть информацию

activate MainSection

MainSection --> Player: Показать информацию

Player -> MainSection: Посмотреть награды

MainSection --> Player: Показать награды

deactivate MainSection

Player -> PlaySection: Выбрать турнир

activate PlaySection

loop [ Пока не завершится турнир]

Player -> PlaySection: Сесть за стол

PlaySection --> Player : Занять место

Player -> PokerGame: Сделать ставку

activate PokerGame

PokerGame --> Player: Показать ставку

Player -> PokerGame: Показать комбинацию карт

PokerGame --> Player: Показать наибольшую комбинацию карт игроков

Player -> PokerGame: Закончить раунд

PokerGame --> PlaySection: Сохранение результата

deactivate PokerGame

PlaySection --> Player: Показать таблицу участников

Player -> PlaySection: Покинуть турнир

PlaySection --> Player: Показать занятое место

end

deactivate PlaySection

@enduml

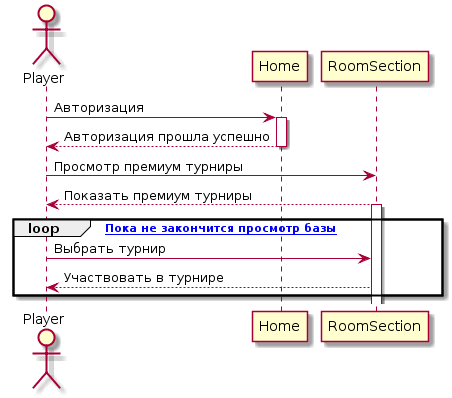


Рисунок 3 – Диаграмма последовательности для сценария использования «Премиум пользователь**»**

Код:

@startuml

actor Player

Player -> Home: Авторизация

activate Home

Home --> Player: Авторизация прошла успешно

deactivate Home

Player -> RoomSection: Открыть премиум турниры

RoomSection --> Player: Показать премиум турниры

activate RoomSection

loop [ Пока не закончится просмотр базы]

Player -> RoomSection: Выбрать турнир

RoomSection --> Player: Участвовать в турнире

end

@enduml

**Диаграммы деятельности:**

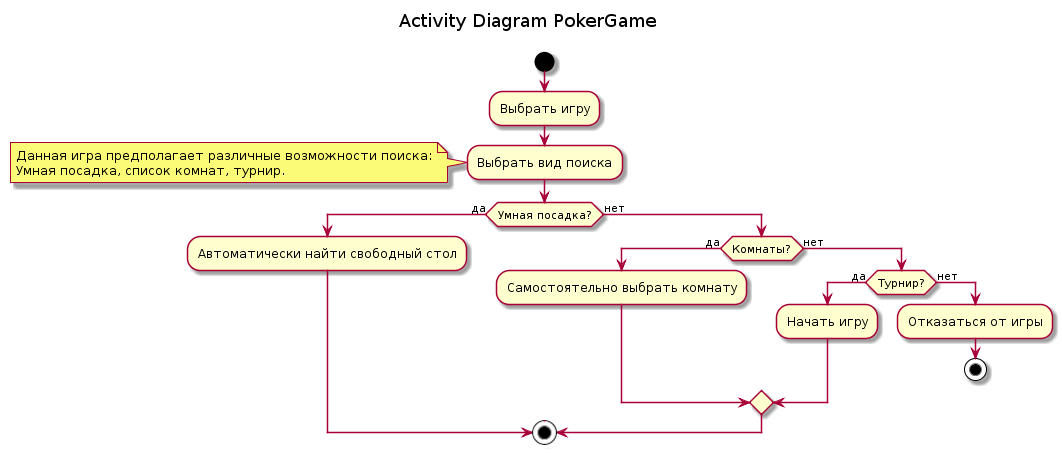


Рисунок 4 – Сценарий на выбор поиска игры

Код:

@startuml

title Activity Diagram PokerGame \n

start

:Выбрать игру;

:Выбрать вид поиска;

note left

Данная игра предполагает различные возможности поиска:

Умная посадка, список комнат, турнир.

end note

if (Умная посадка?) then (да)

:Автоматически найти свободный стол;

else (нет)

if (Комнаты?) then (да)

:Самостоятельно выбрать комнату;

else (нет)

if (Турнир?) then (да)

:Начать игру;

else (нет)

:Отказаться от игры;

stop

@enduml

**Диаграмма классов:**

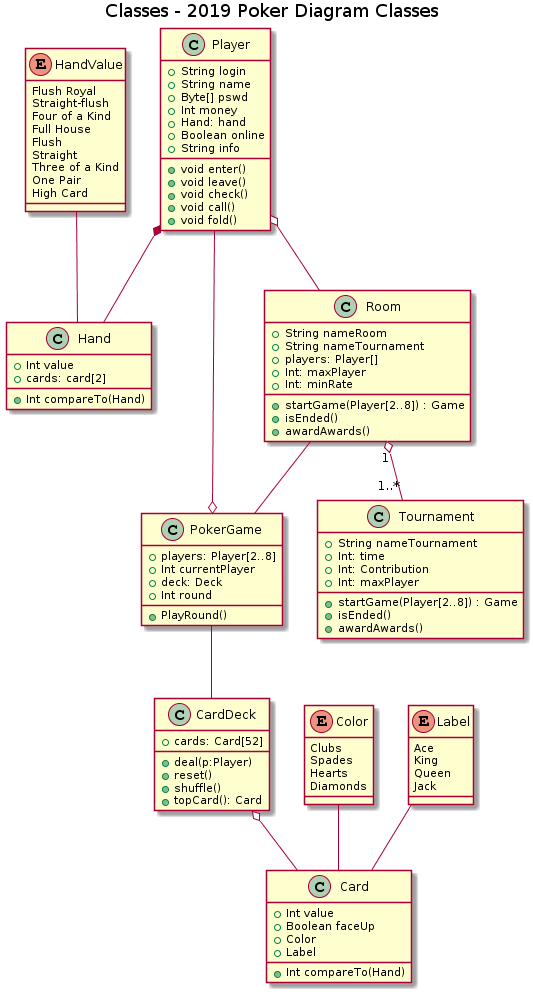


Рисунок 6 – Диаграмма классов

Код:

@startuml

title Classes - 2019 Poker Diagram Classes

class Player {

+String login

+String name

+Byte[] pswd

+Int money

+Hand: hand

+Boolean online

+String info

+void enter()

+void leave()

+void check()

+void call()

+void fold()

}

class Hand {

+Int value

+cards: card[2]

+Int compareTo(Hand)

}

class Card {

+Int value

+Boolean faceUp

+Color

+Label

+Int compareTo(Hand)

}

class PokerGame {

+players: Player[2..8]

+Int currentPlayer

+deck: Deck

+Int round

+PlayRound()

}

enum Color {

Clubs

Spades

Hearts

Diamonds

}

enum Label {

Ace

King

Queen

Jack

}

class CardDeck {

+cards: Card[52]

+deal(p:Player)

+reset()

+shuffle()

+topCard(): Card

}

class Room {

+String nameRoom

+String nameTournament

+players: Player[]

+Int: maxPlayer

+Int: minRate

+startGame(Player[2..8]) : Game

+isEnded()

+awardAwards()

}

class Tournament {

+String nameTournament

+Int: time

+Int: Сontribution

+Int: maxPlayer

+startGame(Player[2..8]) : Game

+isEnded()

+awardAwards()

}

enum HandValue {

Flush Royal

Straight-flush

Four of a Kind

Full House

Flush

Straight

Three of a Kind

One Pair

High Card

}

Player \*-- Hand

Player --o PokerGame

Room -- PokerGame

PokerGame -- CardDeck

CardDeck o-- Card

Color -- Card

Label -- Card

Player o-- Room

HandValue -- Hand

Room "1" o-- "1..\*" Tournament

@enduml

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ ДНР

ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра ПИ

ОТЧЕТ

по лабораторной работе № 2

по теме

“ Разработка UML-диаграмм с использованием PlantUML ”

Выполнил:

ст. группы ПИ-15А

Ефременко Г. О.

Преподаватели:

Грищенко В. И.

Грищенко Д. А.

Донецк - 2018