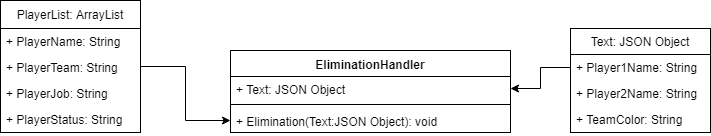
1. Elimination System
   1. Objectives

한 플레이어가 특정 플레이어를 상대 팀의 왕으로 지목했을 경우 이를 처리해주는 시스템의 설계를 설명한다. Class Diagram, Sequence Diagram, State Diagram을 통해 본 시스템의 구조를 설명한다.

* 1. Class Diagram



1. EliminationHandler
   1. Attributes

Text: 누가 누굴 지목했는지에 대한 정보

* 1. Methods

Elimination(Text): Text에 담긴 정보를 토대로 Elimination 과정을 진행한다.

1. Test: JSON Object
   1. Attributes

Player1Name: 지목한 플레이어의 이름

Player2Name: 지목당한 플레이어의 이름

TeamColor: 지목한 플레이어의 소속 팀

* 1. Methods

해당없음.

1. PlayerList: ArrayList
   1. Attributes

PlayerName: 플레이어의 이름

PlayerTeam: 플레이어의 소속 팀

PlayerJob: 플레이어의 게임 내 지위

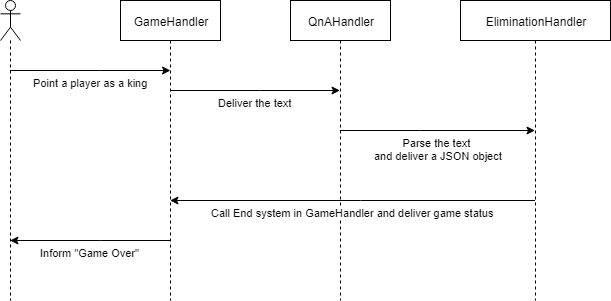
PlayerStatus: 플레이어의 생존 여부

* 1. Methods

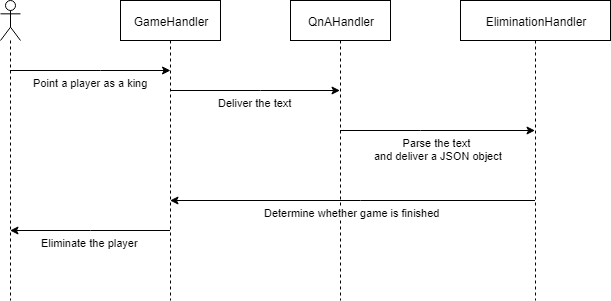
해당없음.

* 1. Sequence Diagram

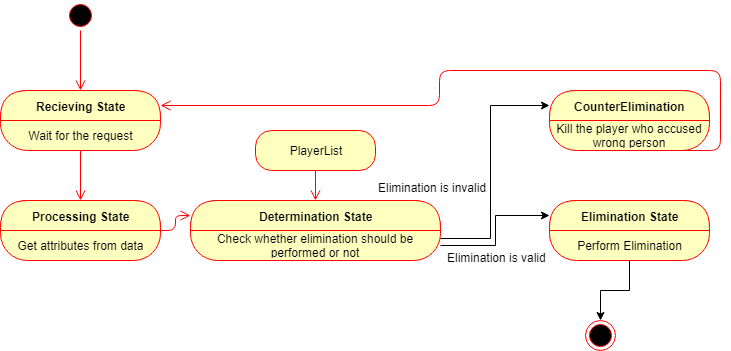
1. “Valid” Elimination: the player accused right person



1. “Invalid” Elimination: the player accused wrong person



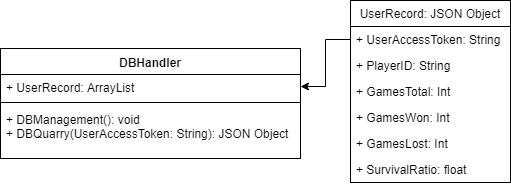
* 1. State Diagram



1. DB Management System
   1. Objectives

본 봇의 Database를 관리해주는 시스템의 설계를 설명한다. Class Diagram, Sequence Diagram, State Diagram을 통해 본 시스템의 구조를 설명한다.

* 1. Class Diagram



1. DBHandler
   1. Attributes

UserRecord: 각 User의 전적사항을 담은 객체의 ArrayList

* 1. Methods

DBManagement(): 게임이 끝난 후 플레이어들의 전적을 갱신한다.

DBQuarry(UserAccessToken): UserAccessToken을 통해 DB에서 전적을 가져온다.

1. UserRecord: JSON Object
   1. Attributes

UserAccessToken: 유저에게 부여된 고유한 식별번호

PlayerID: 플레이어의 ID

GamesTotal: 총 플레이한 게임 수

GamesWon: 이긴 게임의 수

GamesLost: 진 게임의 수

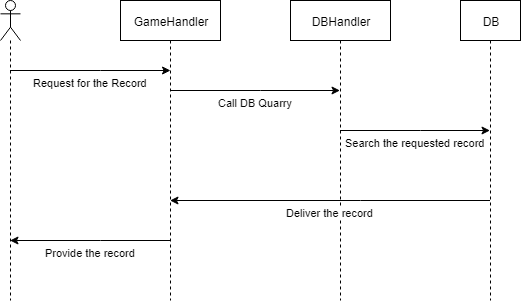
SurvivalRatio: 게임 내 생존율

* 1. Methods

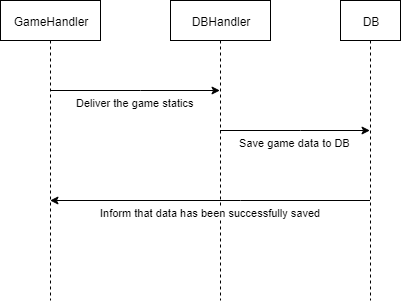
해당없음

* 1. Sequence Diagram

1. DB Quarry Sequence: 플레이어가 자신의 전적을 요청했을 때

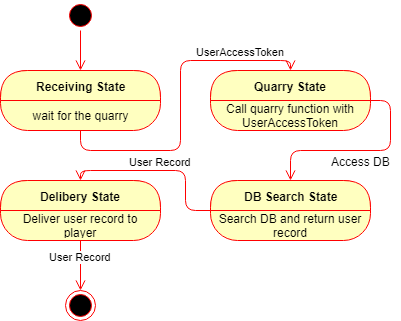


1. DB Save Sequence: 게임 종료 후 플레이어들의 전적을 갱신할 때



* 1. State Diagram

1. DB Quarry State



1. DB Save State

