

Project Overview

1. 프로젝트 소개

- 윷놀이 게임을 JAVA를 이용해 OOAD 설계로 구현하였다.
- UI를 변경해도 문제없이 실행할 수 있게 설계를 했다.

2. Requirement

- 기능적 요구사항 (Functional Requirements)
 - 판 선택 기능
 - 사각형, 오각형, 육각형 중 하나를 고른다.
 - 플레이어 수 선택 기능
 - 플레이어 수를 2명에서 4명까지 선택할 수 있어야 한다.
 - 말의 수 선택 기능
 - 각 플레이어는 2개에서 5개의 말을 선택할 수 있어야 한다.
 - 윷 던지기 기능
 - 플레이어는 매 턴마다 윷을 던지고, 윷이나 모가 나올 경우 한 번 더 던질 수 있어야 한다.
 - 테스트 윷 던지기 기능
 - 윷을 던져 나오는 6가지 결과가 제대로 반영되지 테스트 기능을 통해 확인하여야 한다.
 - 잡기 기능
 - 상대팀의 말을 잡을 수 있으며, 잡은 말은 시작점으로 돌아가야 한다.
 - 업기 기능
 - 자신의 말에 있는 곳에 도착 시, 그 다음 턴부터 두 말은 함께 이동하게 된다.
- 비기능적 요구사항 (Non-Functional Requirements)
 - Usability 요구사항

- 게임의 UI는 직관적이어야 하고, UI가 바뀌어도 사용자는 문제없이 플레이 할 수 있어야 한다.
- Supportability 요구사항
 - 시스템은 Windows 및 Mac OS에서 모두 실행되어야 한다.

2. 빌드 및 실행 방법

- 빌드 방법
 1. YUT.java 파일을 실행하여 프로젝트를 빌드한다.
- 실행 방법
 1. 판의 종류를 선택한다.
 - 사각형
 - 오각형
 - 육각형
 2. 사용자 수와 말의 수를 설정한다.
 3. 시작 버튼을 클릭하여 게임을 시작한다.
 4. 우승자가 나오면 게임이 종료된다.
 - a. 중간에 종료 혹은 재시작을 하고 싶을 경우, ~~을 한다.

3. 게임 규칙

- 사용자는 2...4인, 말의 개수는 2...5개이다.
- 사용자들은 순서를 정하기 위해 윷을 굴린다.
 - 모 > 윷 > 걸 > 개 > 도 > 백도 순서
- 이후 순서대로 윷을 던지고, 그 결과만큼 말을 선택해서 이동한다.
 - 백도 - -1 칸 / 도 - 1칸 / 개 - 2칸 / 걸 - 3칸 / 윷 - 4칸 / 모 - 5칸
 - 윷, 모일 경우, 한 번 더 굴린다.
- 말이 분기점에 정확히 도착했을 경우
 - 그 상황에서의 shortest path로 이동한다.
- 그렇지 않을 경우

- 항상 직선으로 이동해야 한다.

3. 설계 및 기술 요소

- 이게임은 OOAD 방식을 사용하여 class 기반으로 구현되었다.
- 게임은 Java로 구성되었고, MVC 구조를 적용하여 게임 진행과 UI를 분리하였다.
- 사용자와 말은 각각 Player, Horse 클래스로 구현되며, 게임판과 칸들은 각각 Board, Cell 클래스로 구현된다.