|  |
| --- |
| **Report** |

**Team Meta.**

**조장** 심재경 2014010916

백형진 2014010911

박수빈 2014010908

김세훈 2014010904

* **INDEX** -

Report #2

* 1. **Use Case 명세서** - p.3 -
  2. **Sequence Diagram** - p.8 -
  3. **Class Diagram** - p.13 -
  4. **UI** - p.14 -

|  |
| --- |
| **1. Use Case 명세서** |

|  |
| --- |
| **Use Case #1** |

|  |
| --- |
| **▶ 유스케이스 명** : 메뉴 선택  **▶ 엑터 명** : User1  **▶ 유스케이스 개요 및 설명**  - 사용자가 컨트롤러의 회전기능으로 메뉴 선택을 한다.  **▶ 사전 조건** : 게임을 실행시킨다.  **▶ 작업 흐름**  - 정상흐름  1. 사용자는 컨트롤러의 회전 기능을 사용한다.  2. 컨트롤러의 회전기능은 회전각을 감지한다.  3. 컨트롤러의 통신기능은 회전각이 15도를 넘으면 게임 소프트웨어로 "회전각 값"을 통신한다.  4. 게임 소프트웨어의 통신은 통신된 정보를 분석한다.  5. 게임 소프트웨어의 메뉴는 선택 포인터를 이동한다.  6. 사용자가 컨트롤러의 버튼을 누른다.  7. 컨트롤러의 버튼기능은 버튼 입력을 감지한다.  8. 컨트롤러의 통신기능은 게임 소프트웨어로 "버튼 입력" 통신을 한다.  9. 게임 소프트웨어의 메뉴는 현재 포인터가 있는 메뉴를 실행한다.  - 대안 흐름  - 예외 흐름  1. 정상 흐름 3에서 측정된 회전각도가 15도이하일 경우, 방향전환 신호를 보내지 않는다. |

|  |
| --- |
| **Use Case #2** |
|  |

|  |
| --- |
| **▶ 유스케이스 명** : 게임플레이  **▶ 엑터 명** : User1, User2  **▶ 유스케이스 개요 및 설명**  - 사용자는 게임 시작 메뉴를 선택 후, 싱글 모드 또는 듀얼 모드로 게임을 플레인 한다.  **▶ 사전 조건** : 메뉴 선택  **▶ 작업 흐름**  - 정상흐름  1. 사용자는 컨트롤러의 회전 기능을 사용한다.  2. 컨트롤러의 회전기능은 회전각을 감지한다.  3. 컨트롤러의 통신기능은 회전각이 15도를 넘으면 게임 소프트웨어로 "회전각 값"을 통신한다.  4. 게임 소프트웨어의 통신은 통신된 정보를 분석한다.  5. 게임 소프트웨어의 게임플레이는 게임 모드 선택 포인터를 이동한다.  6. 사용자가 컨트롤러의 버튼을 누른다.  7. 컨트롤러의 버튼기능은 버튼 입력을 감지한다.  8. 컨트롤러의 통신기능은 게임 소프트웨어로 "버튼 입력" 통신을 한다.  9. 게임 소프트웨어의 게임플레이는 현재 포인터가 있는 게임모드를 실행한다.  10. 사용자는 컨트롤러의 센서 기능을 사용한다.  11. 컨트롤러는 게임 소프트웨어로 해당 기능 센서 값을 통신한다.  12. 게임 소프트웨어의 통신는 통신된 정보를 분석한다.  13. 게임 소프트웨어는 통신된 정보에 맞게 캐릭터를 움직인다.  14. 캐릭터가 도착지에 도착하면 게임이 클리어되고 클리어타임이 출력된다.  15. 게임 소프트웨어는 랭킹 등록화면을 출력한다.  - 대안 흐름  - 예외 흐름  1. 정상 흐름 3에서 측정된 회전각도가 15도이하일 경우, 방향전환 신호를 보내지 않는다. |

|  |
| --- |
| **Use Case #3** |
|  |

|  |
| --- |
| **▶ 유스케이스 명** : 방향전환  **▶ 엑터 명** : User1, User2  **▶ 유스케이스 개요 및 설명**  - 사용자는 컨트롤러의 회전기능을 이용하여, 캐릭터의 방향을 전환한다.  **▶ 사전 조건** : 메뉴 선택  **▶ 작업 흐름**  - 정상흐름  1. 사용자는 컨트롤러의 회전기능을 사용한다.  2. 컨트롤러의 회전기능은 회전각을 감지한다.  3. 컨트롤러의 통신기능은 회전각이 15도를 넘으면 게임 소프트웨어로 "회전각 값"을 통신한다.  4. 게임 소프트웨어의 통신기능은 통신된 정보를 분석한다.  5. 게임 소프트웨어는 캐릭터의 방향을 전환한다.  - 대안 흐름  - 예외 흐름  1. 정상 흐름 3에서 측정된 회전각도가 15도이하일 경우, 방향전환 신호를 보내지 않는다. |

|  |
| --- |
| **Use Case #4** |
|  |

|  |
| --- |
| **▶ 유스케이스 명** : 함정탈출  **▶ 엑터 명** : User1, User2  **▶ 유스케이스 개요 및 설명**  - 사용자는 컨트롤러의 흔들기 기능을 이용하여, 캐릭터를 함정에서 탈출한다.  **▶ 사전 조건** : 게임 플레이  **▶ 작업 흐름**  - 정상흐름  1. 사용자는 컨트롤러의 흔들기 기능을 사용한다.  2. 컨트롤러의 흔들기 기능은 가속도 센서로 가속도를 감지한다.  3. 컨트롤러의 통신기능은 x,y,z축 가속도 값 ax, ay, az를 제곱근(ax^2 + ay^2 + az^2)이 3이상 게임 소프트웨어로 "가속도값"을 통신한다.  4. 게임 소프트웨어의 통신기능은 통신된 정보를 분석한다.  5. 게임 소프트웨어는 캐릭터를 함정탈출 시킨다.  - 대안 흐름  - 예외 흐름  1. 정상흐름3에서 계산된 가속도 값이 3이하일 경우, 방향전환 신호를 보내지 않는다. |

|  |
| --- |
| **Use Case #5** |
|  |

|  |
| --- |
| **▶ 유스케이스 명** : 달리기  **▶ 엑터 명** : User1, User2  **▶ 유스케이스 개요 및 설명**  - 사용자는 컨트롤러의 두드리기 기능을 이용하여, 캐릭터를 달리게 한다.  **▶ 사전 조건** : 게임 플레이  **▶ 작업 흐름**  - 정상흐름  1. 사용자는 컨트롤러의 두드리기 기능을 사용한다.  2. 컨트롤러의 두드리기 기능은 압력 센서로 가속도를 감지한다.  3. 컨트롤러의 통신기능은 압력 센서 값이 입력되면 게임 소프트웨어로 "압력 센서 값"을 통신한다.  4. 게임 소프트웨어의 통신기능은 통신된 정보를 분석한다.  5. 게임 소프트웨어의 캐릭터는 달린다.  - 대안 흐름  - 예외 흐름 |

|  |
| --- |
| **Use Case #6** |
|  |

|  |
| --- |
| **▶ 유스케이스 명** : 점프  **▶ 엑터 명** : User1, User2  **▶ 유스케이스 개요 및 설명**  - 사용자는 컨트롤러의 버튼 기능을 이용하여, 캐릭터를 점프시킨다.  **▶ 사전 조건** : 게임 플레이  **▶ 작업 흐름**  - 정상흐름  1. 사용자는 컨트롤러의 버튼 기능을 사용한다.  2. 컨트롤러의 버튼 기능은 버튼 입력을 감지한다.  3. 컨트롤러의 통신기능은 버튼이 입력이 되면 게임 소프트웨어로 "버튼 입력"을 통신한다  4. 게임 소프트웨어의 통신기능은 통신된 정보를 분석한다.  5. 게임 소프트웨어의 캐릭터는 점프한다.  - 대안 흐름  - 예외 흐름 |

|  |
| --- |
| **Use Case #7** |
|  |

|  |
| --- |
| **▶ 유스케이스 명** : 랭킹확인  **▶ 엑터 명** : User1  **▶ 유스케이스 개요 및 설명**  - 사용자는 컨트롤러의 회전기능을 이용하여 랭킹확인 메뉴를 선택한다.  **▶ 사전 조건** : 메뉴 선택  **▶ 작업 흐름**  - 정상흐름  1. 게임 소프트웨어의 랭킹확인은 파일로부터 랭킹을 읽어온다.  2. 게임 소프트웨어는 사용자에게 랭킹을 출력해준다.  - 대안 흐름  - 예외 흐름  1. 정상 흐름 3에서 사용자가 클리어 실패했을 경우, 클리어 실패 문구를 출력한다. |

|  |
| --- |
| **Use Case #8** |
|  |

|  |
| --- |
| **▶ 유스케이스 명** : 랭킹등록  **▶ 엑터 명** : User1, User2  **▶ 유스케이스 개요 및 설명**  - 사용자는 게임이 클리어되면 랭킹을 등록한다.  **▶ 사전 조건** : 게임 플레이  **▶ 작업 흐름**  - 정상흐름  1. 사용자는 게임플레이에서 클리어한다.  2. 게임플레이는 랭킹등록 확인 메뉴를 출력한다.  3. 사용자는 컨트롤러의 회전기능을 사용한다.  4. 컨트롤러의 회전기능은 회전값을 감지한다.  5. 컨트롤러의 통신기능은 회전각이 15도를 넘으면 게임 소프트웨어로 "회전각 값"을 통신한다.  6. 게임 소프트웨어의 통신기능은 통신된 정보를 분석한다.  7. 게임 소프트웨어는 "확인" "취소" 선택을 전환한다.  8. 사용자가 컨트롤러의 버튼을 누른다.  9. 컨트롤러의 버튼기능은 버튼 입력을 감지한다.  10. 컨트롤러의 통신기능은 게임 소프트웨어로 "버튼 입력" 통신을 한다.  11. 게임 소프트웨어의 랭킹등록은 랭킹을 등록한다.  - 대안 흐름  - 예외 흐름  1. 정상흐름 11에서 선택된 메뉴가 취소이면 랭킹을 등록하지 않는다. |

|  |
| --- |
| **Use Case #9** |
|  |

|  |
| --- |
| **▶ 유스케이스 명** : 클리어타임 측정  **▶ 엑터 명** : Timer  **▶ 유스케이스 개요 및 설명**  - 게임 소프트웨어는 게임플레이 클리어 시간을 측정하고 있다.  **▶ 사전 조건** : 게임 플레이  **▶ 작업 흐름**  - 정상흐름  1. 사용자는 게임플레이를 시작한다.  2. 게임플레이는 게임 시작 시간을 저장한다.  3. 사용자는 게임을 클리어한다.  4. 게임플레이는 게임 종료 시간과 시작 시간의 차이를 계산한다.  5. 게임플레이는 사용자에게 클리어 타임을 출력한다.  - 대안 흐름  - 예외 흐름 |

|  |
| --- |
| **2. Sequence Diagram** |

|  |
| --- |
| **Sequence Diagram #1 : 메뉴 선택** |
|  |

|  |
| --- |
|  |

|  |
| --- |
| **Sequence Diagram #2 : 게임플레이** |
|  |

|  |
| --- |
|  |

|  |
| --- |
| **Sequence Diagram #3 : 방향전환** |
|  |

|  |
| --- |
|  |

|  |
| --- |
| **Sequence Diagram #4 : 함정탈출** |
|  |

|  |
| --- |
|  |

|  |
| --- |
| **Sequence Diagram #5 : 달리기** |
|  |

|  |
| --- |
|  |

|  |
| --- |
| **Sequence Diagram #6 : 점프** |
|  |

|  |
| --- |
|  |

|  |
| --- |
| **Sequence Diagram #7 : 랭킹확인** |
|  |

|  |
| --- |
|  |

|  |
| --- |
| **Sequence Diagram #8 : 랭킹등록** |
|  |

|  |
| --- |
|  |

|  |
| --- |
| **Sequence Diagram #9 : 클리어타임 측정** |
|  |

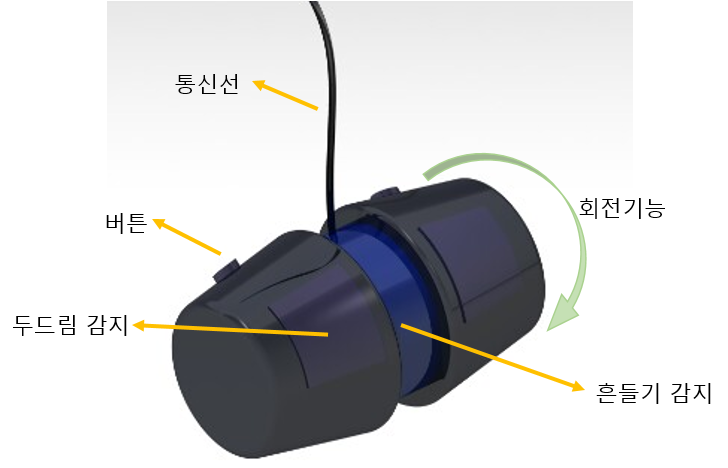
|  |
| --- |
|  |

|  |
| --- |
| **3. Class Diagram** |

|  |
| --- |
|  |

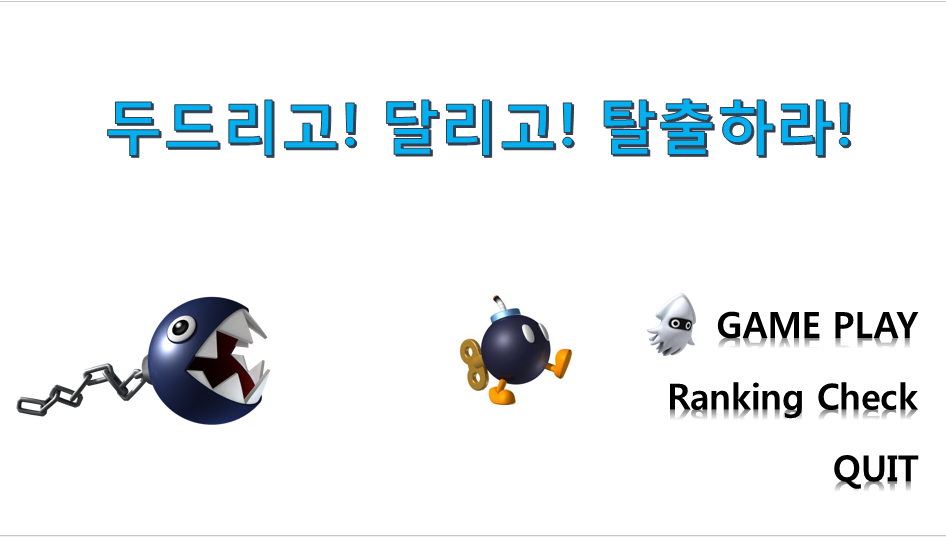
|  |
| --- |
| **4. UI** |

|  |
| --- |
| **4.1 게임 컨트롤러** |
|  |

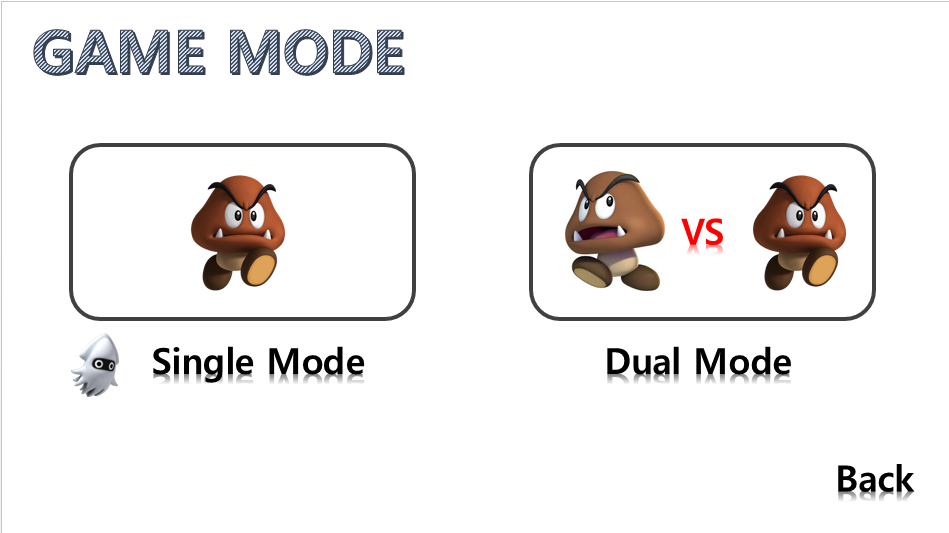


|  |
| --- |
| **4.2 게임 소프트웨어** |
|  |

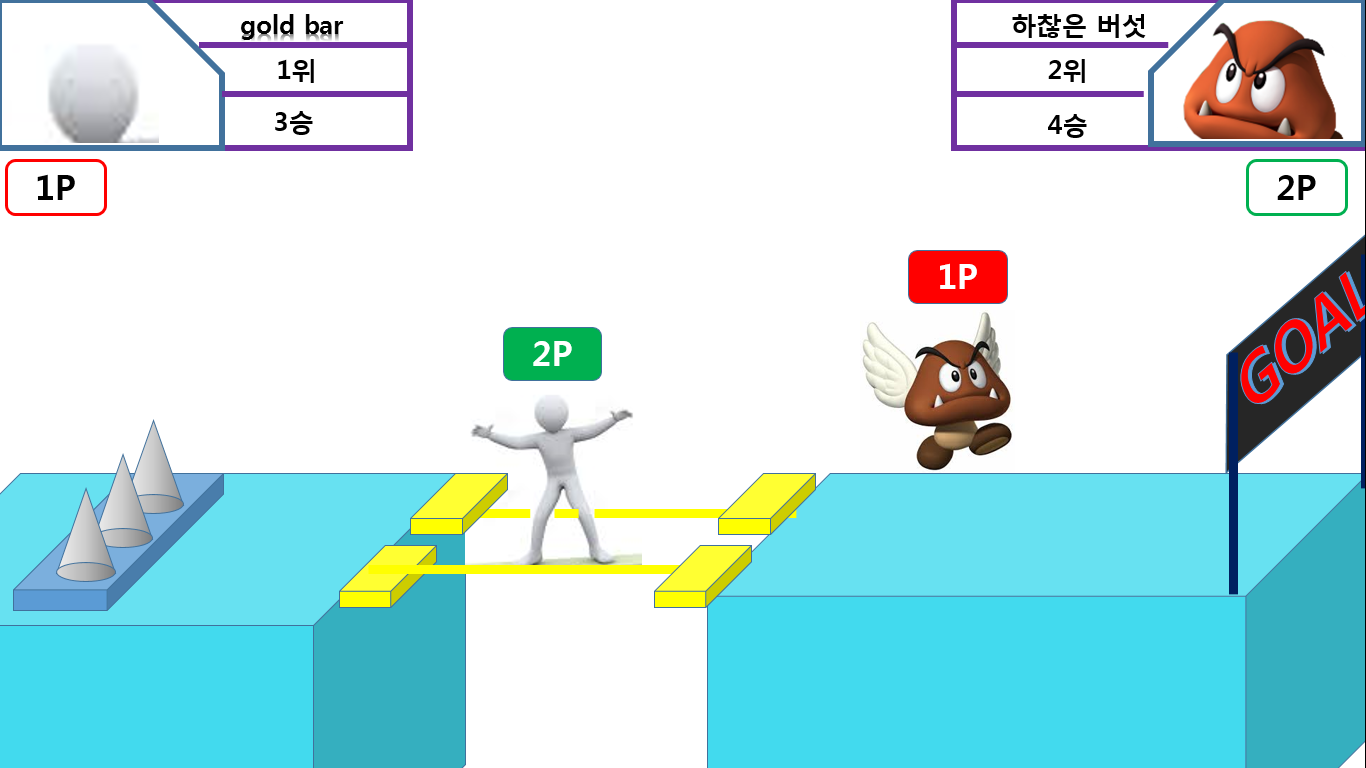
**4.2.1 메뉴화면**



**4.2.1 게임플레이** - 게임 모드선택 -



**4.2.2 게임플레이** - 게임 중 -



**4.2.3 랭킹확인**

