

## 요구사항 분석

### 1. 프로그램 소개

#### □ 목적

- 2017년 1학기 컴퓨터응용 및 실습 1 강의의 팀 프로젝트 과제를 설명한다.

#### □ 작동방식

- Robot은 사용자의 입력에 따라 상태를 변화시키며 상태에 따라 동작한다.
- Robot은 다른 Game Object 상호작용을 하며 데미지를 입거나, 땅에 착지하는 등 실제 세계에 존재하는 물체처럼 행동한다.

### 2. 요구사항 상세

#### □ 로봇의 기능

- 로봇은 이동이 가능하다
- 로봇은 점프 및 착지가 가능하다.
- 로봇은 총알을 발사할 수 있다.
- 로봇은 중력을 적용받는다.
- 로봇의 하체는 이동 기능을 담당한다.
- 로봇의 상체는 총알 종류에 따른 데미지 연산, 고유 스킬 발동의 기능이 있다.
- 로봇의 무기는 총알 종류를 결정한다.
- 로봇의 무기는 총알 발사 각도를 조절할 수 있다.
- 총알은 다른 충돌 가능한 객체와 충돌가능하다.

#### □ 제한사항

- 모든 명령 및 입력은 키보드와 마우스 입력으로 수행된다.
- 모든 출력은 Java swing GUI를 통해 수행된다.
- 파일 저장이 필요한 경우 프로젝트 내 Package 내부에서 저장하여 관리한다.

#### □ 가정 및 의존사항

- 로봇은 점프 후 땅을 밟기까지 이동, 중복 점프가 불가능하다.
- 총알은 총알끼리 충돌하지 않는다.
- 총알의 비행거리는 제한이 있으며 거리를 벗어날 경우 소멸된다.

### 3. 요구사항 명세

#### □ 유저의 행동기능

##### – UI Widgets 1. Build Room

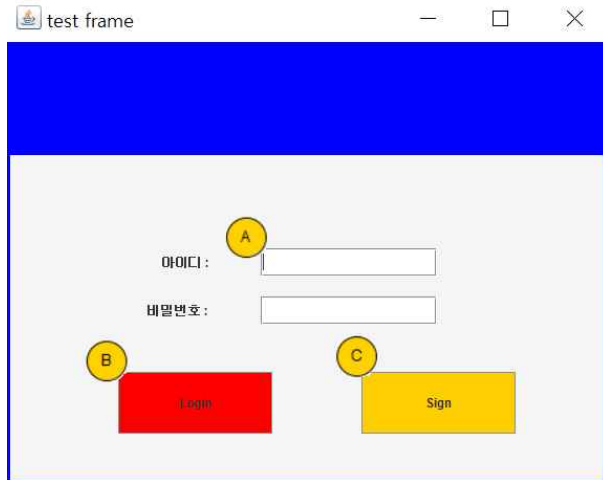


Use Case	Machine Choice
Actor	User
Pre-Condition	N/A
Purpose	조립할 로봇을 선택한다. 로봇 조립은 최대 3 종류까지 할 수 있다.
Typical Courses of Event	(A): Actor, (C): Client System (A): A컴포넌트의 상단 탭에서 조립할 로봇 번호를 선택한다. (C): A컴포넌트의 하단 박스에 선택한 로봇의 이미지를 띄운다. (C): B컴포넌트의 박스에 현재 선택한 로봇이 착용중인 파츠와 조립된 로봇의 종합적인 정보가 출력된다.

Use Case	Parts Select
Actor	User
Pre-Condition	N/A
Purpose	선택한 로봇의 변경할 파츠를 선택한다.
Typical Courses of Event	(A): Actor, (C): Client System, (S): Server System (A): C컴포넌트의 상단 탭에서 변경할 파츠 종류를 선택한다. (C): C컴포넌트 하단 박스에 선택한 파츠 종류에 해당하는 파츠 모델들의 리스트를 그린다.

Use Case	Parts Model Select
Actor	User
Pre-Condition	조립할 로봇과 변경할 파츠가 선택되어 있어야 한다. (디폴트 값으로 1번 로봇과 상체 파츠가 선택되어 있을 예정이다.)
Purpose	선택한 로봇의 변경할 파츠의 모델을 선택한다.
Typical Courses of Event	(A): Actor, (C): Client System, (S): Server System (A): C컴포넌트의 하단 박스에 나열된 파츠 모델 리스트 중 원하는 모델을 선택한다. (C): Actor가 선택한 모델이 조립에 적합한지 판정한다. (C): 조립하기에 적합하다고 판정된다면 A컴포넌트의 하단 박스에 선택한 모델로 조립된 로봇의 모습을 출력하고, B컴포넌트의 박스에 변경된 로봇의 정보를 갱신해준다. (S): 변경된 로봇의 정보를 갱신한다.

## - UI Widgets 2. Login Page



Use Case	Login
Actor	User
Pre-Condition	N/A
Purpose	멀티플레이를 하기위해 네트워크에 접속한다.
Typical Courses of Event	<p>(A): Actor, (C): Client System, (S): Server System</p> <p>(A): A컴포넌트의 ID, PW TextField를 채운다.</p> <p>(A): B컴포넌트를 클릭한다.</p> <p>(S): Actor입력한 정보와 일치하는 계정 정보가 서버에 있는지 확인한다.</p> <p>(S): 일치하는 정보가 있다면 로그인을 허용한다.</p> <p>(C): 로그인이 허용되면 Room List 페이지를 띄운다.</p> <p>(S): 클라이언트 계정정보를 네트워크 접속리스트에 갱신 시킨다.</p>

Use Case	Sign
Actor	User
Pre-Condition	N/A
Purpose	멀티플레이를 위한 네트워크 계정정보를 생성한다.
Typical Courses of Event	<p>(A): Actor, (C): Client System, (S): Server System</p> <p>(A): A컴포넌트의 ID, PW TextField를 채운다.</p> <p>(A): C컴포넌트를 클릭한다.</p> <p>(C): Actor가 입력한 정보와 중복되는 계정 정보가 서버에 있는지 확인한다.</p> <p>(S): 중복되는 계정 정보가 없다면 계정 생성을 허용하고 서버에 해당하는 계정 정보를 등록한다.</p>

### - UI Widgets 3. Room List



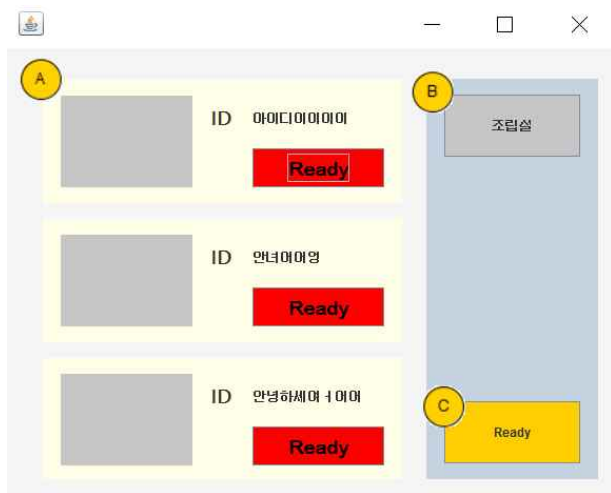
Use Case	Room Join
Actor	User
Pre-Condition	N/A
Purpose	멀티플레이 게임 방에 참여한다.
Typical Courses of Event	(A): Actor, (C): Client System, (S): Server System (A): A컴포넌트의 게임 방 리스트 중 하나를 선택한다. (A): B컴포넌트를 클릭한다. (C): 자리가 남는다면 Actor를 게임 방에 참여시키고 Game Room 페이지를 띄운다. (S): 클라이언트 계정정보의 Room 상태를 변경한다.

Use Case	Room Create
Actor	User
Pre-Condition	N/A
Purpose	멀티플레이 게임 방을 생성한다.
Typical Courses of Event	(A): Actor, (C): Client System, (S): Server System (A): C컴포넌트를 클릭한다. (C): Game Room 페이지를 띄운다. (S): 클라이언트 계정정보의 Room 상태를 변경한다.

Use Case	Machine Setting
Actor	User
Pre-Condition	N/A
Purpose	Build Room에서 로봇을 조립한다.
Typical Courses of Event	(A): Actor, (C): Client System (A): D컴포넌트를 클릭한다. (C): Build Room 페이지를 띄운다.

Use Case	Room Exit
Actor	User
Pre-Condition	N/A
Purpose	이전 페이지인 Login 페이지로 돌아간다.
Typical Courses of Event	(A): Actor, (C): Client System, (S): Server System (A): E컴포넌트를 클릭한다. (C): Login 페이지를 띄운다. (S): 클라이언트 계정 정보를 서버에서 제거한다.

- UI Widgets 4. wait List



Use Case	Machine Setting
Actor	User
Pre-Condition	N/A
Purpose	Build Room에서 로봇을 조립한다.
Typical Courses of Event	(A): Actor, (C): Client System (A): B컴포넌트를 클릭한다. (C): Build Room 페이지를 띄운다.

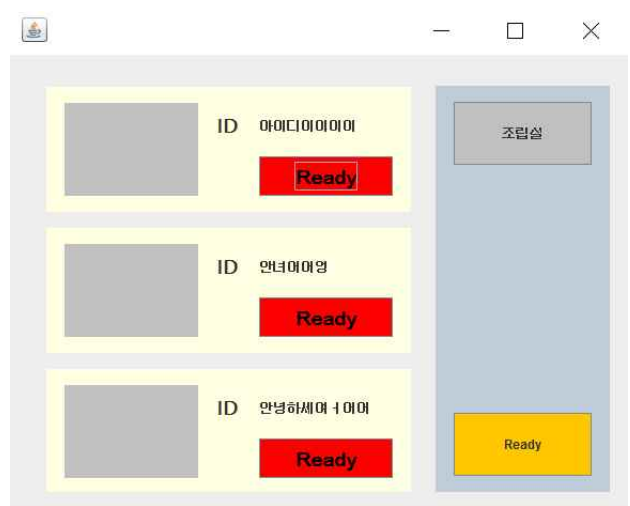
Use Case	
Actor	User
Pre-Condition	N/A
Purpose	게임의 준비가 완료됨을 알린다.
Typical Courses of Event	(A): Actor, (C): Client System, (S): Server System (A): C컴포넌트를 클릭한다. (S): 클라이언트 계정정보의 Room 상태를 변경한다. (C): A컴포넌트에서 준비가 된 User들의 Ready 박스의 색상을 변경한다.

## □ 인터페이스

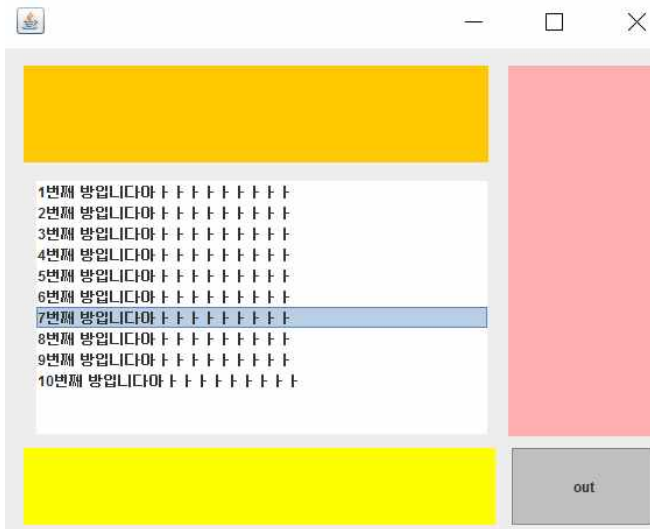
### - 유저 인터페이스



로그인 화면



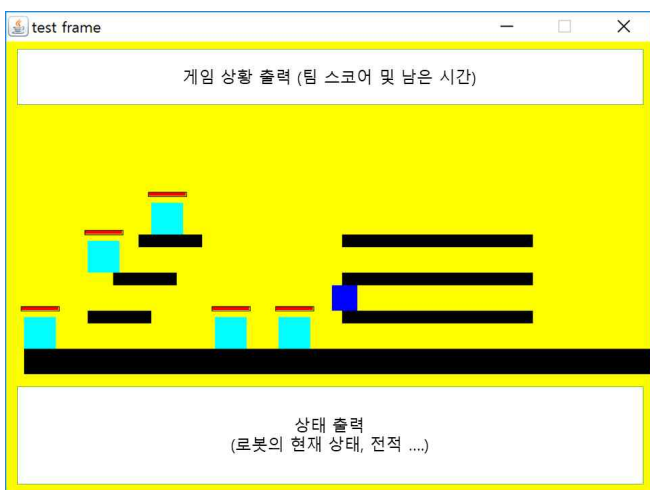
대기방 화면



대기방 열람 화면



로봇 조합 선택 화면

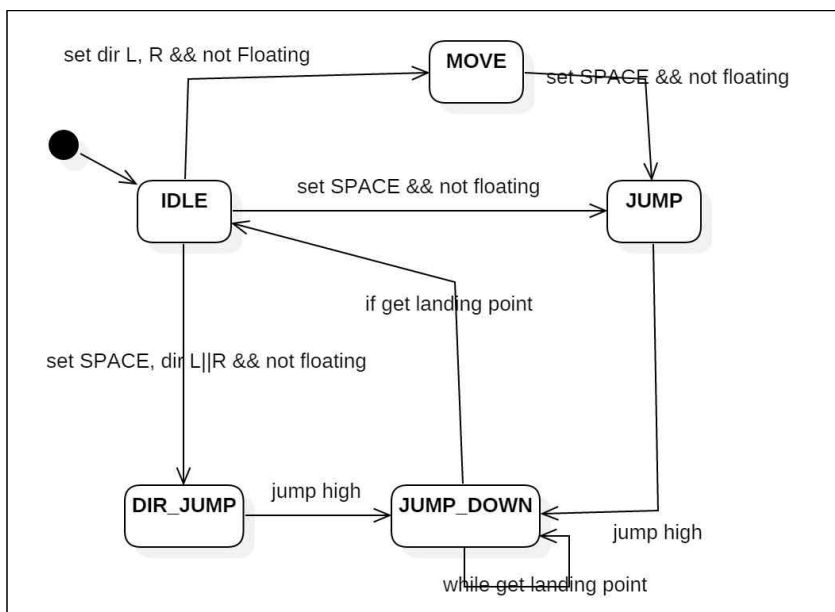


In Game Display

- 입력 인터페이스 : 좌우 방향키(이동) 상하 방향키(각도조절), SPACE(발사), ZXC(점프, 스킵1, 2), F(재장전)
- 출력 인터페이스 : 디스플레이 화면
- Communication : Dialog popup

## □ 기능적 요구사항

1. 로봇은 땅을 밟고 있고, 방향키 좌우방향키 입력 시 이동하여야 한다.
2. 로봇은 땅을 밟고 있고, 점프키 입력 시 점프를 해야 한다.
3. 로봇의 점프와 착지는 다음과 같은 과정을 통해 이루어진다.
  - 로봇의 이동에 관한 상태는 다음과 같이 정리될 수 있다.



4. 로봇은 상체, 하체, 무기의 총 3가지 파트로 구성되어 있다.
5. 로봇은 선택한 파트에 따라 성능이 결정된다.
6. 로봇의 하체는 이동 기능을 담당하며, 상태를 통해 수행된다.
7. 로봇의 상체는 데미지 연산 및 특수기능을 담당한다.
8. 로봇의 무기는 총알객체의 생성, 장탄 수, 각도조절 기능을 담당한다.
9. 키보드 조작을 통해 이동, 조준, 발사, 특수기능을 수행한다.
10. UI조작은 마우스를 통해 수행된다.
11. 무기의 발사횟수(장탄수)는 정해져 있고, 재장전의 개념을 적용하여 장전 중엔 발사가 불가능하다.
12. 무기의 연속적인 사용 입력 시, 무기마다 정해진 Delay에 따라 다음 사용시간까지 대기시간이 존재한다.
13. 총알은 정해진 시간동안 존재하며, 시간이 지나면 게임 내에서 소멸된다.

#### □ 비기능적 요구사항

1. 로봇의 특성과 상성을 이용하여 다양한 파츠를 조립하고, 즐거움을 느낄 수 있다.
2. 플레이 환경이 유저에게 최적화 되어 있어야 한다.
3. 플레이가 지루해지지 않도록 로봇을 이용한 전투만이 아닌 다른 게임내의 목표가 있어야 한다.

#### □ 기타

- 게임 내 프레임 딜레이, 재장전 대기시간, 발사 대기시간 등은 모두 밀리세컨드 단위로 수행한다.