mud_game 보고서

213235 소프트웨어공학과 이재륜

1. 서론

-프로젝트 목적 및 배경 : 7주차까지 배운 내용을 기반으로 간단한 콘솔 기반 MUD 게임을 구현하여 실습함으로써 객체 이동, 조건문, 반복문, 함수 분리 및 설계에 대한 이해를 높입니다. 특히 게임을 실제로 하는 테스트과정을 함으로써 사용자와의 상호작용 또한 경험해봅니다.

-목표 : 간단한 Mud 게임 구현

2. 요구사항

(1) 사용자 요구사항

-유저가 상하좌우로만 이동하며 목적지에 도착하는 게임

(2) 기능 계획

1. 유저는 체력 20을 가지고 게임을 시작합니다.
2. 사용자가 이동할때마다 체력이 1씩 감소합니다.
3. 처음 명령문을 입력 받을 때 마다 HP를 함께 출력합니다.
4. HP가 0이 되면 실패를 출력하고 종료합니다.
5. 무기/갑옷, 포션, 적을 만났을 때 그에 대한 메시지를 출력합니다. (적을 만날 경우 HP가 2 줄어들고 그에 대한 추가 메시지를 출력)
6. 코드 최적화를 합니다

(3) 함수 계획

1. 메인 함수 : 사용자에게 값을 계속 입력받고, 그에 대한 함수 호출
2. 지도와 현재 위치 출력 함수 : displayMap()
3. 사용자 위치 체크 함수 : checkXY()
4. 목적지에 도착 체크 함수 : checkGoal()
5. 이동 처리 함수 : moveUser()

3. 설계 및 구현

-기능 별 구현 사항

(1) 유저는 체력 20을 가지고 게임을 시작합니다.

```
const int initialHP = 20; // 초기 체력 설정

int user_HP = initialHP; // 초기 체력 20 설정
```

1. 입력
- initialHP = 초기 체력을 설정하는 전역변수

2. 결과
- 사용자는 게임이 시작될 때 20의 체력으로 시작함.

3. 설명
- initialHP라는 상수이자 전역 변수를 선언하여 초기 체력을 20으로 설정하였음. 이를 통해 체력을 수정할 때 용이하도록 했으며 체력 값이 코드 전반에 걸쳐 일관되게 유지될 수 있도록 함.

(2) 사용자가 이동할때마다 체력이 1씩 감소합니다.

```
void moveUser(int &user_x, int &user_y, int dx, int dy, int &user_HP, int map[][mapX]) {
    int new_x = user_x + dx; // 새로운 x 좌표 계산
    int new_y = user_y + dy; // 새로운 y 좌표 계산

    // 이동 후 좌표가 맵 안에 있는지 확인
    if (!checkXY(new_x, mapX, new_y, mapY)) {
        cout << "맵을 벗어났습니다. 다시 돌아갑니다." << endl; // 맵을 벗어나면 복귀 메시지 출력
    } else {
        user_x = new_x; // x 좌표 갱신
        user_y = new_y; // y 좌표 갱신
        cout << "이동했습니다." << endl;
        user_HP--; // 이동 시 체력 1 감소
        displayMap(map, user_x, user_y); // 이동 후 지도 출력
```

1. 입력
- user_HP = 현재 사용자의 체력을 저장하는 변수

2. 결과
- 사용자가 이동할 때마다 체력이 1씩 감소함.

3. 설명
- moveUser 함수 내에서 user_HP를 1씩 감소시켜 사용자가 이동할 때마다 체력이 줄어들도록 구현함.

(3) 처음 명령문을 입력 받을 때 마다 HP를 함께 출력합니다.

```
while (1) { // 사용자에게 계속 입력받기 위해 무한 루프
    // 체력 상태와 명령어 입력
    cout << "현재 HP: " << user_HP << " | 명령어를 입력하세요 (up,down,left,right,map,finish): ";
```

1. 입력
- user_HP = 현재 사용자의 체력을 저장하는 변수
2. 결과
- 사용자에게 명령어 입력을 요청할 때마다 현재 체력이 함께 출력됨.
3. 설명
- 명령어 입력 요청 시 cout을 통해 user_HP를 출력하여 사용자가 이동하기 전에 현재 체력을 확인할 수 있도록 하였음.

(4) HP가 0이 되면 실패를 출력하고 종료합니다.

```
// 체력이 0이 되면 게임 실패
if (user_HP <= 0) {
    cout << "HP가 0입니다. 실패했습니다." << endl;
    cout << "게임을 종료합니다." << endl;
    break;
}
```

1. 입력
- user_HP = 현재 사용자의 체력을 저장하는 변수
2. 결과
- 체력이 0이 되면 HP가 0이며 실패했다는 메시지를 출력하고 게임이 종료됨.
3. 설명
- 사용자가 이동했을 때마다 user_HP가 0 이하인지 확인하여 체력이 0이 되면 실패 메시지를 출력하고 게임을 종료하도록 하였음.

(5) 무기/갑옷, 포션, 적을 만났을 때 그에 대한 메시지를 출력합니다. (적을 만날 경우 HP가 2 줄어들고 그에 대한 추가 메시지를 출력)

```
// 현재 위치에 따른 상호작용 처리
int posState = map[user_y][user_x];
if (posState == 1) { // 아이템 위치일 경우
    int item = rand() % 2; // 무기(0) 또는 갑옷(1) 랜덤 결정
    if (item == 0) {
        cout << "무기를 획득했습니다!" << endl;
    } else {
        cout << "갑옷을 획득했습니다!" << endl;
    }
    map[user_y][user_x] = 0; // 아이템을 한 번 획득한 후 빈 공간으로 설정
}
else if (posState == 2) { // 적을 만났을 경우
    cout << "적을 만났습니다! HP가 2 감소합니다." << endl;
    user_HP -= 2; // 체력 2 감소
}
else if (posState == 3) { // 포션을 발견한 경우
    cout << "포션을 발견했습니다! 체력이 2 회복됩니다." << endl;
    user_HP += 2; // 체력 2 회복
    map[user_y][user_x] = 0; // 포션을 한 번 획득한 후 빈 공간으로 설정
}
```

1. 입력
- user_HP = 현재 사용자의 체력을 저장하는 변수
- map = 게임 맵 배열로 각 위치에 아이템, 적, 포션, 목적지 정보가 저장하는 배열 변수
- posState = 사용자가 도착한 위치의 상태를 나타내는 배열변수 (아이템, 적, 포션 여부)

2. 결과
- 사용자가 특정 위치에 도착하면 상황에 맞는 메시지가 출력됨.
- 아이템 : 무기 또는 갑옷 중 하나가 무작위로 드랍되며 메시지가 출력됨.
- 적: "적을 만났습니다! HP가 2 감소합니다."라는 메시지가 출력되고 체력이 2 감소함.
- "포션을 발견했습니다! 체력이 2 회복됩니다."라는 메시지가 출력되고 체력이 2 증가함.

3. 설명
- moveUser 함수 내에서 위치 상태에 따라 특정 이벤트가 발생하도록 구현하였음. rand() 함수를 사용하여 아이템을 주웠을 경우 무기와 갑옷 중 하나가 무작위로 선택되도록 하였으며 적을 만날 경우 체력이 2 감소하고 포션을 만나면 체력이 2 증가하도록 하였음.

(6) 코드 최적화를 합니다

```
// 유저 이동을 처리하는 함수
// - moveUser 함수를 추가하여 이동 중복 코드를 줄임
// - 이동 방향(dx, dy)을 받아 새로운 위치를 계산하고 맵 안에 있는지 확인
void moveUser(int &user_x, int &user_y, int dx, int dy, int &user_HP, int map[][mapX]) {
    int new_x = user_x + dx; // 새로운 x 좌표 계산
    int new_y = user_y + dy; // 새로운 y 좌표 계산

    // 이동 후 좌표가 맵 안에 있는지 확인
    if (!checkXY(new_x, mapX, new_y, mapY)) {
        cout << "맵을 벗어났습니다. 다시 돌아갑니다." << endl; // 맵을 벗어나면 복귀 메시지 출력
    } else {
        user_x = new_x; // x 좌표 갱신
        user_y = new_y; // y 좌표 갱신
        cout << "이동했습니다." << endl;
        user_HP--; // 이동 시 체력 1 감소
        displayMap(map, user_x, user_y); // 이동 후 지도 출력

        // 현재 위치에 따른 상호작용 처리
        int posState = map[user_y][user_x];
        if (posState == 1) { // 아이템 위치일 경우
            int item = rand() % 2; // 무기(0) 또는 갑옷(1) 랜덤 결정
            if (item == 0) {
                cout << "무기를 획득했습니다!" << endl;
            } else {
                cout << "갑옷을 획득했습니다!" << endl;
            }
            map[user_y][user_x] = 0; // 아이템을 한 번 획득한 후 빈 공간으로 설정
        }
        else if (posState == 2) { // 적을 만났을 경우
            cout << "적을 만났습니다! HP가 2 감소합니다." << endl;
            user_HP -= 2; // 체력 2 감소
        }
        else if (posState == 3) { // 포션을 발견한 경우
            cout << "포션을 발견했습니다! 체력이 2 회복됩니다." << endl;
            user_HP += 2; // 체력 2 회복
            map[user_y][user_x] = 0; // 포션을 한 번 획득한 후 빈 공간으로 설정
        }
    }
}
```

1. 입력
- user_x = 현재 사용자의 x 좌표
- user_y = 현재 사용자의 y 좌표
- dx, dy = 이동 방향을 나타내는 변수
- user_HP = 현재 사용자의 체력을 저장하는 변수
- map = 게임 맵 배열변수

2. 결과
- 이동 처리 코드의 중복이 줄어들어 가독성이 높아지고 유지보수가 쉬워졌음.

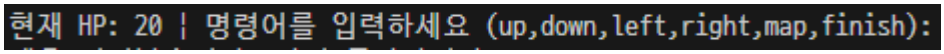
3. 설명
- moveUser 함수를 정의하여 이동 방향(dx, dy)을 인자로 받아 처리하는 방식으로 코드 중복을 제거하였음. 이 함수는 이동할 좌표가 유효한지 확인한 후 유효하면

좌표를 업데이트하고 체력을 감소시킴. 이를 통해 이동 관련 코드가 간결해짐.

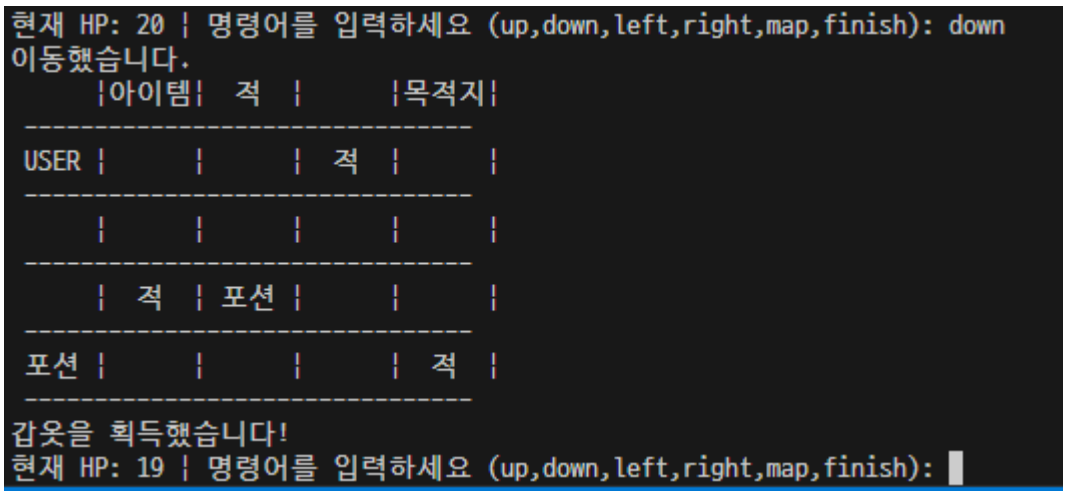
4. 테스트

-기능 별 테스트 결과

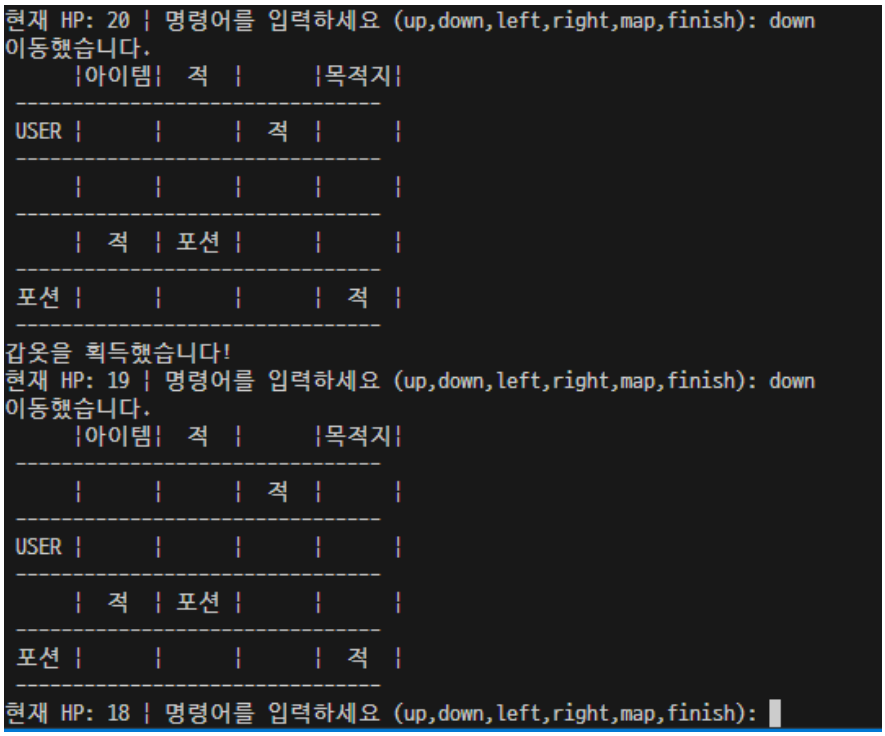
1. 유저는 체력 20을 가지고 게임을 시작합니다.



2. 사용자가 이동할때마다 체력이 1씩 감소합니다.



3. 처음 명령문을 입력 받을 때 마다 HP를 함께 출력합니다.



4. HP가 0이 되면 실패를 출력하고 종료합니다.

```
현재 HP: 3 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): down
이동했습니다.
     ¦아이템¦  적  ¦     ¦목적지¦
 -------------------------------
     ¦     ¦     ¦ 적 ¦     ¦
 -------------------------------
     ¦     ¦     ¦     ¦     ¦
 -------------------------------
     ¦ USER ¦ 포션 ¦     ¦     ¦
 -------------------------------
 포션 ¦     ¦     ¦     ¦ 적 ¦
 -------------------------------
적을 만났습니다! HP가 2 감소합니다.
HP가 0입니다. 실패했습니다.
게임을 종료합니다.
PS C:\CPP2409>
```

5. 무기/갑옷, 포션, 적을 만났을 때 그에 대한 메시지를 출력합니다. (적을 만날 경우 HP가 2 줄어들고 그에 대한 추가 메시지를 출력)

```
현재 HP: 20 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): down
이동했습니다.
     ¦아이템¦  적  ¦     ¦목적지¦
 -------------------------------
 USER ¦     ¦     ¦ 적 ¦     ¦
 -------------------------------
     ¦     ¦     ¦     ¦     ¦
 -------------------------------
     ¦ 적  ¦ 포션 ¦     ¦     ¦
 -------------------------------
 포션 ¦     ¦     ¦     ¦ 적 ¦
 -------------------------------
갑옷을 획득했습니다!
현재 HP: 19 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish):
```

```
현재 HP: 20 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): down
이동했습니다.
     ¦아이템¦  적  ¦     ¦목적지¦
 -------------------------------
 USER ¦     ¦     ¦ 적 ¦     ¦
 -------------------------------
     ¦     ¦     ¦     ¦     ¦
 -------------------------------
     ¦ 적  ¦ 포션 ¦     ¦     ¦
 -------------------------------
 포션 ¦     ¦     ¦     ¦ 적 ¦
 -------------------------------
무기를 획득했습니다!
```

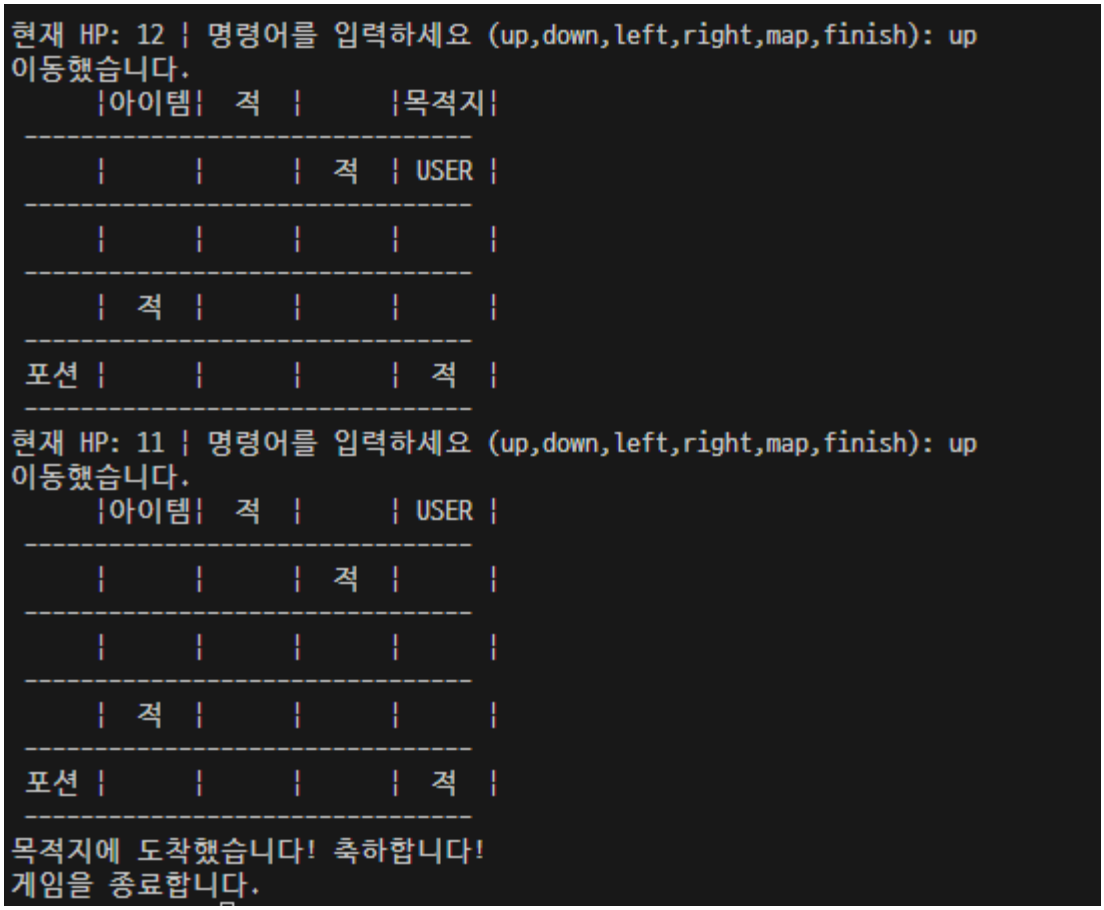
```
현재 HP: 17 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): right
이동했습니다.
     ¦      ¦ USER ¦      ¦목적지¦
 --------------------------------
     ¦      ¦      ¦  적  ¦      ¦
 --------------------------------
     ¦      ¦      ¦      ¦      ¦
 --------------------------------
     ¦  적   ¦ 포션 ¦      ¦      ¦
 --------------------------------
 포션 ¦      ¦      ¦      ¦  적  ¦
 --------------------------------
적을 만났습니다! HP가 2 감소합니다.
현재 HP: 14 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish):
```

```
현재 HP: 12 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): down
이동했습니다.
     ¦      ¦  적  ¦      ¦목적지¦
 --------------------------------
     ¦      ¦      ¦  적  ¦      ¦
 --------------------------------
     ¦      ¦      ¦      ¦      ¦
 --------------------------------
     ¦  적   ¦ USER ¦      ¦      ¦
 --------------------------------
 포션 ¦      ¦      ¦      ¦  적  ¦
 --------------------------------
포션을 발견했습니다! 체력이 2 회복됩니다.
현재 HP: 13 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish):
```

-최종 테스트 스크린샷

```
현재 HP: 20 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): down
이동했습니다.
     ¦아이템¦  적  ¦     ¦목적지¦
 ------------------------------
 USER ¦     ¦     ¦  적  ¦     ¦
 ------------------------------
      ¦     ¦     ¦     ¦     ¦
 ------------------------------
      ¦  적  ¦ 포션 ¦     ¦     ¦
 ------------------------------
 포션 ¦     ¦     ¦     ¦  적  ¦
 ------------------------------
무기를 획득했습니다!
현재 HP: 19 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): down
이동했습니다.
     ¦아이템¦  적  ¦     ¦목적지¦
 ------------------------------
      ¦     ¦     ¦  적  ¦     ¦
 ------------------------------
 USER ¦     ¦     ¦     ¦     ¦
 ------------------------------
      ¦  적  ¦ 포션 ¦     ¦     ¦
 ------------------------------
 포션 ¦     ¦     ¦     ¦  적  ¦
 ------------------------------
현재 HP: 18 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): down
이동했습니다.
     ¦아이템¦  적  ¦     ¦목적지¦
 ------------------------------
      ¦     ¦     ¦  적  ¦     ¦
 ------------------------------
      ¦     ¦     ¦     ¦     ¦
 ------------------------------
 USER ¦  적  ¦ 포션 ¦     ¦     ¦
 ------------------------------
 포션 ¦     ¦     ¦     ¦  적  ¦
 ------------------------------
현재 HP: 17 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): right
이동했습니다.
     ¦아이템¦  적  ¦     ¦목적지¦
 ------------------------------
      ¦     ¦     ¦  적  ¦     ¦
 ------------------------------
      ¦     ¦     ¦     ¦     ¦
 ------------------------------
      ¦ USER ¦ 포션 ¦     ¦     ¦
 ------------------------------
 포션 ¦     ¦     ¦     ¦  적  ¦
 ------------------------------
적을 만났습니다! HP가 2 감소합니다.
```

```
현재 HP: 14 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): right
이동했습니다.
     ¦아이템¦  적  ¦      ¦목적지¦
 ------------------------------
     ¦      ¦      ¦  적  ¦      ¦
 ------------------------------
     ¦      ¦      ¦      ¦      ¦
 ------------------------------
     ¦  적  ¦ USER ¦      ¦      ¦
 ------------------------------
 포션 ¦      ¦      ¦      ¦  적  ¦
 ------------------------------
포션을 발견했습니다! 체력이 2 회복됩니다.
현재 HP: 15 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): right
이동했습니다.
     ¦아이템¦  적  ¦      ¦목적지¦
 ------------------------------
     ¦      ¦      ¦  적  ¦      ¦
 ------------------------------
     ¦      ¦      ¦      ¦      ¦
 ------------------------------
     ¦  적  ¦      ¦ USER ¦      ¦
 ------------------------------
 포션 ¦      ¦      ¦      ¦  적  ¦
 ------------------------------
현재 HP: 14 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): right
이동했습니다.
     ¦아이템¦  적  ¦      ¦목적지¦
 ------------------------------
     ¦      ¦      ¦  적  ¦      ¦
 ------------------------------
     ¦      ¦      ¦      ¦      ¦
 ------------------------------
     ¦  적  ¦      ¦      ¦ USER ¦
 ------------------------------
 포션 ¦      ¦      ¦      ¦  적  ¦
 ------------------------------
현재 HP: 13 ¦ 명령어를 입력하세요 (up,down,left,right,map,finish): up
이동했습니다.
     ¦아이템¦  적  ¦      ¦목적지¦
 ------------------------------
     ¦      ¦      ¦  적  ¦      ¦
 ------------------------------
     ¦      ¦      ¦      ¦ USER ¦
 ------------------------------
     ¦  적  ¦      ¦      ¦      ¦
 ------------------------------
 포션 ¦      ¦      ¦      ¦  적  ¦
 ------------------------------
```



5. 결과 및 결론

-프로젝트 결과 : 요구하는 추가 기능들을 구현하여 mud 게임을 완성하였다.

-느낀 점 : 이번 프로젝트를 통해 코드 구조화의 중요성에 대해 알게 되었습니다. 이전 코드와 구조화한 코드를 봐봤더니 가독성이 확연히 좋아진 걸 느끼게 되었습니다. 또한 게임을 만들고 설계하는 과정에서 재미를 느껴 재미있었습니다. 그래서 기말 프로젝트를 게임쪽으로 진행해보고 싶다는 생각이 들었습니다.