**캡스톤 디자인**

최종 보고서

|  |
| --- |
| VR FPS |

**2019.12.13**

17013141 봉수연

16013083 이주형

16013096 임민규

16011185 송제윤

- 목 차 -

**Ⅰ 개요 ................................................................. 4**

1. 개발배경 ............................................................................... 4

2. 목표설정 및 개발 기간............................................................................... 5

3. 기대효과 및 마케팅............................................................................... 5

**Ⅱ 산출물 체계 ................................................................. 6**

**Ⅲ 산출물 양식 ................................................................. 7**

**1. 분석 단계** ............................................................................................... 7

1.1사용자 요구사항 정의서 ............................................................................... 7

1.2 유스케이스 명세서 ............................................................................... 9

1.3 요구사항 추적표 ............................................................................... 24

**2. 설계 단계** ............................................................................... 25

2.1설계클라스 목록 ............................................................................... 25

2.2 시퀀스도 ............................................................................... 27

2.3 설계 클래스도 ............................................................................... 40

2.4 설계 클래스 정의............................................................................... 41

**3. 구현 단계** ............................................................................... 46

3.1맵 구현 ............................................................................... 46

3.2 아이템 구현 ............................................................................... 47

3.3 캐릭터 구현 ............................................................................... 49

3.4 interaction 구현 ............................................................................... 37

3.5 네트워크 구현 ............................................................................... 38

3.6 UI구현 ............................................................................... 39

3.7 게임 scene 변환 구현 ............................................................................... 40

**4. 시험 단계** ............................................................................... 23

4.1 통합 시스템 결과서............................................................................... 23

4.2 사용자 지침서............................................................................... 40

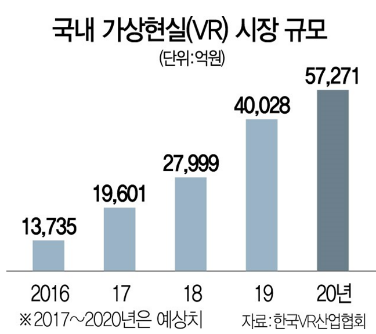
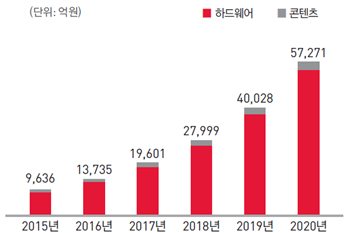
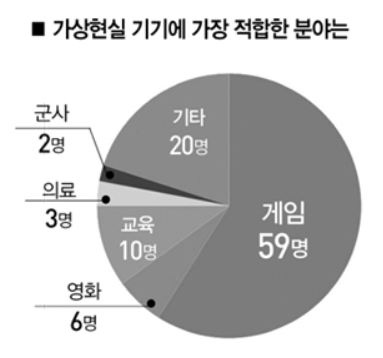
4.3 운영자 지침서 ............................................................................... 39

4.3.1 오큘러스 세팅 ............................................................................... 24

4.3.2 빌드 및 연동 ............................................................................... 37

**Ⅰ 개요**

**1. 개발 배경**



자료 : 헤럴드경제

[](http://www.google.com/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&ved=2ahUKEwjKx5uV0KrmAhVaeXAKHYBGDUwQjRx6BAgBEAQ&url=http%3A%2F%2Fwww.gametrics.com%2Fnews%2FNews02_View.aspx%3Fseqid%3D35112&psig=AOvVaw0KwGoBbHJupSXQoDBt1itm&ust=1576051666620504)다양한 VR HMD (Virtual Reality Head Mounted Display) 기기들이 앞다투어 등장했고, 그에 따라 수많은 VR 게임 컨텐츠들이 쏟아져 나왔습니다. 이 중에서 FPS(First Person Shooting) 장르는 VR기기에서 구현될 경우, 가장 크게 현실감을 느낄 수 있는 장르라고 여겨져 왔습니다. 실제로 VR HMD를 착용했을 때, 사용자는 VR HMD 내에 자신의 시야와 유사한 화면을 제공받을 수 있으므로 현실감이 매우 중요시되는 게임 장르 중 하나인, FPS 장르에 대한 잠재 수요가 매우 컸습니다.

또한 기존의 콘솔게임에서 FPS는 상당한 사랑을 받고 있는 게임 장르 중 하나입니다. 대표적으로 배틀 그라운드, 오버워치를 예를 들 수 있습니다. 이들의 유저는 작년 기준으로 전세계 이용자 수가 4억명, 4000만명을 넘어설 정도라고 합니다. 앞으로 콘솔게임이 고퀄리티 그래픽으로 VR시장이 더 발전하여 VR컨텐츠 시장 또한 활성화됨에 따라, 이러한 기존 FPS 콘솔 유저들이 자연스럽게 대거 유입 될 수 있을 것이라고 예상합니다.

따라서 저희는 게임 장르 중 제일 인기가 많은 장르 중 하나인 FPS를 저희만의 게임으로 만들어 보기로 했습니다. 평소에 사람들이 즐겨보는 영화, 드라마, 게임 등 다양한 매체에서 공통적으로 많은 사랑과 관심을 받는 장르인 공포(horror)에서 영감을 받아 공포을 컨셉으로 단순한 평면 스크린이나 모니터를 통한 감상이나 체험을 뛰어넘어 3D로 생동감 있게 공포를 즐길 수 있다면 많은 사랑을 받을 수 있겠다는 생각을 시작으로 본 프로젝트를 진행했습니다.

**2. 목표설정**

저희가 모티브로 따온 게임은 소위 ‘데바데’로 불리는 ‘DAY BY DEADLIGHT’ 게임과 모바일게임 제 ‘5인격’입니다. 이 두개의 게임은 순위권에 오를 정도로 인기가 있는 게임이라 이것을 바탕으로 FPS 요소를 추가하였습니다. 또한 어렸을 때 했던 게임처럼, 저희 게임만의 특색은 기존의 FPS게임 뿐만 아니라 아케이드적인 요소를 추가한 것에 있습니다. 슈팅 및 액션을 좋아하는 캐릭터인 헌터와 아케이드를 좋아하는 도둑을 선택하여 사용자들의 친숙한 상자잡기 아이템 습득 등을 통해 보다 폭넓은 사람들의 사랑을 받을 수 있는 게임으로 기획했습니다. 그리고 혼자만 즐기는 1인칭 슈팅게임이 아닌 실제 캐릭터가 되어 와이파이만 있다면, 어디서든 친구들과 함께 VR게임을 즐길 수 있도록 네트워크를 추가하는 것을 계획했습니다.

저희가 만들 게임의 이름은 ‘The thieves’로, 장르는 생존멀티플레이 비대칭 게임으로 pun2를 이용하여 네트워크 기반 멀티플레이를 게임으로 플레이 시간은 10-15분입니다.

소프트웨어 적인 부분은Unity 3D 엔진을 사용하여 개발하였고, HMD는 Oculus Quest를 사용합니다.

**3. 기대효과 및 마케팅**

데모영상 및 게임하는 영상 등을 유튜브, 트위치와 같은 게임 유저들이 많이 보는 플랫폼에 배포하여 인지도를 향상시키는 것을 목표로 마케팅 합니다. 실제로 오큘러스로 VR을 즐기는 사람들에게 오큘러스 스토어에 무료 게임으로 등록해 놓으므로 VR시장에서 FPS게임의 선호도가 높은 것에 대비하여 실제 FPS게임 시장점유율은 비교적 높지 않고 유료화가 많다는 점을 이용하여 ‘The Thieves’는 프리마케팅 전략을 사용합니다. 게임을 이용한 사용자들에게 기존에 있는 컨텐츠들과 다른 저희만의 차별성을 어필할 수 있습니다. 본 프로젝트 팀에 대한 인지도를 향상시키는 것을 목표로 하며, 게임은 무료로 제공하고, 게임을 위한 아이템 구입은 유료로 제공하는 등 차후에 진행하게 될 부분적 유료 컨텐츠에 대한 마케팅 기대효과를 확보할 것입니다.

**Ⅱ 산출물 체계**

|  |  |  |
| --- | --- | --- |
| 단계 | 코드 | 산출물 |
| 분석 | R1 | 사용자 요구사항 정의서 |
| R2 | 유스케이스 명세서 |
| R3 | 요구사항 추적표 |
| 설계 | D1 | 클래스 설계서 |
| 구현 | I1 | 맵 구현 |
| I2 | 아이템 구현 |
| I3 | 캐릭터 구현 |
| I4 | interaction 구현 |
| I5 | 네트워크 구현 |
| I6 | UI 구현 |
| I7 | 게임 scene 변환 구현 |
| 시험 | T1 | 단위 시험 결과서 |
| T2 | 통합시험 결과서 |
| T3 | 사용자 지침서 |
| T4 | 운영자 지침서 |
| T5 | 시스템 설치 결과서 |
| T6 | 인수시험 시나리오 |
| T7 | 인수시험 결과서 |

산출물 ID의 기본적인 부여 방법은 다음과 같이 제시한다. 단, 프로젝트에

별도의 산출물 ID 부여 방법이 존재할 경우는 프로젝트에서 정하는 바에 따라 사용할 수 있다.

프로젝트 ID + 분류 ID + 산출물 코드 + 순서번호

【산출물 ID 부여】

프로젝트 ID: SS

요구사항 ID: G(게임), C(일반), S(시스템), D(개발), P(프로젝트)

분류 ID: SUB-서브 시스템, USD – 유스케이스 다이어그램, UC – 유스케이스

DCD: 시퀀스도 SCD: 설계 클래스도

인 경우 산출물 ID는 다음과 같이 부여될 수 있다.

**Ⅲ. 산출물 양식**

**3.1 분석단계**

**3.1.1 사용자요구사항 정의서**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **R1** | **사용자 요구사항 정의서** | | | | | |
| 시스템명 | | Thiefs | 서브시스템명 | 도둑들 | | |
| 단계명 | | 분석 | 작성일자 | 2019.10.06 | 버전 | V1.0 |

**3.1.1.1 기능적 요구사항**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 요구사항 ID | 요구사항 명 | 구분 | 요구사항 설명 | 제약사항 | 해결방안 |
| C001 | 제한시간 | 유저(공통) | 제한시간인 10분을 나타내는 타이머 | x |  |
| C002 | 이동 | 유저(공통) | 휠을 이용해서 움직일 수 있는 기능 | x |  |
| C003 | 슈팅 | 유저(공통) | 버튼을 누를 경우 총을 발사하는 기능 | x |  |
| G001 | 상자 줍기 | 도둑 | 상자 쪽으로 조정해서 버튼을 이용하여 상자를 여는 기능 | 도둑만 가능 | 유저(공통) 클래스를 상속 |
| G002 | 아이템 줍기 | 도둑 | 아이템을 집고 해당 아이템 표시해주는 기능 | 도둑만 가능 | 유저(공통) 클래스를 상속 |
| G003 | 기절 | 도둑 | 술래에게 총 2방을 맞았을 경우, 기절(모든 키 제한)하게 된다. | 도둑만 가능 | 유저(공통) 클래스를 상속 |
| G004 | 도둑 잡기 | 헌터 | 술래가 버튼을 때리는 기능 | 헌터만 가능 | 유저(공통) 클래스를 상속 |
| G005 | 상자 | 아이템 | 상자를 열었을 때 뭐가 나올지 결정하는 기능과 술래에게 상자의 위치가 표시되는 기능(보물 상자 제외) | 도둑만 가능 |  |
| G006 | 독가스 | 아이템 | 상자가 열었을 때 독가스일 경우 도둑이 스턴에 걸리게 되는 기능 | 도둑만 가능 |  |
| G007 | 소생 총알 | 아이템 | 소생 총알을 사용시, 도둑이 풀려나는 기능 | 도둑만 가능 |  |
| G008 | 스턴 총알 | 아이템 | 스턴총알을 헌터에게 맞출 경우, 스턴에 걸리게 하는 기능 | 도둑만 가능 |  |
| G009 | 보물상자 | 아이템 | 보물 상자를 집었을 때 보물상자 위치가 발각되는 기능 | 도둑만 가능 |  |
| G010 | 음향효과 | 아이템 | 상황별 음향 효과 기능 | X |  |
| G011 | 피격 기능 | 유저(공통) | 총알을 맞았을 때 속성값 변화기능 | X |  |

**3.1.1.2 비기능적 요구사항**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 요구사항 ID | 요구사항 명 | 구분 | 요구사항 설명 | 제약사항 | 해결방안 |
| S001 | 성능 | 시스템 | 저사양 디바이스를 위한 성능 제한 | x |  |
| S002 | 용량 | 시스템 | 1GB를 넘지 않는 용량 제한 | x |  |
| S003 | 품질 | 시스템 | 3D에서 느낄 수 있는 어지러움증 개선 | x |  |
| P001 | 일정관리 | 프로젝트 | 정해진 기간안에 프로젝트가 완성 |  |  |
| D001 | H/W | 개발 | 게임을 개발하기 위한 하드웨어 요구사항 |  |  |
| D002 | S/W | 개발 | 게임을 개발하기 위한 소프트웨어 요구사항 |  |  |

**3.1.2 유스케이스 명세서**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **R2** | **유스케이스 명세서** | | | | | |
| 시스템명 | | Thiefs | 서브시스템명 | 도둑들 | | |
| 단계명 | | 분석 | 작성일자 | 2019.10.06 | 버전 | V1.0 |

**3.1.2.1서브시스템 목록**

|  |  |  |
| --- | --- | --- |
| 서브시스템 ID | 서브시스템명 | 서브시스템 설명 |
| SS\_SUB\_00\_01 | Grab\_box | 도둑만 상자를 잡을 수 있고 그랩 버튼 사용시 상자는 사라진다. |
| SS\_SUB\_00\_02 | Grab\_item\_01~04 | 도둑만 아이템을 잡을 수 있다. |
| SS\_SUB\_00\_03 | shot\_thief\_01 | 도둑만 쏠 수 있으며 헌터를 맞춘다. |
| SS\_SUB\_00\_04 | shot\_thief\_02 | 도둑만 쏠 수 있으며 기절 도둑을 맞춘다. |
| SS\_SUB\_00\_05 | shot\_hunter\_01 | 헌터만 쏠 수 있으며 도둑을 맞춘다. |
| SS\_SUB\_00\_06 | Trajectory\_stun | 맞은 액터 (헌터, 도둑)는 기능키 제한(5초간)된다. |
| SS\_SUB\_00\_07 | Trajectory\_recover | 목숨이 없는 도둑만 소생할 수 있다. |
| SS\_SUB\_00\_08 | hit\_hunter | 헌터만 때릴 수 있으며 도둑을 때린다. |
| SS\_SUB\_00\_09 | beat\_thief | 도둑만 발생하며 헌터가 때릴 때 목숨이 줄어든다. |
| SS\_SUB\_00\_10 | Shock | 도둑이 헌터에게 2번 맞고 모든(제한시간&시선 제외)키가 제한된다. |
| SS\_SUB\_00\_11 | shot\_possible | 도둑, 헌터 모두 사용 가능하며 방아쇠 버튼 시 탄환의 여부와 종류를 판단한다. |
| SS\_SUB\_00\_12 | bullet\_diminish | 총 사용 후 탄의 개수를 감소시킨다. |
| SS\_SUB\_00\_13 | trajectory\_appear | 탄의 종류에 따라 총구 앞에 스턴 궤적obj와 소생 obj를 생성시킨다. |
| SS\_SUB\_00\_14 | character\_move | 도둑, 헌터의 움직임 정보 |
| SS\_SUB\_00\_15 | character\_hunter | 헌터의 기본값 정보 |
| SS\_SUB\_00\_16 | character\_thief | 도둑의 기본값 정보 |
| SS\_SUB\_00\_16 | limit\_time | 제한시간 정보 |

**3.1.2.2 유스케이스 다이어그램(UCD)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 유스케이스 ID | 유스케이스명 | 유스케이스 설명 | 관련액터ID | 관련 SUB ID |
| SS\_USD\_00\_01 | Grab | 도둑이 상자를 열 때와 그 안의 아이템과의 관계 | Thief | SS\_SUB\_00\_01  SS\_SUB\_00\_02 |
| SS\_USD\_00\_02 | Shot | 도둑 과 헌터가 슈팅을 하였을 때의 관계 | Thief,hunter | SS\_SUB\_00\_03  SS\_SUB\_00\_04  SS\_SUB\_00\_05  SS\_SUB\_00\_11  SS\_SUB\_00\_12 |
| SS\_USD\_00\_03 | Shot\_effect | 도둑이 총을 쏠 때 어떤 종류의 총알인지의 따른 관계 | Thief,Hunter | SS\_SUB\_00\_06  SS\_SUB\_00\_07  SS\_SUB\_00\_13 |
| SS\_USD\_00\_04 | Hit | 헌터가 도둑을 때렸을 때의 관계 | Thief,Hunter | SS\_SUB\_00\_08  SS\_SUB\_00\_09  SS\_SUB\_00\_10 |
| SS\_USD\_00\_05 | Character | 도둑, 헌터의 기본값 정보 관계 | Thief,Hunter | SS\_SUB\_00\_14  SS\_SUB\_00\_15 |
| SS\_USD\_00\_06 | limit\_time\_gameover | 제한시간과 게임종료의 관계 | End\_system | SS\_SUB\_00\_16 |

|  |  |  |  |
| --- | --- | --- | --- |
| UCD ID | SS\_USD\_00\_01 | UCD 명 | Grab |
| 관련 서브시스템 ID | SS\_SUB\_00\_01  SS\_SUB\_00\_02 | 관련 서브시스템명 | Grab\_box  Grab\_item\_01~04 |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 유스케이스ID | SS\_UC\_00-01 | 유스케이스명 | 아이템 줍기 |
| **1. 주요 액터**  - 도둑, 시스템  **2. 이해관계자와 관심사항**  - 그랩 버튼을 눌렀을 때, 상자 obj가 사라져야 한다.  - 상자 obj가 사라지면서 item obj가 생성되어야 한다.  - 독가스가 생성될 시 도둑이 스턴에 걸려야 한다.  - 생성된 obj에 그랩 버튼을 누르면, 아이템이 주워진 걸로 처리된다.  **3. 전제조건**  - 상자 obj나 아이템 obj에 그랩 버튼을 누르면 이벤트가 발생해야 된다.  **4. 종료조건**  - 아이템에 obj에 그랩 버튼을 눌렀을 때 이벤트가 발생하고, 발생했다는 게  시스템에 전달되어야 한다  **5. 기본 시나리오**  1) 도둑이 지나가다가 상자를 발견한다.  2) 상자 근처로 가서 그랩 버튼을 사용한다.  3) 이벤트가 발생하면서, 상자가 사라진다.  4) 상자가 사라진 자리에 아이템 obj가 생성된다.  5) 독가스의 경우 도둑은 5초간 스턴에 걸린다.  6) 다른 아이템의 경우에는 도둑은 아이템 근처로 가서 그랩을 할 수 있다.  7) 그랩 버튼을 누르면, 눌렀다는게 시스템에 전송되고 아이템을 주운걸로 처리한다.  8) 아이템이 사라진다.  **6. 대안 시나리오**  1) 헌터가 상자에 그랩 버튼을 사용할 시에는 아무런 이벤트가 발생하지 않는  다.  2) 도둑이 나온 아이템에 그랩 버튼을 안 사용할 수도 있다. 7. 구현 시 고려사항 - Obj에 그랩 버튼의 작동유무  - 시스템에서 변경된 정보가 obj에 제대로 재현되는가  - 독가스가 나올 경우, 즉발적으로 스턴에 걸리는가 8. 발생 빈도 - 게임내에 배치된 상자의 개수 | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| UCD ID | SS\_USD\_00\_02 | UCD 명 | shot |
| 관련 서브시스템 ID | SS\_SUB\_00\_03  SS\_SUB\_00\_04  SS\_SUB\_00\_05  SS\_SUB\_00\_11  SS\_SUB\_00\_12 | 관련 서브시스템명 | shot\_thief\_01  shot\_thief\_02  shot\_hunter\_01  shot\_possible  bullet\_diminish |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 유스케이스ID | SS\_UC\_00-02 | 유스케이스명 | 총 맞은 후 효과 |
| **1. 주요 엑터**  - 도둑, 헌터  **2. 이해관계자와 관심사항**  - 슈팅하는 버튼을 누르면 발생 여부에 따라 이벤트 진행  - 총알이 발사되면서 궤적도 obj 생성.  - 총알이 발사되면서 애니메이션 효과 생성  - 총알이 발사되면서 슈팅 effect 생성  - 총알이 발사되면서 해당 액터의 총알 개수 감소  **3. 전제조건**  - 버튼을 누르면 이벤트가 발생해야 된다.  **4. 종료조건**  - 총을 쏜 후 총알의 개수가 줄어든 것이 헌터나 도둑의 정보에 반영되어야 한다.  **5. 기본 시나리오**  1) 헌터나 도둑이 상대를 향해 겨냥하고 버튼을 누른다.  2) 발생여부에 따라, 총알이 없으면 아무런 이벤트가 발생하지 않는다.  3) 총알이 있을 경우, 총알이 발사되면서 애니메이션 효과 발생  4) 궤적도 obj가 생성된다.  5) 슈팅 effect 생성된다.  6) 해당 액터의 총알 개수가 감소한다.  **6. 대안시나리오**  **7. 구현 시 고려사항**  - 버튼의 작동유무  - 총을 쏘고 난 후에 총알에 개수 같은 정보가 바로 헌터나 도둑에게  적용되는가.  **8. 발생 빈도**  - 총을 쏠 때마다 발생 | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| UCD ID | SS\_USD\_00\_03 | UCD 명 | Shot\_effect |
| 관련 서브시스템 ID | SS\_SUB\_00\_06  SS\_SUB\_00\_07  SS\_SUB\_00\_13 | 관련 서브시스템명 | Trajectory\_stun  Trajectory\_recover  Trajectory\_appear |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 유스케이스ID | SS\_UC\_00-03 | 유스케이스명 | 총 맞은 후 효과 |
| **1. 주요 액터**  - 궤적도 obj, 총에 맞는 obj  **2. 이해관계자와 관심사항**  - 궤적도 obj랑 총에 맞은 obj 충돌 여부 확인.  - 스턴 총알에 충돌했을 경우 해당 obj 5초간 스턴.  - 소생 총알과 충돌했을 경우 해당 obj의 기능 제한 해제.  **3. 전제조건**  - Obj와 obj의 충돌함수가 구현되어 있어야한다.  **4. 종료조건**  - 총알에 맞은 후 나타나는 효과가 총알에 맞은 obj에 구현되면 종료.  **5. 기본 시나리오**  1) 궤적도 obj가 플레이어 obj에 충돌함.  2) 발사된 총알이 무슨 총알인지 여부를 따짐.  3) 스턴 총알일 경우에는 해당 obj 기능을 5초간 제한.  4) 소생 총알일 경우 해당 obj 목숨 값을 확인.  5) 목숨 값이 0일때는 맞은 obj에 기능키제한 해제  **6.대안시나리오**  - 발사된 궤적도 obj가 플레이어 obj와 충돌하지 않을 경우에는 아무런  일도 발생하지 않는다  - 소생 총알을 맞았을 때 기절 상태가 아닐 경우에는 아무런 효과가 일어나지 않는다.  **7. 구현 시 고려사항**  - 궤적도 obj의 길이가 어느 정도까지 구현해야 할지 고려해야 된다.  - 총을 쏘고 나서 생겨난 궤적도 obj가 몇 초뒤에 사라질지도 고려해야 된다.  **8. 발생 빈도**  - 총을 쏠 때마다 발생 | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| UCD ID | SS\_USD\_00\_04 | UCD 명 | Hit |
| 관련 서브시스템 ID | SS\_SUB\_00\_08  SS\_SUB\_00\_09  SS\_SUB\_00\_10 | 관련 서브시스템명 | hit\_hunter  beat\_thief  Shock |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 유스케이스ID | SS\_UC\_00-04 | 유스케이스명 | 때리기 |
| **1. 주요 액터**  - 헌터, 도둑  **2. 이해관계자와 관심사항**  - 헌터가 도둑 근처에 있을 때, 그랩 버튼을 누르면 이벤트 발생.  - 이벤트 발생으로 헌터 hit 애니메이션  - 도둑 obj랑 충돌했는지 여부.  - 맞았을 경우 도둑 obj 체력 감소.  - 도둑의 목숨이 0이되면 도둑 obj의 모든 기능키 제한.  **3. 전제조건**  - Obj와 obj의 충돌함수가 구현되어 있어야한다.  **4. 종료조건**  - 도둑이 맞은 다음, 감소된 목숨 값이 도둑 obj에 전달되면 종료.  **5. 기본 시나리오**  1) 헌터가 도둑을 쫓다가 때리면 닿을 거리가 되면 그랩 버튼을 누른다.  2) 때리는 애니메이션이 구현된다.  3) 헌터 딜레이가 발생한다.  4) 만약 때리는 모션과 도둑 obj가 충돌했을 경우 맞은 도둑 obj 체려감소.  5) 목숨 값이 0일때는 맞은 도둑 obj의 기능 제한  **6. 대안 시나리오**  - 헌터가 도둑을 못 맞추면 아무런 일도 일어나지 않는다.  **7. 구현 시 고려사항**  - 헌터 hit 애니메이션을 구현할 때 히트박스의 영역을 고려해봐야 된다.  **8. 발생 빈도**  - 헌터가 hit 버튼을 누를 때마다 발생. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| UCD ID | SS\_USD\_00\_05 | UCD 명 | character |
| 관련 서브시스템 ID | SS\_SUB\_00\_14  SS\_SUB\_00\_15  SS\_SUB\_00\_16 | 관련 서브시스템명 | character\_move  character\_hunter  character\_thief |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 유스케이스ID | SS\_UC\_00-05 | 유스케이스명 | 캐릭터 |
| **1. 주요 액터**  - 헌터, 도둑  **2. 이해관계자와 관심사항**  - 각 obj는 공통적으로 총알에 대한 정보를 가지고 있어야한다.  - 도둑과 헌터의 이동속도가 달라야한다.  - 맞았을 경우 도둑 obj 목숨이 감소해야한다.  - 도둑의 목숨이 0이되면 도둑 obj의 모든 기능키 제한.  **3. 전제조건**  - 도둑과 헌터에 해당하는 obj가 개별적으로 구현되어 있어야한다.  **4. 종료조건**  - 게임이 종료되면 각 obj는 기능을 종료한다.  **5. 기본 시나리오**  1) 헌터가 도둑을 쫓다가 때리면 닿을 거리가 되면 그랩 버튼을 누른다.  2) 도둑이 맞았을 경우 도둑 obj내에 목숨 값이 감소한다.  3) 기절이 되었을 경우 도둑 obj 내에 이동속도는 0이 된다.  4) 총을 사용시 도둑, 헌터 obj내에 총알에 대한 정보가 변경된다.  5) 도둑 obj의 목숨 값이 0일경우 소생 총알을 통하여 값이 1 증가한다.  **6. 대안 시나리오**  **7. 구현 시 고려사항**  - 도둑 obj의 목숨 값이 0이 아닐 때에 소생 총알을 통한 목숨 값 증가를 제한한다.  **8. 발생 빈도**  - 총을 쏘거나 hit가 발생하면 일어난다. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| UCD ID | SS\_USD\_00\_06 | UCD 명 | limit\_time\_gameover |
| 관련 서브시스템 ID | SS\_SUB\_00\_16 | 관련 서브시스템명 | limit\_time |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 유스케이스ID | SS\_UC\_00-06 | 유스케이스명 | 시간제한 |
| **1. 주요 액터**  - 헌터, 도둑  **2. 이해관계자와 관심사항**  - 제한시간이 0이 되면 도둑과 헌터의 obj내에 이동속도가 0이된다.  - 모든 동작을 제한하고 나면 end\_system에 돌입하게 된다.  **3. 전제조건**  - 게임이 종료되기 위하여 제한시간을 이용한 종료함수가 필요하다.  **4. 종료조건**  - 제한시간이 0이 되면 종료  **5. 기본 시나리오**  1) 도둑들의 목숨 값이 0이 되기전까지 제한시간은 0으로 수렴한다.  2) 제한시간이 0이되는 순간 플레이어 obj들의 이동속도는 모두 0이된다.  3) 게임 end\_system을 호출하여 게임을 정상적으로 종료한다.  **6. 대안 시나리오**  **7. 구현 시 고려사항**  - 제한시간이 남아있어도 도둑들의 모든 obj의 목숨 값이 0 이되면 종료되어야한다.  **8. 발생 빈도**  - 게임에 정해진 시간이 되었을 때 1번 발생 | | | |

**3.1.2.3 액터 목록**

|  |  |  |  |
| --- | --- | --- | --- |
| 액터 ID | 액터명 | 액터유형 | 액터설명 |
| End\_system | 게임종료시스템 | 보조 | 게임이 조건에 달하면, 게임을 종료시킨다. |
| Time | 시계 | 보조 | 10분으로 시작하여 0분시  End\_system 를 부른다. |
| Thief | 도둑 플레이어 | 주요 | 게임내에서 도둑을 플레이하는 사람  (목숨 값, 스피드, 총알 유무, 총알 종류, 보물 보유) |
| Hunter | 헌터 플레이어 | 주요 | 게임내에서 헌터를 플레이하는 사람  (스피드, 탄 개수) |
| Trajectory  \_stun\_obj | 스턴총알의 궤적 | 주요 | Shot\_effect를 주는 object |
| Trajectory  \_recovery\_obj | 소생총알의 궤적 | 주요 | Shot\_effect를 주는 object |

**3.1.2.4 통합 유스케이스 다이어그램**

|  |
| --- |
| **텍스트, 지도이(가) 표시된 사진  자동 생성된 설명** |

**3.1.3 요구사항 추적도**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **R3** | **요구사항 추적도** | | | | | |
| 시스템명 | | Thiefs | 서브시스템명 | 도둑들 | | |
| 단계명 | | 분석 | 작성일자 | 2019.10.06 | 버전 | V1.0 |

|  |  |  |  |
| --- | --- | --- | --- |
| **분석단계** | | | |
| **사용자 요구사항 명세서** | | **유스케이스 모형 기술서** | |
| **요구사항 ID** | **요구사항 명** | **유스케이스 ID** | **유스케이스 명** |
| G001, G002 | 상자열기, 아이템 줍기 | SS\_USD\_00\_01 | Grab |
| C003 | 슈팅 | SS\_USD\_00\_02 | Shot |
| G007, G008, G011 | 소생, 스턴, 피격기능 | SS\_USD\_00\_03 | Shot\_effect |
| G003, G004 | 기절, 도둑 잡기 | SS\_USD\_00\_04 | Hit |
| C002 | 이동 | SS\_USD\_00\_05 | character |
| C001 | 제한시간 | SS\_USD\_00\_06 | limit\_time\_gameover |

**3.2 설계단계**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **D1** | **클래스 설계서** | | | | | |
| 시스템명 | | Thiefs | 서브시스템명 | 도둑들 | | |
| 단계명 | | 설계 | 작성일자 | 2019.10.06 | 버전 | V1.0 |

**3.2.1 설계 클래스 목록**

|  |  |  |
| --- | --- | --- |
| **설계 클래스 ID** | **설계 클래스명** | **관련 유스케이스ID** |
| **SS\_DCD\_00\_01** | **도둑 기절 정보** | SS\_USD\_00\_02  SS\_USD\_00\_04 |
| **SS\_DCD\_00\_02** | **히트 정보** | SS\_USD\_00\_04 |
| **SS\_DCD\_00\_03** | **독가스 상자 정보** | SS\_USD\_00\_01 |
| **SS\_DCD\_00\_04** | **보물 상자 정보** | SS\_USD\_00\_01 |
| **SS\_DCD\_00\_05** | **상자 정보** | SS\_USD\_00\_01 |
| **SS\_DCD\_00\_06** | **상자 줍기 정보** | SS\_USD\_00\_01 |
| **SS\_DCD\_00\_07** | **소생 총알 정보** | SS\_USD\_00\_01 |
| **SS\_DCD\_00\_08** | **스턴 총알 정보** | SS\_USD\_00\_01 |
| **SS\_DCD\_00\_09** | **아이템 줍기 정보** | SS\_USD\_00\_01 |
| **SS\_DCD\_00\_10** | **이동 정보** | SS\_USD\_00\_05 |
| **SS\_DCD\_00\_11** | **제한시간 정보** | SS\_USD\_00\_06 |
| **SS\_DCD\_00\_12** | **총알 슈팅 정보** | SS\_USD\_00\_02  SS\_USD\_00\_03 |
| **SS\_DCD\_00\_13** | **캐릭터 정보** | SS\_USD\_00\_05 |

**3.2.2 시퀀스도**

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_01** | **시퀀스도명** | **도둑 기절 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_02, SS\_USD\_00\_04** | | |
| **주요 액터** | **thief** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_02** | **시퀀스도명** | **히트 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_04** | | |
| **주요 액터** | **hunter, thief** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_03** | **시퀀스도명** | **독가스 상자 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_01** | | |
| **주요 액터** | **thief** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_04** | **시퀀스도명** | **보물 상자 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_01** | | |
| **주요 액터** | **thief, hunter** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_05** | **시퀀스도명** | **상자 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_01** | | |
| **주요 액터** | **thief** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_06** | **시퀀스도명** | **상자 줍기 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_01** | | |
| **주요 액터** | **thief** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_07** | **시퀀스도명** | **소생 총알 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_01** | | |
| **주요 액터** | **thief** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_08** | **시퀀스도명** | **스턴 총알 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_01** | | |
| **주요 액터** | **thief, hunter,**  **Trajectory\_stun\_obj** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_09** | **시퀀스도명** | **아이템 줍기 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_01** | | |
| **주요 액터** | **thief** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_10** | **시퀀스도명** | **이동 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_05** | | |
| **주요 액터** | **thief, hunter** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_11** | **시퀀스도명** | **제한시간 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_06** | | |
| **주요 액터** | **thief,hunter,End\_system** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_12** | **시퀀스도명** | **총알 슈팅 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_02, SS\_USD\_00\_03** | | |
| **주요 액터** | **thief, hunter** | **주요 클래스** |  |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **시퀀스도 ID** | **SS\_DCD\_00\_13** | **시퀀스도명** | **캐릭터 정보** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_05** | | |
| **주요 액터** | **thief, hunter** | **주요 클래스** |  |
|  | | | |

**3.2.3 설계 클래스도**

|  |  |  |  |
| --- | --- | --- | --- |
| **설계 클래스도 ID** | **SS\_SCD\_00\_01** | **설계 클래스도명** | **통합 클래스** |
| **관련 유스케이스 ID** | **SS\_USD\_00\_01~06** | | |
|  | | | |
|

**3.2.4 설계 클래스 정의**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 클래스 id | SS\_DCD\_001\_01\_01 | | 클래스명 | End\_system |
| 속성 | | | | |
| 속성명 | 가시성 | 타입 | 기본값 | 설명 |
| Limit time | public | int | N/A | stopwatch 클래스를 이용하여 제한시간을 준다. |
| Theif\_life | protected | int | N/A | 도둑의 남은 목숨 수 |
| treasure | private | bool | N/A | 보물의 여부 |
| 오퍼레이션 | | | | |
| 오퍼레이션명 | 가시성 | 파라미터 | 반환타입 | 설명 |
| case\_win\_hunter | public | limit time  Theif\_life | void | 헌터가 승리하였을 때  표시한다. |
| case\_win\_thief | public | treasure | void | 도둑이 승리하였을 때  표시한다 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 클래스 id | SS\_DCD\_001\_01\_02 | | 클래스명 | Hunter |
| 속성 | | | | |
| 속성명 | 가시성 | 타입 | 기본값 | 설명 |
| Limit time | public | int | N/A | stopwatch 클래스를 이용하여 제한시간을 준다. |
| speed\_hunter | protected | int | N/A | 헌터의 이동속도 |
| wait\_bullet\_time | private | int | N/A | 총알의 대기시간 |
| 오퍼레이션 | | | | |
| 오퍼레이션명 | 가시성 | 파라미터 | 반환타입 | 설명 |
| shot | public | wait\_bullet\_time | void | 슈팅 |
| shot\_effect | public |  | void | 헌터가 총에 맞았을 때 일어나는 영향 |
| Hit | public |  | void | 헌터가 도둑을 때릴 시 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 클래스 id | SS\_DCD\_001\_01\_03 | | 클래스명 | thief |
| 속성 | | | | |
| 속성명 | 가시성 | 타입 | 기본값 | 설명 |
| Limit time | public | int | N/A | stopwatch 클래스를 이용하여 제한시간을 준다. |
| speed\_thief | protected | int | N/A | 도둑의 이동속도 |
| bullet | private | int | N/A | 총알의 종류 |
| Theif\_life | protected | int | N/A | 도둑의 남은 목숨 수 |
| treasure | private | bool | N/A | 보물의 여부 |
| 오퍼레이션 | | | | |
| 오퍼레이션명 | 가시성 | 파라미터 | 반환타입 | 설명 |
| Grab | public | bullet  treasure | void | 상자& 아이템 습득 |
| shot | public |  | void | 슈팅 |
| shot\_effect | public |  | void | 도둑이 총에 맞았을 때 일어나는 영향 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 클래스 id | SS\_DCD\_001\_01\_04 | | 클래스명 | Hit |
| 속성 | | | | |
| 속성명 | 가시성 | 타입 | 기본값 | 설명 |
| Hit\_action | public |  | N/A | 그랩버튼 사용시 헌터가 때리는 모션을 취한다. |
| Thief\_life | protected | int | N/A | 도둑의 남은 목숨 수 |
| 오퍼레이션 | | | | |
| 오퍼레이션명 | 가시성 | 파라미터 | 반환타입 | 설명 |
| Theif\_crash | public | Thief\_life | void | 도둑의 목숨이 줄어든다. |
| controll\_limit | public | Thief\_life | void | 목숨이 0이 된 도둑은  모든 기능 키를 제한받는다. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 클래스 id | SS\_DCD\_001\_01\_05 | | 클래스명 | Grab |
| 속성 | | | | |
| 속성명 | 가시성 | 타입 | 기본값 | 설명 |
| box[obj] | protected |  | N/A | 상자가 사라지고 아이템이 나타난다. |
| item | protected | int | N/A | 해당 아이템에게 부여된 번호에 맞는 속성 값 부여 |
| 오퍼레이션 | | | | |
| 오퍼레이션명 | 가시성 | 파라미터 | 반환타입 | 설명 |
| Grab\_box | public | box[obj] | void | 상자 오브젝트가 사라지고 아이템이 나타난다. |
| Grab\_item | public | Thief\_life | void | 각 아이템 고유의 속성 값을 받는다. |
| shot\_effect\_stun | public | Thief | void | 독가스의 경우  스턴효과가 발생한다. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 클래스 id | SS\_DCD\_001\_01\_06 | | 클래스명 | shot |
| 속성 | | | | |
| 속성명 | 가시성 | 타입 | 기본값 | 설명 |
| shot\_action | public |  | N/A | 슈팅버튼 사용 시  슈팅하는 모션을 취한다. |
| Trajectory\_stun[obj] | public |  | N/A | 스턴총알의 궤적도 생성 |
| Trajectory\_recover[obj] | public |  | N/A | 소생총알의 궤적도 생성 |
| Distance | private | int | N/A | 헌터와 도둑간의 거리 |
| Stun\_bullet\_count | public | int | N/A | 스턴총알갯수 |
| Recover\_bullet\_count | public | int | N/A | 소생총알갯수 |
| 오퍼레이션 | | | | |
| 오퍼레이션명 | 가시성 | 파라미터 | 반환타입 | 설명 |
| Shot\_possible | public | Stun\_bullet\_count  Recover\_bullet\_count | void | 총알 갯수를 확인 후,  총을 쏠 수 있는 지 확인한다. |
| shot\_effect\_stun | public | Stun\_bullet\_count | void | 맞은 상대가 일정시간동안 스턴된다. |
| shot\_thief\_01 | public | Stun\_bullet\_count | void | 스턴 총알의 갯수가 감소한다. |
| shot\_thief\_02 | public | Recover\_bullet\_count | void | 소생 총알의 갯수가 감소한다. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 클래스 id | SS\_DCD\_001\_01\_06 | | 클래스명 | shot\_effect |
| 속성 | | | | |
| 속성명 | 가시성 | 타입 | 기본값 | 설명 |
| shot\_effect\_action | public |  | N/A | 총알에 맞았을때의 해당되는 모션을 취한다. |
| Trajectory[obj]] | public |  | N/A | 궤적도 오브젝트 |
| Distance | private | int | N/A | 헌터와 도둑간의 거리 |
| stun\_bullet | public | int | N/A | 스턴 총알 |
| recover\_bullet | public | int | N/A | 소생 총알 |
| 오퍼레이션 | | | | |
| 오퍼레이션명 | 가시성 | 파라미터 | 반환타입 | 설명 |
| shot\_effect\_stun | public | Stun\_bullet | void | 맞은 상대가 일정시간동안 스턴된다. |
| shot\_effect\_recover | public | recover\_bullet | void | 목숨이 0인 상대의 목숨이 1증가한다. |
| release\_limit | public | recover\_bulletThief | void | 목숨이 0이였던 도둑의 기능키 제한이 없어진다. |

**3.3구현단계**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I1** | **맵 구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | | 충돌감지 및 성능 최적화 | | |
| 관련 유스케이스 ID | | - | | |
| 구현 설명 | | 게임에 필요한 Map에 필요한 setting값을 넣어줍니다. | | |
| 프로그램 파일 | | | Inspector & setting  [Add Component] | 비고 |
| ID | 파일명(Assets) | |
| Map | Mega Fantasy Props Pack | | Mesh Renderer  Mesh Collider  Box collider  Rigidbody |  |
| Mapsky | UrbanNightSky | | Skybox |  |
| MapAudio | HorrorElements | | Audio Source |  |

2. 구현 모습

|  |
| --- |
|  |
| 그림 1.1 맵 충돌 감지 값 적용 |
| 캐릭터들이 한정적으로 돌아다닐 맵의 사이즈를 결정하고 컴포넌트를 추가하여 각각의 성과 성벽, 집과 여러 아이템에 collider(충돌체)값과 rigidbody(물리력을 영향받도록 하는 컴포넌트)값에 mass(질량)과 gravity(중력)을 필요한 부분에 개별적으로 추가합니다. |

|  |
| --- |
|  |
| 그림 1.2 light bake 및 성능 최적화 |
| Directional Light만 mode를 Realtime두고 나머지 Point Light 및 Spot Light는 bake모드로 soft shadow를 만들어 주도록 한다. 이때 캐릭터를 제외한 모든 물체들은 Static으로 두어 bake할 수 있도록 한다.  Realtime된 조명은 매 프레임마다 조명 연산이 되어 화면을 업데이트 하기 때문에 연산 부하가 크고 간접 광원을 사용하지 않아 그림자가 검은색으로 표현된다. 따라서 성능 최적화를 위해 대부분의 조명은 bake한다.  bake된 조명은 런타임시 따로 연산을 하지 않아 표면의 조명과 그림자를 생성한다. |

3.고려사항

|  |
| --- |
| 그림1.1에서의 고려사항은 collider를 전부 설정해 주면 바닥과 문에도 충돌 값이 발생하여 진입하지 못하게 되기 때문에 바닥과 문이 있는 부분에는 is trigger를 설정해주어 충돌 값을 발생하지 않도록 설정해주거나 collider를 추가하지않아야 하므로 개별적으로 설정해주었다. 그림1.2에서의 고려사항은 성능 저하를 방지하기 위해 실시간 라이트 매핑 옵션을 비활성화 해야 하며 Lightmap Resolution(유닛 당 텍셀 값으로 라이트의 품질과 베이크시간이 비례한다)과 lIghtmap Size(라이트맵의 크기)를 하드웨어의 성능에 맞게 적절한 값을 찾아야했다. |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I2** | **아이템 구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | | 상자 이벤트 발생 및 아이템 습득 | | |
| 관련 유스케이스 ID | | SS\_UC\_00-01 | | |
| 구현 설명 | | 도둑이 상자에 다가가 특정 키를 누르면 상자가 열리고 해당아이템이 나타난다. 2가지의 bullet아이템은 특정키를 눌러 습득이 가능하며 독가스 상자의 경우는 상자가 열림과 동시에 5초간 기능키가 제한된다. | | |
| 서브시스템명 | 서브시스템 ID | 프로그램 파일 | | 비고 |
| ID | 파일명(Assets) |
| Grab\_box | SS\_SUB\_00\_01 | Box[prefab] | Treasure chest closed | Audio effect |
| Grab\_item | SS\_SUB\_00\_02 | stunBullet[prefab]  recoverBullet[prefab]  TreasureBox[prefab]  파티클뭐시기써야함 | Ammunition pack (demo)  ChestFree |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| BoxDisapper script |

도둑이 상자에 다가가 F 키를 누르면 Gameobiect 뭐뭐뭐에 해당하는 아이템이 나타나게 된다.

3. 구현 모습

|  |
| --- |
|  |
| 그림 1.3 BOX |
| . |

4. 고려사항

|  |
| --- |
| . |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I3** | **캐릭터 구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | |  | | |
| 관련 유스케이스 ID | | SS\_USD\_00\_05 | | |
| 구현 설명 | | 도둑 캐릭터 기본정보 | | |
| 서브시스템명 | 서브시스템 ID | 프로그램 파일 | | 비고 |
| ID | 파일명(Assets) |
| character\_move | SS\_SUB\_00\_14 | Thief(prefabs) | Thief\_move(script) |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Thief\_move(Script) |

그림 1.1 Thief의 기본 정보 코드

4. 고려사항

|  |
| --- |
| 도둑의 기본 정보 값으로 이동속도(moveSpeed)는 5f, 회전속도(turnspeed)는 350f이며 스턴 총알의 변수는 stun이고 최초 5개를 주어 총을 쏘는 행위를 테스트 할 수 있게 주었고 recover총알은 0개, 목숨값은 heart이며 개수는 5개로 설정하였다.  . |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I3** | **캐릭터 구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | |  | | |
| 관련 유스케이스 ID | | SS\_USD\_00\_02 | | |
| 구현 설명 | | 도둑이 총을 쏠때의 작동구현 | | |
| 서브시스템명 | 서브시스템 ID | 프로그램 파일 | | 비고 |
| ID | 파일명(Assets) |
| trajectory\_appear | SS\_SUB\_00\_13 | Thief(prefabs) | Thief\_Fire(script) |  |
| bullet\_diminish | SS\_SUB\_00\_12 | Thief(prefabs) | Thief\_Fire(script) |  |
|  |  |  |  |  |
|  |  |  |  |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Thief\_fire(Script) |

그림 1.1 오른쪽 방아쇠를 누를 시 총알 생성

|  |
| --- |
|  |
| Thief\_move(Script) |

그림1.1 도둑의 기본정보 값 중 총알개수 줄이기

4. 고려사항

|  |
| --- |
| Thief\_Fire스크립트에서 오큘러스 퀘스트 오른쪽 방아쇠 버튼을 누를 시 s\_bullet혹은 r\_bullet prefabs를 총구 앞에 배치 된다. 그후 Thief\_move스크립트의 기본 정보값의 stun,혹은 recover의 개수를 줄여준다. |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I3** | **캐릭터 구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | |  | | |
| 관련 유스케이스 ID | | SS\_USD\_00\_05 | | |
| 구현 설명 | | 헌터 기본값 정보 | | |
| 서브시스템명 | 서브시스템 ID | 프로그램 파일 | | 비고 |
| ID | 파일명(Assets) |
| character\_move | SS\_SUB\_00\_14 | hunter(prefabs) | hunter\_move(script) |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Hunter\_move(script) |

1.1 헌터 기본정보

4. 고려사항

|  |
| --- |
| 도둑의 기본 정보 값으로 이동속도(moveSpeed)는 7f, 회전속도(turnspeed)는 350f이며 스턴 총알의 변수는 stun이고 최초15개를 주어 총을 쏘는 행위를 테스트 할 수 있게 설정해 주었다. |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I3** | **캐릭터 구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | | SS\_USD\_00\_02 | | |
| 관련 유스케이스 ID | | 헌터의 총을 쏠때의 작동구현 | | |
| 구현 설명 | |  | | |
| 서브시스템명 | 서브시스템 ID | 프로그램 파일 | | 비고 |
| ID | 파일명(Assets) |
| trajectory\_appear | SS\_SUB\_00\_13 | Hunter(prefabs) | Hunter\_Fire(script) |  |
| bullet\_diminish | SS\_SUB\_00\_12 | hunter(prefabs) | Hunter\_Fire(script) |  |
|  |  |  |  |  |
|  |  |  |  |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Hunter\_Fire(script) |

주요코드를 편집하여 넣을 것 그림 1.1 충돌처리 코드

설명을 쓸 것 설명: 그림1.1에서 ~를 선언하여 ~값을 넣어 ~설정해준다 라고 쓰거나  
시험항목 및 처리절차에 관하여 쓸 것

3. 구현 모습

|  |
| --- |
|  |
| Hunter\_move(script) |
| . |

4. 고려사항

|  |
| --- |
| hunter\_Fire스크립트에서 오큘러스 퀘스트 오른쪽 방아쇠 버튼을 누를 시 s\_bullet 총구 앞에 배치 된다. 그후 hunter\_move스크립트의 기본 정보값의 stun의 개수를 줄여준다.  . |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I4** | **Interaction구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | |  | | |
| 관련 유스케이스 ID | | SS\_USD\_00\_02 | | |
| 구현 설명 | | 스턴 총알을 맞을 때 의 헌터와 도둑의 제한 | | |
| 서브시스템명 | 서브시스템 ID | 프로그램 파일 | | 비고 |
| ID | 파일명(Assets) |
| Trajectory\_stun | SS\_SUB\_00\_06 | Thief(prefabs) | thief\_collider(script) |  |
| Trajectory\_stun | SS\_SUB\_00\_06 | Hunter(prefabs) | Hunter \_collider(script) |  |
|  |  |  |  |  |
|  |  |  |  |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Thief\_collider(Script) |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Hunter\_collider(script) |

----

4. 고려사항

|  |
| --- |
| 헌터와 도둑이 발사한 총알이 헌터 혹은 도둑에 맞을 시 hunter\_move혹은 thief\_move안의 stun함수를 불러온다. 함수의 내용은 5초간 멈추게 한다.  . |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I4** | **Interaction구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | |  | | |
| 관련 유스케이스 ID | | SS\_USD\_00\_02 | | |
| 구현 설명 | | 회복 총알을 맞을 때 의 도둑의 life개수 증가 | | |
| 서브시스템명 | 서브시스템 ID | 프로그램 파일 | | 비고 |
| ID | 파일명(Assets) |
| Trajectory\_stun | SS\_SUB\_00\_06 | Thief(prefabs) | thief\_collider(script) |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Thief\_collider(Script) |

4. 고려사항

|  |
| --- |
| 다른 도둑이 총알을 쏘아 다른 도둑이 맞을경우 thief\_move내의 heartplus함수를 불러온다. 함수에는 맞은 도둑의 life의 개수가 증가한다.  . |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I4** | **Interaction구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | |  | | |
| 관련 유스케이스 ID | | SS\_USD\_00\_04 | | |
| 구현 설명 | | 헌터가 도둑을 떄렸을때의 상황 | | |
| 서브시스템명 | 서브시스템 ID | 프로그램 파일 | | 비고 |
| ID | 파일명(Assets) |
| hit\_hunter | SS\_SUB\_00\_08 | Hunter(prefabs) | Hunter \_collider(script) |  |
| beat\_thief | SS\_SUB\_00\_09 | Thief(prefabs) | thief\_collider(script) |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Thief\_collider(Script) |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Hunter\_collider(script) |

----

4. 고려사항

|  |
| --- |
| 헌터는 도둑을 떄리면 hunter\_move의 stun()함수를 불러오며 5초간 정지한다. 또한 thief\_move의 heartsub()스크립트를 불러온다. heartsub()는 도둑 기본값의 heart의 값을 1줄여준다.  . |

|  |
| --- |
| . |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I5** | **네트워크 구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | |  | | |
| 관련 유스케이스 ID | |  | | |
| 구현 설명 | |  | | |
| 프로그램 파일 | | | Inspector & setting  [Add Component] | 비고 |
| ID | 파일명(Assets) | |
|  |  | |  |  |
|  |  | |  |  |
|  |  | |  |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
|  |

----

3. 구현 모습

|  |
| --- |
|  |
| 그림 |
| . |

4. 고려사항

|  |
| --- |
| . |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I6** | **UI구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | | 플레이시, 유저들 앞에 보이게 되는 화면UI 모습 | | |
| 관련 유스케이스 ID | | - | | |
| 구현 설명 | | 플레이 유저가 원활하게 게임을 진행하는데 있어 보게 될 UI 화면으로 현재 남은 목숨, 남은 총알 개수 등을 파악할 수 있다. | | |
| 프로그램 파일 | | | Inspector & setting  [Add Component] | 비고 |
| ID | 파일명(Assets) | |
| SS\_SUB\_00\_15 | Hunter\_infoo | | Text(TMP)bullet |  |
| SS\_SUB\_00\_16 | Thief\_infoo | | Text(TMP)bullet  Text(TMP)life |  |
|  |  | |  |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| Thief\_heart\_info Script |

|  |
| --- |
|  |
| Thief\_bullet \_info Script |

|  |
| --- |
|  |
| Hunter\_bullet\_info Script |

3. 구현 모습

|  |
| --- |
| 목요일에 |
| 그림 |
| . |

4. 고려사항

|  |
| --- |
| 콘솔 화면에서 보여지는거랑 VR 화면에서 보는거랑 다르기 때문에 텍스트들의 위치 조정을 빌드해서 확인해야한다.  . |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **I7** | **게임 Scene 변환 구현** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 구현 | 작성일자 |  | 버전 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험 유형 | | 게임 시작부터 종료까지의 Scene 변환 | | |
| 관련 유스케이스 ID | | - | | |
| 구현 설명 | | 게임 서버에 접속되고, 영상을 확인 후, 게임 화면으로 전환된다. | | |
| 프로그램 파일 | | | Inspector & setting  [Add Component] | 비고 |
| ID | 파일명(Assets) | |
| FirstScene | Photon | | Canvas  Button |  |
| SecndScene | Video | | Video Player  Canvas  Button |  |
| DemoScene | - | | - | 본 게임 화면 |
| EndScene | Video | | Video Player |  |

2. 프로그램 코드 및 알고리즘

|  |
| --- |
|  |
| changescene script |

using UnityEngine.SceneManagement;를 추가하면 해당 scene으로 이동가능한 SceneManager.LoadScene 기능을 사용할 수 있다.

3. 구현 모습

|  |
| --- |
|  |
| 그림 7.1 Scene 전환 순서  FirstScene[게임 로비]🡪 SecondScene[스토리 영상]🡪 DemoScene[게임 창]🡪 EndScene[게임 종료] |
| 버튼 컴포넌트에 On Click()이 존재한다. 이곳에 위에서 쓴 스크립트가 들어간 오브젝트를 추가하여 해당 버튼을 클릭하면 원하는 Scene으로 변경이 가능한 메소드가 실행된다. |

**4.1 시험단계**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **T1** | **통합시험 결과서** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 시험 | 작성일자 |  | 버전 |  |

1. 시험 시나리오

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 시험시나리오 ID | |  | | |
| 시험시나리오명 | |  | | |
| 시험시나리오설명 | |  | | |
| 관련 유스케이스ID | |  | | |
| 시험케이스  ID | 시험케이스  설명 | 시험 절차 | 시나리오 설명 | 비고 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

2. 시험 케이스

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 차수 | |  | | | | | | |
| 시험시나리오 ID | |  | | | | | | |
| 시험케이스 ID | |  | | | | | | |
| 시험 절차 | | 시험 항목 | 사전조건 | 입력자료 | 예상결과 | 화면ID | 시험결과 | 비고 |
| 순번 | 업무처리  내용 |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **T3** | **사용자 지침서** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 시험 | 작성일자 |  | 버전 |  |

1. 개요

2. 사용 지침

2.1 설치 방법

2.2 시작 및 종료 방법

2.3 공통 사항

3. 기능별 사용법

3.1 기능 이름

3.1.1 기능 설명

3.1.2 주의 사항

3.1.3 오류 메시지 및 원인 부록

【작성 목적】

개발한 시스템을 사용자가 효과적으로 사용할 수 있도록 기능적인 설명과 기타 예외처리에 대한 지침을 기술한다.

【작성 방법】

개발 시스템의 기능 및 예외처리 사항을 빠짐없이 기술하며, 이해하기 쉽고 구체적인 언어표현을 사용한다. 사용자지침서의 경우 시스템의 특성에 따라 기술하는 항목이 차이가 많이 발생 수 있으므로 내용이 빠지지 않도록 관련항목을 수정하여 사용한다.

【항목 설명】

• 개요

시스템의 사용자 측면의 전체적인 개요를 기술한다.

• 사용 지침

시스템의 설치방법, 시작 및 종료 방법과 공통적인 내용을 화면 캡처를 이용하여 사용자가 쉽게 사용할 수 있도록 기술한다.

• 기능별 사용법

기능 이름, 기능 설명, 주의 사항, 오류 메시지 및 원인 등을 빠짐없이 완전하고 일관된 언어표현으로 화면 캡처를 이용하여 이해하기 쉽게 기술한다.

• 부록

오류 메시지 및 처리 절차, 용어 및 색인 등을 제공한다.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **T4** | **운영자 지침서** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 시험 | 작성일자 |  | 버전 |  |

1. 개요

2. 시스템 설치

2.1 시스템 구성

2.1.1 시스템 구성도

2.1.2 하드웨어 구성

2.1.3 소프트웨어 구성

2.1.4 네트워크 구성

2.2 시스템 설치 절차

3. 시스템 운용

3.1 자료 관리

3.2 백업 관리

3.3 장애 내역 및 복구 절차

4. 사용 지침

4.1 시작 및 종료 방법

4.2 공통 사항

5. 기능별 사용법

5.1 기능명

5.1.1 기능 설명

5.1.2 주의 사항

5.1.3 오류 메시지 및 해결 방법 부록

【작성 목적】

개발한 시스템을 운영자가 효과적으로 관리와 운영을 할 수 있도록 시스템 관리 및 운영에 대한 지침을 기술한다.

【작성 방법】

개발 시스템의 관리 및 운영 측면에서 시스템의 지속적인 관리가 가능하도록 상세하게 작성하며 이해하기 쉽고 구체적인 언어표현을 사용한다. 운영자지침서의 경우 시스템의 특성에 따라 기술하는 항목이 차이가 많이 발생 수 있으므로 내용이 빠지지 않도록 관련항목을 수정하여 사용한다.

【항목 설명】

• 개요

시스템의 운영자 측면의 전체적인 개요를 기술한다.

• 시스템 설치

하드웨어, 소프트웨어, 네트워크를 포함한 시스템의 구성, 시스템 설치 절차 등의 내용을 화면 캡처를 이용하여 운영자가 쉽게 사용할 수 있도록 기술한다.

• 시스템 운용

자료 관리, 백업 관리, 장애 내역 및 복구 절차 등의 내용을 화면 캡처를 이용하여 운영자가 쉽게 사용할 수 있도록 기술한다.

• 사용 지침

시작 및 종료 방법과 공통적인 내용을 화면 캡처를 이용하여 운영자가 쉽게 사용할 수 있도록 기술한다.

• 기능별 사용법

기능 이름, 기능 설명, 주의 사항, 오류 메시지 및 해결 방법 등을 빠짐없이 완전하고 일관된 언어표현으로 화면 캡처를 이용하여 이해하기 쉽게 기술한다.

• 부록

기타 필요사항 등을 제공한다.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **T5** | **시스템 설치 결과서** | | | | | |
| 시스템명 | |  | 서브시스템명 |  | | |
| 단계명 | | 시험 | 작성일자 |  | 버전 |  |

1. 시스템 개요

1.1 시스템 개요 및 특징

1.2 시스템 구성

1.3 시스템 운영 환경

2. 시스템 설치 내역

2.1 서버 주소

2.2 작업내역 및 결과

2.3 상세 작업내역

3. 플랫폼 설치 결과

4. 데이터 전환 결과

5. 시스템 설치 결과

6. 백업 목록표

7. 문제점 및 조치사항

8. 기타사항

【작성 목적】

최종적으로 개발한 시스템을 사업자가 발주자에게 인도하여 시스템을 설치하기 위하여 설치환경의 내용을 기술한다.

【작성 방법】

시스템 개발 결과 및 이를 설치하고 운영할 수 있는 운영 환경을 구체적이고 상세하게 기술한다. 시스템 설치결과서의 경우 시스템의 특성에 따라 기술하는 항목이 차이가 많이 발생 수 있으므로 내용이 빠지지 않도록 관련항목을 수정하여 사용한다.

【항목 설명】

• 시스템 개요

시스템 개요 및 특징, 시스템 구성과 하드웨어, 소프웨어를 포함한 시스템 운영 환경을 기술한다.

• 시스템 설치 내역

서버 주소, 작업내역 및 결과, 상세 작업내역 등을 기술한다.

• 플랫폼 설치 결과

개발한 소프트웨어 및 데이터베이스의 설치 내역을 기술한다.

• 데이터 전환 결과

기존 데이터베이스의 데이터를 전환 프로그램을 이용하여 개발한 데이터베이스로 전환한 결과를 기술한다.

• 시스템 설치 결과

시스템 설치에 대한 상세한 내역을 기술한다.

• 백업 목록표

데이터 백업 항목 및 내용을 기술한다.

• 문제점 및 조치사항

시스템 설치 과정상의 문제점과 조치내역을 기술한다.

• 기타사항

추가적으로 필요한 내용을 기술한다.

【작성 목적】

총괄시험 계획서와 통합시험 시나리오에 따라서 수행한 시험 결과를 기술한다.

【작성 방법】

유스케이스별로 통합 시험 시나리오를 기술하고 시나리오에 대한 시험 케이스를 기술하며 시험 예상결과와 시험결과를 비교한다.

【항목 설명】

▣ 시험 시나리오

설계단계의 “통합시험 시나리오”의 시험 시나리오를 현행화하고, 통합시험을 수행한다.

▣ 시험 케이스

설계단계의 “통합시험 시나리오”의 시험 케이스를 현행화하여 시험을 수행 후, 시험결과를 기록한다.