

BossLocker

레이드 형식의 액션 슈팅 게임

Team Goddun

기획 & 서버: 김홍일

클라이언트: 양태윤

모델링 & UI: 조소연

게임공학과 2015180048

엔터테인먼트 2013184048

엔터테인먼트 2015184042



BossLocker

01

프로젝트 개요

BossLocker

02

게임 소개

- 게임 목표
- 게임 소개
- 게임 특징
- 기본 조작
- 월드 설정
- 그래픽 컨셉

03

기술적 요소

- 중점 연구분야
- 타 게임과의 비교

04

개발목표 및 내용

- 개발 환경
- 서버
- 클라이언트
- 그래픽

05

개발 일정

- 서버
- 클라이언트
- 그래픽

팀 구성

기획 & 서버: 김홍일

클라이언트: 양태운

모델링 & UI: 조소연



프로젝트 개요

B
O
S
S
L
O
C
K
E
R

BossLocker

장르: 레이드 형식의 액션 슈팅 게임

특징: 빠른 진행, 화려한 탄막

조작: 마우스 & 키보드

시점: 3인칭 쿼터뷰



게임소개
게임 목표

B
O
S
S
L
O
C
K
E
R

BossLocker

Boss가 튀어나오는 신비한 Locker

영웅이 되어 Boss를 Locker로 다시 봉인하자!



게임소개

게임소개

B
O
S
S
L
O
C
K
E
R



서로 다른 3인의 플레이어가 대기방에서
영웅과 공략할 보스를 선택 후,
키보드를 이용해 적의 공격을 피하고
랜덤하게 드랍하는 룬을 획득하면서
마우스로 적을 조준해 공격하는 게임



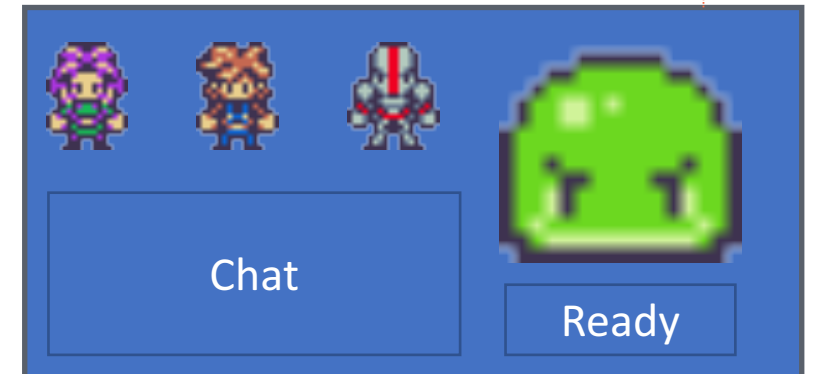
3인 멀티플레이어 게임



키보드를 이용한 캐릭터 이동



마우스를 이용한 논-타겟 방식 공격

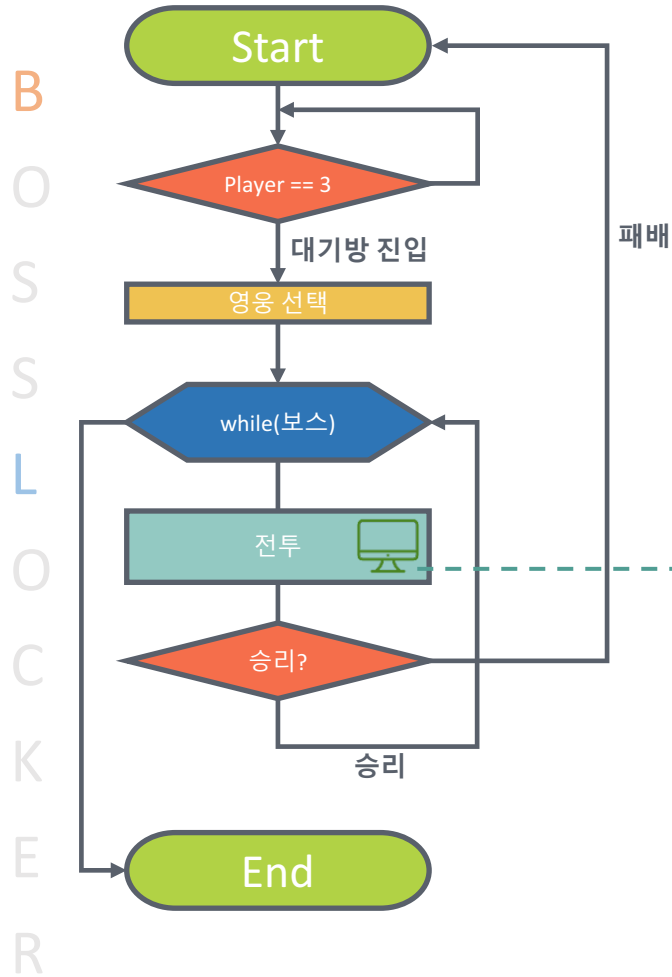


(대기방 예시)



게임소개

게임특징



1. 보스를 공략하기 전까지, 잡몹들을 잡아야 하는 지루한 과정을 생략하고 순수 보스만을 공략한다.



2. 보스는 화려하고 다양한 탄막으로 플레이어를 공격한다.

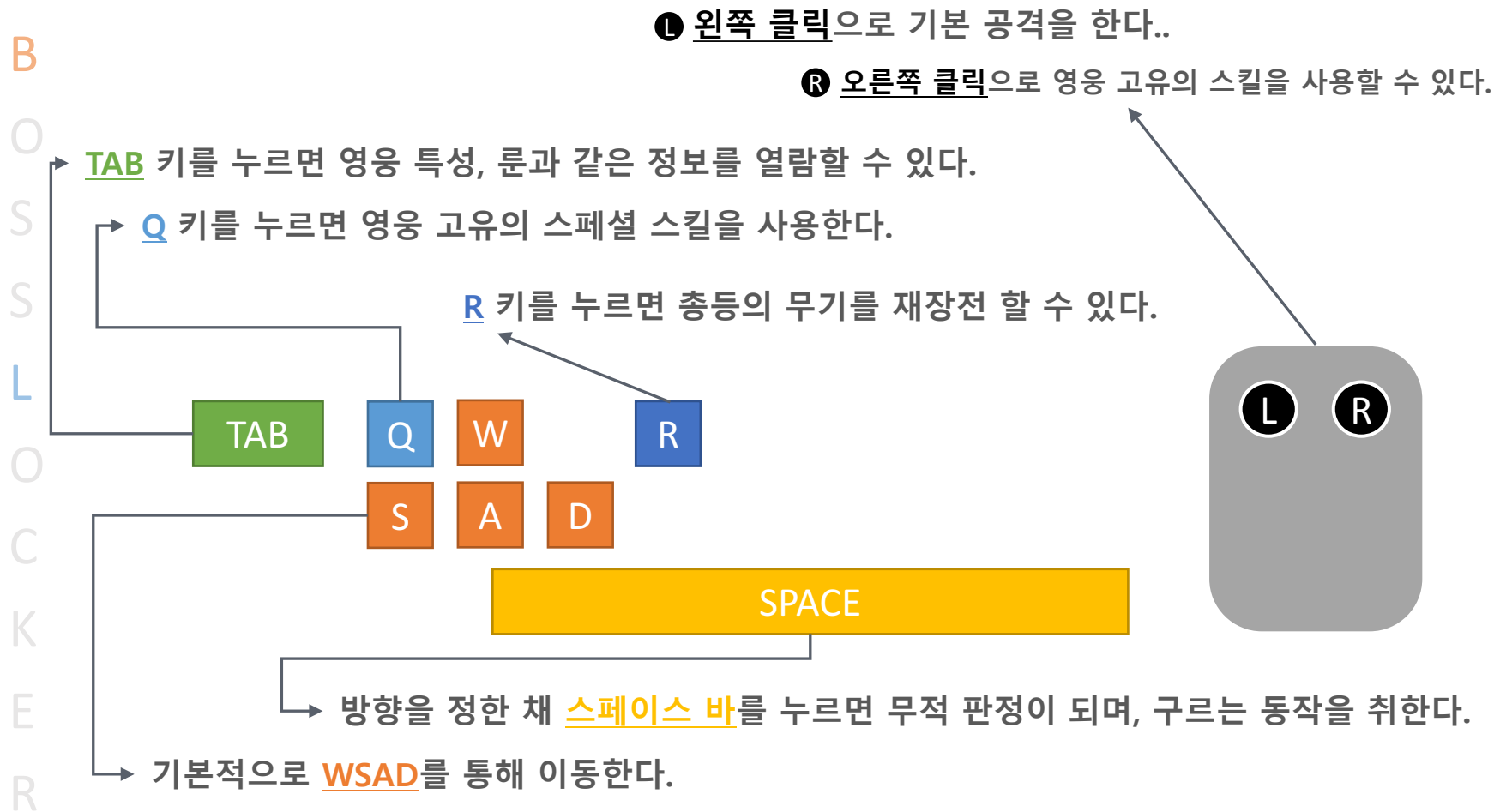
3. 랜덤하게 드랍하는 룬의 획득으로 성장한다.

4. 플레이어 개인의 컨트롤등 기량도 중요하지만 플레이어 3명의 팀워크가 맞지 않으면 점점 힘든 구조로 되어있다.



게임소개

기본 조작





게임소개

월드 설정

B

O

S

S

L

O

C

K

E

R



300m x 300m 3개
보스와 던전



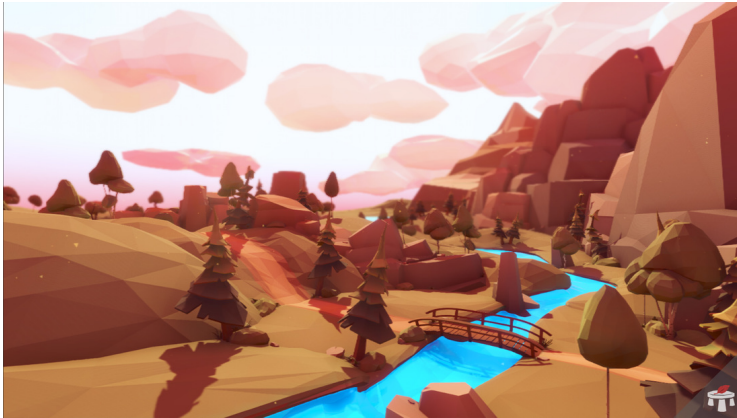
1.5m의 키 설정
2m/s의 이동속도



게임소개

그래픽 컨셉

B
O
S
S
L
O
C
K
E
R



로우폴리의 동화적인 그래픽 연출



기술적 요소

타 게임과의 비교

B
O
S
S
L
O
C
K
E
R



2D

탐류 카메라

2인 멀티플레이

아이템을 통한 성장

논-타겟 시스템

잡몹 처리후 보스 공략

VS

BossLocker

3D

쿼터뷰 카메라

3인 멀티플레이

룬을 통한 스탯 성장

논-타겟 시스템

곧바로 보스 공략





기술적 요소

중점 연구분야

B

O

S

S

L

O

C

K

E

R



1 탄만

	CGCII (C++)	Supersocket (C#)	asio (C++)	CGSF(C++)	node.JS(Java Script)
Messages/sec	4600만	800만	100만	11만	20만
BPS	2.88G bps (360M bytes/s)	496M bps (62M bytes/s)	60M bps (7.5M bytes/s)	6.64M bps (0.83M bytes/s)	12M bps (1.5M bytes/s)
CPU 사용률	15%~18%	75%	60%	2%~3%	8%
메모리 사용량	40M Bytes~70M Bytes	16M Bytes	11M Bytes	450M Bytes ~	40M Bytes

전반적인 성능은 아무런 처리를 하지 않은 asio(C++)에 비해 1/5정도의 성능을 나타내며 C#의 라이브러리인 SuperSocket에 비해서는 1/40 정도의 성능을 나타냅니다. 또 CGCII에 비해서는 1/230 정도의 성능을 나타냅니다.

높은 생산성을 지닌 NodeJS로 서버 구축
NoSQL 특징을 가진 Mongo DB 활용
디지털오션 클라우드 서비스를 이용한 서버



개발목표 및 내용

개발 환경

B Window 10, Mac OS 10.xx

O 클라이언트

S DirectX 11

S MS Visual Studio 17

L 서버

L NodeJS, socketIO, MongoDB

O WebStorm 17, RoboMongo

C 그래픽

K 3ds MAX 2016, Photoshop

E 협업

R Github, Trello, KakaoTalk



개발목표 및 내용

1_{/2} 기획 / 서버 김홍일

화려한 탄막 알고리즘 기획

게임의 메인 컨셉인 긴장감을 살려줄 수 있는
다양한 탄막 알고리즘을 기획



개발목표 및 내용

2_{/2} 기획 / 서버 김홍일

NodeJS를 활용한 서버

NodeJS와 외부 라이브러리 SocketIO를 활용하여
OS와 관계없이 작동되는 서버



개발목표 및 내용

1_{/2} 클라이언트 양태윤

DirectX 11을 활용한 자체 프레임워크 제작

DirectX 11에서 제공하는 파이프 라인과 HLSL을 익혀
게임에 적합한 3D환경의 프레임워크 제작



개발목표 및 내용

2_{/2} 클라이언트 양태윤

게임의 연출을 위한 셰이더 제작

Lighting, Diffuse Specular Mapping, Toon, Shadow Mapping



개발목표 및 내용

1_{/2} 그래픽 조소연

게임의 컨셉과 어울리는 모델 / 애니메이션 제작

빠르게 반응하는 애니메이션과
이를 살려 줄 수 있는 로우폴리 데이터 제작



개발목표 및 내용

2_{/2} 그래픽 조소연

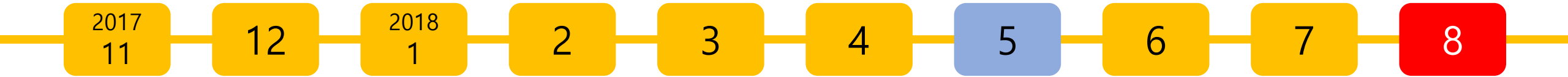
직관적인 UI 디자인

게임의 화려한 탄막 효과를 방해하지 않으면서
학습이 필요없는 직관적인 UI 디자인



개발 일정
기획 / 서버

기획 / 서버
김홍일



BossLocker 기획

Node JS
서버 제작

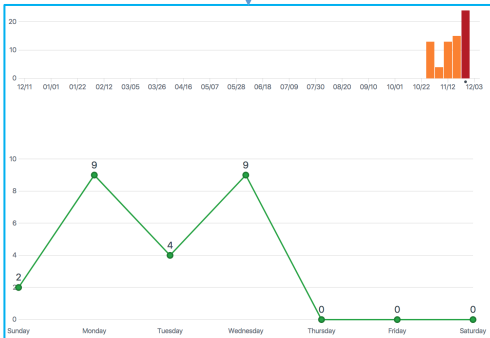
클라이언트
동기화

충돌 처리

탄막 알고리즘 기획 / 구현

필요한 AI 기능 구현 및 테스트

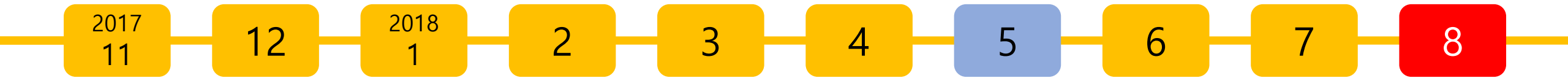
리팩토링 / 최적화





개발 일정
클라이언트

클라이언트 양태윤



DirectX 11
프레임워크 제작

대기방

영웅 캐릭터

스킬

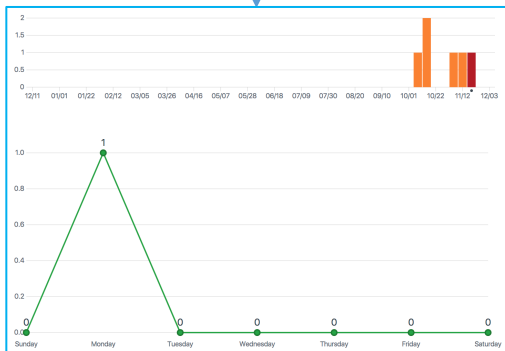
보스

AI

문

이펙트

리팩토링 / 최적화





개발 일정
그래픽

그래픽
조소연



감사합니다.

