

Unity Backend

게임 서버 “뒤끝” 기능 소개

Created in 2023-01-30

Last Updated 2023-01-30

Unity Version 2022.2.2f1

Index

- ◆ 뒤끝소개

뒤끝소개

- 뒤끝이란?
- 뒤끝베이스



뒤끝소개

■ 뒤끝이란?

- PC, 콘솔, 모바일 등의 플랫폼에서 온라인 게임 개발 및 운영을 위해 필요한 서버와 데이터베이스 구축을 손쉽게 할 수 있는 서버 시스템

유저 데이터 유지

- 📍 기기 변경 등으로 쉽게 유실될 수 있는 유저 데이터를 안전하게 유지

데이터 분석 관리

- 📍 유저 데이터 분석 및 관리
- 📍 로그 및 퍼널 분석으로 유저 이탈 구간 확인 및 개선
- 📍 주요 지표 관리

유저 충성도 강화

- 📍 푸시, 우편, 이벤트, 랭킹, 소셜 기능 활용으로 유저 리텐션 및 플레이타임 개선

뒤끝

게임 캐시 관리

- 📍 캐시 재화 충전 및 캐시 상품 관리
- 📍 영수증 검증

잘은 앱 업데이트 방지

- 📍 주요 밸런싱 데이터를 서버에서 관리하여 앱 업데이트 없이 게임 밸런스 수정

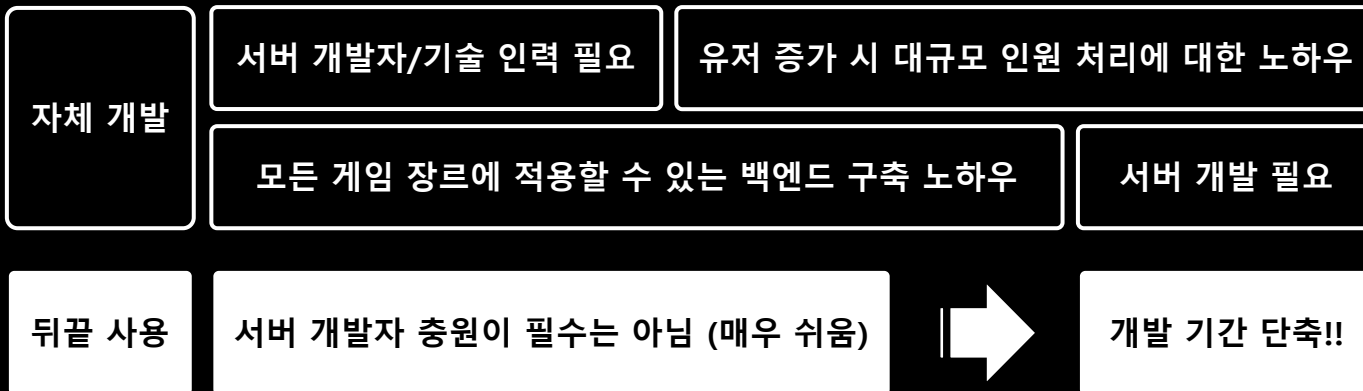
보안 강화

- 📍 부정 결제 방지 및 환불 대응을 위한 유저 결제 정보 관리
- 📍 주요 확률 및 아이템 획득 정보를 서버에서 관리



뒤끝소개

■ 뒤끝 사용의 장점



■ 뒤끝은

- 게임 개발에 필수적인 비동기 서버 기능들을 제공하는 **뒤끝베이스**
- 모바일 게임에 특화된 실시간 채팅 서비스를 제공하는 **뒤끝챗**
- 1:1부터 팀 전까지 국내 최초 실시간 PvP 서비스를 지원하는 **뒤끝매치**
- 뒤끝베이스에 보안을 더한 서버 평션 기능을 제공하는 **뒤끝평션**
- 의 4가지 기능을 제공하고 있습니다.



뒤끝소개

■ 이용 사례

- 많은 게임 개발사가 뒤끝을 사용해 게임 서버를 구축, 운영



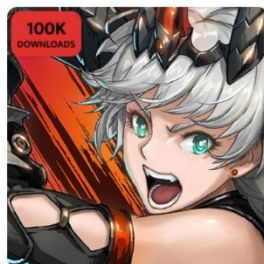
1M
DOWNLOADS

프리스트 아일랜드: 자연과 함께하는
방치형 힐링게임
Nanali Studios



100K
DOWNLOADS

미스터 히어로: 방치형 키우기
MONSTER PLANET Corp.



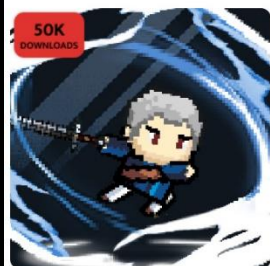
100K
DOWNLOADS

다크 히어로 대쉬: 방치형 키우기
BluePotion Games



100K
DOWNLOADS

픽셀 판타지아: 이세계 소녀 키우기
방치형RPG
SUPERBOX.Inc



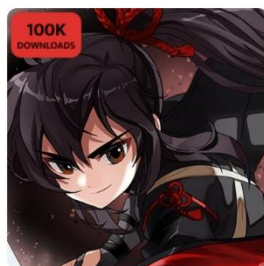
50K
DOWNLOADS

모험가 키우기 - 검사편
WhiteJupiter Studio



10K
DOWNLOADS

성 키우기 타이쿤 - 방치형 타워 RPG
IOI Games



100K
DOWNLOADS

요괴사냥꾼 키우기: 방치형RPG
MoreAsh



10K
DOWNLOADS

가디언 아카데미 - 방치형 가디언 키우기
hoduro games



50K
DOWNLOADS

철구 키우기
DwG



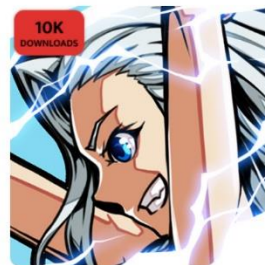
100K
DOWNLOADS

그랑젠카 키우기
gameberry studio (Idle RPG, Simulation)



50K
DOWNLOADS

누맞고타운
MONSTER PLANET Corp.



10K
DOWNLOADS

번개의 신 키우기
DwG



뒤끝소개

- 3000이 넘는 누적 가입 개발사와 6천만명이 넘는 게임 유저 보유

축적된 기술과 경험으로 서비스합니다.		
누적 가입 개발사	1일 트랜잭션	누적 게임 유저
3,163	100,976,417	60,345,005

- 인디 개발사부터 대형 퍼블리셔까지 다양한 개발사가 뒤끝 사용





뒤끝소개

■ 뒤끝 베이스

■ 게임 유저 관리

- 게임 유저 관리는 회원가입/로그인, 유저 정보 관리, 닉네임 설정, 국가 코드 관리, 회원 탈퇴와 같은 기능을 제공합니다.

회원가입/로그인

회원가입/로그인

유저 정보 관리

유저 닉네임 설정

국가 코드 관리

회원 탈퇴

3가지 회원가입 방법이 제공됩니다.

- ✓ 유저가 직접 아이디/비밀번호를 이용하여 가입하는 커스텀 계정
- ✓ 구글/애플/페이스북 계정을 이용한 페더레이션 계정
- ✓ 기기 내부에 생성된 정보를 이용한 게스트 로그인

개발문서 읽기 [🔗](#)

```
// 회원가입
BackendReturnObject bro =
Backend.BMember.CustomSignUp( "id", "password" );
if(bro.IsSuccess())
{
    Debug.Log("회원가입에 성공했습니다");
}

// 로그인
BackendReturnObject bro =
Backend.BMember.CustomLogin( "id", "password" );
if(bro.IsSuccess())
{
    Debug.Log("로그인에 성공했습니다");
}
```




뒤끝소개

■ 게임 운영 관리

- 게임 운영 관리는 공지사항, 이벤트, 1:1문의, 정책과 같은 기능을 제공

공지사항

공지사항

이벤트

1:1 문의

정책

공지사항을 등록하고 관리할 수 있습니다.

- ✓ 등록 형태는 즉시 게시 및 예약 게시 형태로 제공됩니다.
- ✓ 이미지 첨부 및 URL링크 버튼 생성 기능을 활용할 수 있습니다.
- ✓ 일반 공지사항 외 임시 공지사항을 등록하고 관리할 수 있습니다.
- ✓ 다국어 지원
국가별로 다른 언어(내용)의 공지사항을 노출시킬 수 있습니다.

개발문서 읽기 [🔗](#)



```
// 공지사항 조회
BackendReturnObject bro = Backend.Notice.NoticeList(2);
if (bro.IsSuccess())
{
    Debug.log("공지사항 조회 성공");
}
```



뒤끝소개

■ 결제 관리

- 결제 관리는 인앱 결제 영수증 검증과 게임 캐시 관리 기능을 제공합니다.

인앱 결제 영수증 검증

인앱 결제 영수증 검증

게임 캐시 관리

뒤끝에서는 영수증 검증을 통해 확실한 결제 인증을 진행합니다.

- ✓ 영수증 자체의 유효성과 구매한 ProductID를 검증합니다.
- ✓ 구글/애플 영수증 검증이 가능합니다.

개발문서 읽기 [🔗](#)



```
// 구글 영수증 검증
BackendReturnObject bro =
Backend.Receipt.IsValidateGooglePurchase ( "영수증
토큰", "receiptDescription", false);
if(bro.IsSuccess())
{
    Debug.Log("구글 영수증 검증 성공");
}

// 애플 영수증 검증
BackendReturnObject bro =
Backend.Receipt.IsValidateApplePurchase ( "영수증 토큰",
"receiptDescription" );
if(bro.IsSuccess())
{
    Debug.Log("애플 영수증 검증 성공");
}
```



뒤끝소개

■ 랭킹 관리

- 뒤끝에서는 유저랭킹과 길드랭킹 두 가지 종류의 랭킹을 제공
- 뒤끝의 랭킹 기능을 사용해 손쉽게 랭킹을 생성하고 관리할 수 있으며, 랭킹에 따른 보상 지급 가능 (랭킹 초기화 기간 : 일간/주간/월간/누적/일회성)

유저 랭킹

유저 랭킹

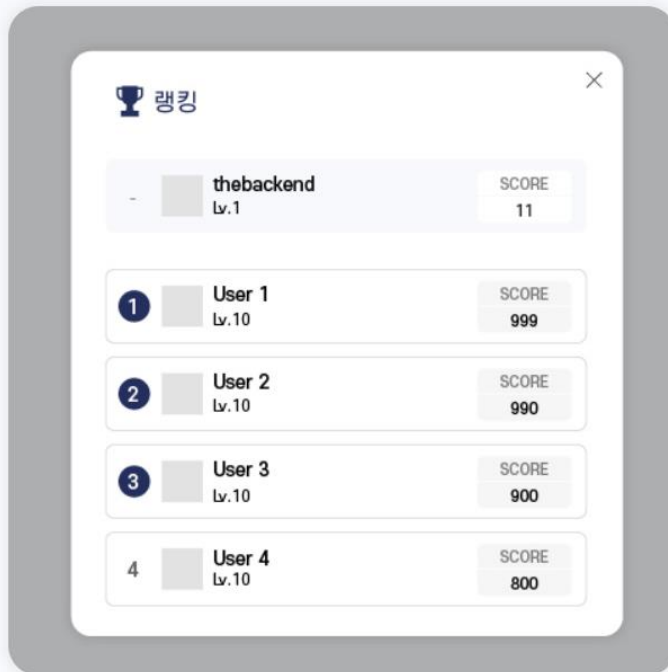
길드 랭킹

랭킹 보상

유저들 간 경쟁을 위한 랭킹을 지원합니다.

- ✓ 실시간으로 갱신되는 랭킹입니다.
- ✓ 유저 랭킹 검색 및 수동 삭제가 가능합니다.
- ✓ 랭킹, 점수 외에 추가 항목 정보 노출을 지원합니다.
- ✓ 랭킹 정보 엑셀 다운로드를 지원합니다.

[개발문서 읽기](#)



```
// 실시간으로 갱신되는 유저 랭킹 조회
BackendReturnObject bro =
Backend.URank.User.GetRankList ( 유저 랭킹 uuid );
if(bro.IsSuccess())
{
  Debug.Log("랭킹 리스트 조회 성공");
}
```



뒤끝소개

- 소셜 기능 (친구, 길드, 쪽지, 우편)
 - 소셜 기능은 친구, 길드, 쪽지, 우편과 같은 기능을 제공합니다.

친구

친구

길드

쪽지

우편

게임 유저 간의 친구 기능을 제공합니다.

- ✓ 유저 간에는 친구 요청을 보내고, 이를 수락 및 거절할 수 있습니다.
- ✓ 콘솔 내에서는 친구 최대 수를 설정할 수 있습니다.
- ✓ 친구를 맺은 유저는 닉네임과 키값 (inDate)을 쉽게 확인할 수 있습니다.
- ✓ 친구를 맺은 유저의 접속, 접속 종료를 실시간으로 확인할 수 있습니다.

개발문서 읽기 [🔗](#)



```
// 전체 친구 리스트 조회
BackendReturnObject bro =
Backend.Social.Friend.GetFriendList();
if (bro.IsSuccess())
{
  Debug.log("전체 친구 리스트 조회 성공");
}

// 친구 요청하기
BackendReturnObject bro =
Backend.Social.Friend.RequestFriend("2020-02-
11T01:08:12.603Z");
if (bro.IsSuccess())
{
  Debug.log("친구 요청하기 성공");
}
```

실시간 알림

실시간 알림을 통해 다음과 같은 알림을 받을 수 있습니다.



User123님께서 친구 요청을 수락하셨습니다!





뒤끝소개

■ 푸시 알람

- 푸시 알람은 푸시 등록하기와 푸시 테스트 기능을 제공합니다.

푸시 등록하기

푸시 등록하기

푸시 테스트

뒤끝 콘솔에서 보내는 푸시를 받는 기능을 제공합니다.

- ✓ 즉시 발송, 예약 발송 모두 지원합니다.
- ✓ 다국어 지원
국가별로 다른 언어(내용)의 공지사항을 노출시킬 수 있습니다.

THEBACKEND.IO now

[이벤트 공지]
지금 랭킹 이벤트에 참여해서 다이아를 획득하세요!

THEBACKEND.IO now

하트가 모두 충전되었습니다!
다시 한 번 도전해보시는 건 어때요?

THEBACKEND.IO now

보물 상자 오픈
무료 보물 상자가 기다리고 있습니다!

푸시 등록하기

발송 일시
 즉시 발송 예약 발송

발송 대상
 Android iOS[Development] iOS[Production]

기본 국가 선택 × +

국가 선택

- US
- TW
- KR
- ES

제목

내용



뒤끝소개

■ 게임 정보 관리

- 게임 정보 관리는 데이터 저장과 트랜잭션 처리 기능을 제공합니다.

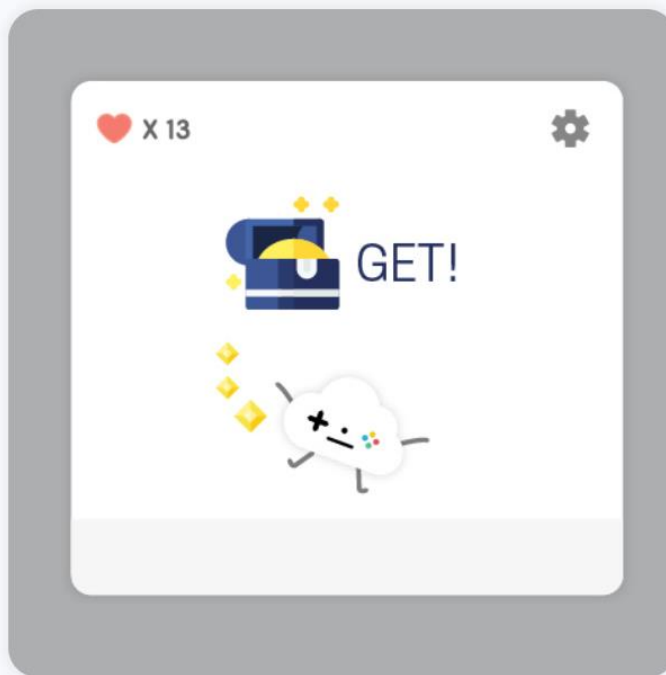
데이터 저장

스키마를 지정한 SQL 형식, 스키마를 지정하지 않은 NoSQL 형식 모두 지원합니다. 데이터 CRUD를 손쉽게 수행할 수 있습니다.

개발문서 읽기 [🔗](#)

데이터 저장

트랜잭션 처리



```
// content 컬럼의 값을 testData로 수정하기 위한 준비
Param param = new Param ();
param.Add ( "content", "testData" );

// 테이블 수정
// tableName 테이블에 inDate(key)가 inDate인 row에 param
값을 적용
BackendReturnObject bro = Backend.GameData.Update(
"tableName", inDate, param );
if(bro.IsSuccess())
{
    Debug.Log("테이블 수정 성공");
}
```



뒤끝소개

■ 차트 관리

- 차트 관리는 게임 내에서 사용되는 데이터를 서버에 테이블 형태로 저장하고, 게임 내에서 불러와 사용하는 기능을 제공합니다.

차트 관리

차트 관리

차트는 게임 정보와 달리 모든 유저가 공통적으로 조회할 수 있는 데이터입니다.

- ✓ 아이템 차트 파일을 만들어 업로드 후, 게임 내 상점에서 아이템 리스트를 불러와 이를 출력할 수 있습니다. 차트는 콘솔 내에서 간단한 수정이 가능합니다.

개발문서 읽기 [📄](#)

상점				
아이템 리스트				
아이템 이름	구분	수량	가격	
Item 1	단일	1	15	10000
Item 2	단일	1000	0	100000
Item 3	단일	20	100	0
Item 4	단일	100	10	0
Item 5	묶음	50	0	50000
Item 6	단일	1	0	300000
Item 7	단일	5	30	0

```
// 차트 리스트 조회하기
BackendReturnObject bro = Backend.Chart.GetChartList();
if (bro.IsSuccess())
{
    Debug.log("차트 리스트 조회하기 성공");
}

// 차트 내용 조회하기
BackendReturnObject bro =
Backend.Chart.GetChartContents("selectedChartFileId");
if (bro.IsSuccess())
{
    Debug.log("차트 내용 조회하기 성공");
}
```



뒤끝소개

■ 확률 관리

- 확률 관리는 랜덤 보상 상자 등에 사용되는 확률을 제공하는 기능입니다.

확률 관리

랜덤 보상 상자 등에 사용되는 확률을 제공하는 기능입니다.

- ✓ 뒤끝 콘솔에 각 행(row)의 퍼센트를 지정한 확률 파일을 업로드하고, 뒤끝 SDK를 통해 해당하는 확률로 도출된 결과를 받아와 보다 손쉽게 활용할 수 있습니다.

[개발문서 읽기](#)

확률 관리



```
// "CardFileUuid" 확률 차트에서 뽑기를 10회 실행
BackendReturnObject bro =
Backend.Probability.GetProbabilitys("CardFileUuid", 10);
if(bro.IsSuccess())
{
    Debug.Log("뽑기 10회 성공");
}
```




뒤끝소개

■ 쿠폰 관리

- 쿠폰 관리는 뒤끝 콘솔에 등록된 쿠폰의 리스트를 받아오고, 사용하는 기능을 제공합니다.

쿠폰 관리

쿠폰 관리

뒤끝 콘솔에 등록된 쿠폰의 리스트를 받아오고, 사용하는 기능을 제공합니다.

- ✓ 쿠폰의 유효기간을 지정할 수 있습니다.
- ✓ 시리얼 쿠폰 / 단일 쿠폰 지정이 가능하며 중복 사용 여부를 설정할 수 있습니다.
- ✓ 발행 개수와 지급 아이템 테이블을 변경할 수 있습니다.

개발문서 읽기 [🔗](#)



```
// 쿠폰 리스트 조회하기
BackendReturnObject bro = Backend.Coupon.CouponList();
if (bro.IsSuccess())
{
    Debug.log("쿠폰 리스트 조회하기 성공");
}

// 쿠폰 사용하기
BackendReturnObject bro = Backend.Coupon.UseCoupon (
    "18c7c2e1d16e779b" );
if (bro.IsSuccess())
{
    Debug.log("쿠폰 사용하기 성공");
}
```