

CloudBread

<http://aka.ms/cbp>

Facebook CloudBread 그룹 가입

<http://aka.ms/cfb>

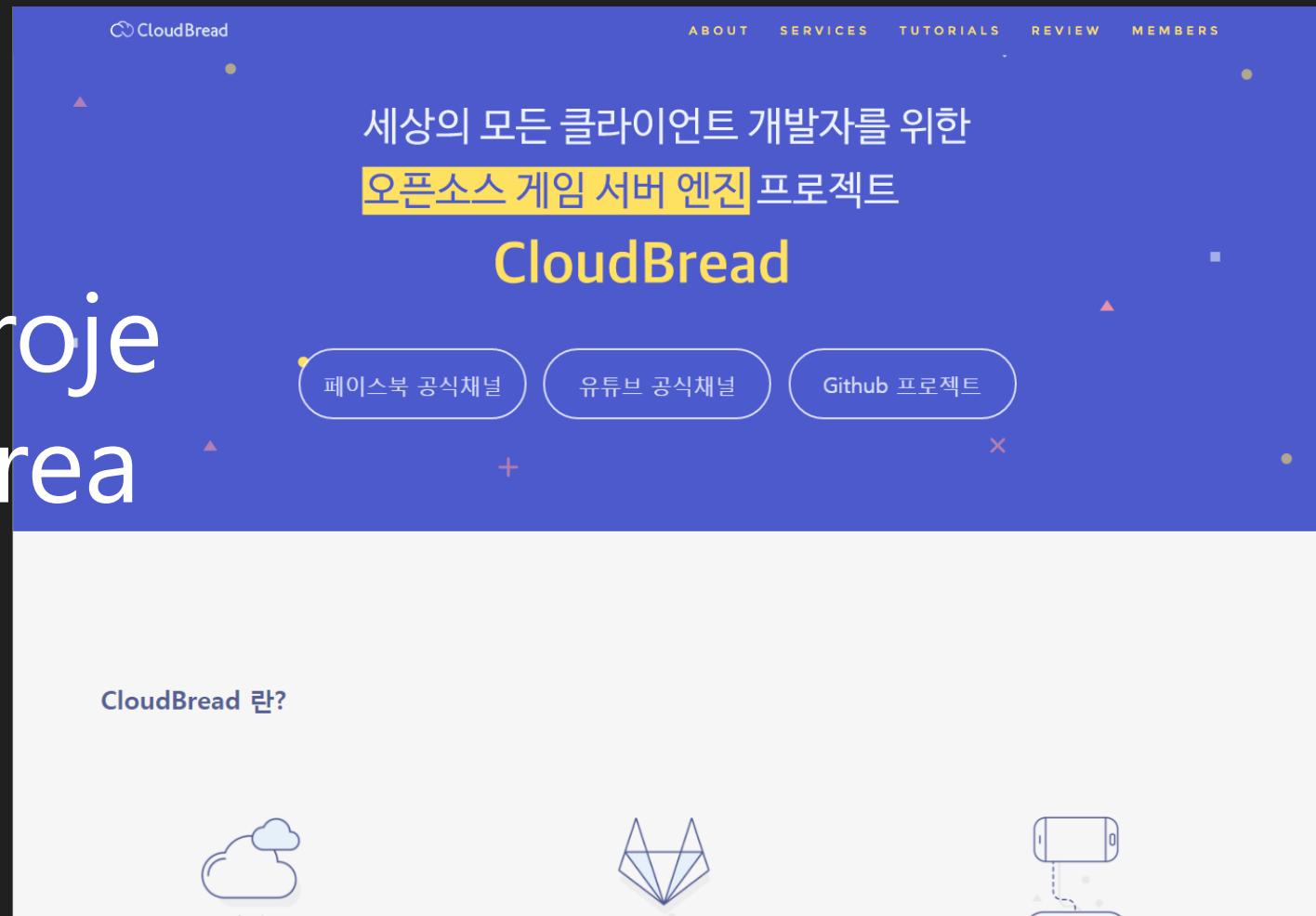
또는 "CloudBread"로 페이스북 검색 후 가입

<http://cbp.cu.cc>에서 페이스북 링크 누르기

CloudBread 메인 페이지

<http://cbp.cu.cc>

<http://cloudbreadproject.github.io/CloudBread.github.io/>



CloudBread 그룹에 후기를 적으면~

<http://aka.ms/cfb>

- CloudBread 페이스북 그룹에 후기를 남겨 주시고
- 자신의 페이스북 타임라인에 “공유”해 주세요!
- Floppy Bird 미션을 완료하신 5명에게는 무선 키보드 마우스 세트를 드립니다.

세션 발표자



김장윤



김대우



홍윤석



이정엽

캠프 중간 질문 있으면 언제든 찾아주세요!

캠프 아젠다

| 시간 | 내용 | 발표자 |
|---------------|--|-----|
| 10:00 ~ 10:10 | CloudBread 캠프 유의사항 안내 | |
| 10:10 ~ 10:30 | CloudBread 프로젝트 소개 | |
| 10:30 ~ 10:45 | CloudBread 진행중인 프로젝트, 컨트리뷰터 소개 | |
| 10:45 ~ 11:00 | 실제 게임을 가지고 게임 서버 설명 쿠키런 게임에서 어떻게 서버를 구축할지 | |
| 11:00 ~ 11:30 | CloudBread API 호출 방법 안내 DB 설명 및 API 규칙 소개 | |
| 11:30 ~ 11:45 | CloudBread API 사용하기 실습 Post Man 사용하여 호출 해보기 | |
| 11:45 ~ 12:30 | Unity에서 HTTP 호출 실습 (유니티에서 Post Man 만들기) | |
| 12:30 ~ 1:30 | 점심식사 및 네트워킹 | |
| 1:30 ~ 1:45 | ARM 배포 과정 소개 | |
| 1:45 ~ 2:00 | Unity SDK 설명 및 실습 | |
| 2:00 ~ 4:00 | CloudBread 실습 (플래피버드 게임 만들기) | |
| 4:00 ~ 4:30 | 캠프 마무리 | |

오늘의 경품!



マイクロソフト 무선 키보드 마우스 세트

CloudBread Project 소개

무료 오픈소스 프로젝트 - CloudBread

(대부분 이미 다 만들어진) 무료 오픈소스(MIT) 프로젝트

모바일 게임과 앱에 최적화된 게임 서버 엔진

100개의 BL 기본 제공

최소~최대 규모 백엔드 서버

보안, 관리, 기본 통계

- 앱 개발에 집중
- 파라미터에 맞춰 호출
- 예제 테스트 앱 제공
- REST 호출 함수 만들 필요 없음

- 앱의 코드는 바뀌지 않음
- IaaS보다 진화한 PaaS 클라우드
- Scale Up - Out 선택은 자유
- 서비스 구조에 맞는 기능 선택

- 보안 구조 / 데이터 암호화 선택 가능
- GM 관리자 도구 제공
- 기본 배치(DAU, HAU, DARPU) 제공
- 무료 기술 교육 제공(헤카톤/캠프)



기존 게임 서버 서비스와의 차이점

IFunFactory Hive 5

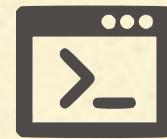
1.



개인 서버에 배포

IaaS 방식의 클라우드나
개인 서버에 배포해야 하므로
클라이언트 개발자에게 어려움이 있음

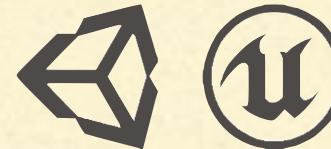
2.



제공하는 프레임워크와 툴을
사용하여 필요한 BL 개발

프레임워크를 사용하여 개발이 쉬워질 뿐
백엔드 개발이 필요함!

3.



게임 개발 플랫폼 SDK 사용

제공되는 SDK를 사용하여
쉽게 서버 통신을 끝낼 수 있음 개발과 운영을 함께 할 수 있도록

4.



게임 관리 웹페이지

게임 개발 할 시, 어차피 서버 구축과 백엔드 BI 코딩이 필요함

진짜 클라이언트 개발자를 위한 게임 서버 엔진

클라우드 게임 서버를 사용한 개발 시나리오

1.



클라우드 서버에 배포

각각 클라우드 서비스에
해당 프로젝트를 배포
Auzre 사용시 배포툴 제공)

2.



Develop Document 참고

게임에서 필요한 BL 와
사용방법에 대해 공부

3.



게임 개발 플랫폼 SDK 사용

제공되는 라이브러리를
사용하여 클라이언트 개발

4.



게임 관리 웹페이지

유저 데이터 통계, 분석
푸시 알람이나 공지등 운영

진짜 클라이언트 개발자를 위한 게임 서버 엔진

클라우드 게임 서버를 사용한 개발 시나리오

1.



클라우드 서버에 배포

각각 클라우드 서비스에
해당 프로젝트를 배포
(Auzre 사용시 배포툴 제공)

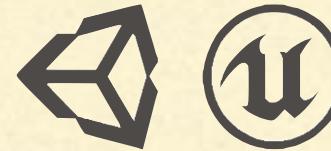
2.



Develop Document 참고

게임에서 필요한 BL 와
사용방법에 대해 공부

3.



게임 개발 플랫폼 SDK 사용

제공되는 라이브러리를
사용하여 클라이언트 개발

4.



게임 관리 웹페이지

유저 데이터 통계, 분석
푸시 알람이나 공지등 운영

배포 툴과 제공되는 BL 를 사용하면, 서버 개발을 할 필요가 없음!

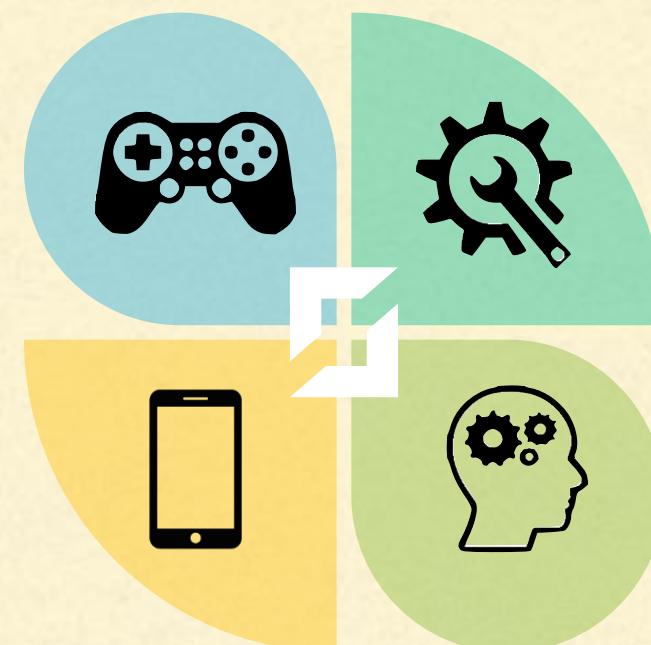
Cloud Game Server 특징

다양한 게임 라이브러리 제공

- 앱 개발에만 집중 할 수 있도록 테스트 앱 (클라이언트 소스) 제공
- 아이템 관리, 스테이지 관리 등
- 100여가지 BL 제공

모바일 특화 기능 제공

- 모바일 게임에서 중요시 되는 기능 제공
- 사용자 인증, 소셜 기능, 채팅 기능
- 푸시 알람 메시지, 공지 관리



Dev-Ops 관리자 웹

- 관리자 웹페이지 제공
- 게임 데이터를 관리 할 수 있는 툴 제공
- 게임 사용자를 분석한 데이터와 푸시메시지 관리, 공지 관리 등
- 게임 운영에 필요한 툴 제공

머신러닝 (추후 제공)

- 유저 데이터를 학습하여 머신러닝에 사용
- 게임 이탈 정도 예측, 블랙컨슈머 예측을 통해 게임 프로모션에 도움

다양한 Document!



Introduce

클라우드 브래드 소개 및 배포

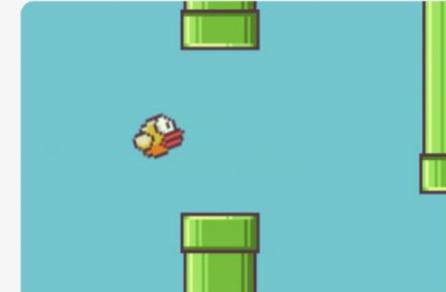
[Git wiki 바로가기 >](#)



Interview

클라우드 브래드를 사용한 대표님들의 인터뷰

[Youtube 바로가기 >](#)



Example

클라우드 브래드를 사용하여 게임 만들기

[Git wiki 바로가기 >](#)



Communication

Unity와 클라우드 브래드간 HTTP 통신

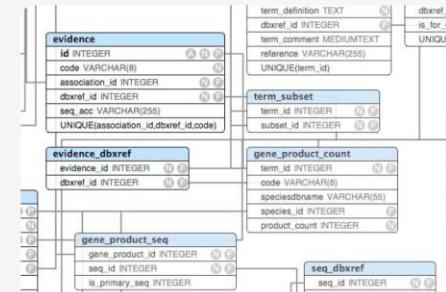
[Youtube 바로가기 >](#)



UnitySDK

UnitySDK를 클라우드 브래드에 적용하기

[Git wiki 바로가기 >](#)



Database

클라우드 브래드로 DB스키마 사용하기

[Git wiki 바로가기 >](#)

오늘의 목표 :

CloudBread 연동!!!

게임에서 CloudBread 를
어떻게 사용할까?

제가 참 좋아하는
“국민 게임”

게임 데이터 처리



공지사항 - Notice



이벤트 - Event



선물 - Gift



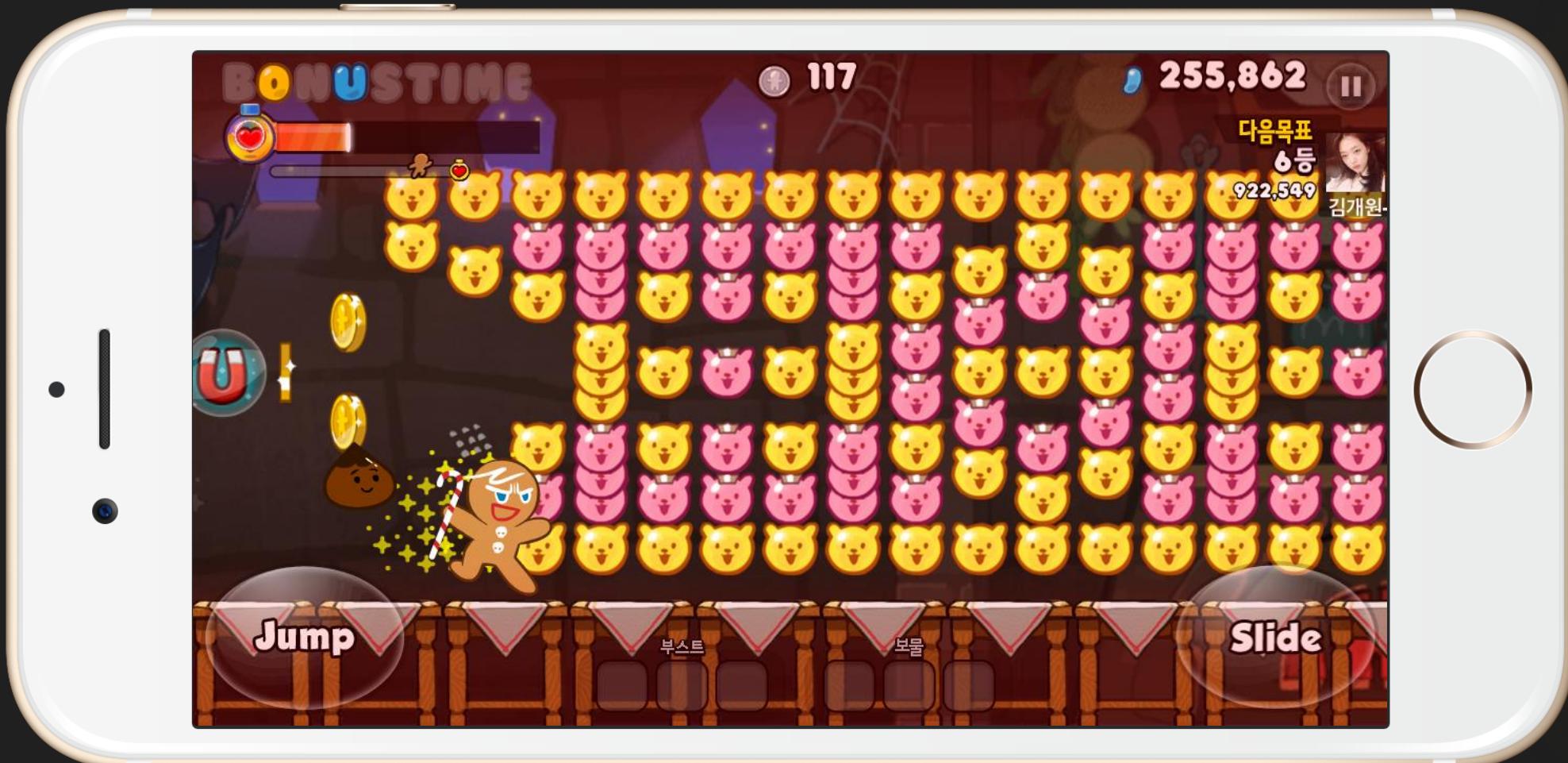
게임정보-GamelInfo, 랭킹-Rank, 아이템-Item



멤버아이템-memberitem



게임내 아이템 추가+상태 변경



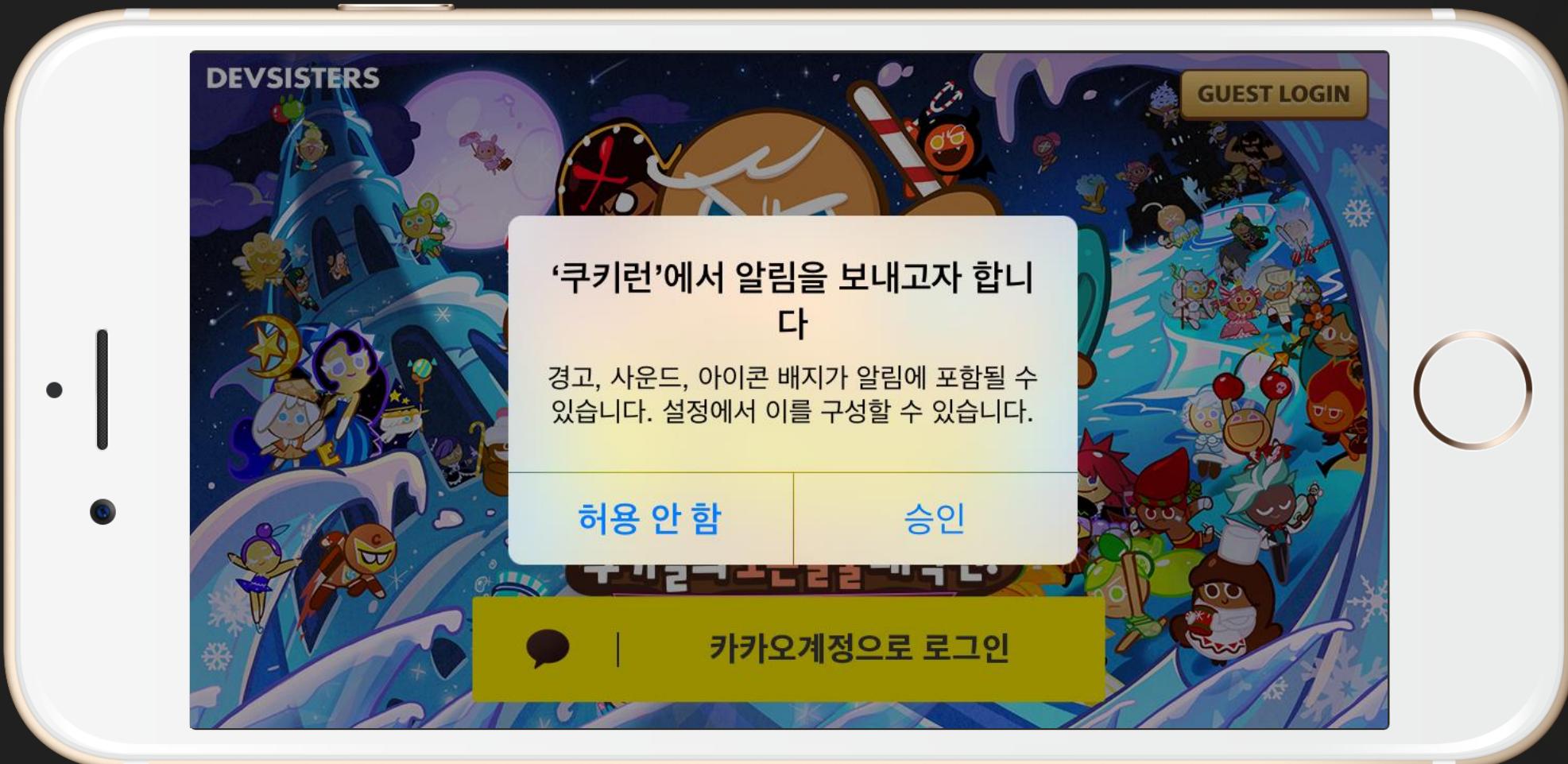
게임후 아이템 추가+상태 변경



아이템 앱 내 구매 - Purchase



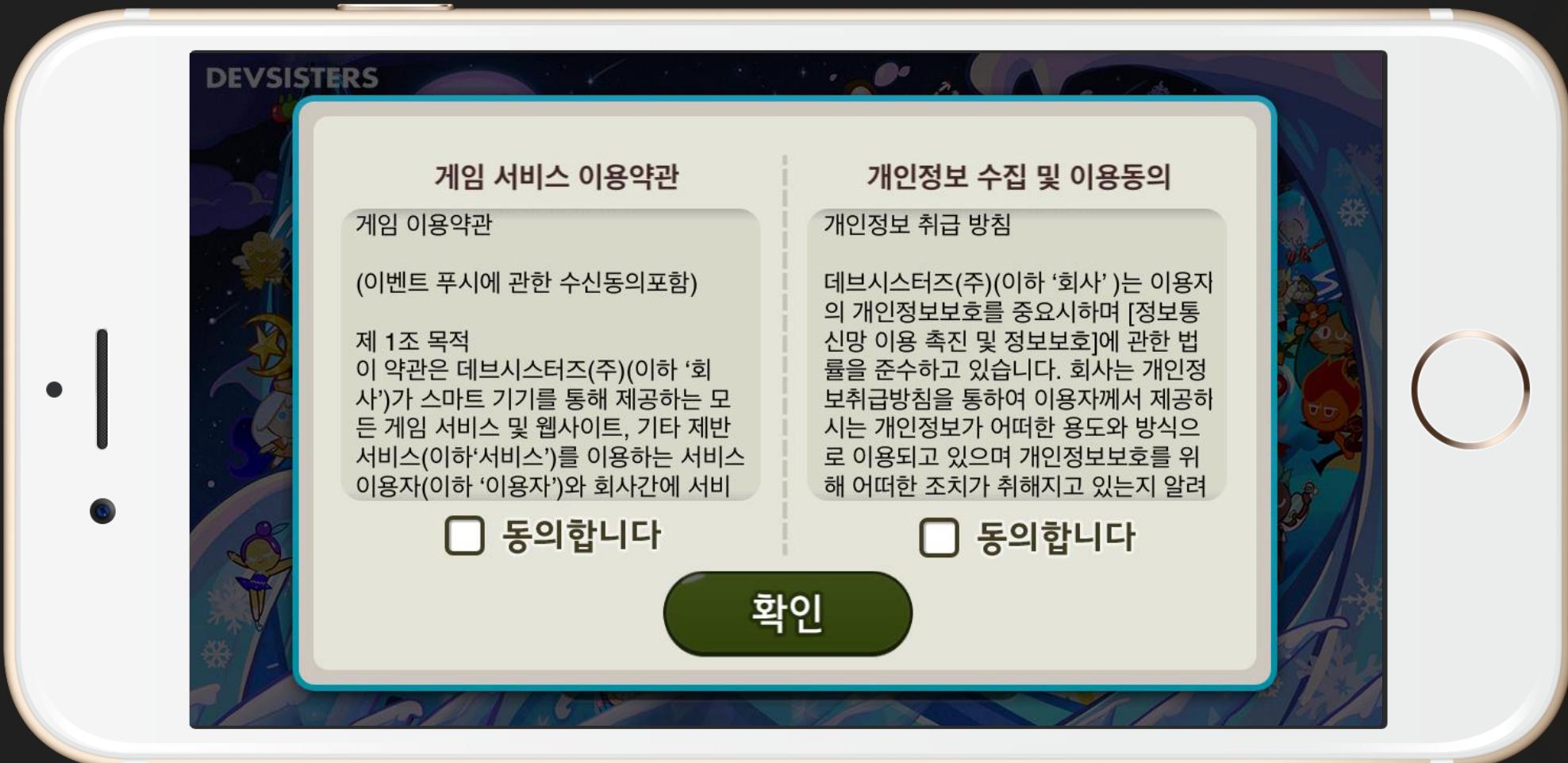
Push



인증 - 3rd party authentication



서버정보+리소스 - serverinfo



그렇다면...

어떻게 서버와 통신을?

CloudBread Unity App

DEMO

클라이언트 디바이스와
CloudBread 게임서버의 은밀한~ 통신

HTTP
REST API
JSON
+ 실시간 통신(P2P)

Unity App에서 본 그대로~!

이벤트 조회 API : SelGameEvents 를 호출한다.
HTTP Request 요청 구조

```
{  
    memberID => "member1"  
}
```

HTTP Response 응답 구조

```
{  
    EventID = "eventId1", EventName = "크리스탈1", 기간 = ...  
    EventID = "eventId2", EventName = "좋은템1", 기간 = ...  
    EventID = "eventId13", EventName = "보너스아이템", 기간 = ...  
}
```

Unity App에서 본 그대로~!

이벤트 조회 API : SelGameEvents 를 호출한다.
HTTP Request 요청 구조

```
{  
    memberID = "member1"  
}
```

HTTP Response 응답 구조

```
{  
    EventID = "eventID1", EventName = "크리스탈1", 기간 = ...  
    EventID = "eventID2", EventName = "좋은템1", 기간 = ...  
    EventID = "eventID13", EventName = "보너스아이템", 기간 = ...  
}
```

API ???

API 리스트

Game Logic = Behavior

Game Logic → API

API 추가, 수정이 간편!

Behavior 리스트
<http://aka.ms/cbp>

→ wiki → Behavior 리스트 문서

Behavior = 행위
Object = 대상

데이터베이스!

회원 : Member

관리자 : Admin

구매 : Purchase

아이템 : Item

멤버아이템 :

MemberItem

게임정보 : GameInfo

스테이지 : Stage

선물 : Gift

공지사항 : Notice

이벤트 : Event

쿠폰 : Coupon

CloudBread DB 디자인

<http://aka.ms/cbp>

→ wiki → database 디자인 문서

포스트맨(Postman) 소개

Postman 데모

<http://aka.ms/cfb>

Postman collection(20160326)

다운로드 :

<https://goo.gl/3F4By5>

Postman → import → import from file

공지사항 - Notice

- 회원 : Member
- 관리자 : Admin
- 구매 : Purchase
- 아이템 : Item
- 멤버아이템 : MemberItem
- 게임정보 : GameInfo
- 스테이지 : Stage
- 선물 : Gift
- 공지사항 : Notice
- 이벤트 : Event
- 쿠폰 : Coupon



공지사항
CBSelNotices

이벤트 - Event

- 회원 : Member
- 관리자 : Admin
- 구매 : Purchase
- 아이템 : Item
- 멤버아이템 : MemberItem
- 게임정보 : GameInfo
- 스테이지 : Stage
- 선물 : Gift
- 공지사항 : Notice
- 이벤트 : Event
- 쿠폰 : Coupon



이벤트
CBSelGameEvents

선물 - Gift

- 회원 : Member
- 관리자 : Admin
- 구매 : Purchase
- 아이템 : Item
- 멤버아이템 :
MemberItem
- 게임정보 : GameInfo
- 스테이지 : Stage
- 선물 : Gift
- 공지사항 : Notice
- 이벤트 : Event
- 쿠폰 : Coupon



이벤트

CBSelGiftItemToMe

게임정보-GamelInfo, 랭킹-Rank, 아이템-Item

- 회원 : Member
- 관리자 : Admin
- 구매 : Purchase
- 아이템 : Item
- 멤버아이템 : MemberItem
- 게임정보 : GameInfo
- 스테이지 : Stage
- 선물 : Gift
- 공지사항 : Notice
- 이벤트 : Event
- 쿠폰 : Coupon



게임정보

CBCComSelMemberGameInfoes

랭킹

GetMemberRankbyMeberID

GetRangeRankerInfo

CBRank-GetTopXRanker

멤버 아이템-memberitem

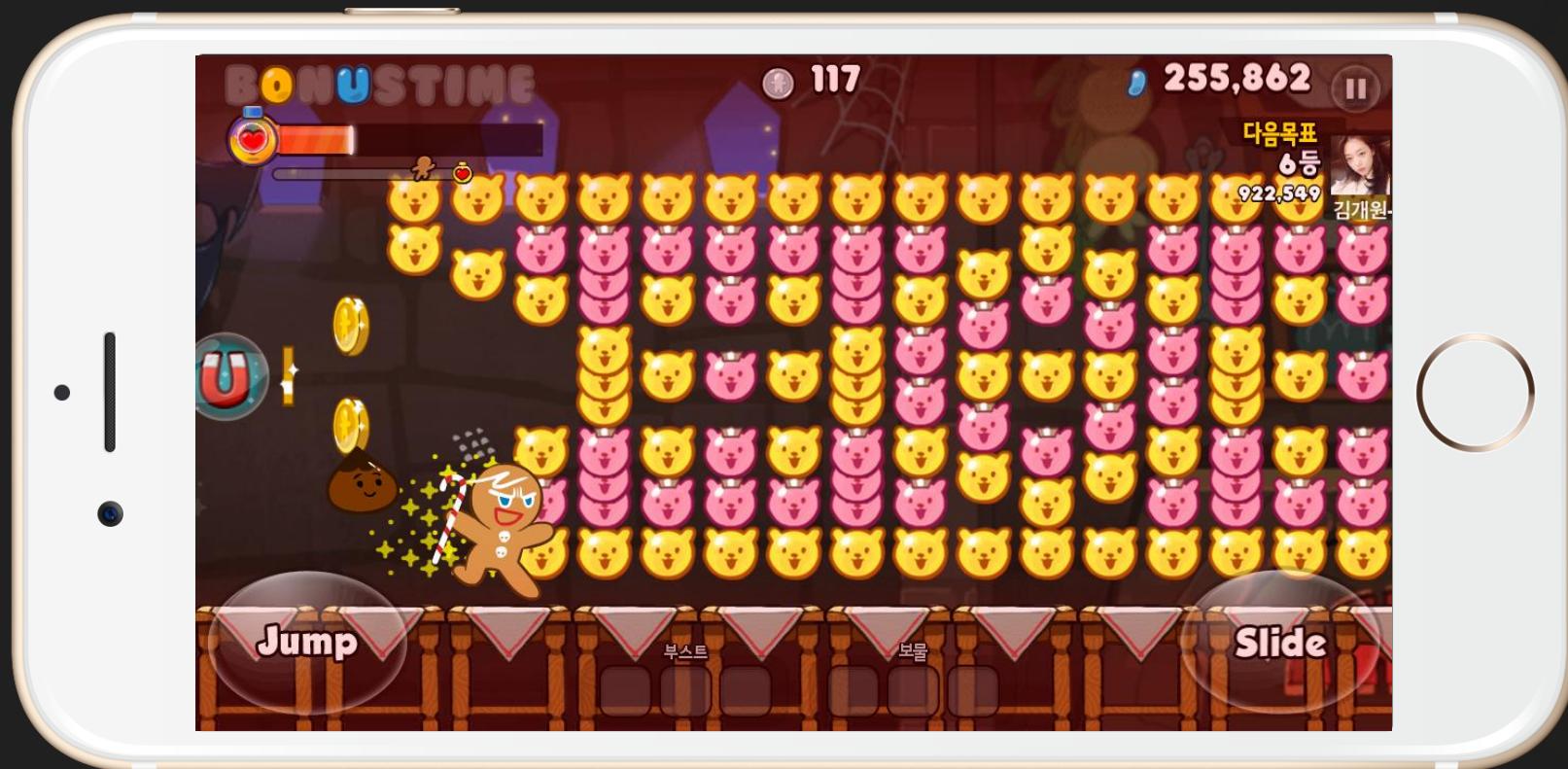
- 회원 : Member
- 관리자 : Admin
- 구매 : Purchase
- 아이템 : Item
- 멤버아이템 : MemberItem
- 게임정보 : GameInfo
- 스테이지 : Stage
- 선물 : Gift
- 공지사항 : Notice
- 이벤트 : Event
- 쿠폰 : Coupon



멤버아이템
CBSelMemberItems

게임내 아이템 추가+상태 변경

- 회원 : Member
- 관리자 : Admin
- 구매 : Purchase
- 아이템 : Item
- 멤버아이템 : MemberItem
- 게임정보 : GameInfo
- 스테이지 : Stage
- 선물 : Gift
- 공지사항 : Notice
- 이벤트 : Event
- 쿠폰 : Coupon



게임후 아이템 추가+상태 변경

- 회원 : Member
- 관리자 : Admin
- 구매 : Purchase
- 아이템 : Item
- 멤버아이템 : MemberItem
- 게임정보 : GameInfo
- 스테이지 : Stage
- 선물 : Gift
- 공지사항 : Notice
- 이벤트 : Event
- 쿠폰 : Coupon



아이템 사용

AddUseMemberItem

아이템 앱 내 구매 - Purchase

- 회원 : Member
- 관리자 : Admin
- 구매 : Purchase
- 아이템 : Item
- 멤버아이템 : MemberItem
- 게임정보 : GameInfo
- 스테이지 : Stage
- 선물 : Gift
- 공지사항 : Notice
- 이벤트 : Event
- 쿠폰 : Coupon



구매
CBCComSelMemberItemPurchase

Post Man

직접 해보기

CloudBread 개발자 가이드

Postman Hands on :

<http://aka.ms/cbp>

→ wiki → Home kor 페이지

CloudBread 사용하기 위한 기초공사

- Unity 코딩

발표자 : 홍윤석

//build/

HTTP Request 실습

```
// Headers
/*
 * Accept-Encoding:gzip
 * Accept:application/json
 * X-ZUMO-VERSION:ZUMO/2.0 (lang=Managed; os=Windows Store; os_version=--; arch=X86; version=2.0.31217.0)
 * X-ZUMO-FEATURES:AJ
 * ZUMO-API-VERSION:2.0.0
 * User-Agent:ZUMO/2.0 (lang=Managed; os=Windows Store; os_version=--; arch=X86; version=2.0.31217.0)
 * Content-Type:application/json
 */
private void HTTPRequestSend (){
    var serverEndPoint = ServerAddress + PathString;
}
```

■ 유니티 WWW 클래스 사용

UnityEngine.WWW

```
// Get the latest webcam shot from outside "Friday's" in Times Square
using UnityEngine;
using System.Collections;

public class ExampleClass : MonoBehaviour {
    public string url = "http://images.earthcam.com/ec_metros/ourcams/fridays.jpg";
    IEnumerator Start() {
        WWW www = new WWW(url);
        yield return www;
        Renderer renderer = GetComponent<Renderer>();
        renderer.material.mainTexture = www.texture;
    }
}
```

Unity HTTP Request 실습

```
// Headers
/*
 * Accept-Encoding:gzip
Accept:application/json
X-ZUMO-VERSION:ZUMO/2.0 (lang=Managed; os=Windows Store; os_version=--; arch=X86; version=2.0.31217.0)
X-ZUMO-FEATURES:AJ
ZUMO-API-VERSION:2.0.0
User-Agent:ZUMO/2.0 (lang=Managed; os=Windows Store; os_version=--; arch=X86; version=2.0.31217.0)
Content-Type:application/json
*/
private void HTTPRequestSend (){
    var serverEndPoint = ServerAddress + PathString;
}
```

HTTP Request 실습

```
// Headers
/*
 * Accept-Encoding:gzip
Accept:application/json
X-ZUMO-VERSION:ZUMO/2.0 (lang=Managed; os=Windows Store
X-ZUMO-FEATURES:AJ
ZUMO-API-VERSION:2.0.0
User-Agent:ZUMO/2.0 (lang=Managed; os=Windows Store; os
Content-Type:application/json
*/
private void HTTPRequestSend (){
    var serverEndPoint = ServerAddress + PathString;
}
```

Server Address :

Path :

결과 :

HTTP Request 실습

```
private void HTTPRequestSend (){
    var serverEndPoint = ServerAddress + PathString;

    Dictionary<string, string> Header = new Dictionary<string, string> ();
    Header.Add("Accept-Encoding", "gzip");
    Header.Add ("Accept", "application/json");
    Header.Add ("X-ZUMO-VERSION", "ZUMO/2.0 (lang=Managed; os=Windows Store; os_version=--; arch=X86; version=2.0.31217.0)");
    Header.Add ("X-ZUMO-FEATURES", "AJ");
    Header.Add ("ZUMO-API-VERSION", "2.0.0");
    Header.Add ("User-Agent","ZUMO/2.0 (lang=Managed; os=Windows Store; os_version=--; arch=X86; version=2.0.31217.0)");
    Header.Add ("Content-Type", "application/json");

    WWW www = new WWW(serverEndPoint, null, Header);
    StartCoroutine(WaitForRequest(www));
}

private IEnumerator WaitForRequest(WWW www) {
    yield return www;
    RequestResultJson = www.text;
    www.Dispose();
}
```

CloudBread 를 더 쉽게 쓰기

- Unity SDK 데모

발표자 : 홍윤석

//build/

CloudBread Unity SDK 데모

레포 명 : CloudBread-Unity-SDK

사용법은 CloudBread Wiki 를 참조해 주세요!

CloudBread Socket Demo

라이브 데모 : <http://goo.gl/S5OzJ0>

CloudBread 에서 실시간 채팅 사용하기

- CloudBread Socket 데모



WEB APPS

How to install CloudBread-Socket.IO

Type

```
npm run deploy
```

라이브 데모 : <http://goo.gl/S5OzJ0>

Username

AUTHORIZE

Join Channel

Channel Short ID

JOIN

#테스트방

유저이름 이렇게 메세지 보내면 됨

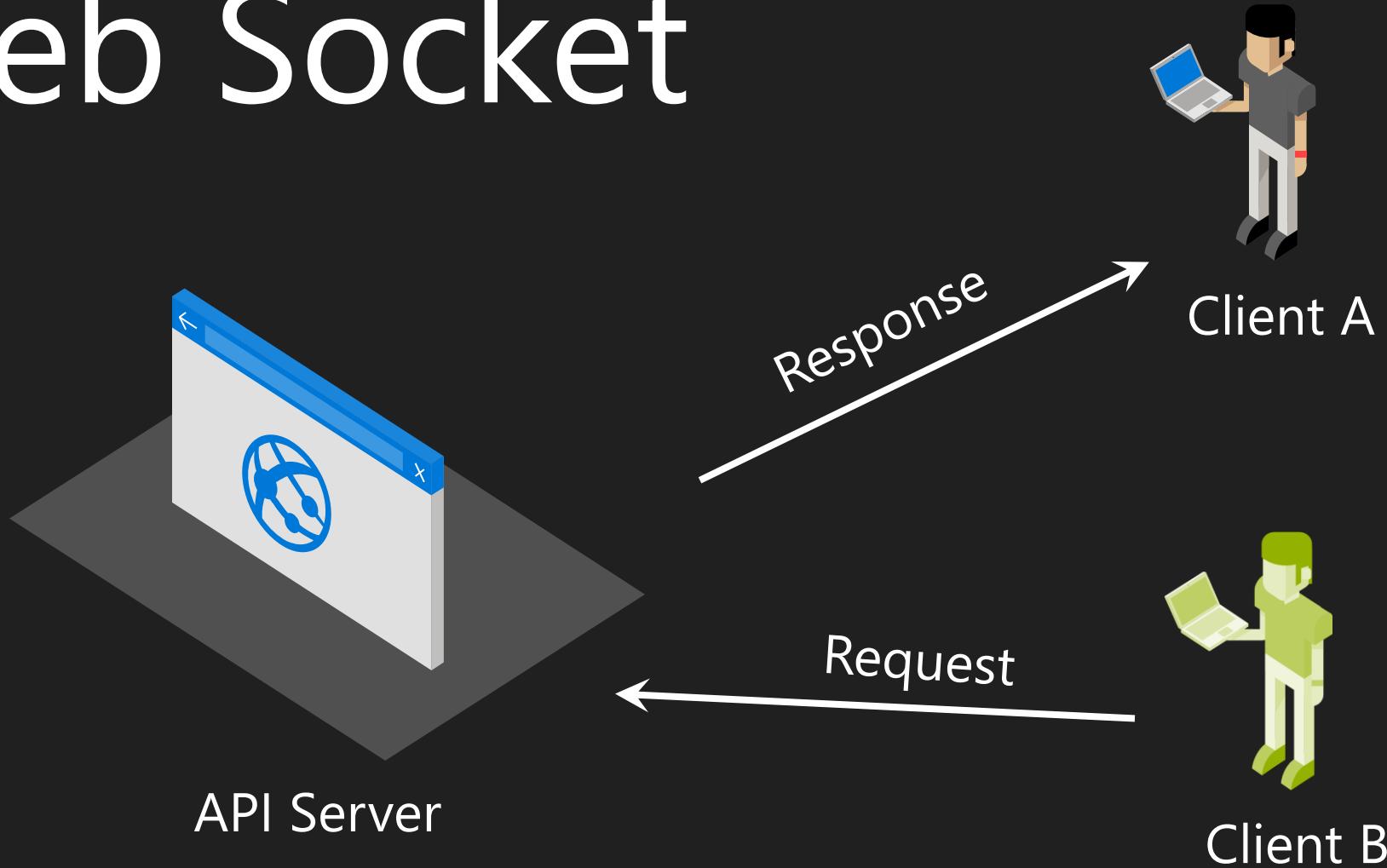
Message

SEND

CloudBread 에서 실시간 채팅 사용하기

- Socket.IO 소개

Web Socket





How to install CloudBread-Socket.IO

WEB APPS

GENERAL

- Quick start >
- Properties >
- Application settings >

Allow web-socket

Web sockets i

Connect to Redis

App settings

| | | | |
|---------------------------|--------------------------------|---------------------------------------|-----|
| WEBSITE_NODE_DEFAULT_V... | 4.2.3 | <input type="checkbox"/> Slot setting | ... |
| REDIS_HOST | dw-cloudbread-redis.redis.c... | <input type="checkbox"/> Slot setting | ... |
| REDIS_PORT | 6379 | <input type="checkbox"/> Slot setting | ... |
| REDIS_AUTH_KEY | EoQ2mU+U7Alkr2hFv6+iQ... | <input type="checkbox"/> Slot setting | ... |

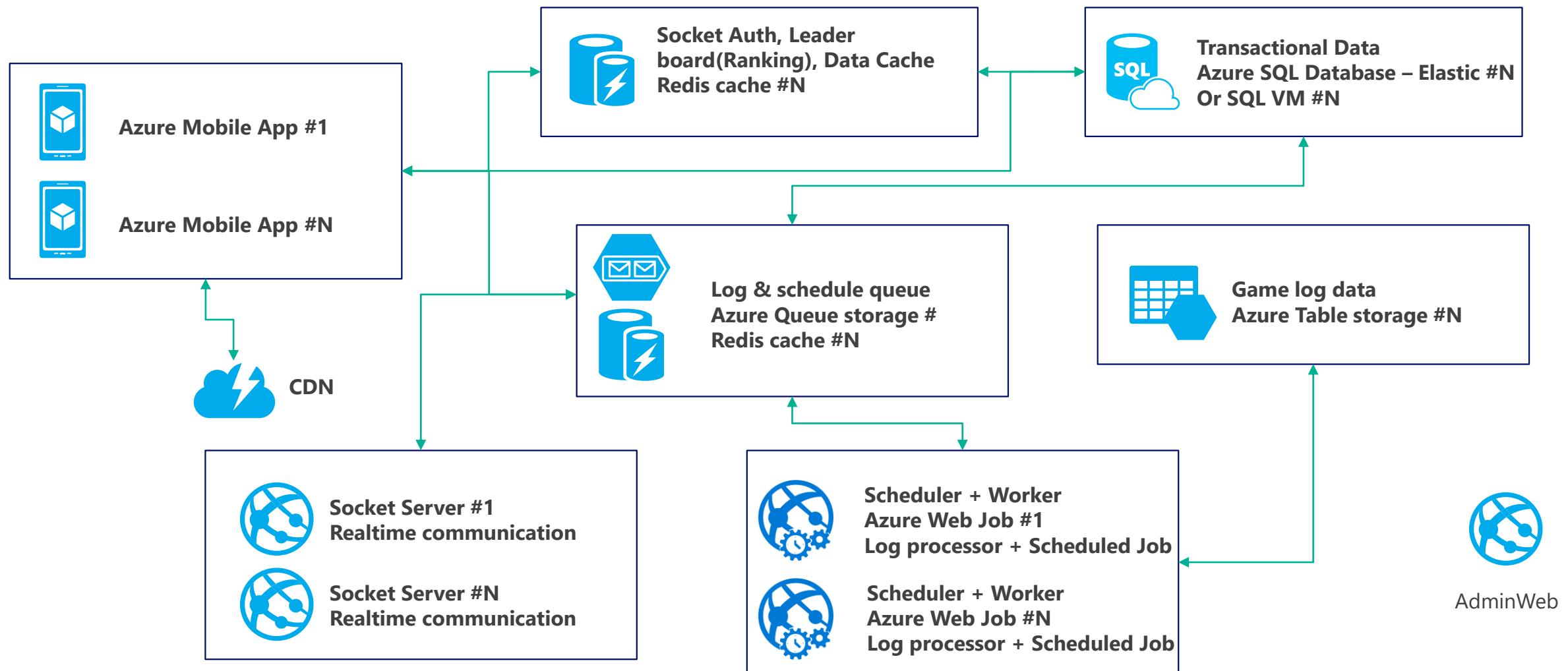
CloudBread 로 게임 운영하기

- CloudBread Admin-Web 데모

CloudBread Service Architecture

CloudBread – (2) “Cumulonimbus” project architecture design

– Full PaaS – most “flexible” & wide range of game & app service



한방에 끝내는 클라우드 배포

- ARM 배포 소개

- Your application code
- ~~Requiring fast socket code~~





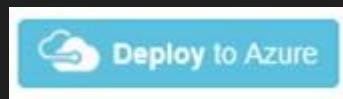
You: Code (application, infrastructure)
Azure: Resources (IaaS, PaaS)

How to deployment

- Fork the Cloudbread-ARM project



- Click the [Deploy to Azure] button



- Input parameters and Create resource group

The screenshots illustrate the deployment process:

- Resource group:** A new resource group named "jh-rg-cloudbread" is created in the "Japan West" location.
- Parameters:** A parameter named "POSTFIXAPPNAME" is defined as a string type.
- OK:** The deployment is confirmed with the "OK" button.

주의사항

- postfix app name
 - 오직 소문자 또는 숫자
 - 16자 이내
- SQL server login ID
 - admin 이란 ID 사용 불가
- SQL server login password
 - 최소 8자
 - 하나이상의 특수문자 필수

ARM DEMO & Hands on

Demo 영상 : <https://youtu.be/TvtXkRAymcg>

사용자 인증 기능 활용하기!
- Authentication 인증 기능 소개

인증 - 3rd party authentication

카카오계정 닫기

 데브시스터즈(주)
쿠키런

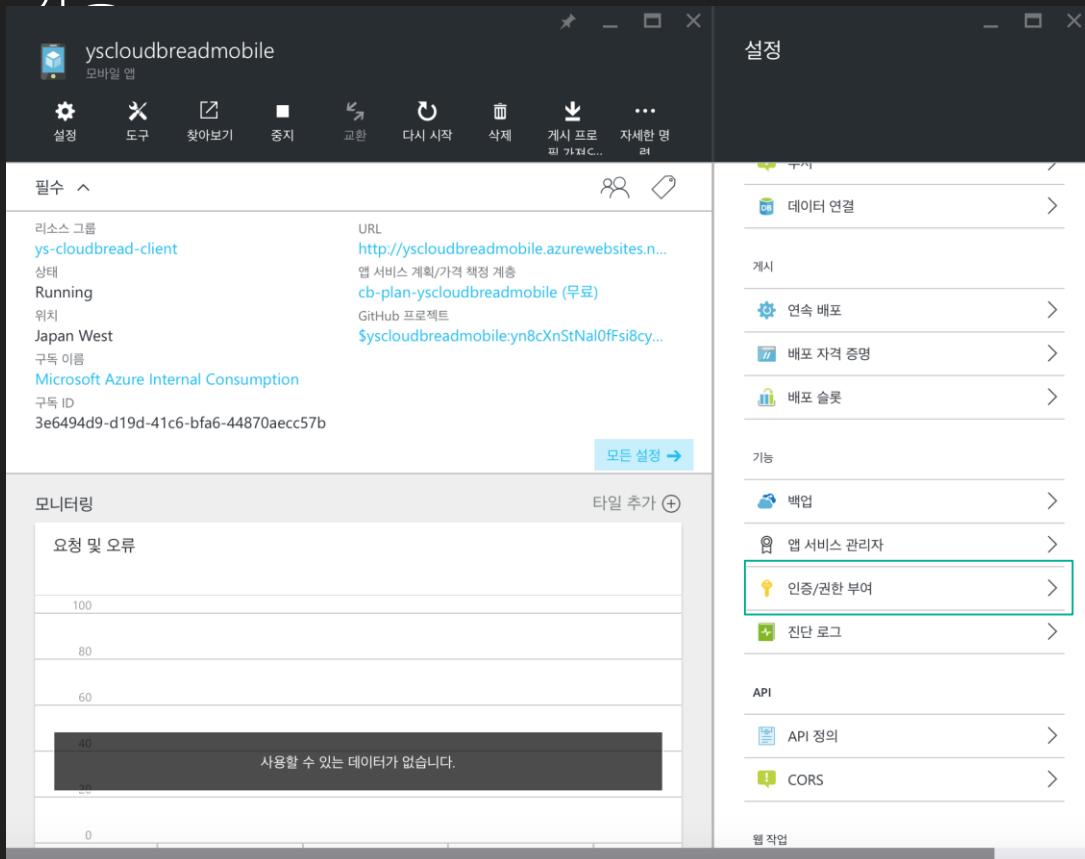
이 앱은 회원님의 카카오톡 이름(닉네임), 프로필 사진, 카카오톡 친구목록 등의 개인정보를 사용할 수 있습니다. 친구목록은 앱 상에서 이용자 본인이 친구들을 찾는 데 한정되어 사용됩니다.

아래에 동의하시면 이 정보를 해당 앱 개발자/개발업체에 제공하게 되며, 해당 앱은 이를 회원님의 앱 탈퇴시까지 보유하게 됩니다.

kim_daewoo@naver.com

Azure Mobile App 인증이란?

- Azure에서 제공하는 회원 아이디 관리, 회원 인증 관리 등의 편리한 기능



어떻게 회원 인증 처리를 할까?

컨셉

HTTP Rest Request 를 할

때마다 암호화된 Token 를 같이

보내자!

CloudBread on Azure가 발급하는

Token 을 가지고 있을때만 대답하자!

(Response)

Request Header 에 올바른 Token을 넣었을 때

The screenshot shows a web-based tool for making API requests. At the top, there are input fields for 'Server Address' (set to <https://dw-cloudbread2.azurewebsites.net/>) and 'Path' (set to /api/ping). Below these are two buttons: 'Send' and 'Auth Send'. The 'Auth Send' button is highlighted with a blue border. To the right of the buttons is a section labeled '결과:' (Result) containing the text "Hello". At the bottom of the result area, the text "정상적인 응답을 받음!" (Received a normal response!) is displayed.

어떻게 회원 인증 처리를 할까?

컨셉

HTTP Rest Request 를 할

때마다 암호화된 Token 를 같이

보내자!

CloudBread on Azure가 발급하는

Token 을 가지고 있을때만 대답하자!

(Response)

Request Header 에 올바른 Token을 넣었을 때

The screenshot shows a web-based tool for making API requests. At the top, there are input fields for 'Server Address' (set to <https://dw-cloudbread2.azurewebsites.net/>) and 'Path' (set to /api/ping). Below these are two buttons: 'Send' and 'Auth Send'. The 'Auth Send' button is highlighted with a blue border. To the right of the buttons is a section labeled '결과:' (Result) containing the text "[Error]" in quotes. At the bottom of the interface, the word "Hello" is displayed.

이렇게 편리한 Token 어떻게 받지?

ID : LoveCloudBread

PW : *****

페이스북 계정

Google + 계정

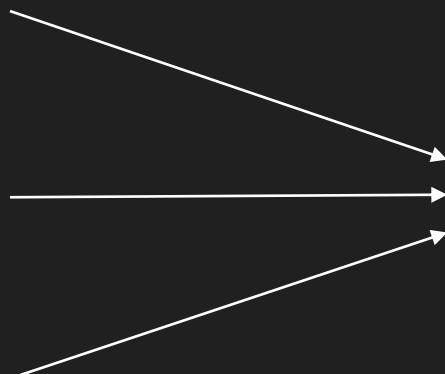
Microsoft 계정

자체 인증... 구현... ㅠㅠ
어차피 유저들이 안써...

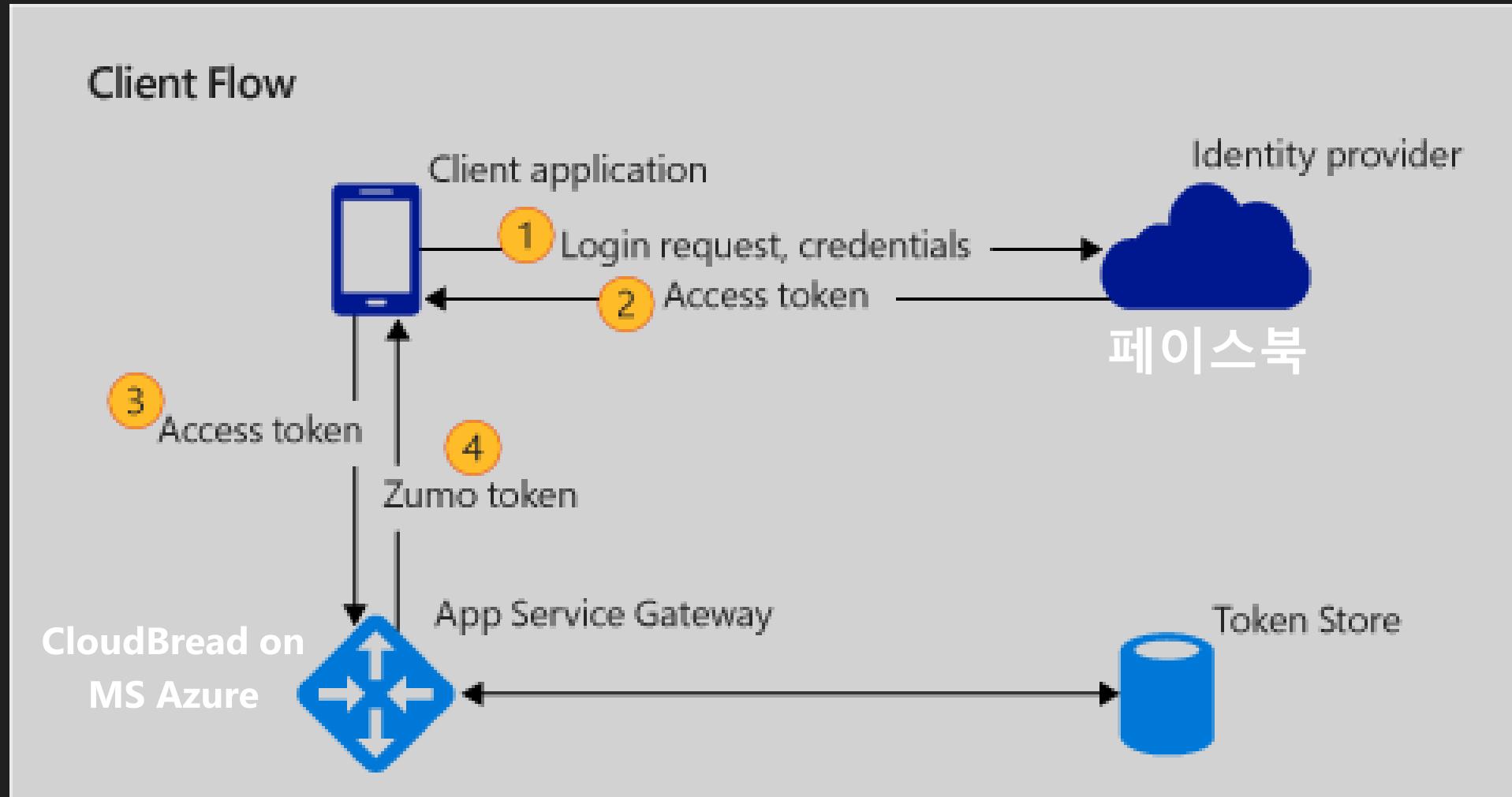


CloudBread(MS Azure)

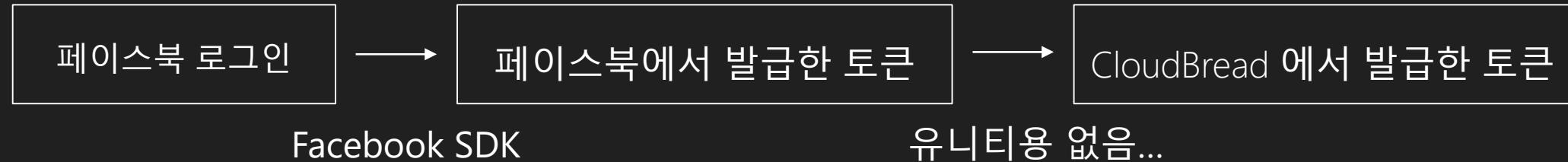
각 사이트에서 발급한 토큰



Azure Mobile App 인증



CloudBread 회원 인증



1. 열심히 게임을 만든다. (Unity로…)
2. 페이스북 개발자 페이지에서 앱을 생성한다.
(<http://developers.facebook.com>)
3. Azure에서 생성된 Mobile App에서 페이스북 인증 등록을 한다.
4. Facebook SDK for Unity를 다운받는다.
(<https://developers.facebook.com/docs/unity>)
5. 예제 Copy & Paste
6. 끝!

<https://developers.facebook.com>

The screenshot shows the homepage of the Facebook Developers website. At the top, there's a navigation bar with links for "Products", "Docs", "Tools & Support", and "News". To the right of the navigation is a search bar and a "My Apps" dropdown menu. The dropdown menu lists several apps: "CloudBreadApp", "Cloudbread", "HelloDoctor", and "Shall We Mate ?". Below the dropdown, there's a link to "Add a New App". Further down, there are links for "Requests", "Developer Settings", "Company Settings", and "Log Out". The main content area features a large, semi-transparent overlay with the text "Connect on a global scale" and "Build, grow and monetize your apps with Facebook". The background of the page shows a blurred image of a person working at a desk with a laptop, phone, and other office supplies.

facebook for developers

Products Docs Tools & Support News

Search

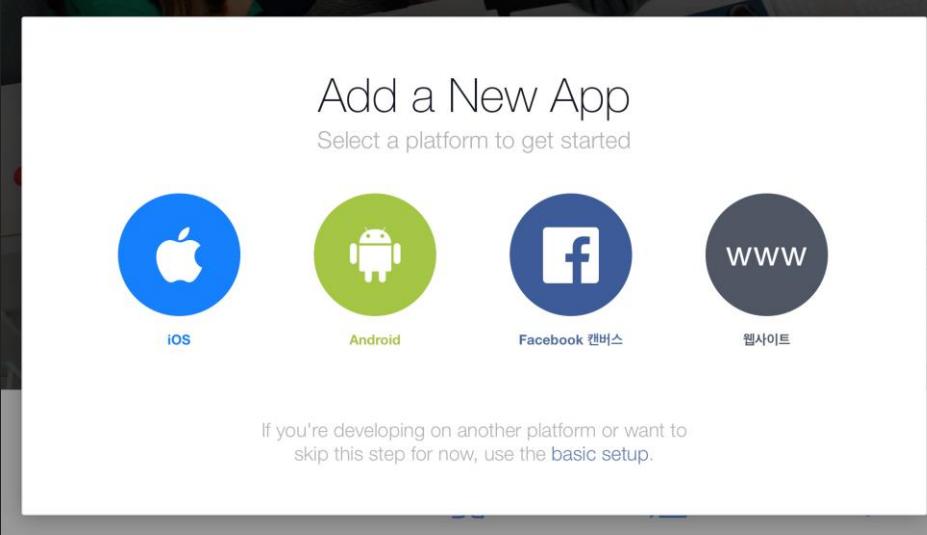
My Apps ▾

CloudBreadApp
Cloudbread
HelloDoctor
Shall We Mate ?

Add a New App

Requests
Developer Settings
Company Settings
Log Out

Connect on a global scale
Build, grow and monetize your apps with Facebook



Tell us about your website

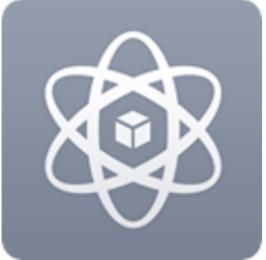
Site URL

URL of your site

Next

- <http://<your ID>.azurewebsites.net>

Dashboard



Cloubread o

This app is in development mode and can only be used by app admins, developers and testers [?]

| App ID | API Version [?] | App Secret |
|------------------|-----------------|---|
| 1737997756487447 | v2.5 | ••••••• Show |

- Settings -> Advanced

Valid OAuth redirect URIs

<http://yscloudbreadmobile.azurewebsites.net/login/facebook> X

- `http://<your ID>.azurewebsites.net/login/facebook`

Azure Portal :

https://portal.azure.com

The screenshot shows three windows from the Azure Portal:

- Left Window (Dashboard):** Shows basic settings like URL (<http://yscloudbreadmobile.azurewebsites.net>), App Service Plan (cb-plan-yscloudbreadmobile (무료)), GitHub Project, and a public IP address (370aecc57b). It also has a "모든 설정" (All Settings) button.
- Middle Window (Settings):** Under the "인증/권한 부여" (Authentication/Authorization) section, it shows the "설정" (Configure) button for the Facebook provider. Other providers listed are Azure Active Directory (Not configured), Google (Not configured), Twitter (Not configured), and Microsoft 계정 (Not configured).
- Right Window (Facebook Authentication Configuration):** A detailed configuration page for the Facebook provider. It includes:
 - Facebook icon:** Facebook logo.
 - Description:** "이 설정을 통해 사용자가 Facebook으로 로그인할 수 있습니다. 자세히 알아보려면 여기를 클릭하세요." (This setting allows users to log in with Facebook. Click here to learn more.)
 - App ID:** 1121658894531533
 - App Secret:** 419f41dbb3bc978e094e8b6289a5f1d8
 - Permissions:** A list of permissions requested from users:
 - public_profile: 개인의 공용 프로필에 포함된 항목의 하위 집합에 대한 액세스를 제공합니다.
 - user_friends: 사용자 앱이 사용하는 친구 목록에 대한 액세스를 제공합니다.
 - email: 사용자 개체의 전자 메일 속성을 통해 개인의 기본 전자 메일 주소에 대한 액세스를 제공합니다.
 - user_actions.books: 사용자 개체의 전자 메일 속성을 통해 개인의 기본 전자 메일 주소에 대한 액세스를 제공합니다.
 - user_actions.fitness: 개인이 사용한 앱에서 게시된 일반적인 모든 오픈 그래프 피트니스 활동에 대한 액세스를 제공합니다.
 - user_actions.music: 개인이 사용한 앱에서 게시된 일반적인 모든 오픈 그래프 피트니스 활동에 대한 액세스를 제공합니다.
 - user_actions.news: 이러한 활동을 게시하는 개인이 사용한 앱에서 게시된 일반적인 모든 오픈 그래프 피트니스 활동에 대한 액세스를 제공합니다.
 - user_actions.video: 이러한 활동을 게시하는 개인이 사용한 앱에서 게시된 일반적인 모든 오픈 그래프 피트니스 활동에 대한 액세스를 제공합니다.
 - Confirm Button:** A blue "확인" (Confirm) button at the bottom right.

Facebook SDK for Unity

Build cross-platform games with
Facebook rapidly and easily

[Download the SDK](#)

Requires Unity 5.

v7.4.0 See Change Log or Upgrade Guide.



[Get Started](#)

Basic guide for Unity SDK



[SDK Reference Docs](#)

API Reference Docs



[View Source](#)

View Source On Github



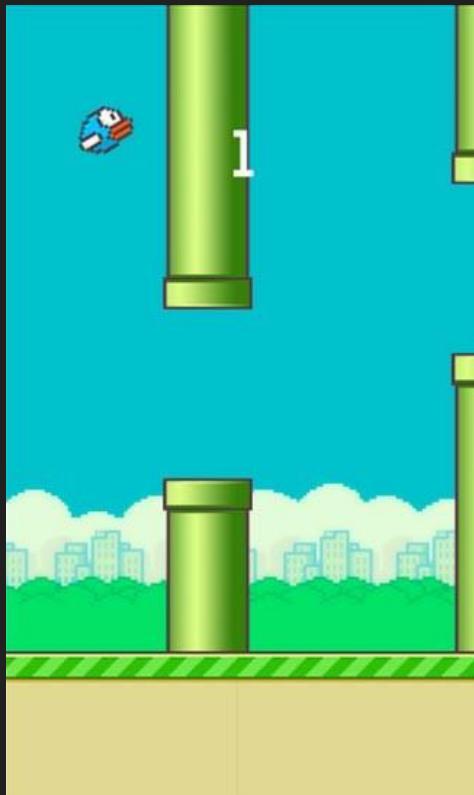
<https://developers.facebook.com/docs/unity>

Floppy Bird - Unity Demo

홍윤석

<https://youtu.be/umWGSm0h8kE>

오늘의 미션! 플래피 버드 게임에 CloudBread 붙이기!



1. CloudBread를 사용한 사용자 로그인 구현
페이스북 로그인 후, CloudBread 에 사용자 정보 등록하기
사용 API : CBInsRegMember
2. 게임 스코어, CloudBread 에 기록하기!
CloudBread API 를 사용하여 DB 에 게임 데이터 저장하기
사용 API : CBCComUdtMmberGameInfoes
3. 내 랭킹 가져오기! (시간 가능하면...)

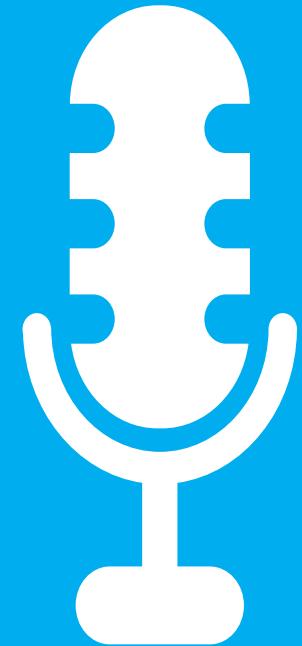
중간에 막히는 부분이 있으면 언제든 질문해주세요!

CloudBread 설치 가이드

<http://aka.ms/cbp>

-> wiki -> install guide kor

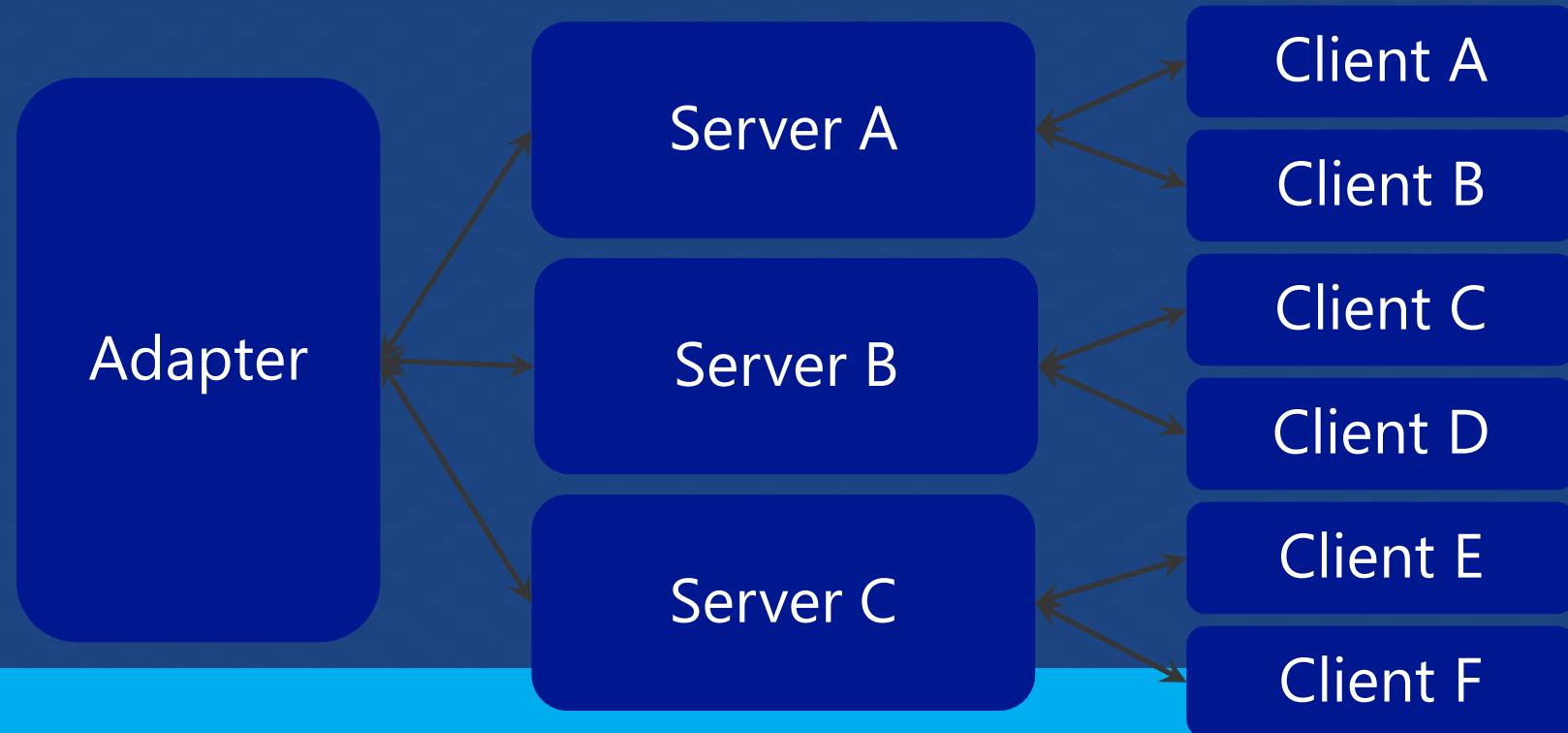
Q&A



Appendix

Why need Socket.IO?

To enable servers real-time communicate with clients





WEB APPS

How to install CloudBread-Socket.IO

Needs



Type

```
git clone https://github.com/CloudBreadProject/CloudBread-Socket.git  
cd CloudBread-Socket  
npm i # alias to install
```



How to install CloudBread-Socket.IO

WEB APPS

Match your web app credentials

/tools/tasks/deploy.js

URL

<http://hbh-cloudbread-socket.azurewebsites.net>

App Service plan/pricing tier

hbh-nodejs (Standard: 1 Small)

Git/Deployment username

beingbook

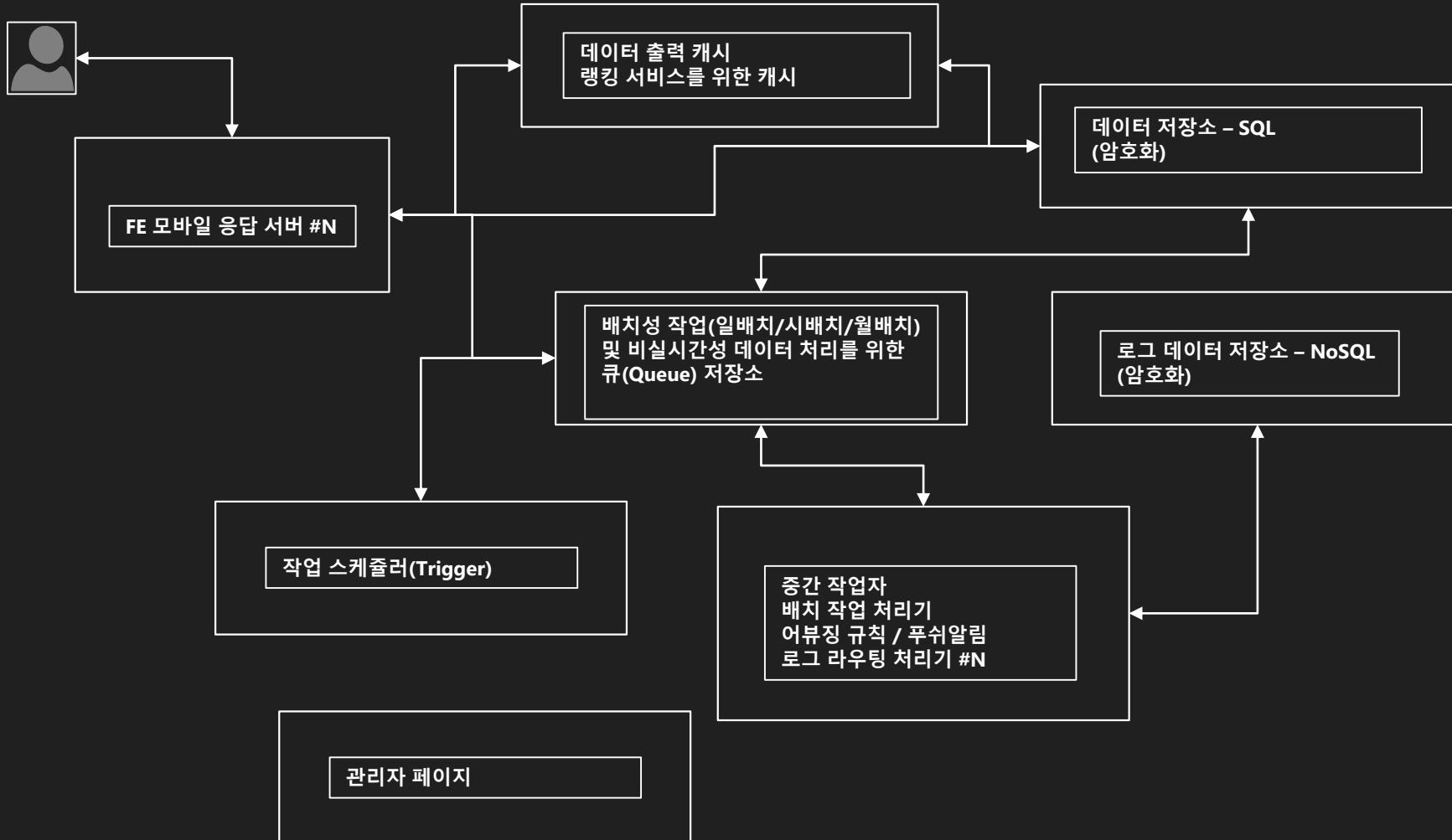
Git clone url

<https://beingbook@hbh-cloudbread-socket.scm.azurewebsites.net:443/hbh-cloudbread-socket.git>

```
10  async function deploy() {
11      // By default deploy to the staging deployment slot
12      const remote = {
13          name: 'azure',
14          url: 'https://beingbook@hbh-cloudbread-socket.scm.azurewebsites.net:443/hbh-cloudbread-socket.git',
15          website: 'http://hbh-cloudbread-socket.azurewebsites.net',
16      };

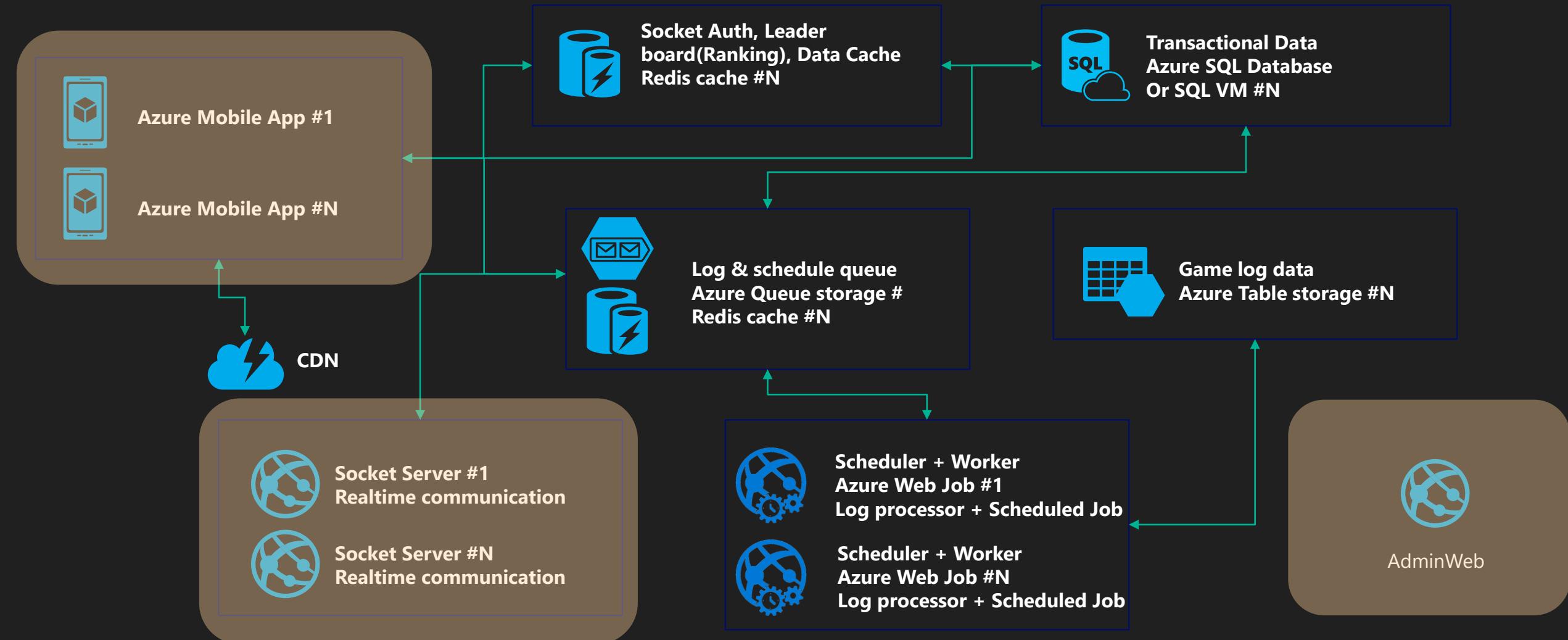
```

CloudBread – 추상화 디자인 아키텍처



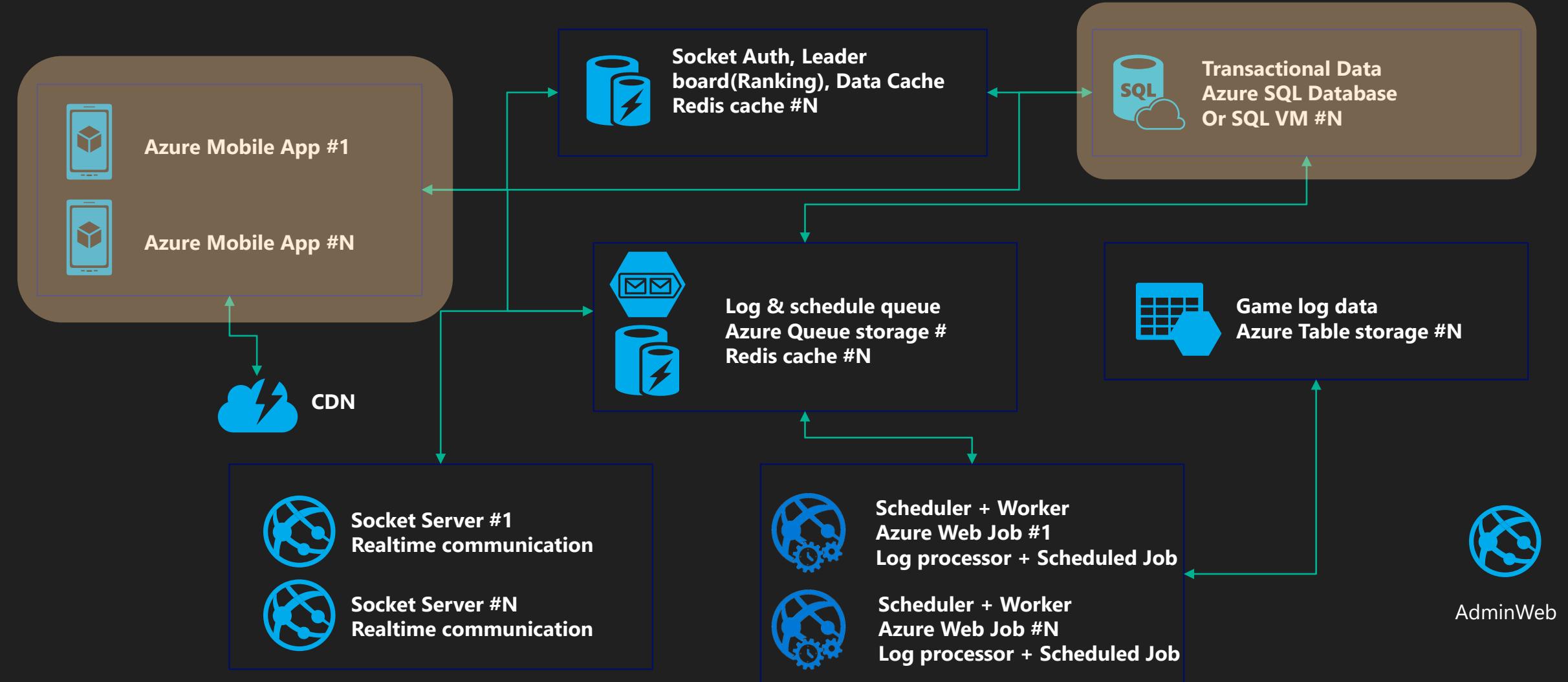
CloudBread – (2) “Cumulonimbus” project architecture design

– CloudBread Front-End 서버, Socket 서버와 Admin webpage 서버



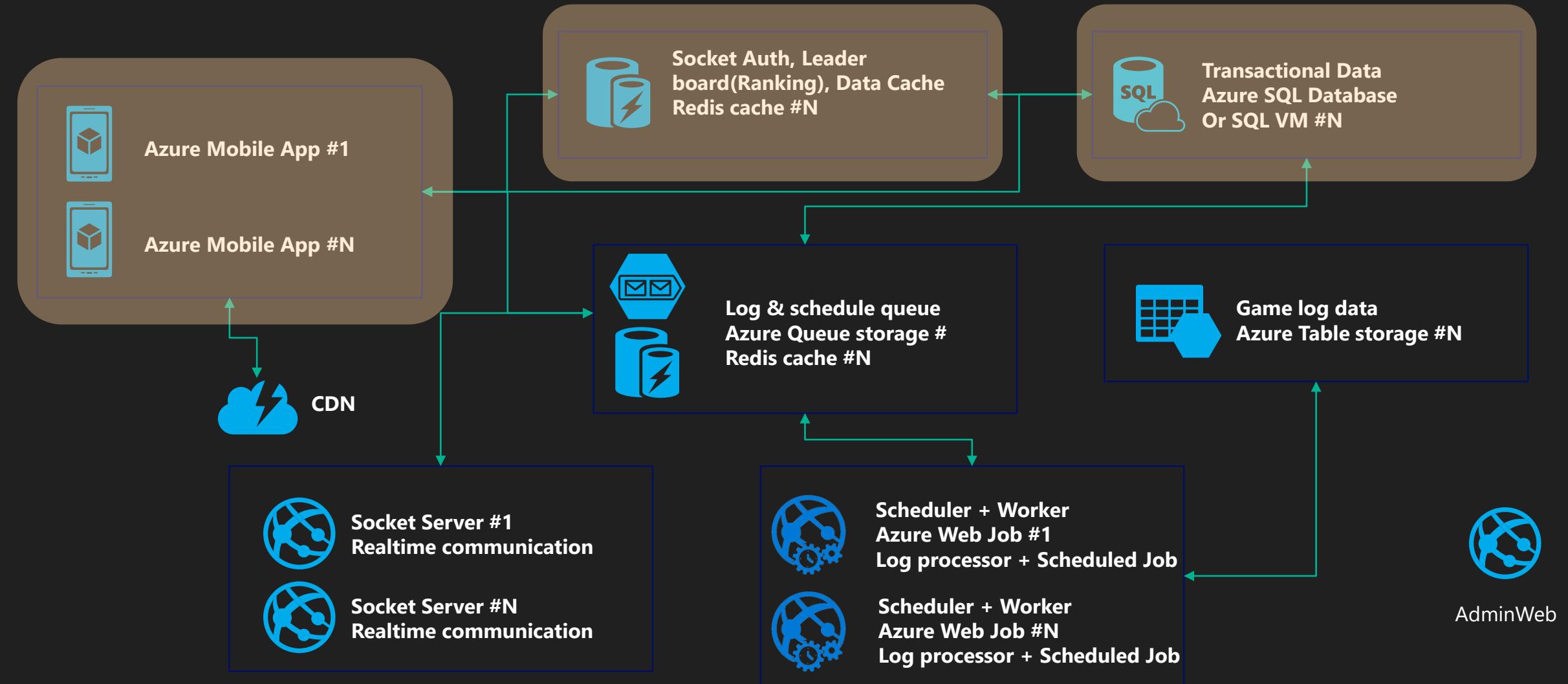
CloudBread – (2) “Cumulonimbus” project architecture design

– Transactional Data 처리



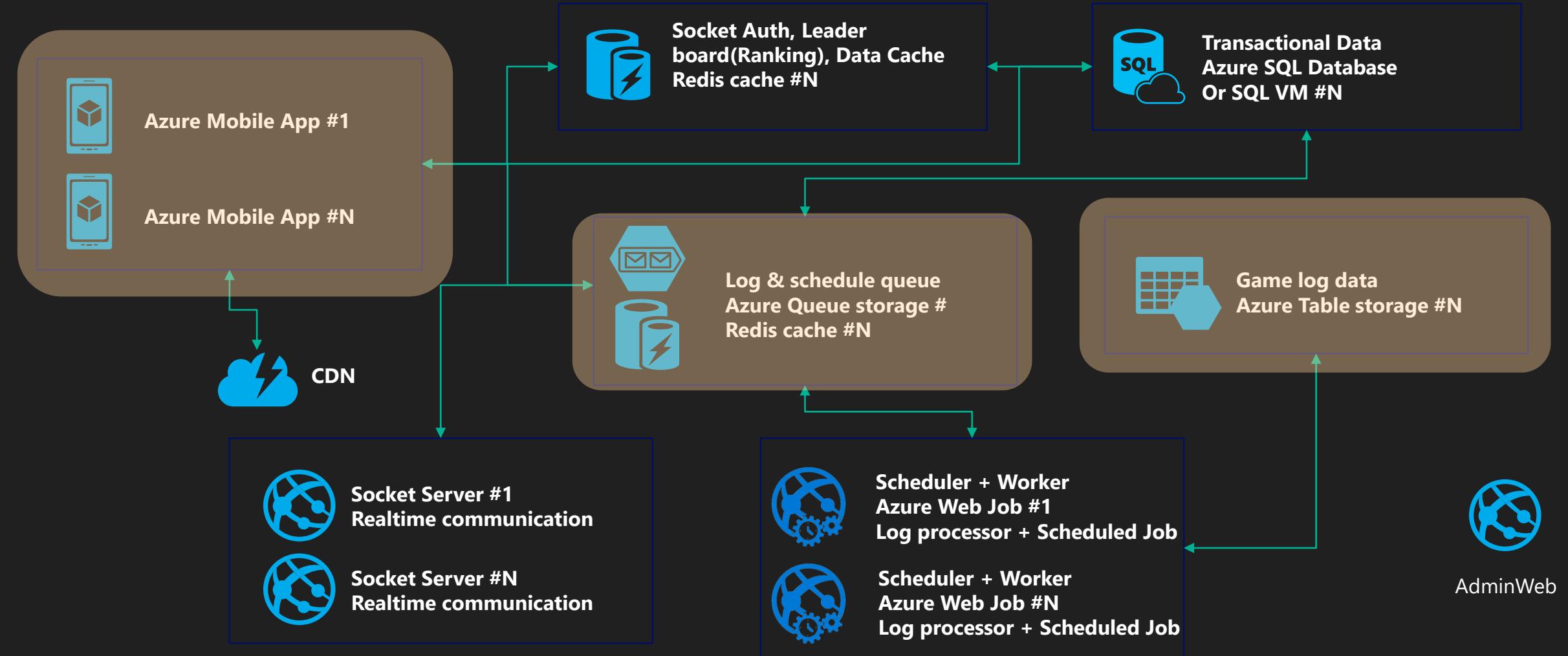
CloudBread – (2) “Cumulonimbus” project architecture design

– DB 캐시



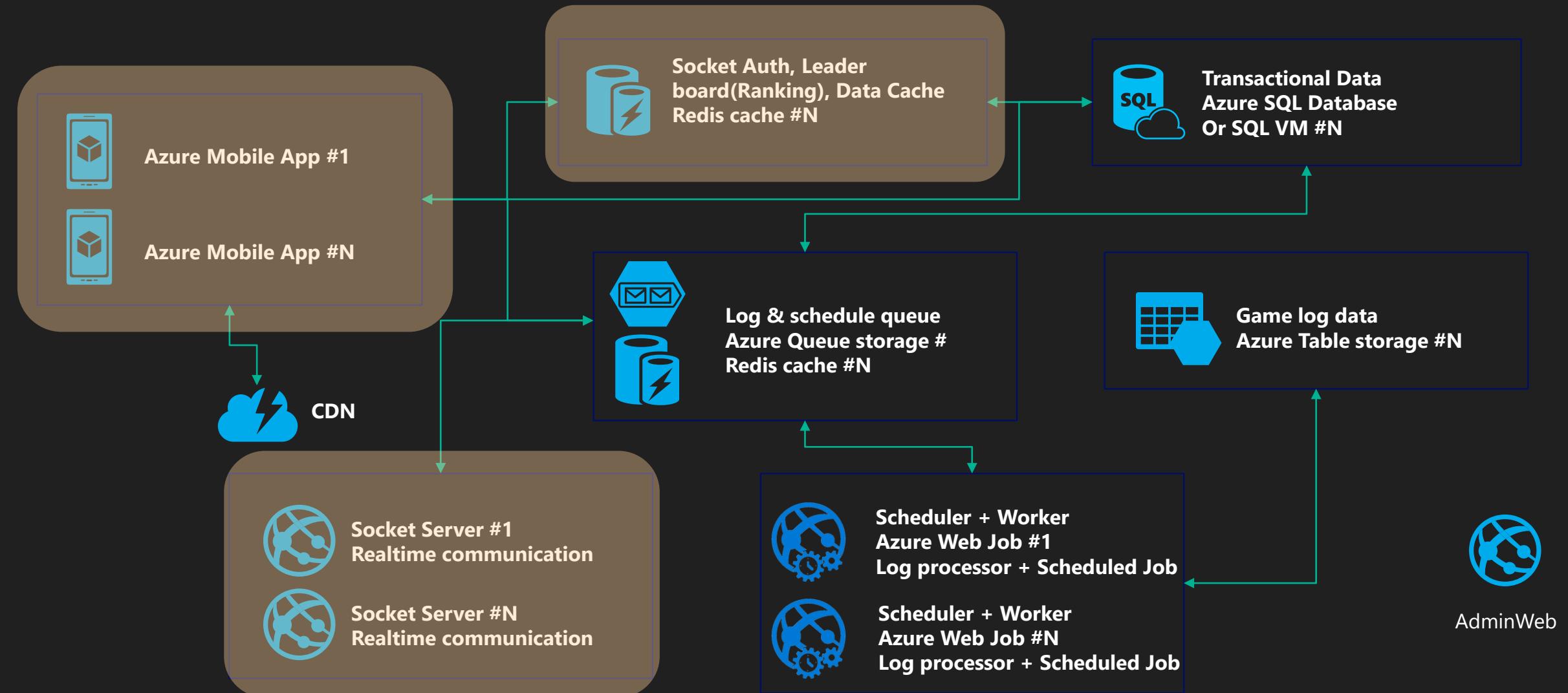
CloudBread – (2) “Cumulonimbus” project architecture design

– 게임 로그 저장



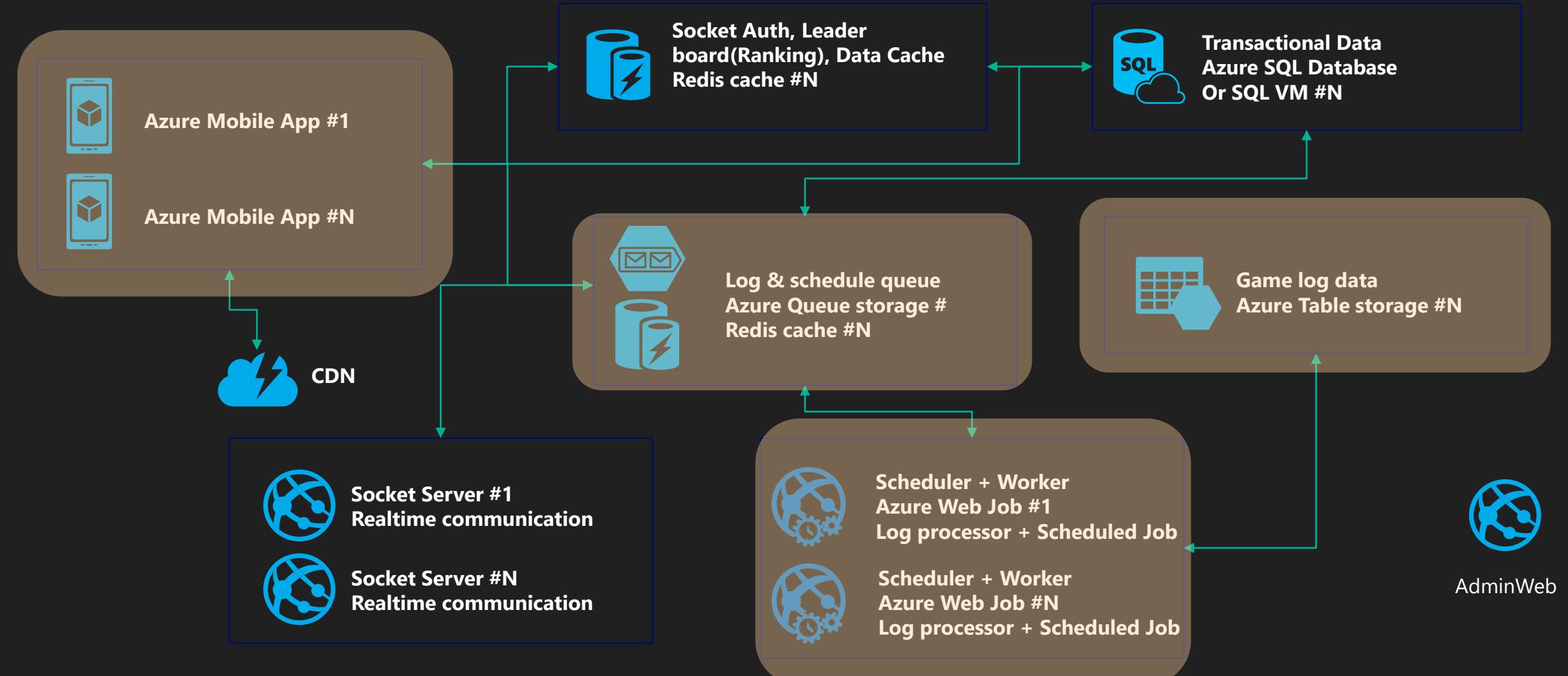
CloudBread – (2) “Cumulonimbus” project architecture design

– 실시간 통신 – Socket 서버



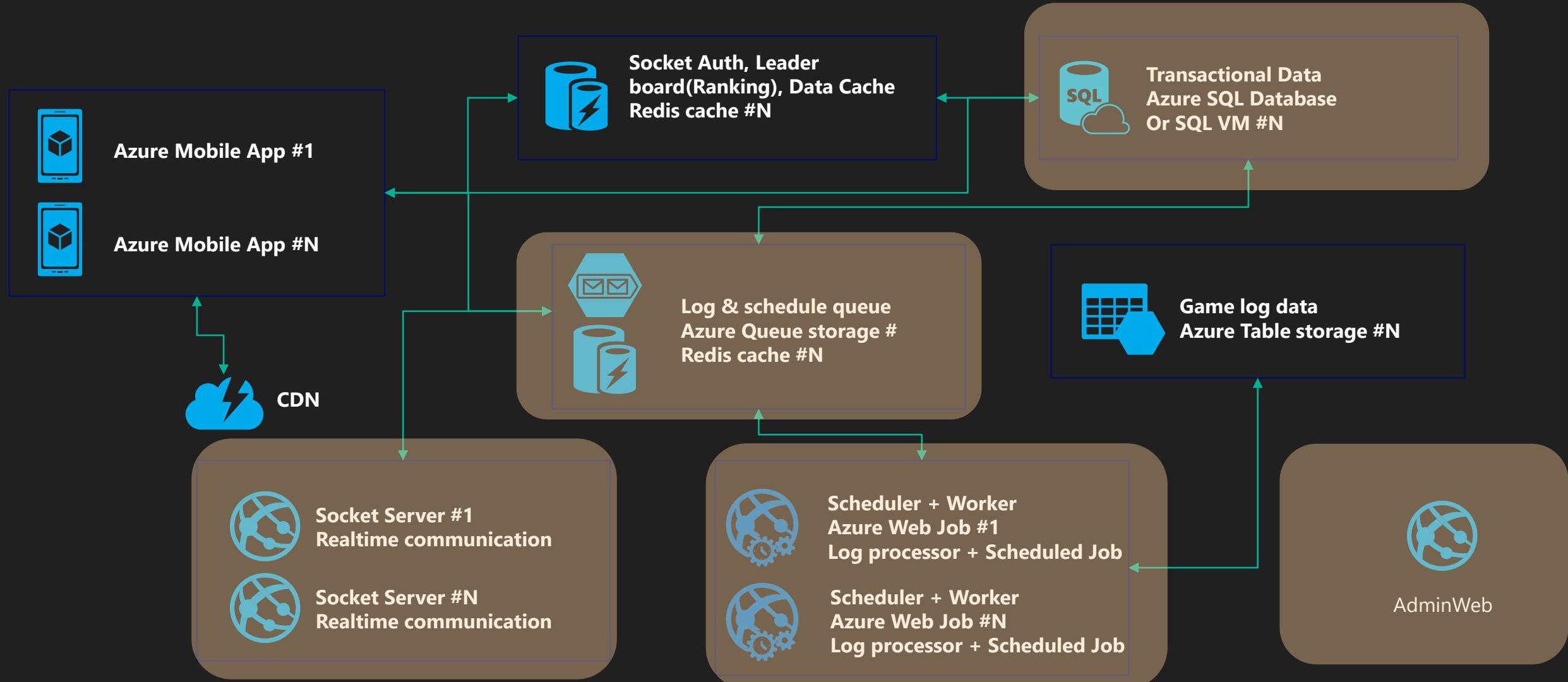
CloudBread – (2) “Cumulonimbus” project architecture design

- 대량 로그 저장



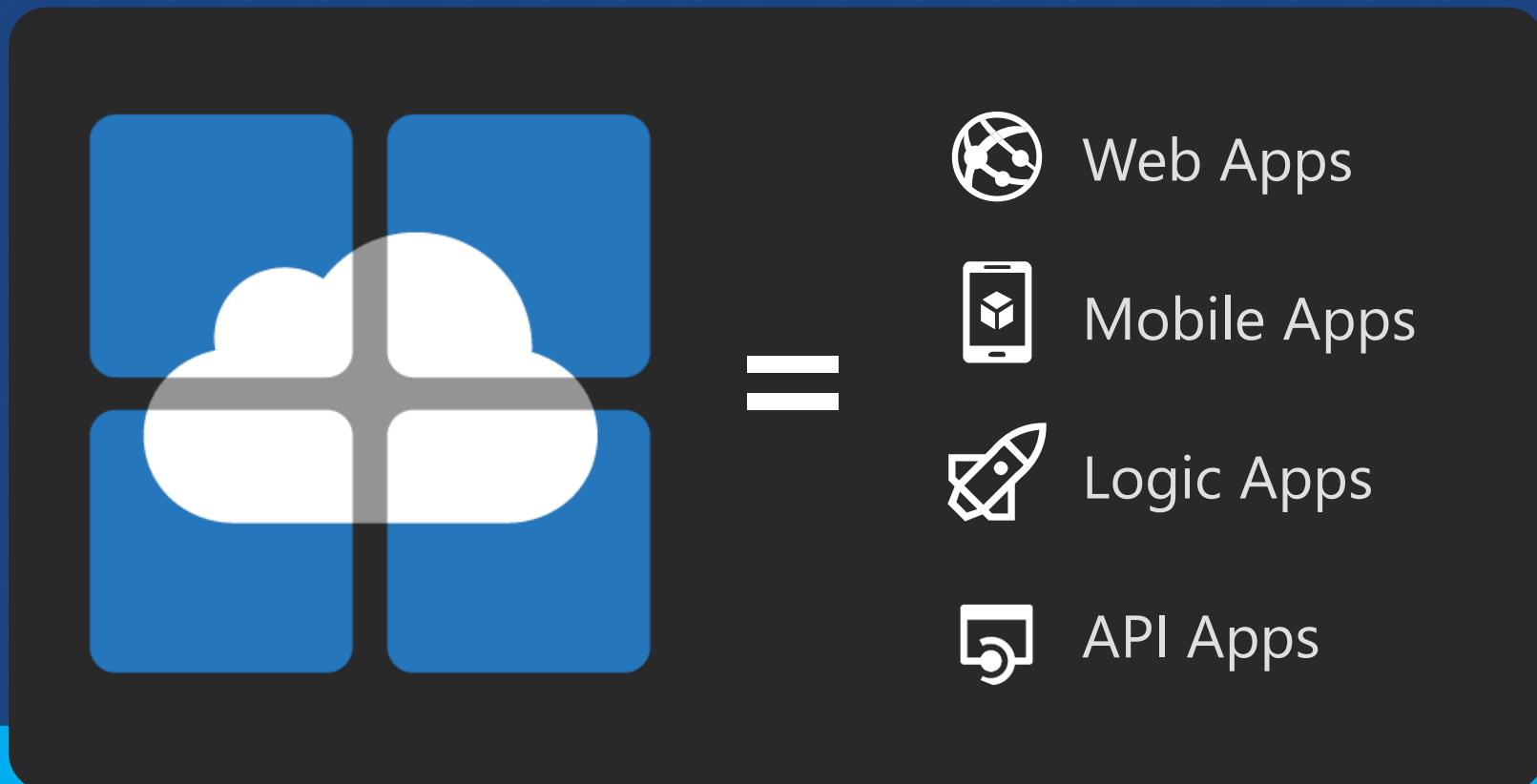
CloudBread – (2) “Cumulonimbus” project architecture design

– Scheduler / Batch 서비스



Azure App Service

Build and scale great cloud apps



App Service – one integrated offering



Web Apps

Web apps that scale
with your business



Mobile Apps

Build Mobile apps
for any device



LOGIC Apps

Automate business process
across SaaS and on-premises



API Apps

Easily build and consume
APIs in the cloud

Develop apps with...

.NET | Node.js | PHP | Python | Java



Benefits of App Services

- Automatic OS patching
- Enterprise-grade security
- High availability
 - Automated scale out/in
 - Built-in load balancing
- Supports many languages and platforms
 - .NET, Node.js, Python, Ruby and many more
- Easy continuous deployment
 - Continuous delivery from third-party source control providers
 - Built-in Git repo

Mobile Apps

New capabilities for Mobile apps:

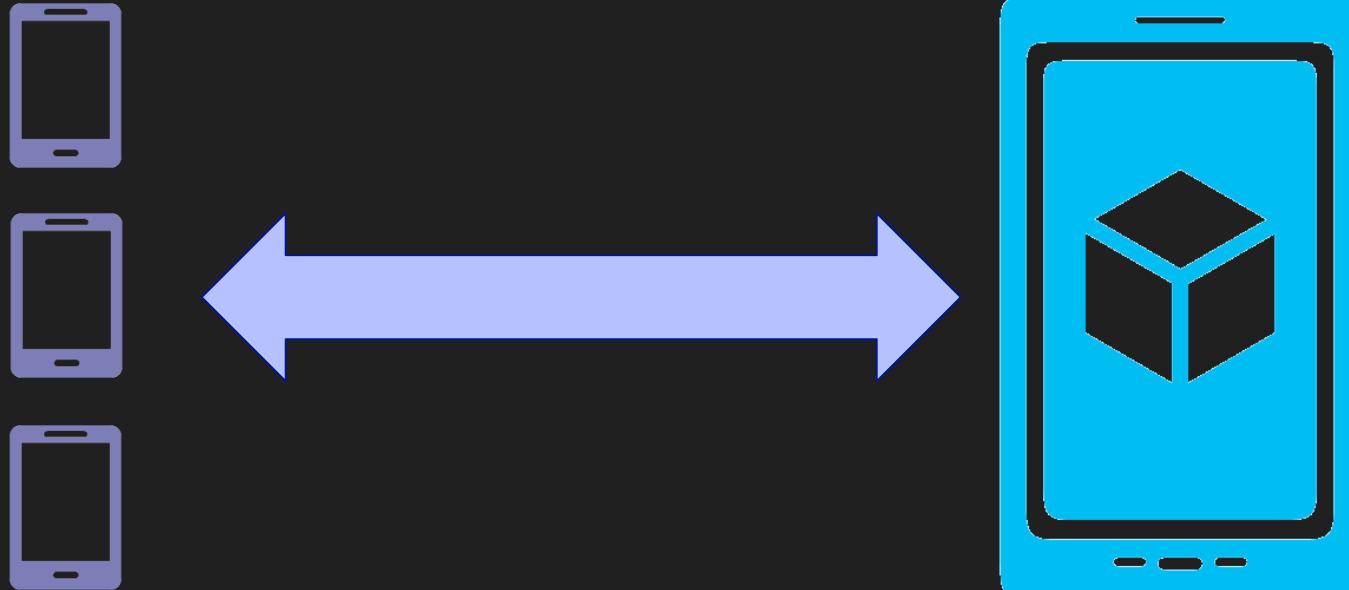


MOBILE APPS

Mobile services plus
a whole lot more

- Webjobs for long running tasks
- CI with GitHub, BitBucket, VSO
- Auto-load balance, Autoscale, Geo DR
- Virtual networking and hybrid connections
- Site slots for staged deployments

Azure Mobile Services



Storage

Authentication

Logic

Push

Scheduler

Structure d Storage

- Powered by SQL Database
- Supports rich querying capabilities
- Dynamic Schematization
- Data management in:
 - Azure Portal
 - SQL Portal (Silverlight)
 - SQL Management Studio
 - REST API
 - Azure CLI Tools
 - SQL CLI

Node.js scripts

.NET Web API
backend in
Visual Studio

Passes through to SQL by
default

MongoDB, Table Storage,
SQL out of the box

Intercept CRUD requests
to tables

Fully customizable

Server Side Table Scripts

Custom API

- Non-table based endpoints
- Accessible from
 - GET
 - POST
 - PUT
 - PATCH
 - DELETE
- For node.js logic in scripts like table endpoints
- For .NET delivered through a WebAPI
- Expose any functionality you want

Web Apps

Full capability set available including:

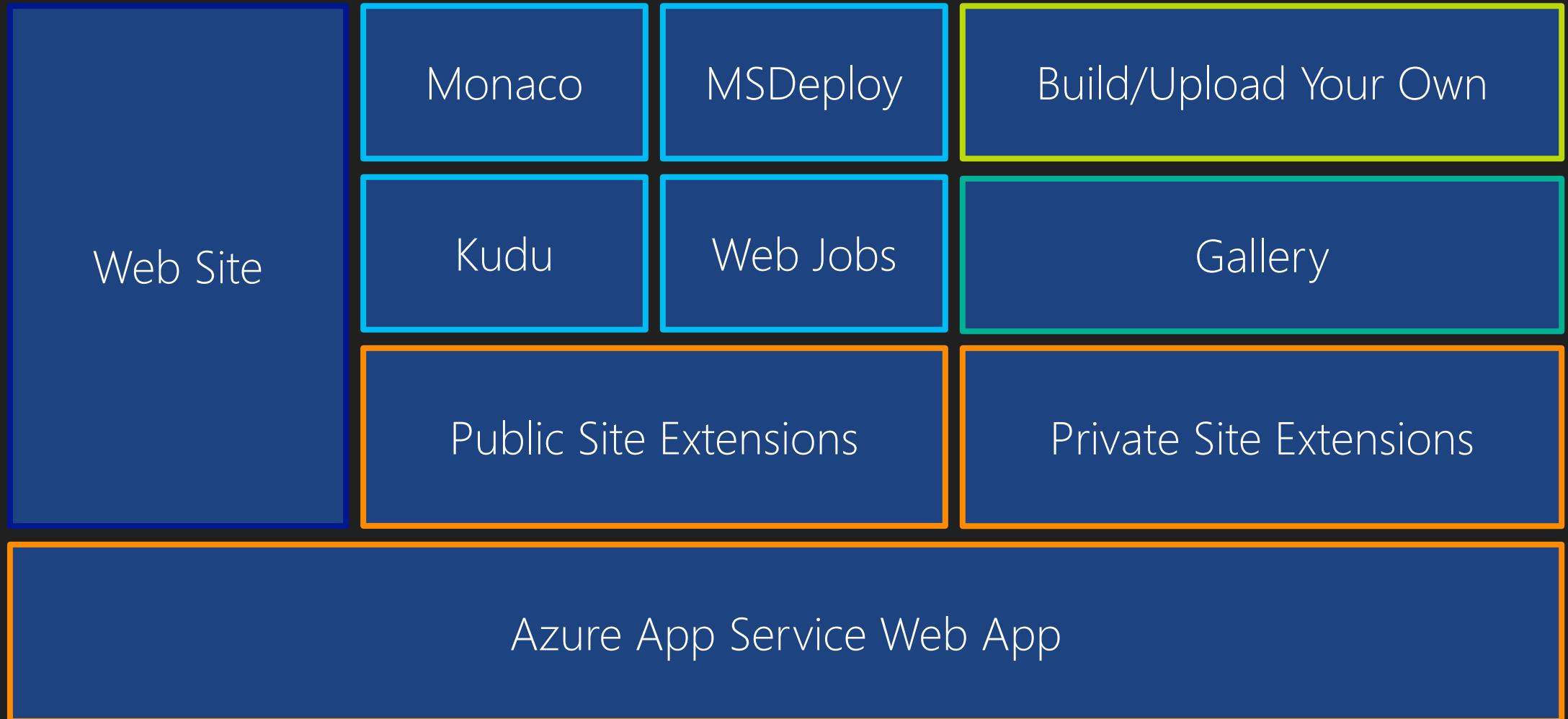


WEB APPS

Web apps run as-is
no changes required

- .NET, Node.js, Java, PHP, and Python
- WebJobs for long running tasks
- Integrated VS publish, remote debug...
- CI with GitHub, BitBucket, VSO
- Auto-load balance, Autoscale, Geo DR
- Virtual networking and hybrid connections
- Site slots for staged deployments

App Service Web App



WebJobs

Light-weight CPU Intensive Tasks



run.cmd, run.bat



run.exe



run.ps1



run.sh



run.php



run.py



run.js

Job Type: On-Demand, Scheduled, Continuous

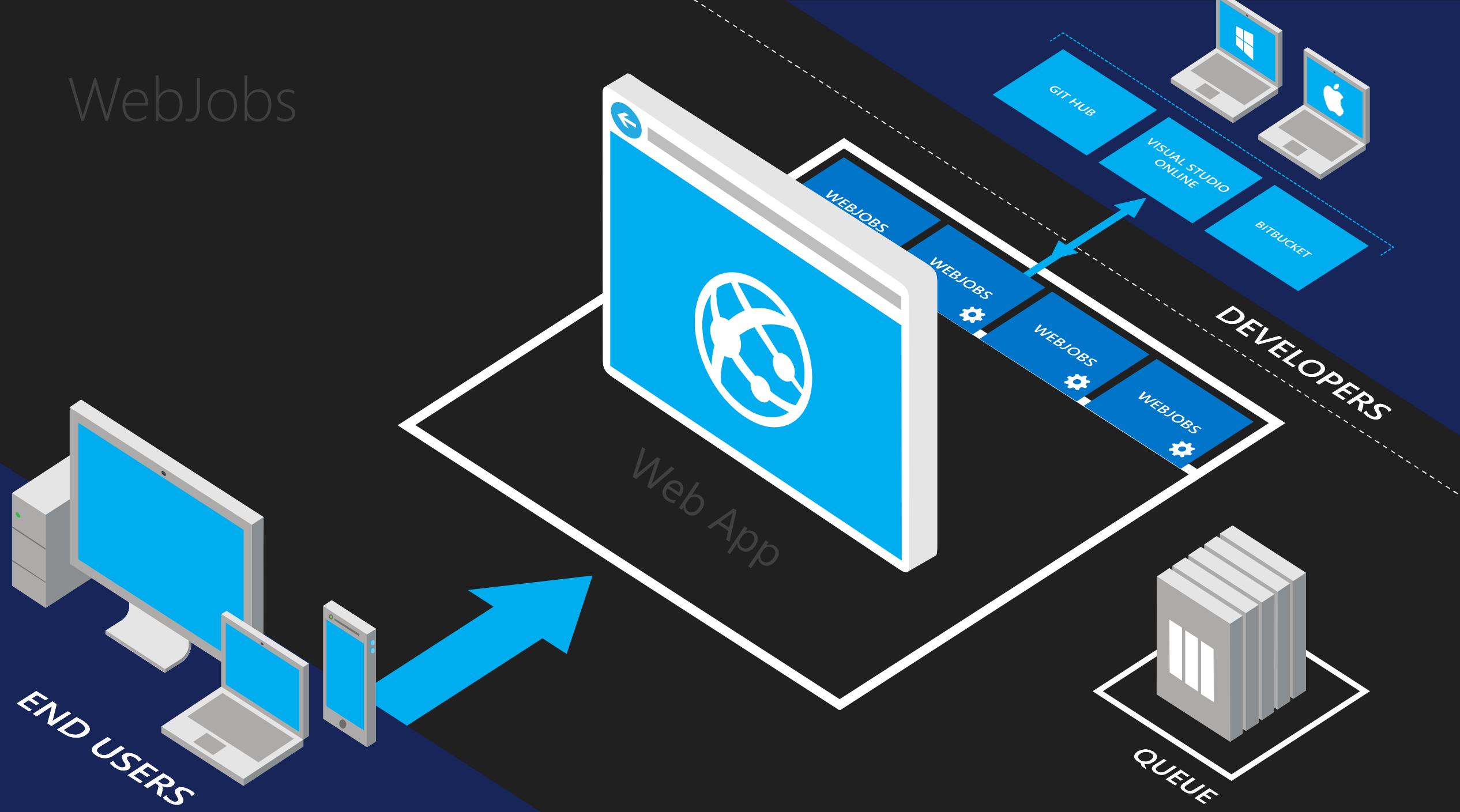
Scale: Singleton, Multi-instance

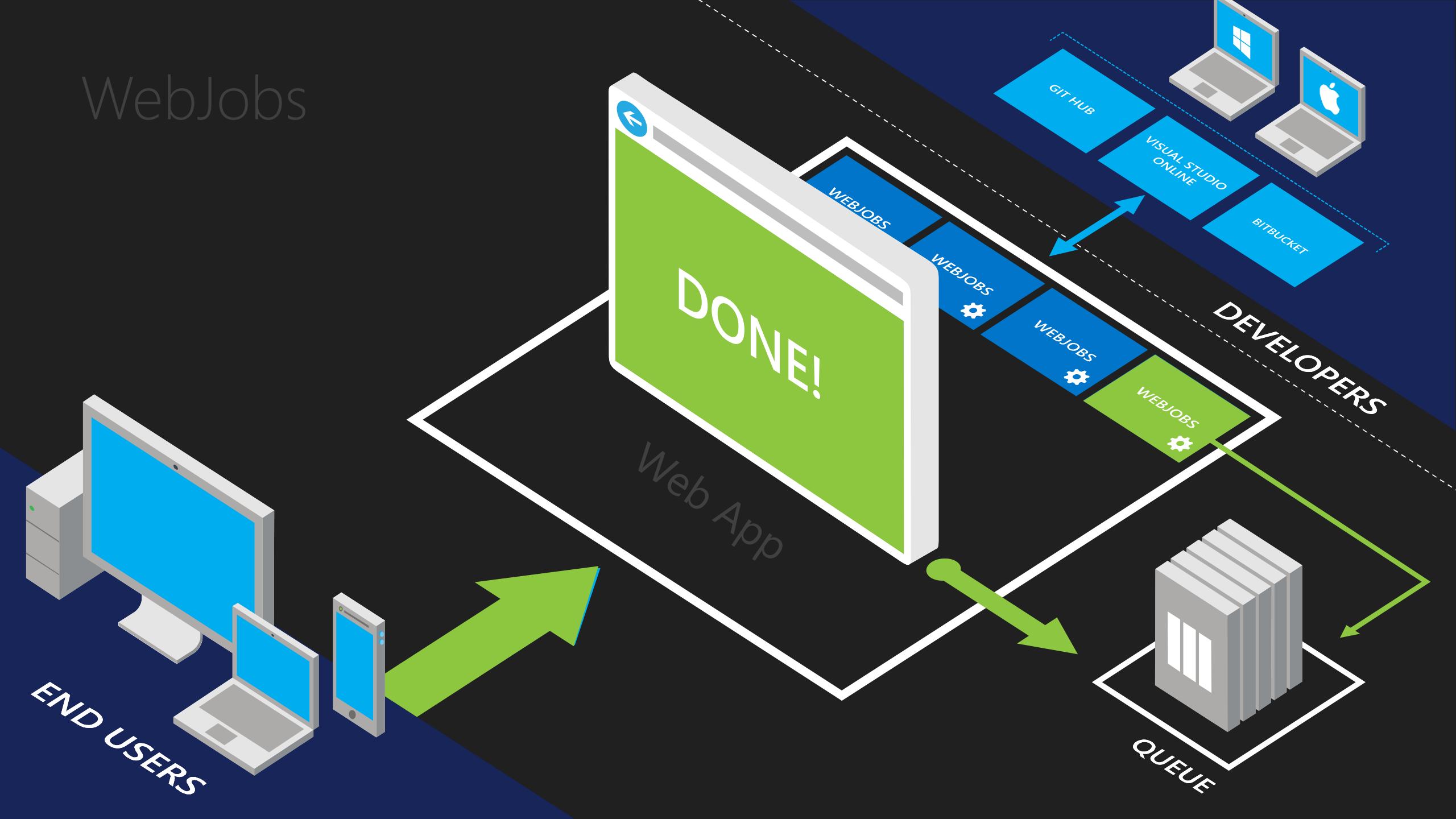
WebJobs SDK Feature: BlobTrigger, TableTrigger, QueueTrigger, ServicebusTrigger

Deployment: Portal, Visual Studio, CLI, Git



WebJobs





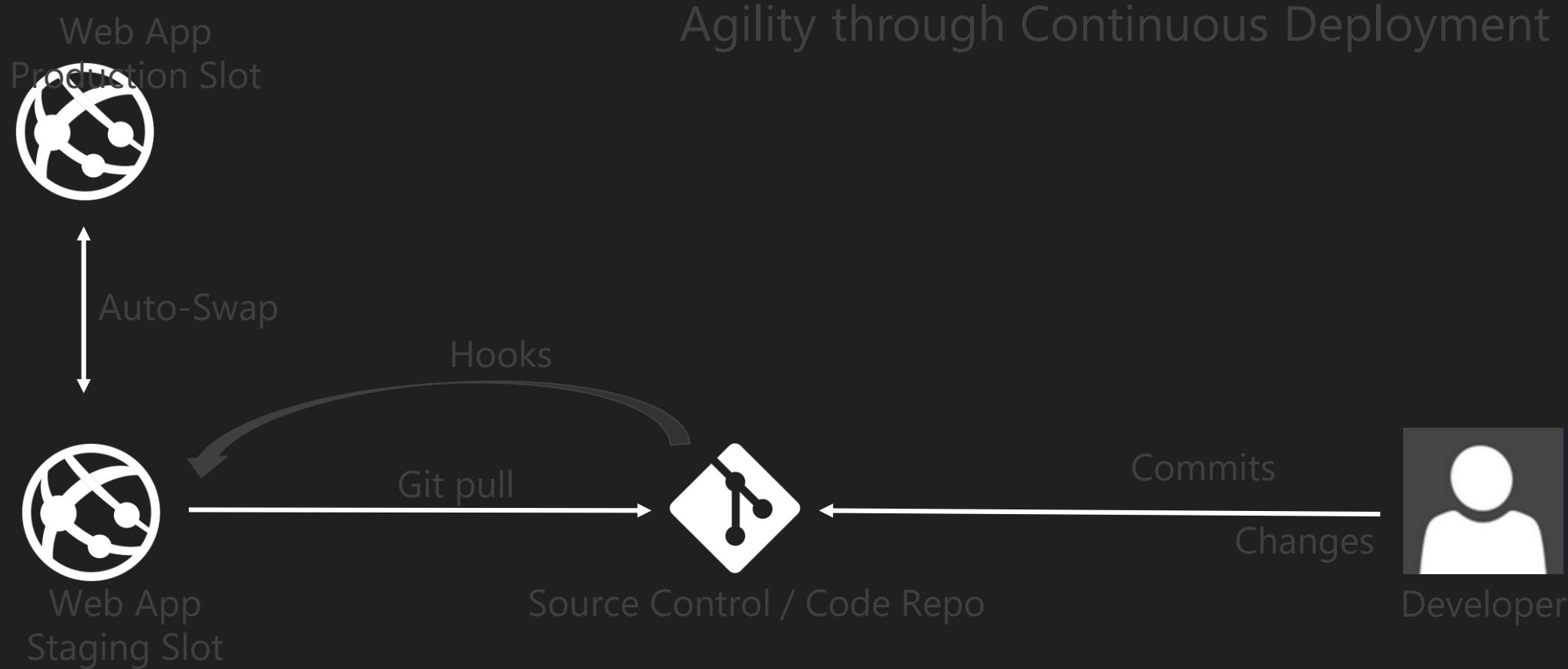
Logic Apps

Continuous Delivery with App Service

Continuous integration



Continuous Deployment for Web Apps



Source Control for Web/API/Mobile Apps



Git



Visual Studio
Online



CodePlex



GitHub



BitBucket



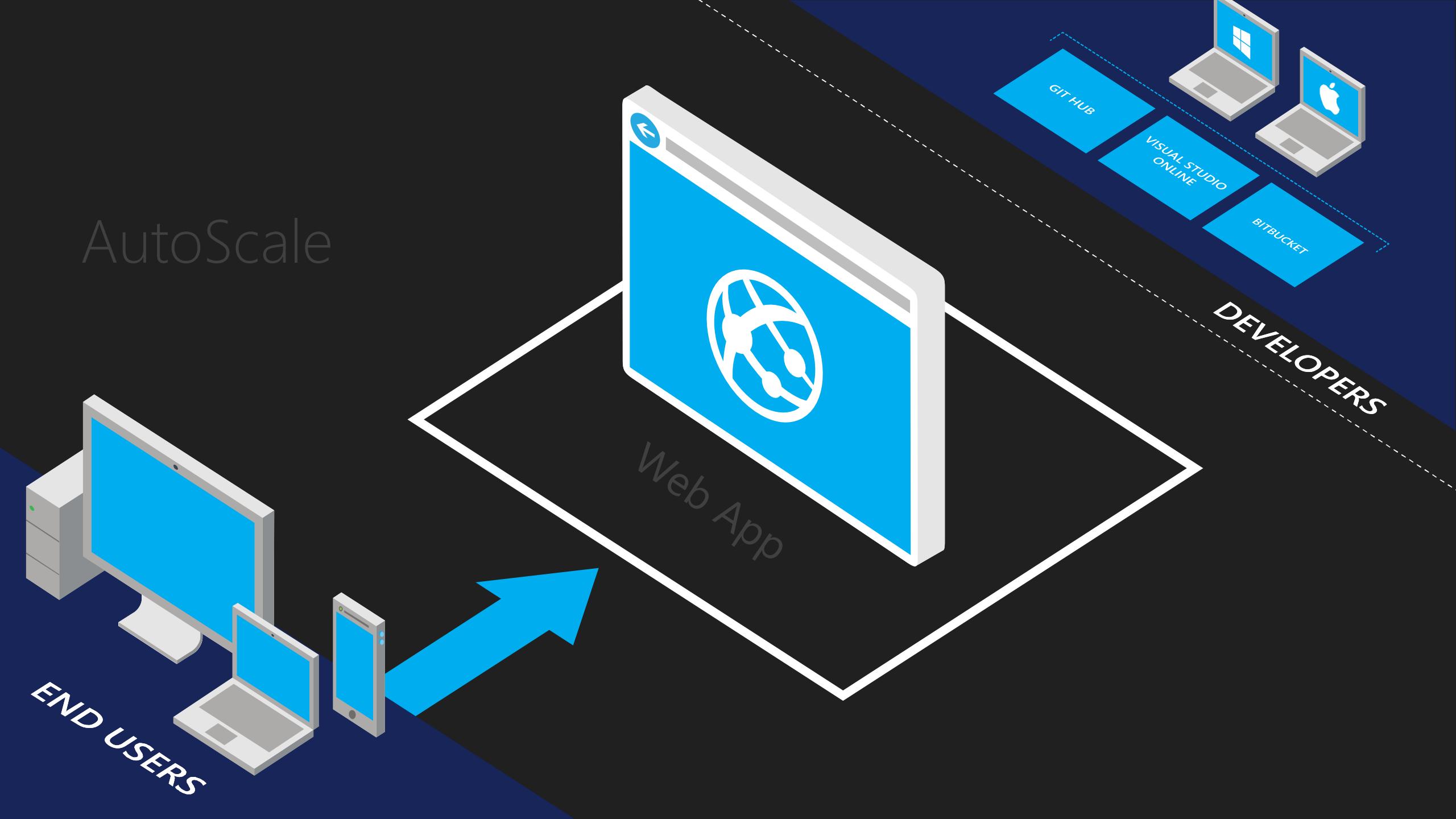
DropBox

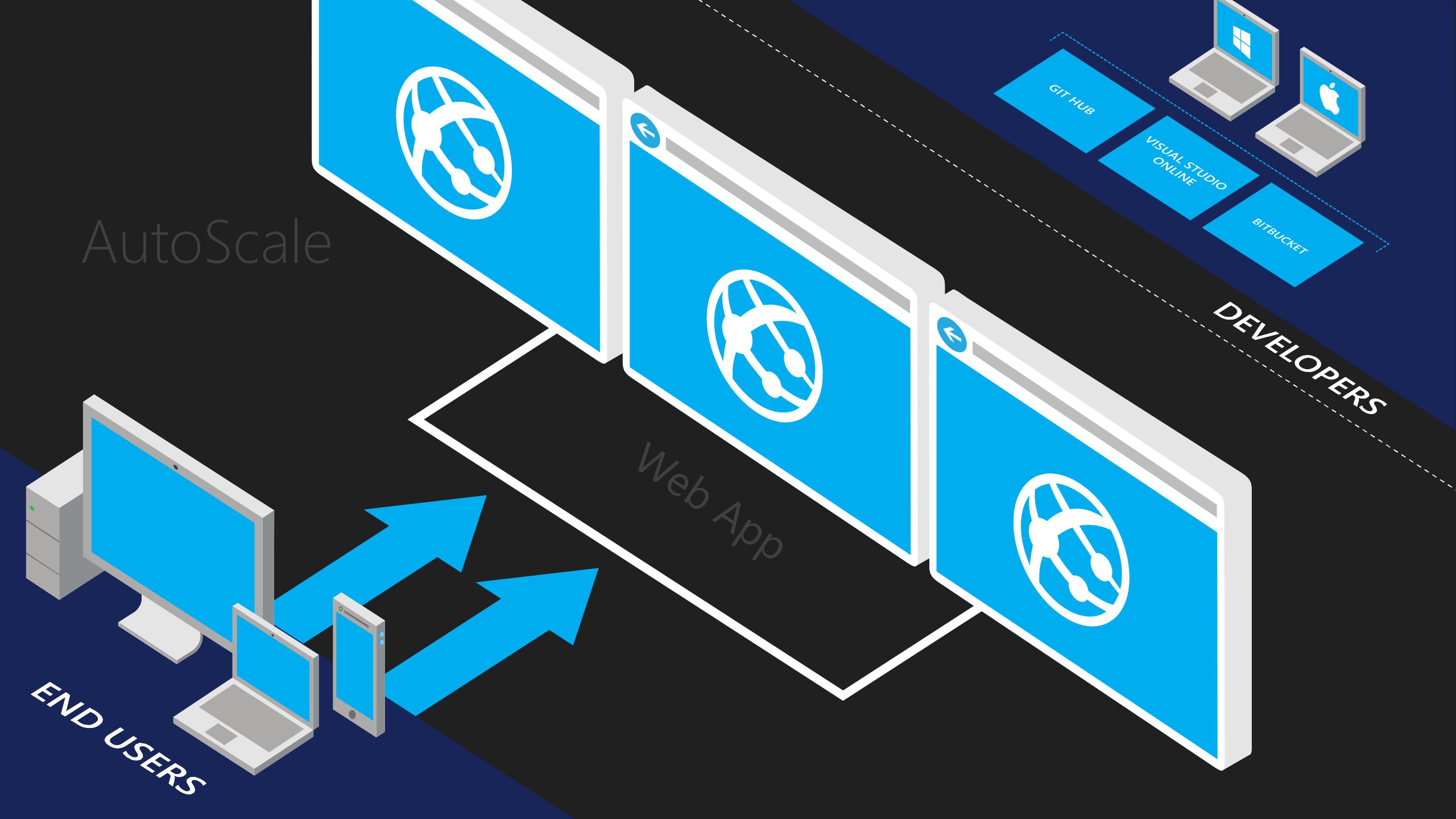


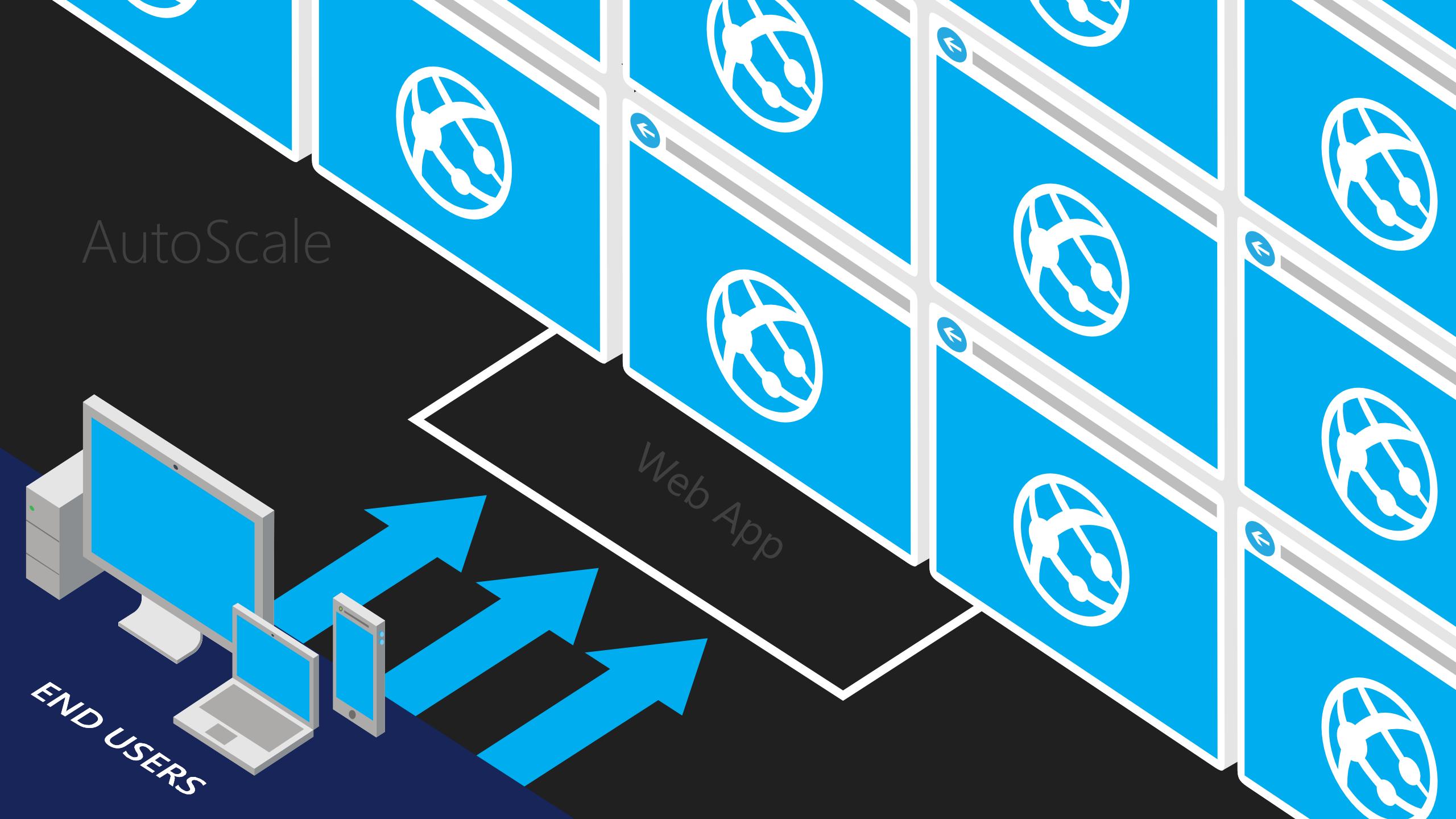
FTP

Choose your own adventure!

AutoScale







App Service Environment

App Service Environment (ASE)

- New Premium Tier Feature
- Dedicated compute resources and network resources
- Increased Scaling Options
- Directly created in a Virtual Network
- Support all Web App features and capabilities
- Support Web, Mobile and API Apps
- Global Scale

Conceptual Model



App Service
App



App Service
App



App Service
App

App Service Plan

SKU: Premium (# of sites, storage, slots...)

Compute Resource: P4 (8 cores, 14 GB memory)

Scale: 8



App Service
App



App Service
App



App Service
App

App Service Plan

SKU: # of sites, storage, slots...

Compute Resource: cores, memory

Scale: number of instances

Note: For App Service Environment the SKU will always be Premium

App Service Environment

