.NET MAUI 앱 개발 실전 가이드

흑우마스터
jojangwon@kakao.com

Agenda

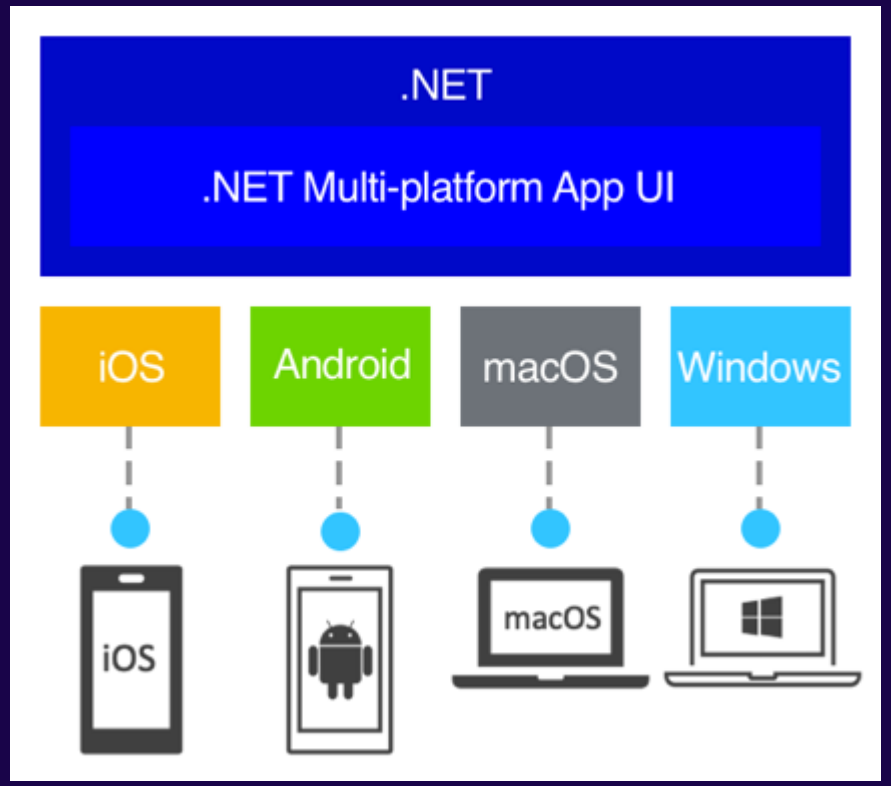
- .NET MAUI?
- MVVM & MVU
- Style
- App Update
- Push Notification
- Authentication
- Cache & Storage
- Monitoring
- Feature Flags
- DeepLink
- Deploy

00

.NET MAUI?

MAUI

.NET MAUI?



Initial Release – May, 2022

Open Source framework

Supported by - Microsoft

.NET MAUI?

Winforms
ASP.NET

WPF
SILVERLIGHT
UWP

XAML

```xml
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="MauiApp21.MainPage">

    <ScrollView>
        <VerticalStackLayout
            Spacing="25"
            Padding="30,0"
            VerticalOptions="Center">

            <Image
                Source="dotnet_bot.png"
                SemanticProperties.Description="Cute dot net bot waving hi to you!"
                HeightRequest="200"
                HorizontalOptions="Center" />

            <Label
                Text="Hello, World!"
                SemanticProperties.HeadingLevel="Level1"
                FontSize="32"
                HorizontalOptions="Center" />

            <Label
                Text="Welcome to .NET Multi-platform App UI"
                SemanticProperties.HeadingLevel="Level2"
                SemanticProperties.Description="Welcome to dot net Multi platform App U I"
                FontSize="18"
                HorizontalOptions="Center" />

            <Button
                x:Name="CounterBtn"
                Text="Click me"
                SemanticProperties.Hint="Counts the number of times you click"
                Clicked="OnCounterClicked"
                HorizontalOptions="Center" />

        </VerticalStackLayout>
    </ScrollView>

</ContentPage>
```

XAML : Extensible Application Markup Language

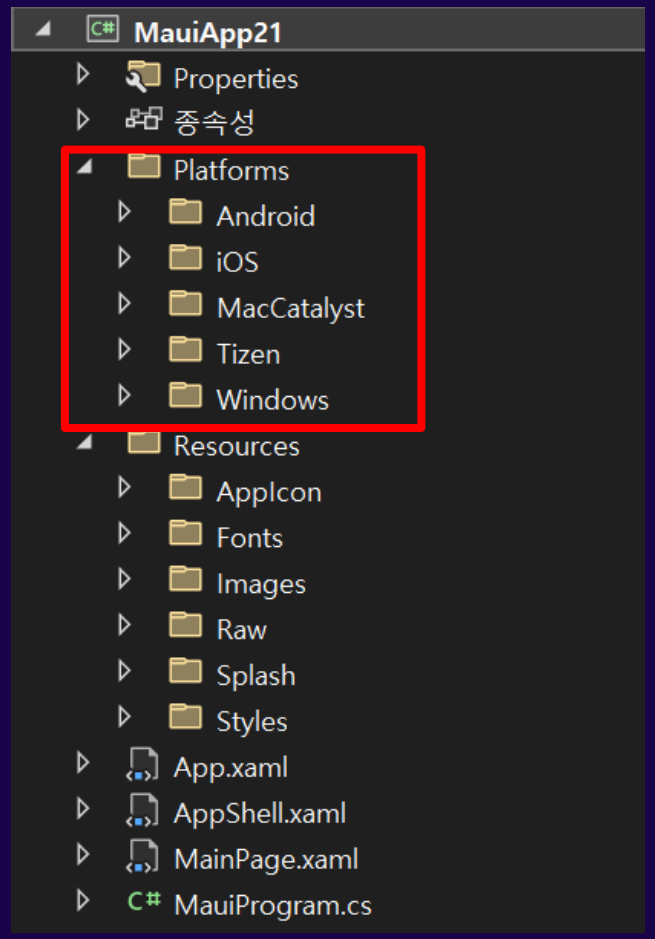
코드 베이스가 아닌 태그 형태를 사용하여 사용자 인터페이스 정의

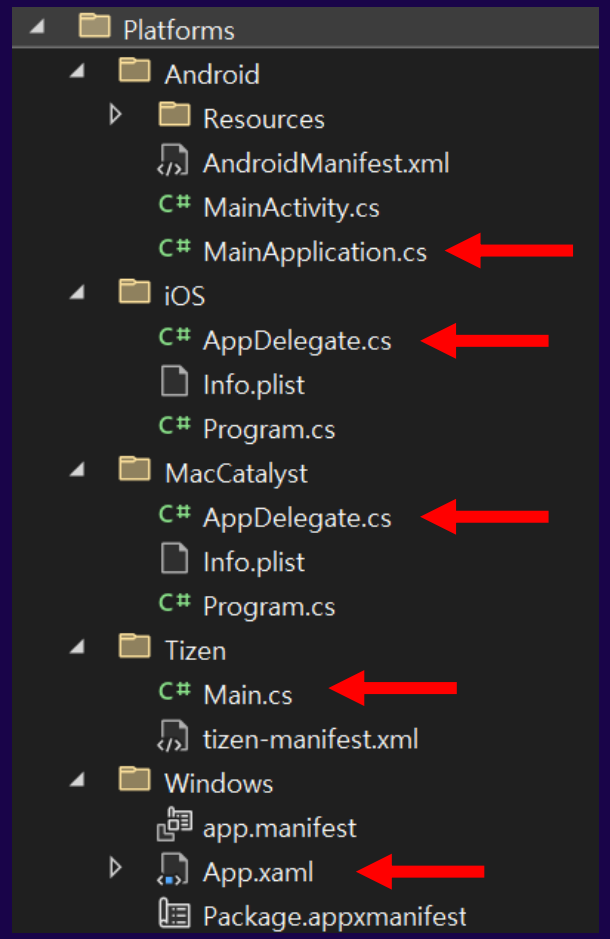
XAML 기반의 데이터 바인딩을 통한 MVVM 구현에도 적합

일반적인 조건부 처리가 아닌 사용자 인터페이스에 목적을 둠

별도의 디자이너 없이 직접 사용해야 하여 학습 난이도가 있는 편

기존 XAML을 사용했던 WPF, UWP, Silverlight 등 개발자들에게 유리

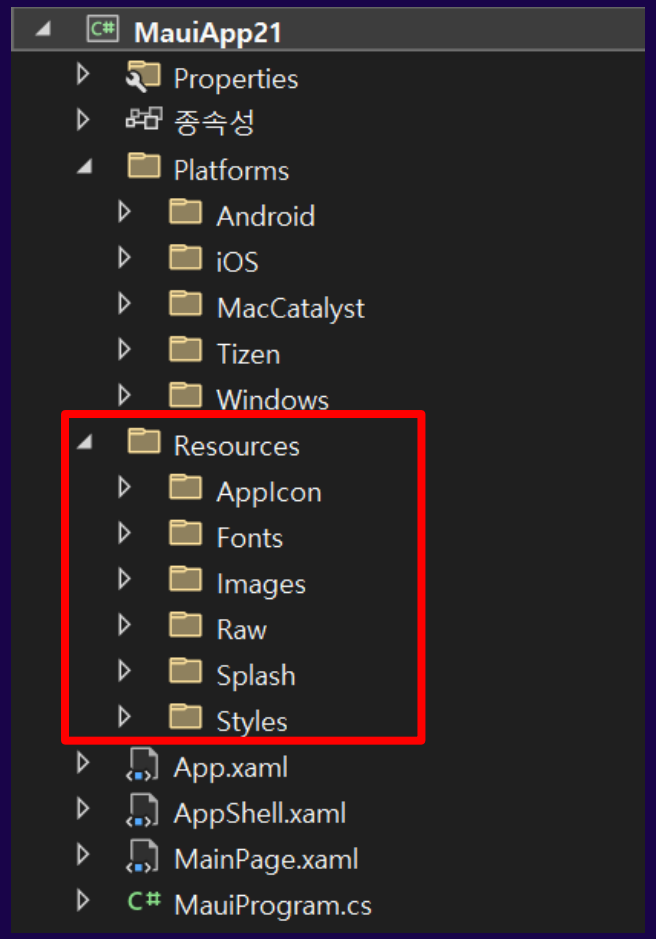


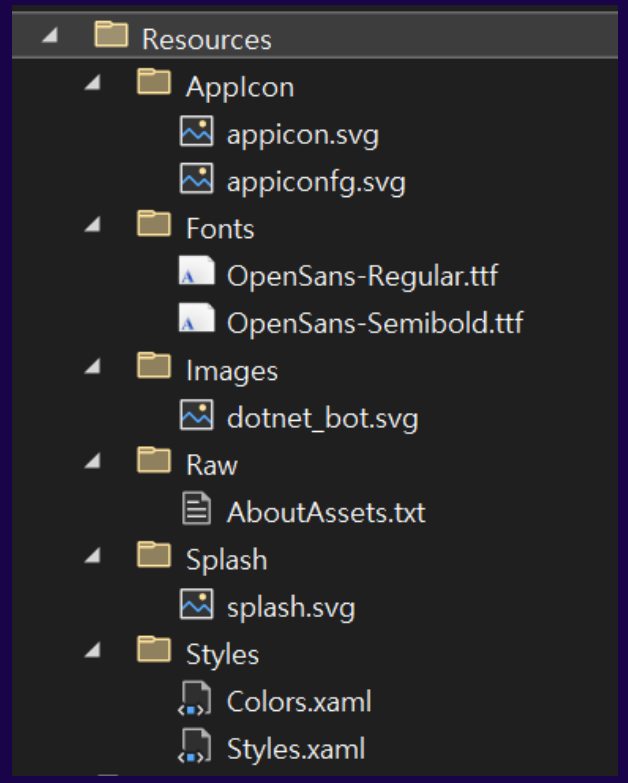


Platform

각 대상 플랫폼에 대한 별도의 폴더

플랫폼 별 앱의 진입점, 관련 종속 세팅 등 개별 설정 가능

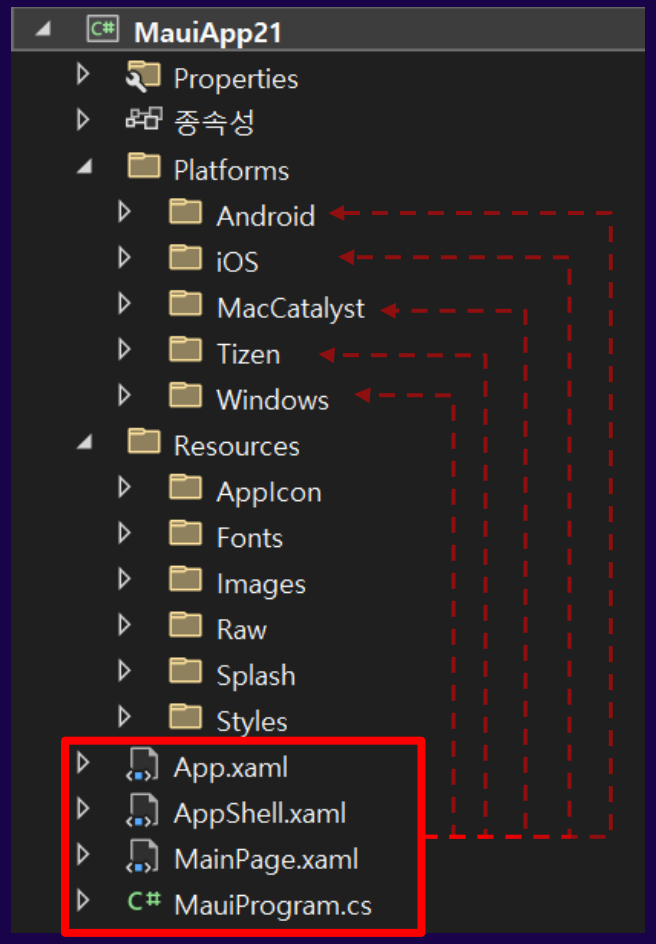




Resources

글꼴, 이미지 등 공통 프로젝트 리소스 폴더

플랫폼별로 배치해야 되는 경우도 있음



ROOT

HostBuilder 패턴으로 각 플랫폼 별 진입점에

CreateMauiApp()을 호출하며 커스텀 폰트, 핸들러, 렌더러 등 지정

다른 선택지

.NET MAUI?

기술 부재의 늪

.NET MAUI?

dotnet/maui

#19200 [regression/8.0.3

Splash screen no longer working in .Net 8 Maui

7 comments

polygonsheep opened on December 4, 2023

낮은 인지도

.NET MAUI?

채용정보 총 3건

일반 채용정보 | 파견/대행 | 헤드헌팅

입사지원 만 보기 | 조회한 공고 제외 | 취업축하금만 보기

기업형태 전체 | 관련도순 | 40개씩

IT개발·데이터 지원 급상승

[서울 문래동] 최고의 .NET/C# 개발자 채용 ~ 01/18(토) 입사지원 — 이엠시티 주식회사 기업정보

서울 영등포구 경력5년↑ 학력무관 정규직 6,000 만원

ASP.NET, .NET, C#, C++, 솔루션 외 | 수정일 24/12/19

연봉상승률 10% 이상

난소프트와 함께 성장할 개발자를 모집합니다 ~ 02/02(일) 입사지원 — (주)난소프트 기업정보

서울 금천구 신입·경력 고졸↑ 정규직

.NET, C#, C++, 유지보수, 솔루션 외 | 수정일 25/01/03

스타트업

- 웹 프론트엔드(React) / Windows 어플리케이션 개발자 모집 ~ 01/19(일) 입사지원 — (주)아고스 기업정보

서울 금천구 경력3년↑ 초대졸↑ 정규직 5,000 만원

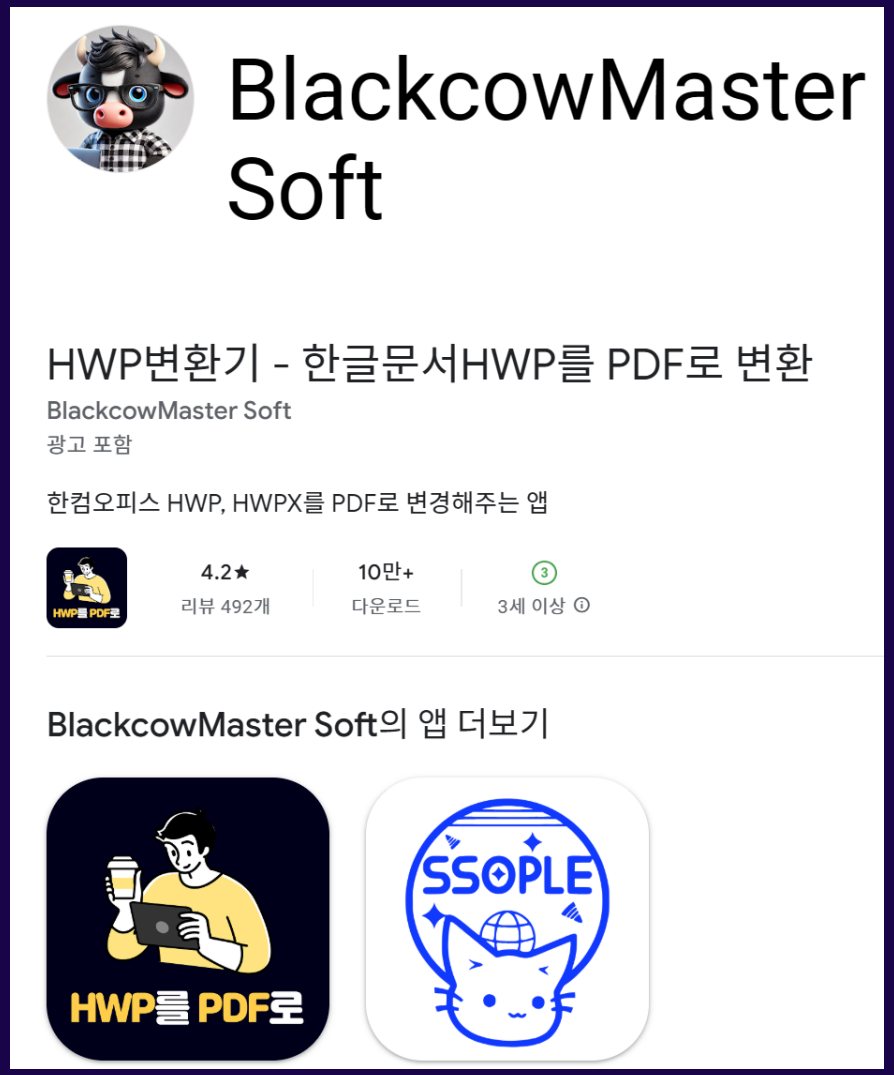
.NET, C#, C++, 솔루션, 웹개발 외 | 수정일 24/12/20

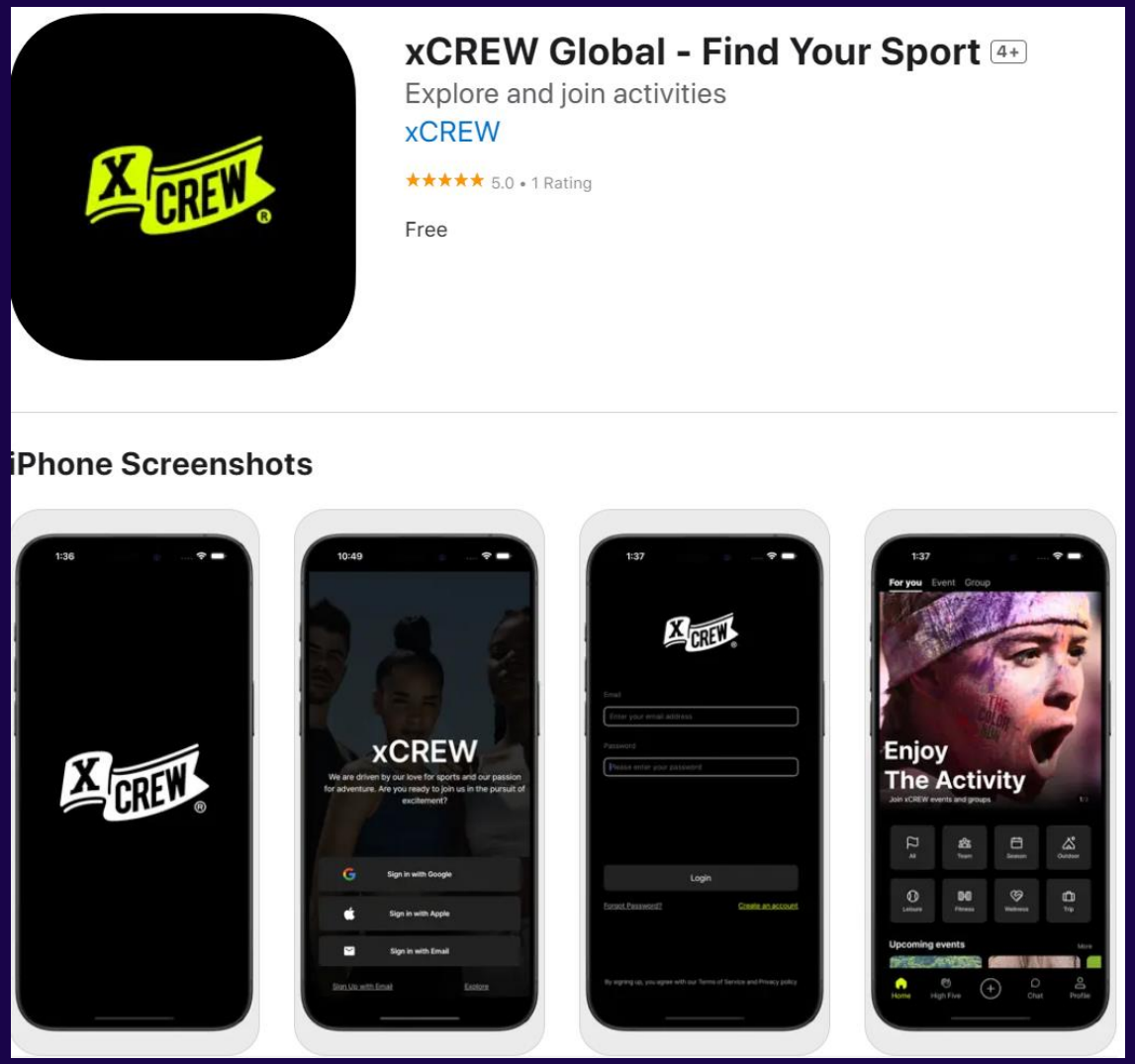
1

17:20 – 18:00	201호	조장원 엑스크루 기술총괄	.NET MAUI와 Supabase로 시작하는 크로스플랫폼 개발
	204호	이승민 에어프레미아랩스 / Android GDE	글로벌 기업은 홈페이지에서 영어 번역을 어떻게 제공할까? 항공사의 다국어 관리 방법 알아보기
	206호	윤창현 MIRI D.I.H / GDG Incheon	당신이 React Native를 선택할때 깊게 고민해봐야 할 이유
	208호 * 멘토링	박준호 디엔데 CPO & CTO	창업 / 사이드프로젝트
	302호	한성민 AL/ML , Cloud , Go GDE	Gemma for your device 2B 모델의 온디바이스 최적화와 Token-based CoT 구현
	305호	남반석 Coupang	Flutter 테스트 코드의 기본 시나리오별로 배우는 실전 가이드와 노하우
	307호	한상준 GDG Songdo	GFS 35만달러 지원과 함께하는 스타트업 성장기 Google Cloud는 덤, 체계를 잡아가는 스타트업 이야기
	308호	임성호 무신사 / GDG Incheon	개발자의 채용과정과 마음가짐

그럼에도 쓸만합니다

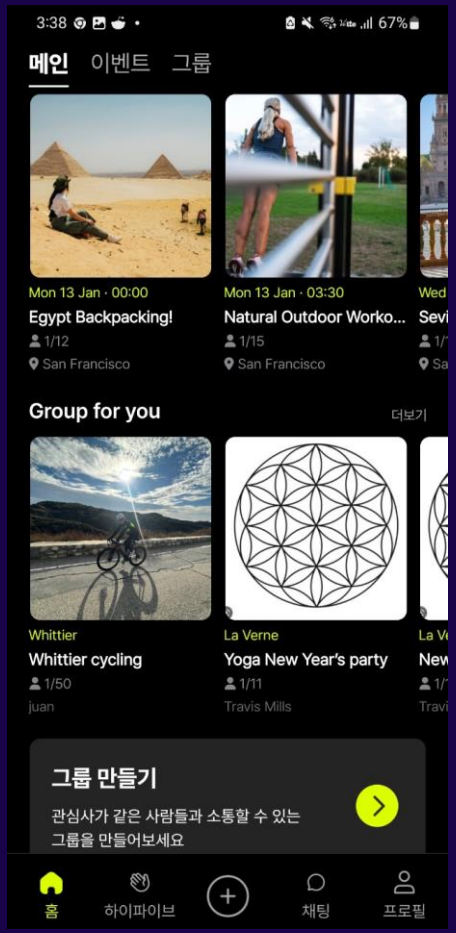
.NET MAUI?

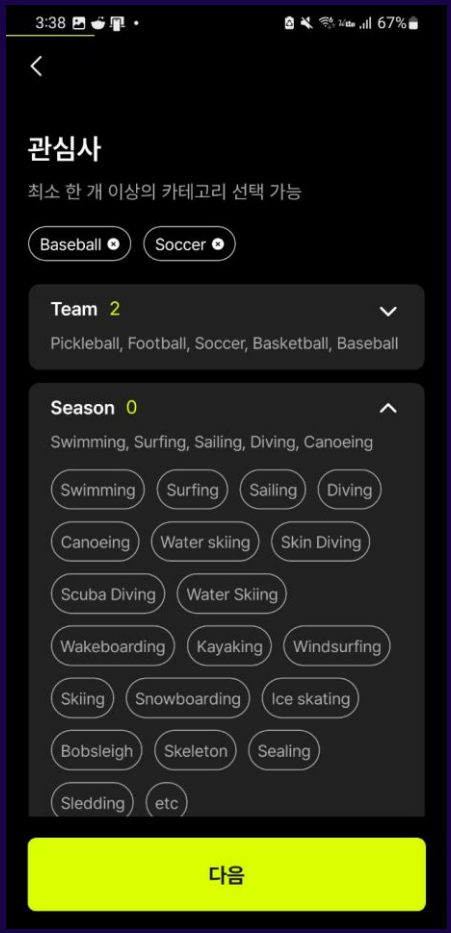


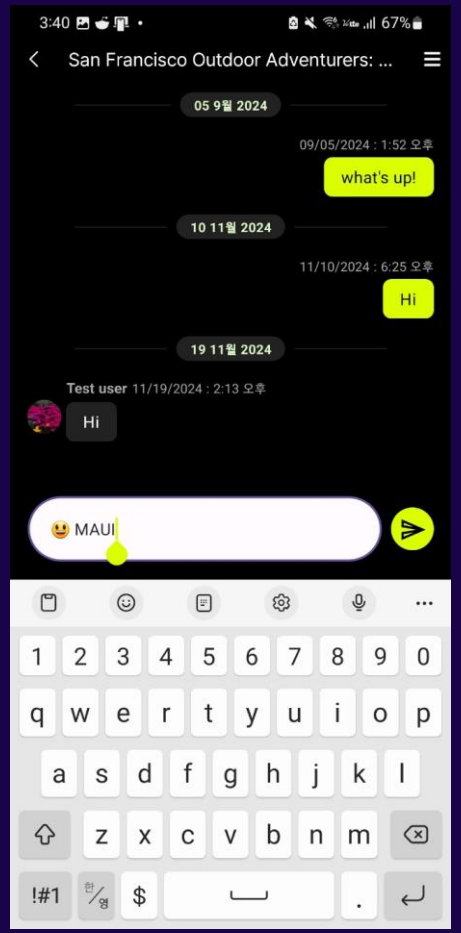


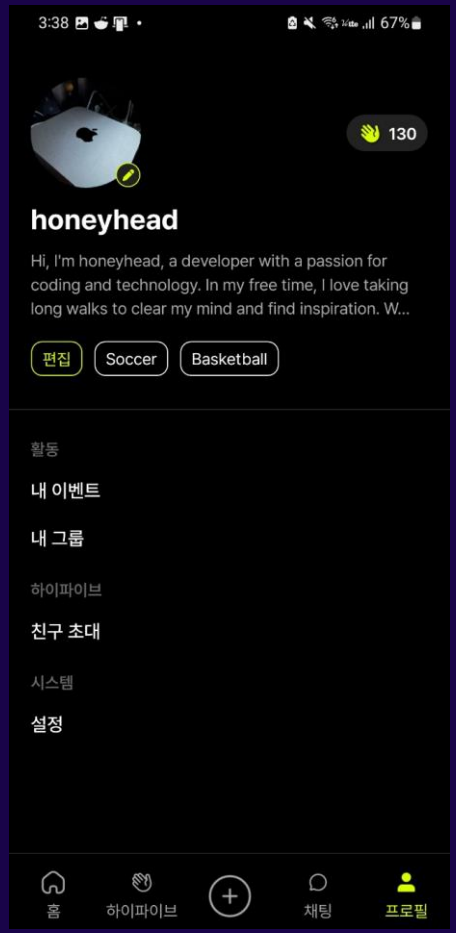
세상에 나쁜 기술은 없습니다

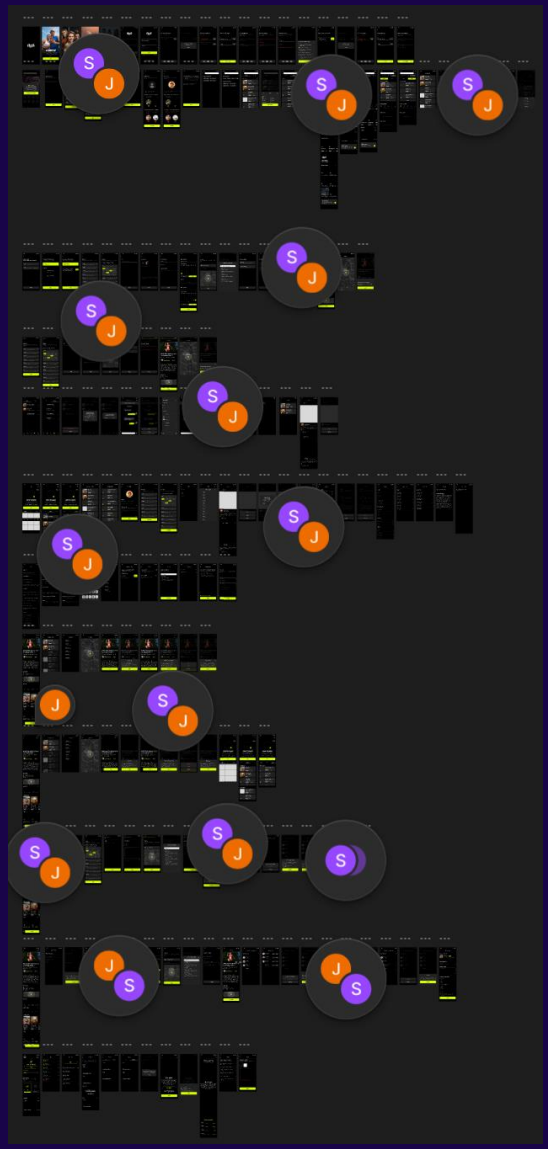
.NET MAUI?





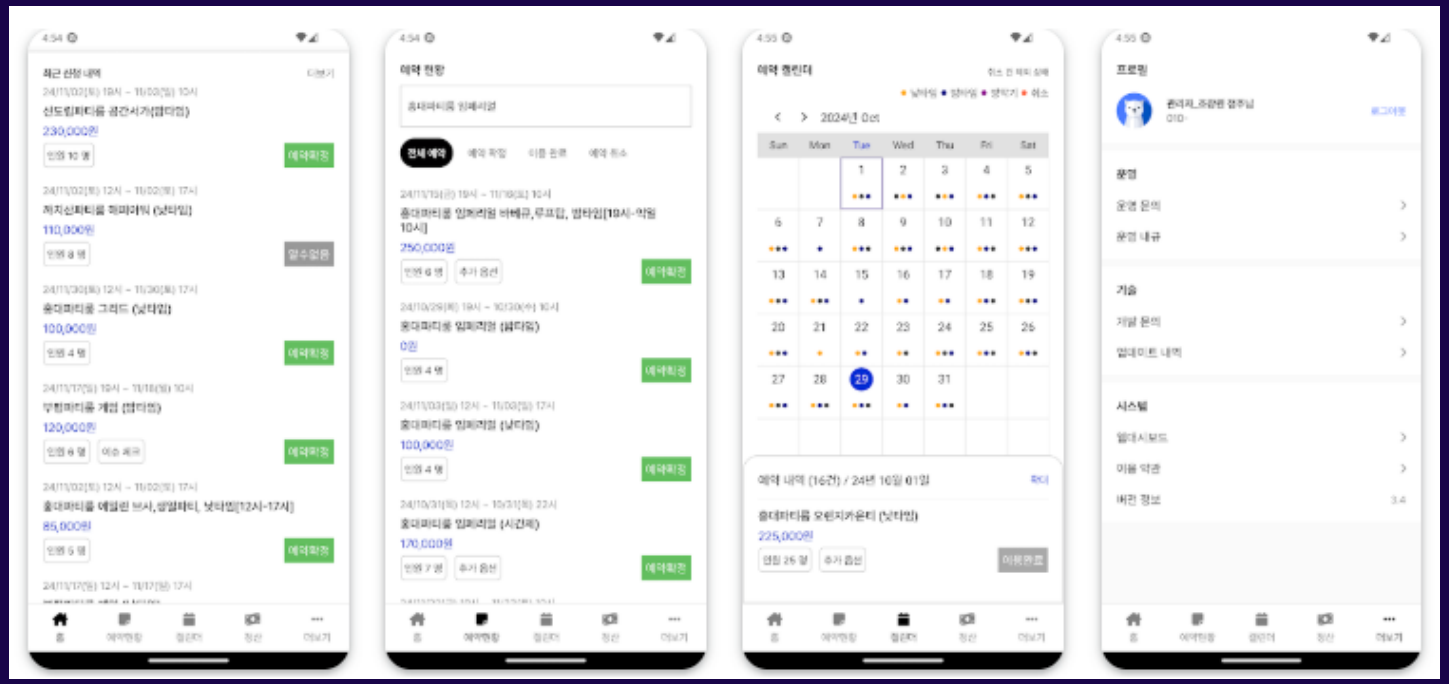


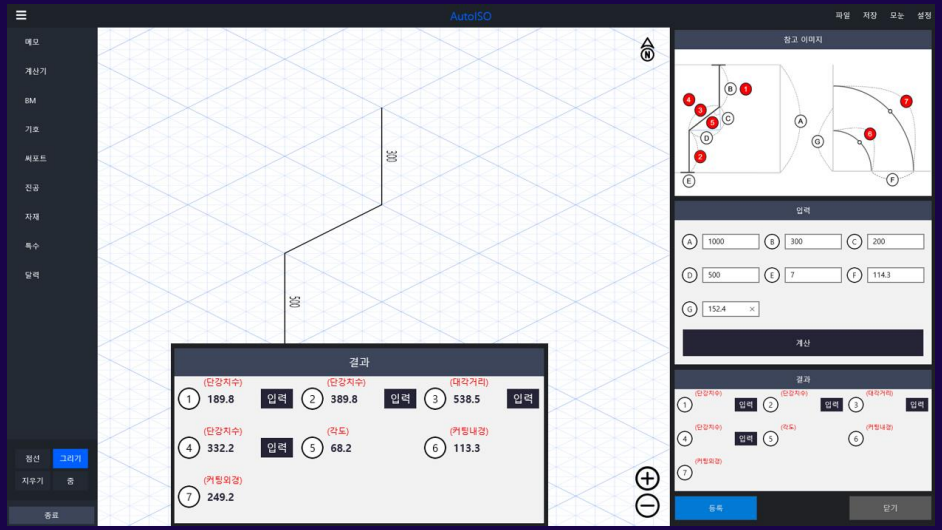




세상에 나쁜 기술은 없습니다

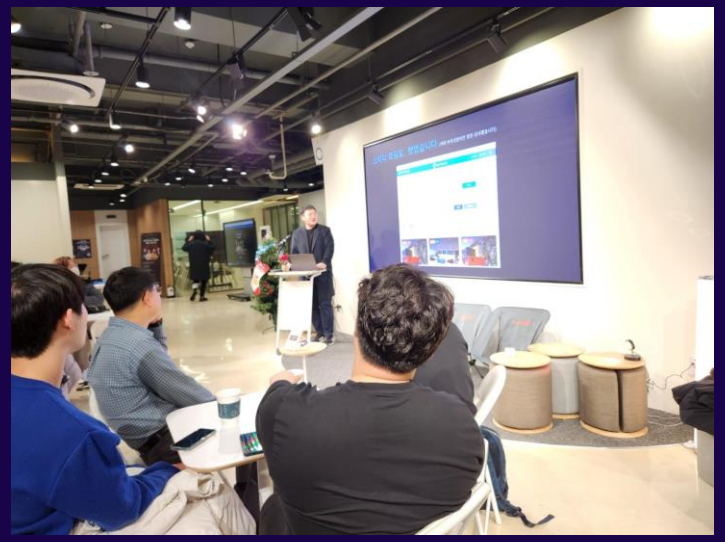
.NET MAUI?





그래서 MAUI를 홍보합니다

.NET MAUI?



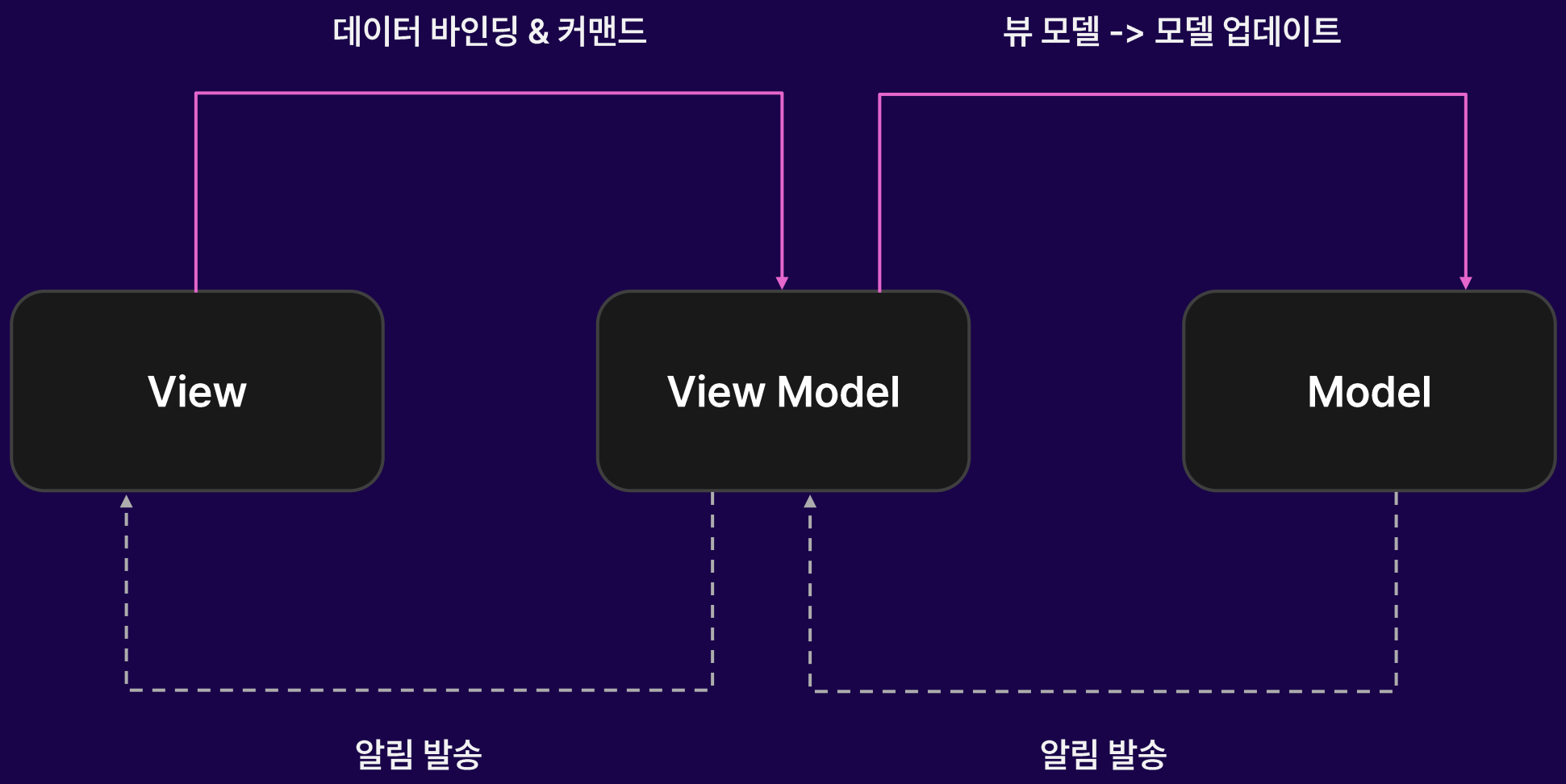
그런 이유로 오늘도 약 팔러 나왔습니다!

대신 기초보단 실전 가이드!

01

MVVM & MVU

MVVM Pattern



C# 복사

```
using CommunityToolkit.Mvvm;
```

VB 복사

```
Imports CommunityToolkit.Mvvm
```

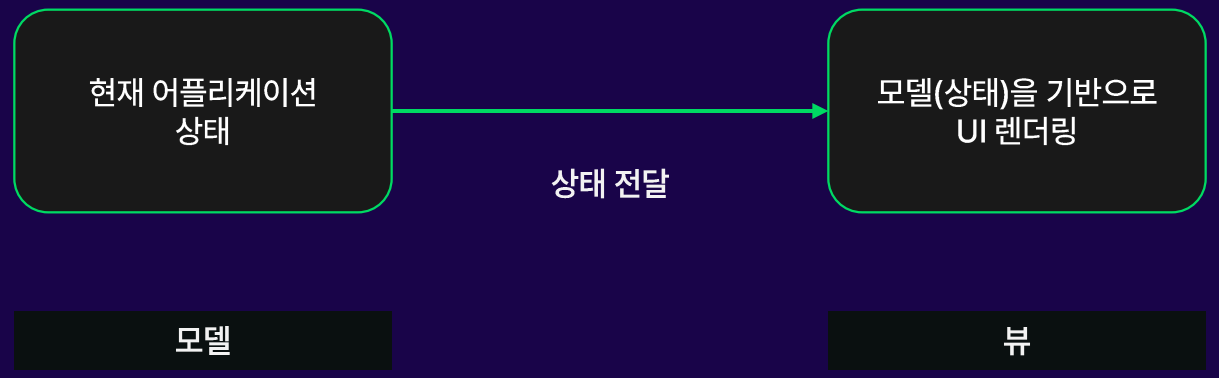
- **CommunityToolkit.Mvvm.ComponentModel**
  - ObservableObject
  - ObservableRecipient
  - ObservableValidator
- **CommunityToolkit.Mvvm.DependencyInjection**
  - Ioc
- **CommunityToolkit.Mvvm.Input**
  - RelayCommand
  - RelayCommand<T>
  - AsyncRelayCommand
  - AsyncRelayCommand<T>
  - IRelayCommand
  - IRelayCommand<T>
  - IAsyncRelayCommand
  - IAsyncRelayCommand<T>
- **CommunityToolkit.Mvvm.Messaging**
  - IMessenger
  - WeakReferenceMessenger
  - StrongReferenceMessenger
  - IRecipient<TMessage>
  - MessageHandler<TRecipient, TMessage>
- **CommunityToolkit.Mvvm.Messaging.Messages**
  - PropertyChangedMessage<T>
  - RequestMessage<T>
  - AsyncRequestMessage<T>
  - CollectionRequestMessage<T>
  - AsyncCollectionRequestMessage<T>
  - ValueChangedMessage<T>

MVU Pattern

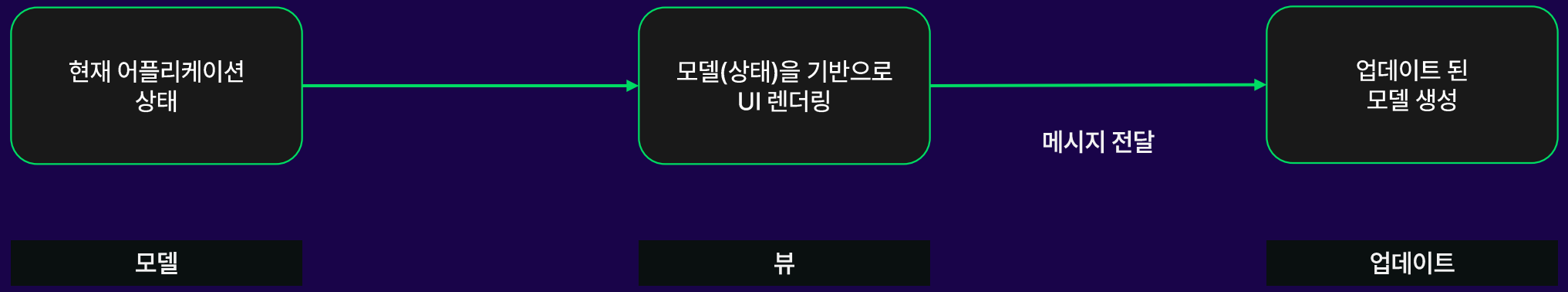
현재 어플리케이션
상태

모델

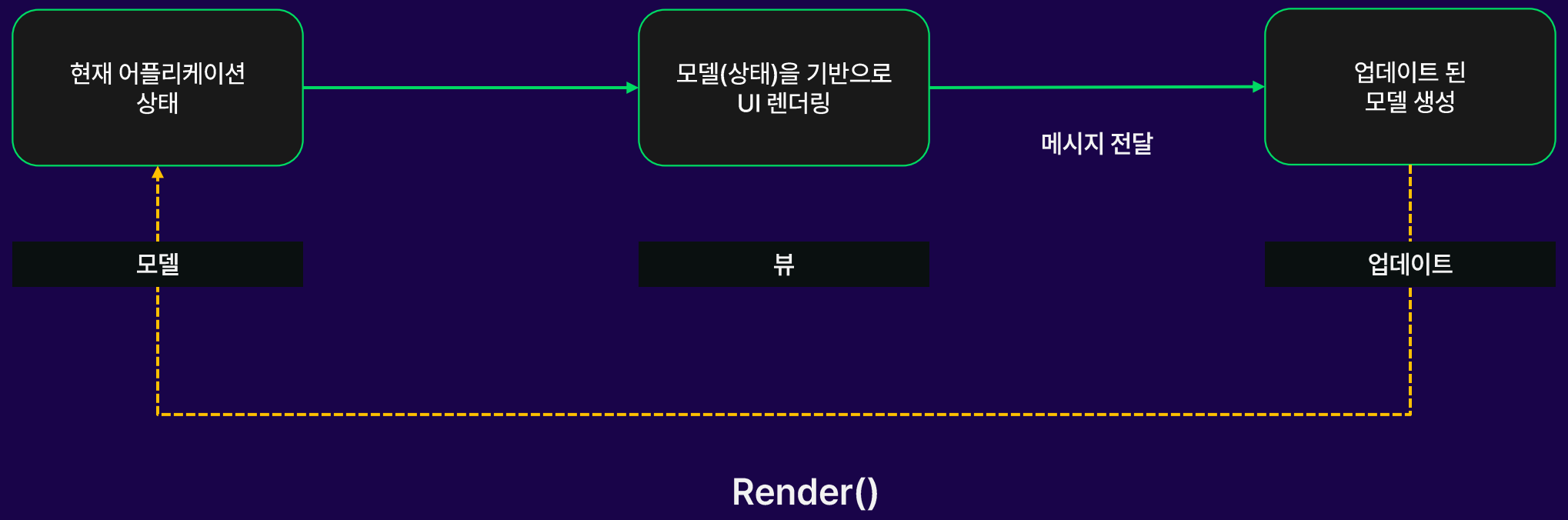
MVU Pattern



MVU Pattern



MVU Pattern


현재 어플리케이션
상태
모델(상태)을 기반으로
UI 렌더링
메시지 전달
업데이트 된
모델 생성
모델
뷰
업데이트
Render()

MVVM VS MVU


```
<Label
    Text="Hello, World!"
    Style="{StaticResource Headline}"
    SemanticProperties.HeadingLevel="Level1"  />
```

```
private void OnCounterClicked(object sender, EventArgs e)
{
    count++;

    if (count == 1)
        CounterBtn.Text = $"Clicked {count} time";
    else
        CounterBtn.Text = $"Clicked {count} times";

    SemanticScreenReader.Announce(CounterBtn.Text);
}
```

MVVM

```
class CounterPageState
{
    public int Counter { get; set; }
}

class CounterPage : Component<CounterPageState>
{
    public override VisualNode Render()
        => ContentPage("Counter Sample",
            VStack(
                Label($"Counter: {State.Counter}"),

                Button("Click To Increment", () =>
                    SetState(s => s.Counter++))
            )
            .Spacing(10)
            .Center()
        );

}
```

MVU

MauiReactor

Component-based UI Library built on top of .NET MAUI

build passing | nuget v2.0.45

MauiReactor is .NET library written on top of .NET MAUI that allows you to write applications in pure C# using an MVU approach.

This is the classic Counter app in MauiReactor:

```
class CounterPageState
{
    public int Counter { get; set; }
}

class CounterPage : Component<CounterPageState>
{
    public override VisualNode Render()
        => ContentPage("Counter Sample",
            VStack(
                Label($"Counter: {State.Counter}"),

                Button("Click To Increment", () =>
                    SetState(s => s.Counter++))
            )
            .Spacing(10)
            .Center()
        );

}
```

Setting up MauiReactor from CLI

1. Install MauiReactor templates

```
dotnet new install Reactor.Maui.TemplatePack
```

2. Install MauiReactor hot reload console command

```
dotnet tool install -g Reactor.Maui.HotReload
```

If you already installed an old version of Reactor.Maui.HotReload you can update it to the latest using this command:

```
dotnet tool update -g Reactor.Maui.HotReload
```

https://github.com/adospace/reactorui-maui

MauiReactor

What is MauiReactor?
What's New in Version 2
Getting Started

COMPONENTS
State-less Components
Stateful Components
Component life-cycle
Component Properties
Component with children
Component Parameters
Theming
Navigation
Controls
Wrap 3rd party controls
Accessing native controls
Animation
Graphics
Window
Testing
XAML Integration

DEEP DIVES
Native tree and Visual tree
Dependency injection
Working with the GraphicsView
Migrating from MVVM Model
Using XAML Resources
Behaviors

RESOURCES
Source and Sample Applications

Q&A
How to deal with state shared across Components?

What is MauiReactor?

Describes what MauiReactor is and why you should be interested

MauiReactor is a .NET library that implements a **component-based UI framework** on top of the .NET MAUI.

> MauiReactor version 2 features a new way to write components. Some code on this documentation site uses the classic format.
> The writing component using the old way is perfectly valid and working, and will be maintained/supported also in the coming versions.
> Choosing between the two formats is up to you and your preferences.

If you already have some experience in React Native, Flutter, or Swift you may find some similarities. MauiReactor borrows some implementation designs from them while maintaining the normal MAUI application development you're used to.

NO XAML needed, just using C# you can write fluent declarative UI with MauiReactor components:

```
class MainPage : Component
{
    public override VisualNode Render()
        => ContentPage("Login",
            VStack( // Vertical Stack
                Label("User:"),
                Entry(),
                Label("Password:"),
                Entry(),
                HStack( // Horizontal Stack
                    Button("Login"),
                    Button("Register")
                )
            )
            .Center()
        );
}
```

ReactorMaui ports .NET MAUI controls to C#:

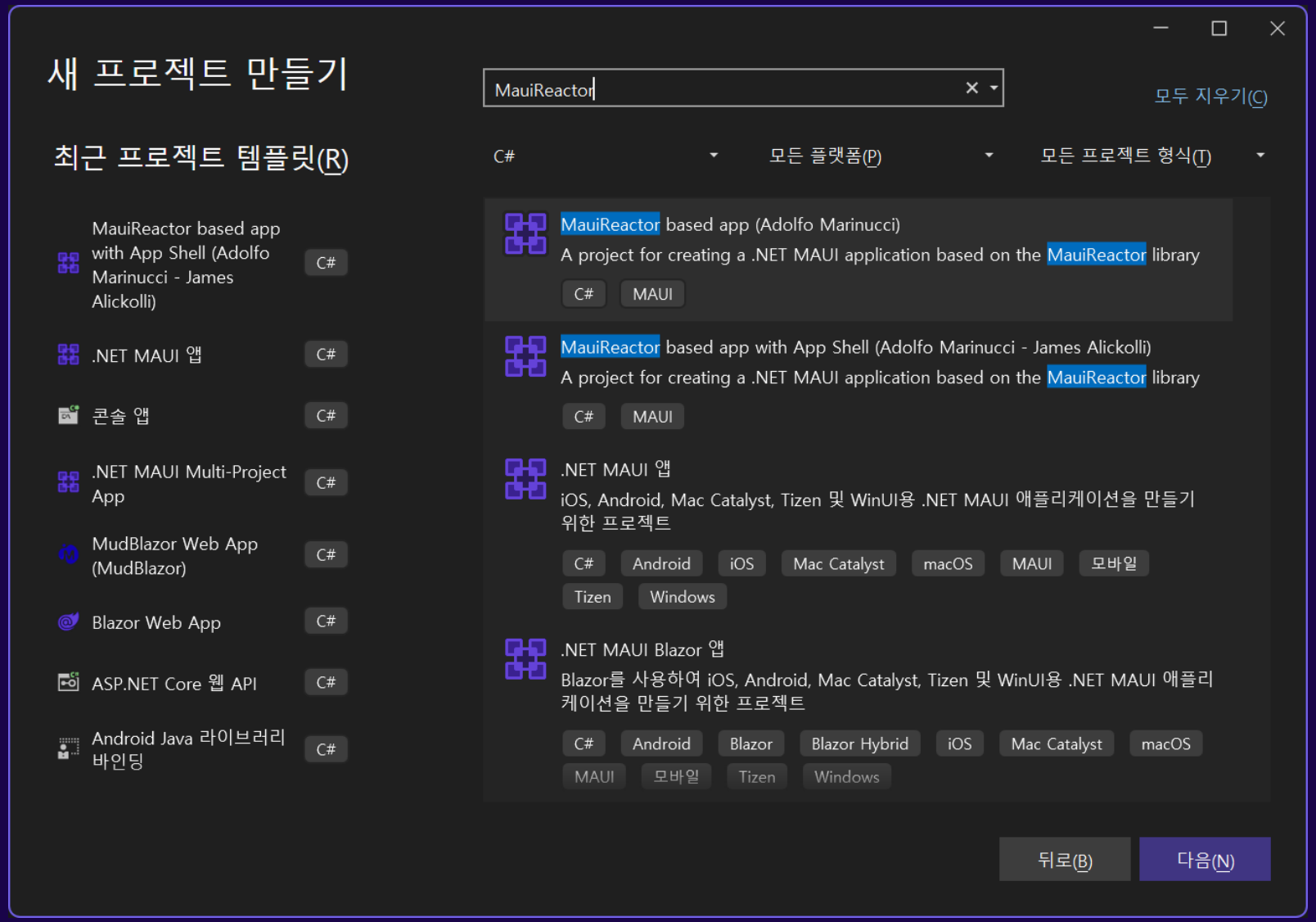
1. Dependency properties that deal with simple types (i.e. excluding templates or views) are translated to Prop fluent methods:

```
Button()
    .Text("Click!")
```

2. Events are translated to methods that accept a callback:

```
Button()
    .OnClicked(()=> ...)
```

https://adospace.gitbook.io/mauireactor


새 프로젝트 만들기
MauiReactor
모두 지우기(C)
최근 프로젝트 템플릿(R)
C#
모든 플랫폼(P)
모든 프로젝트 형식(T)
MauiReactor based app with App Shell (Adolfo Marinucci - James Alickolli)
C#
.NET MAUI 앱
C#
콘솔 앱
C#
.NET MAUI Multi-Project App
C#
MudBlazor Web App (MudBlazor)
C#
Blazor Web App
C#
ASP.NET Core 웹 API
C#
Android Java 라이브러리 바인딩
C#
MauiReactor based app (Adolfo Marinucci)
A project for creating a .NET MAUI application based on the MauiReactor library
C# MAUI
MauiReactor based app with App Shell (Adolfo Marinucci - James Alickolli)
A project for creating a .NET MAUI application based on the MauiReactor library
C# MAUI
.NET MAUI 앱
iOS, Android, Mac Catalyst, Tizen 및 WinUI용 .NET MAUI 애플리케이션을 만들기 위한 프로젝트
C# Android iOS Mac Catalyst macOS MAUI 모바일 Tizen Windows
.NET MAUI Blazor 앱
Blazor를 사용하여 iOS, Android, Mac Catalyst, Tizen 및 WinUI용 .NET MAUI 애플리케이션을 만들기 위한 프로젝트
C# Android Blazor Blazor Hybrid iOS Mac Catalyst macOS MAUI 모바일 Tizen Windows
뒤로(B)
다음(N)

```
public static class MauiProgram
{
    참조 4개
    public static MauiApp CreateMauiApp()
    {
        var builder = MauiApp.CreateBuilder();
        builder
            .UseMauiReactorApp<MainPage>(app =>
            {
                app.AddResource("Resources/Styles/Colors.xaml");
                app.AddResource("Resources/Styles/Styles.xaml");

            })
#if DEBUG
            .EnableMauiReactorHotReload()
            .OnMauiReactorUnhandledException(ex =>
            {
                System.Diagnostics.Debug.WriteLine(ex.ExceptionObject);
            })
#endif
```

	Stateful Component	Stateless Component
상태 관리	내부적으로 상태를 관리하고, 상태 변경에 따라 UI가 변경됨	상태를 관리하지 않음, 고정된 UI를 렌더링
UI 업데이트	상태가 변경될 때마다 렌더링 반복	외부 데이터나 고정된 데이터에 따라 한 번 렌더링
사용 예시	버튼 클릭, 폼 입력, 동적 리스트 등	단순 텍스트, 이미지, 외부 데이터 표시
렌더링 성능	상태 변경 시마다 리 렌더링이 발생하여 상대적으로 비용이 있을 수 있음	상태 변경이 없으므로 리 렌더링 비용이 적음

```
참조 0개
public static class ComponentExtensions
{
    참조 0개
    public static TComponent CustomComponent<TComponent>(this VisualNode node, Action<TComponent>? component = null)
        where TComponent : VisualNode, new()
    {
        var instance = new TComponent();
        component?.Invoke(instance);
        return instance;
    }
}
```

```
public class MainPage : Component
{
    private MauiControls.Entry _entryRef;

    public override VisualNode Render()
    {
        return new ContentPage()
        {
            new VStack(spacing: 10)
            {
                new Entry(entryRef => _entryRef = entryRef),

                new Button("Click here!")
                    OnClick(OnButtonClicked)
                    VCenter()
                    HCenter()
            }
        };
    }

    private void OnButtonClicked()
        => _entryRef?.Focus();
}
```

Microsoft.Maui.Controls.Entry

MauiReactor.Entry

```
[Scaffold(typeof(CommunityToolkit.Maui.Behaviors.StatusBarBehavior))]
참조 4개
partial class StatusBarBehavior
{
    참조 0개
    public CommunityToolkit.Maui.Behaviors.StatusBarBehavior GetNative()
        => NativeControl ?? null;
}
```

```
namespace MauiWanted.Components.Native
{
    참조 5개
    public class CustomEditor : MauiControls.Editor
    {
        참조 0개
        public static void Configure()
        {
            Microsoft.Maui.Handlers.EditorHandler.Mapper.AppendToMapping(nameof(Editor), (handler, view) =>
            {
#if ANDROID
                handler.PlatformView.SetBackgroundColor(Android.Graphics.Color.Transparent);

                // 키보드가 화면을 가리지 않도록 설정
                var activity = Microsoft.Maui.ApplicationModel.Platform.CurrentActivity;
                activity?.Window?.SetSoftInputMode(Android.Views.SoftInput.AdjustResize);
#endif
            });
        }
    }
}

namespace MauiWanted.Components.Controls
{
    [Scaffold(typeof(Native.CustomEditor))]
    참조 3개
    public partial class CustomEditor
    {
        참조 0개
        public Native.CustomEditor GetNative() => NativeControl;
    }
}
```

```
ScrollView(
    HStack(
        [.. State.jobList.Select((item) => RenderPostionItem(item))]
    ).Spacing(10)
).Orientation(ScrollOrientation.Horizontal).HorizontalScrollBarVisibility(ScrollBarVisibility.Never)
```

(X)

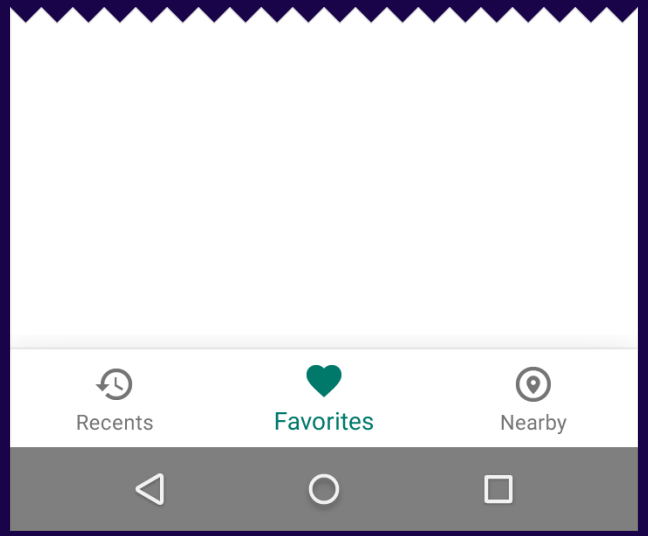
```
CollectionView((sen) =>
{
}).ItemsSource(State.jobList, RenderPostionItem)
.ItemsLayout(new HorizontalLinearItemsLayout().ItemSpacing(10))
.HorizontalScrollBarVisibility(ScrollBarVisibility.Never)
```

(O)

MauiReactor TIP #8

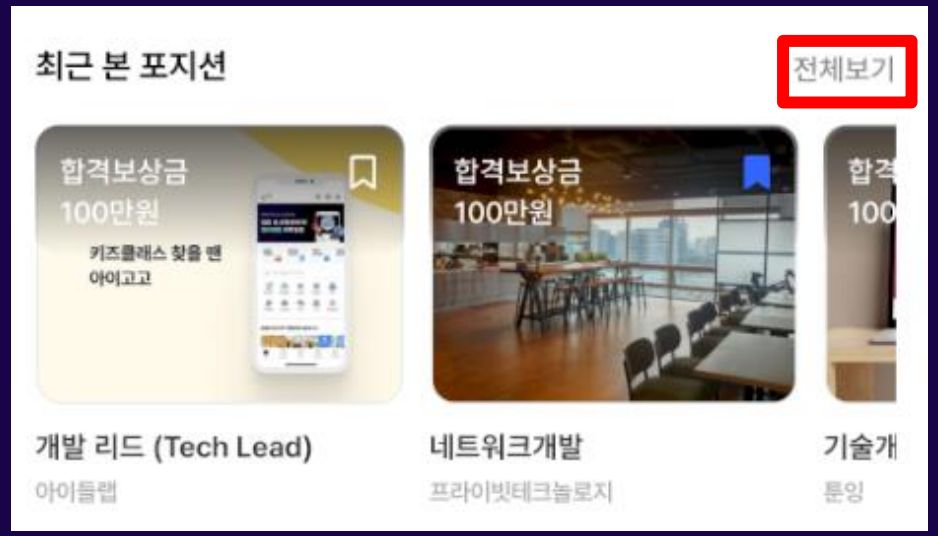

```
private VisualNode RenderContent()
{
    if (!State.CachedPages.TryGetValue(State.selectedIndex, out var page))
    {
        page = CreatePage(State.selectedIndex);
        State.CachedPages[State.selectedIndex] = page;
    }

    return State.CachedPages[State.selectedIndex];
}

private VisualNode CreatePage(int index)
{
    switch (index)
    {
        case 0:
            return new HomePage();
        case 1:
            return new ResvStatePage();
        case 2:
            return new CalendarPage();
        case 3:
            return new PaymentPage();
        case 4:
            return new MorePage();
        default:
            return new HomePage();
    }
}

private void OnTabSelected(int index)
{
    SetState(_ => _.selectedIndex = index);
}
```

(X)

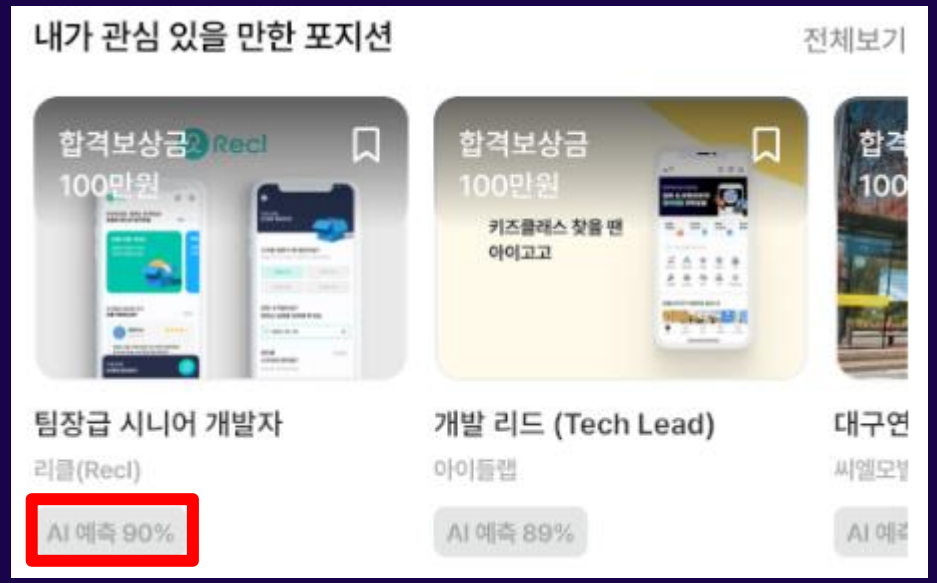



```
Grid RenderPage()
{
    return Grid(
        new HomePage().IsVisible(State.CurrentPage == PageEnum.Home ? true : false),

        new TempPage().IsVisible(State.CurrentPage == PageEnum.Career ? true : false),

        new TempPage().IsVisible(State.CurrentPage == PageEnum.Social ? true : false),

        new TempPage().IsVisible(State.CurrentPage == PageEnum.MyWanted ? true : false),

        new AllPage().IsVisible(State.CurrentPage == PageEnum.All ? true : false)
    );
}
```

(O)



- IsVisibleRightButton , RightButtonText RightButtonTextColor

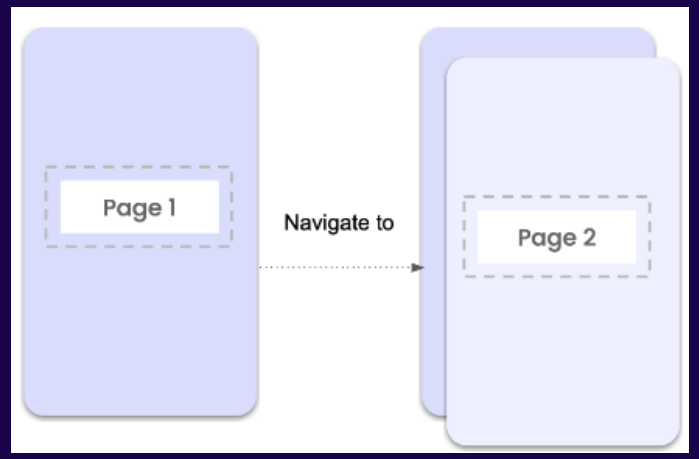


- IsVisiblePrediction, PredictionTextColor, PredictionBackColor

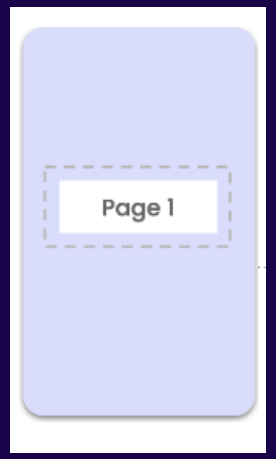
```
public class ComponentCacheService
{
    private Dictionary<string, object> _cache = new Dictionary<string, object>();

    1 reference
    public void CacheComponents(AppServices.DebugService _debugService)...

    7 references
    public T GetComponent<T>(string key) where T : class
    {
        if (_cache.ContainsKey(key))
        {
            return _cache[key] as T;
        }
        return null;
    }
}
```

```
_cache["recentPositions"] = new NormalTypePositionItemComponent()
    .HeaderText("최근 본 포지션")
    .RightButtonText("전체보기")
    .IsRightButtonVisible(true)
    .PropJobList(debugService.LoadRecentPositionsAsync());

_cache["mainCarousel"] = new MainCarouselComponent()
    .PropCarouselSource(debugService.LoadBannerFeedsAsync());

_cache["interestPositions"] = new NormalTypePositionItemComponent()
    .HeaderText("내가 관심 있을 만한 포지션")
    .RightButtonText("전체보기")
    .IsRightButtonVisible(true)
    .PropJobList(debugService.LoadInterestPositionsAsync());
```

MemoryStream



Navigation
Dispose()



Render() + MemoryStream
GoBack()

?

02

COMPONENT

Component & Template

Component

Progress® Telerik®

Syncfusion®

DevExpress®

themeforest

snppts

Component & Template

Component

Progress® Telerik®

Syncfusion®

DevExpress®

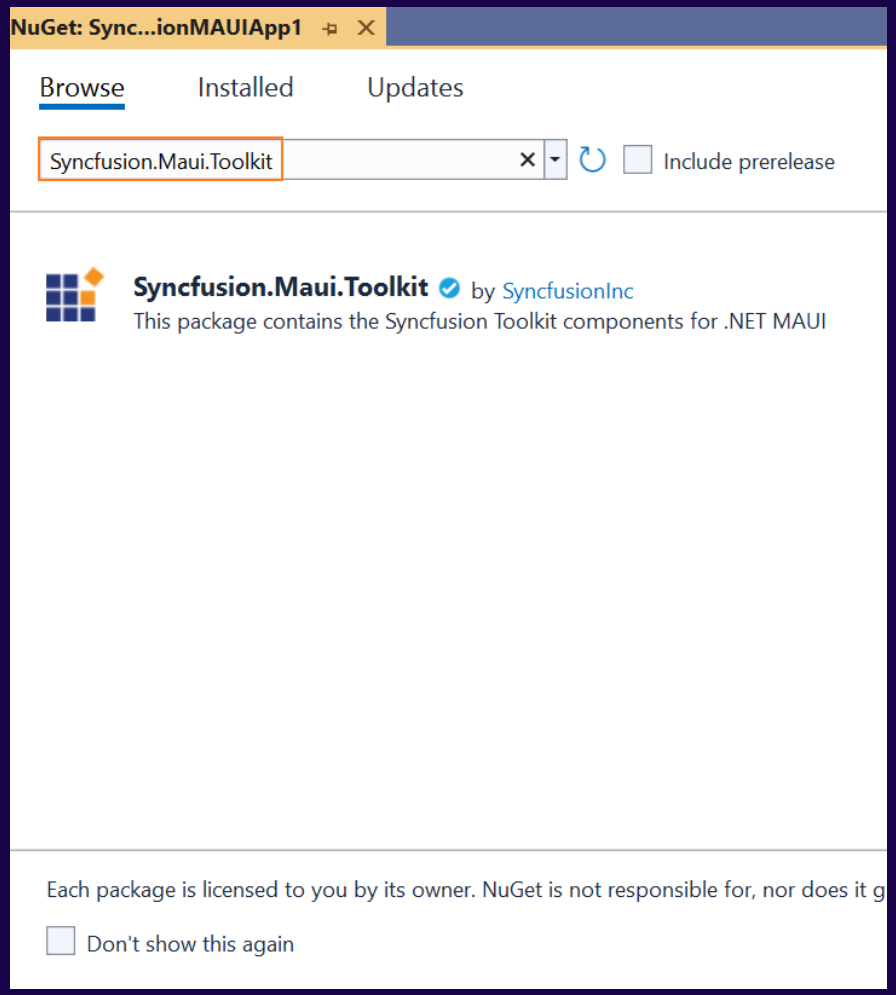
Grial Studio.

themeforest

snppts

Syncfusion .NET MAUI Toolkit

Component



- Syncfusion의 오픈소스 .NET MAUI 컨트롤 세트 (현재 2개 세트 오픈)
- .NET 8.0+ 이상 지원
- BottomSheet, Button, Calendar, Carousel, Catesian Chart,

  Chips, CircularChart, EffectView, FunnelChart,

  NavigationDrawer, NumericEntry, NumericUpDown,

  PolarChart, PulltoRefresh, PyramidChart, SegmentControl,

  Shimmer, TabView, TextInputLayout

03

STYLE

```
<Label
    Text="Hello, World!"
    Style="{StaticResource Headline}"
    SemanticProperties.HeadingLevel="Level1" />
```

```
<Style TargetType="Label" x:Key="Headline">
    <Setter Property="TextColor" Value="White"/>
    <Setter Property="FontSize" Value="32" />
    <Setter Property="HorizontalOptions" Value="Center" />
    <Setter Property="HorizontalTextAlignment" Value="Center" />
</Style>
```

StyleClass

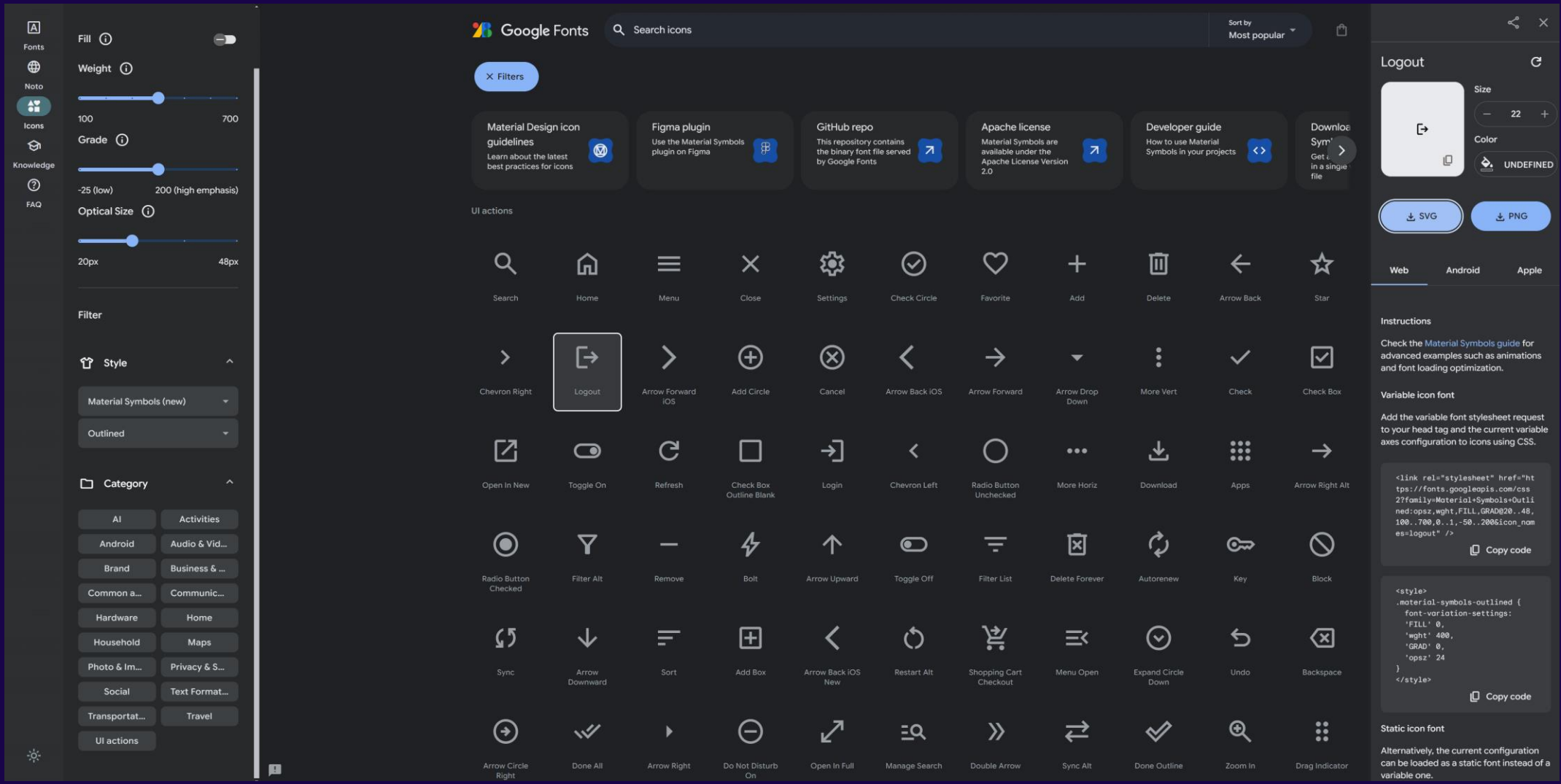
Style

```
<Label
    Text="Welcome to &#10;.NET Multi-platform App UI"
    StyleClass="Important,vs"
    SemanticProperties.HeadingLevel="Level2"
    SemanticProperties.Description="Welcome to dot net Multi platform App UI" />
```

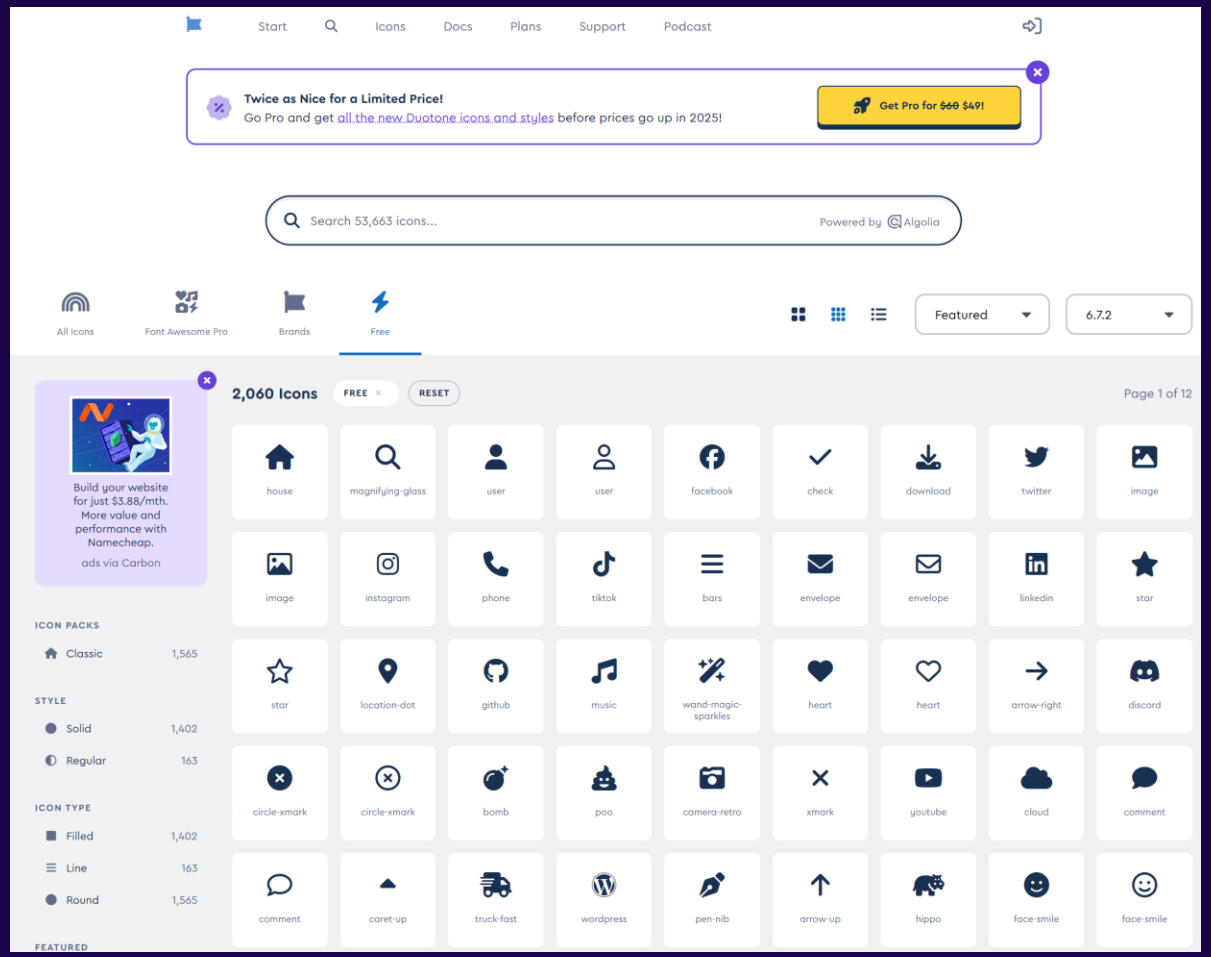
```
<Style TargetType="Label" Class="Important">
    <Setter Property="TextColor" Value="White" />
    <Setter Property="FontSize" Value="32" />
    <Setter Property="HorizontalOptions" Value="Center" />
    <Setter Property="HorizontalTextAlignment" Value="Center" />
</Style>

<Style TargetType="Label" Class="vs">
    <Setter Property="TextColor" Value="Red" />
    <Setter Property="FontAttributes" Value="Bold"/>
</Style>
```

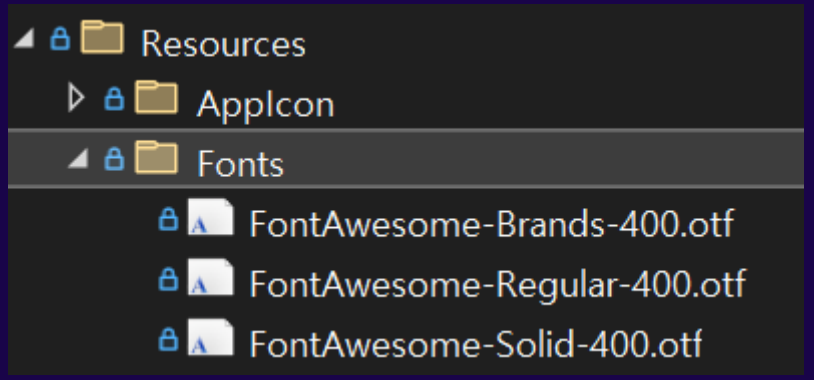
Icon & Font

Style



Google Fonts






```
"FontAwesome-Brands-400.otf", "FontAwsomeBrand");
"FontAwesome-Regular-400.otf", "FontAwsomeRegular");
"FontAwesome-Solid-400.otf", "FontAwsomeSolid");
```

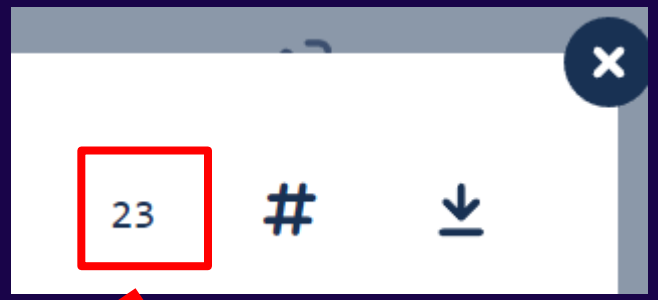
```
static class IconFont
{
    public const string Space = "₩u0020";
    public const string Exclamation = "₩u0021";
    public const string Hashtag = "₩u0023";
```

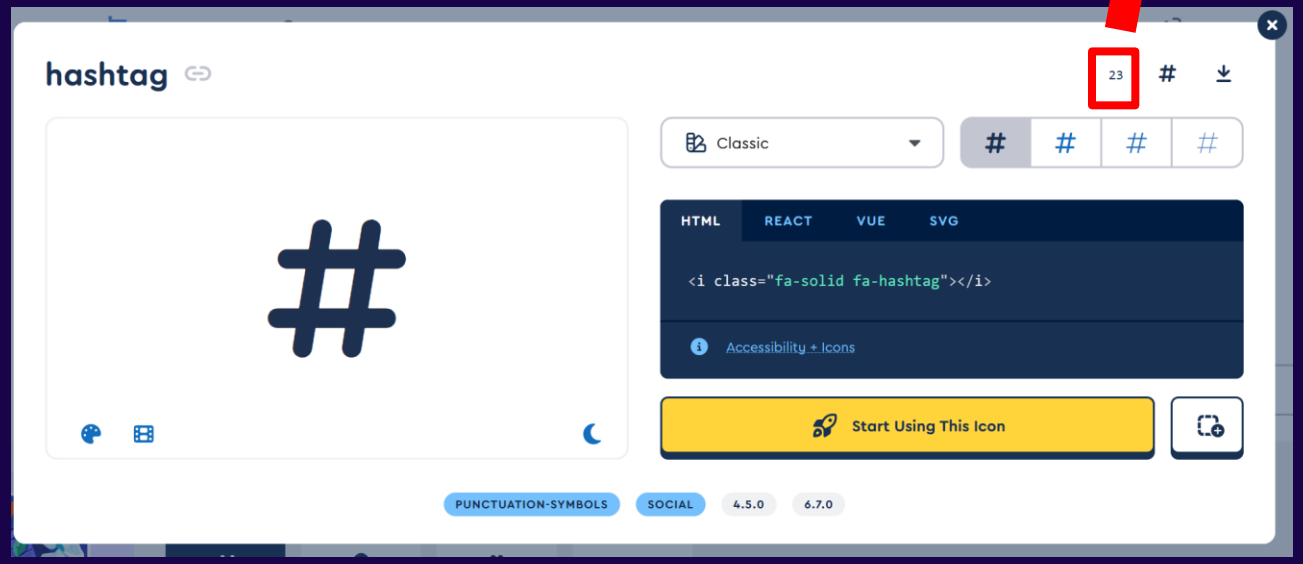
Font Awesome

Icon & Font

Style

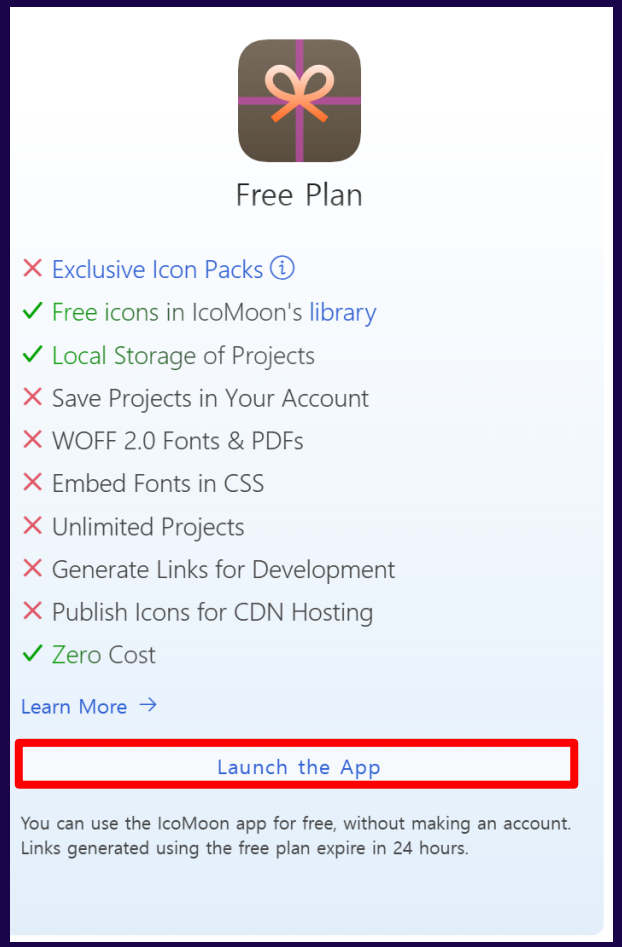
```
Hashtag = "₩u0023";
```

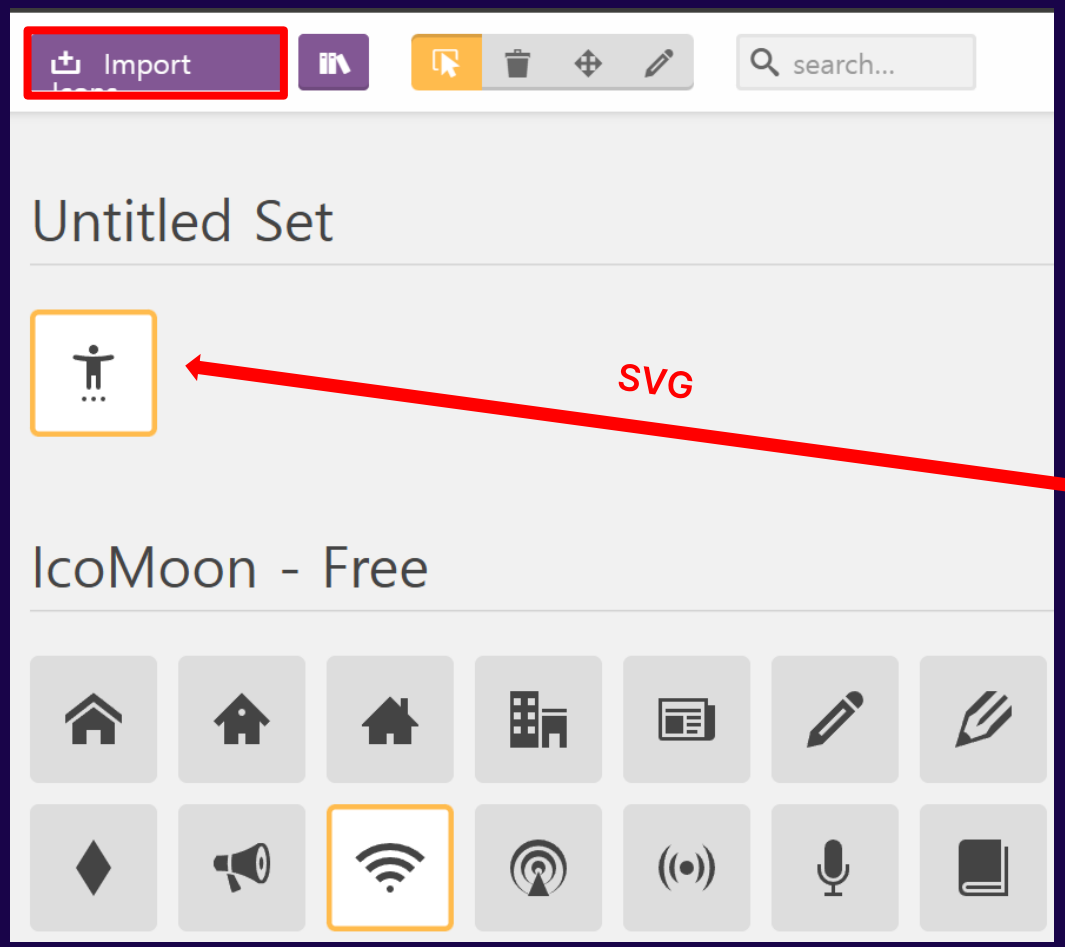




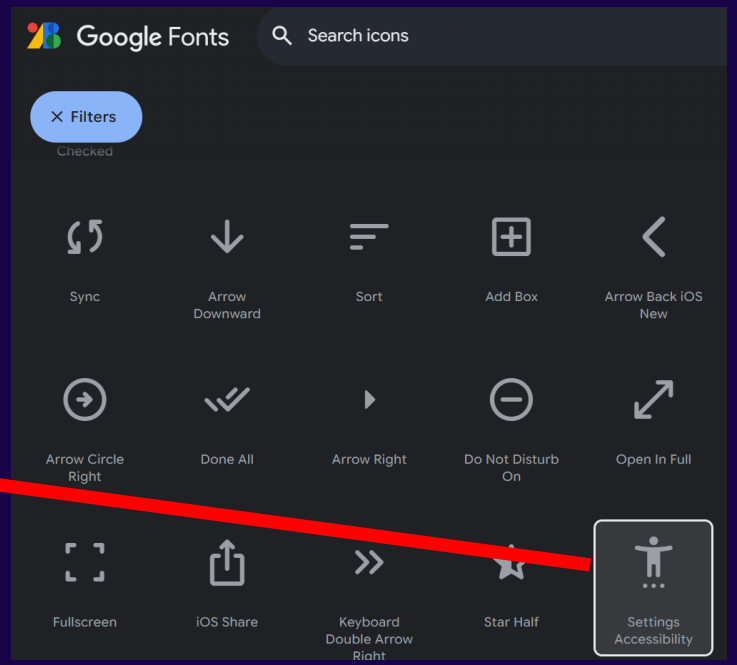
Icon & Font

Style





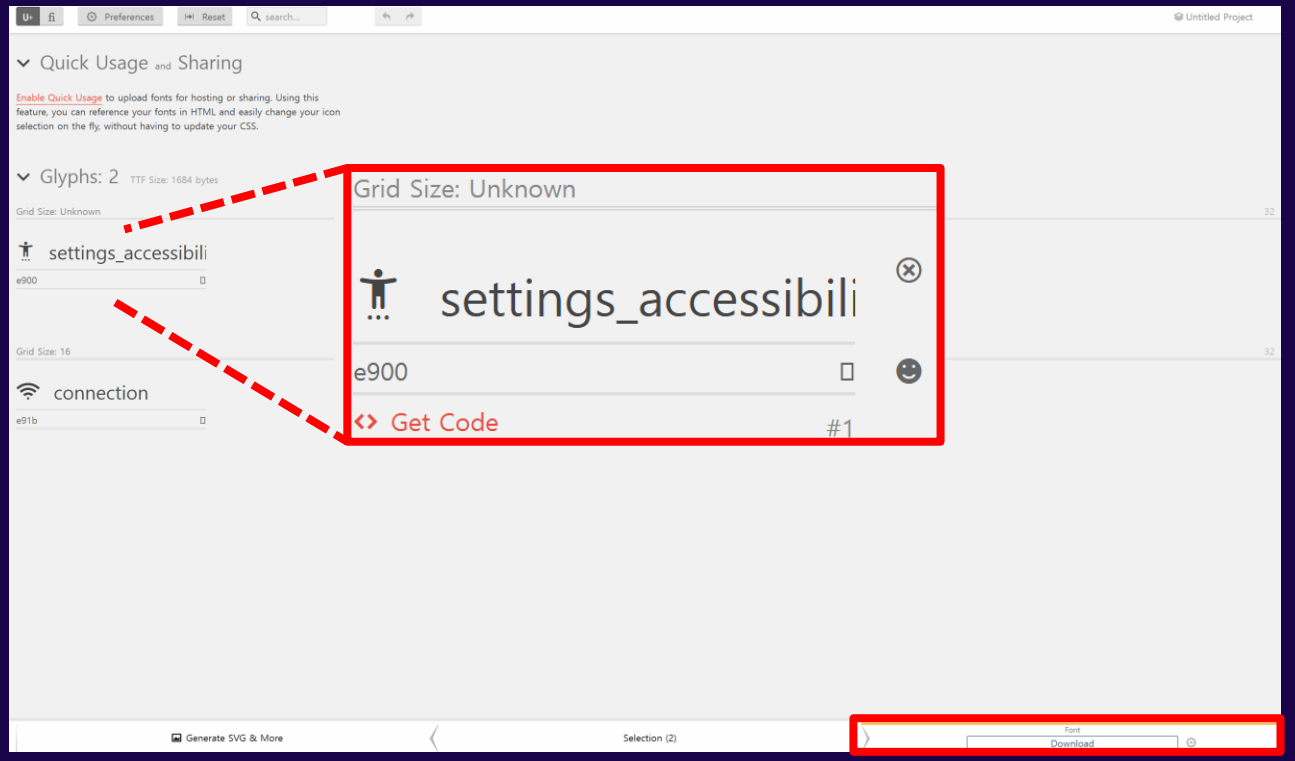
SVG

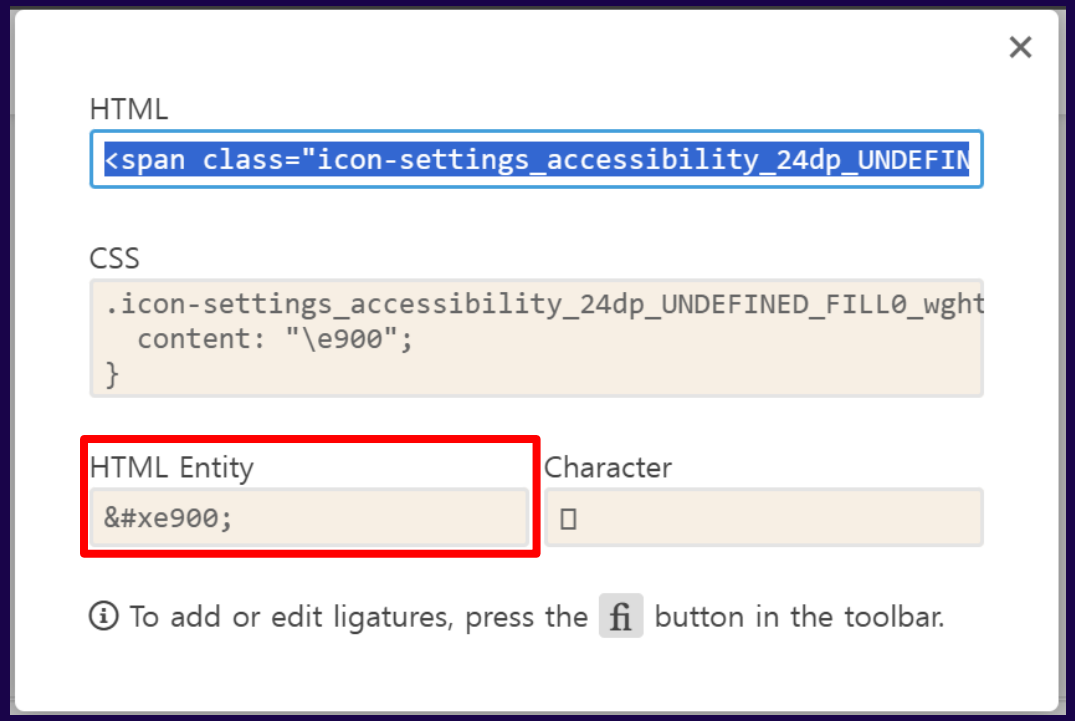


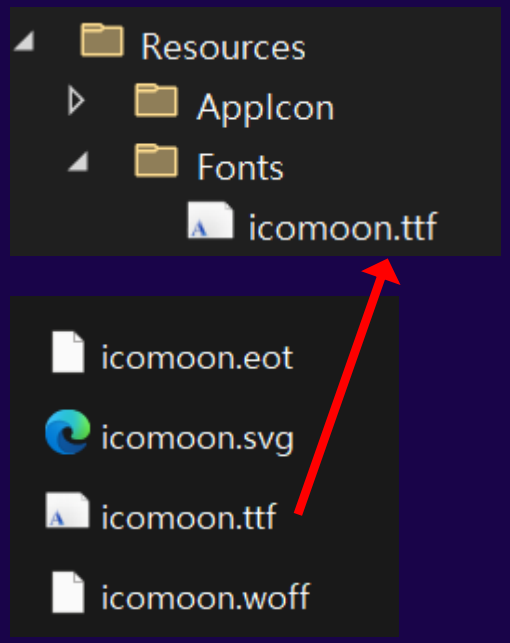
Icomoon.io

Icon & Font

Style

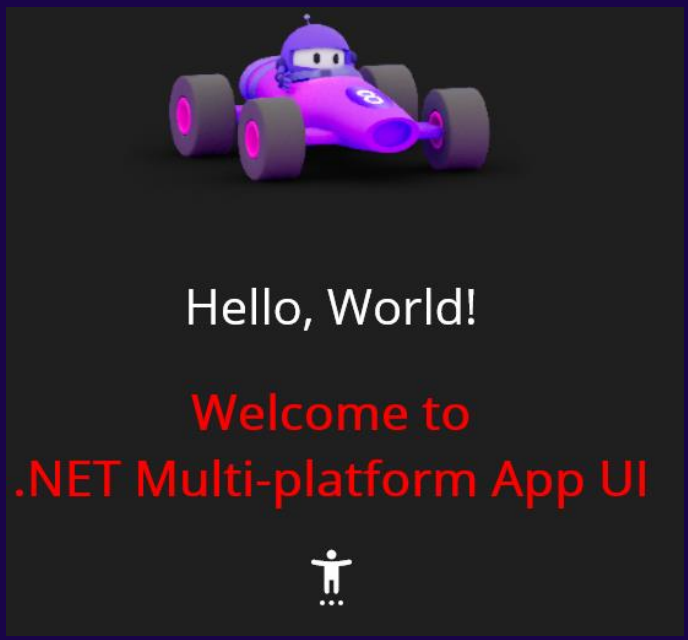







```
builder
    .UseMauiApp<App>()
    .ConfigureFonts(fonts =>
    {
        fonts.AddFont("icomoon.ttf", "dotnetconf");
        fonts.AddFont("OpenSans-Regular.ttf", "OpenS
        fonts.AddFont("OpenSans-Semibold.ttf", "Open
```

```
<Label Text="&#xe900;"
       FontFamily="dotnetconf"
```

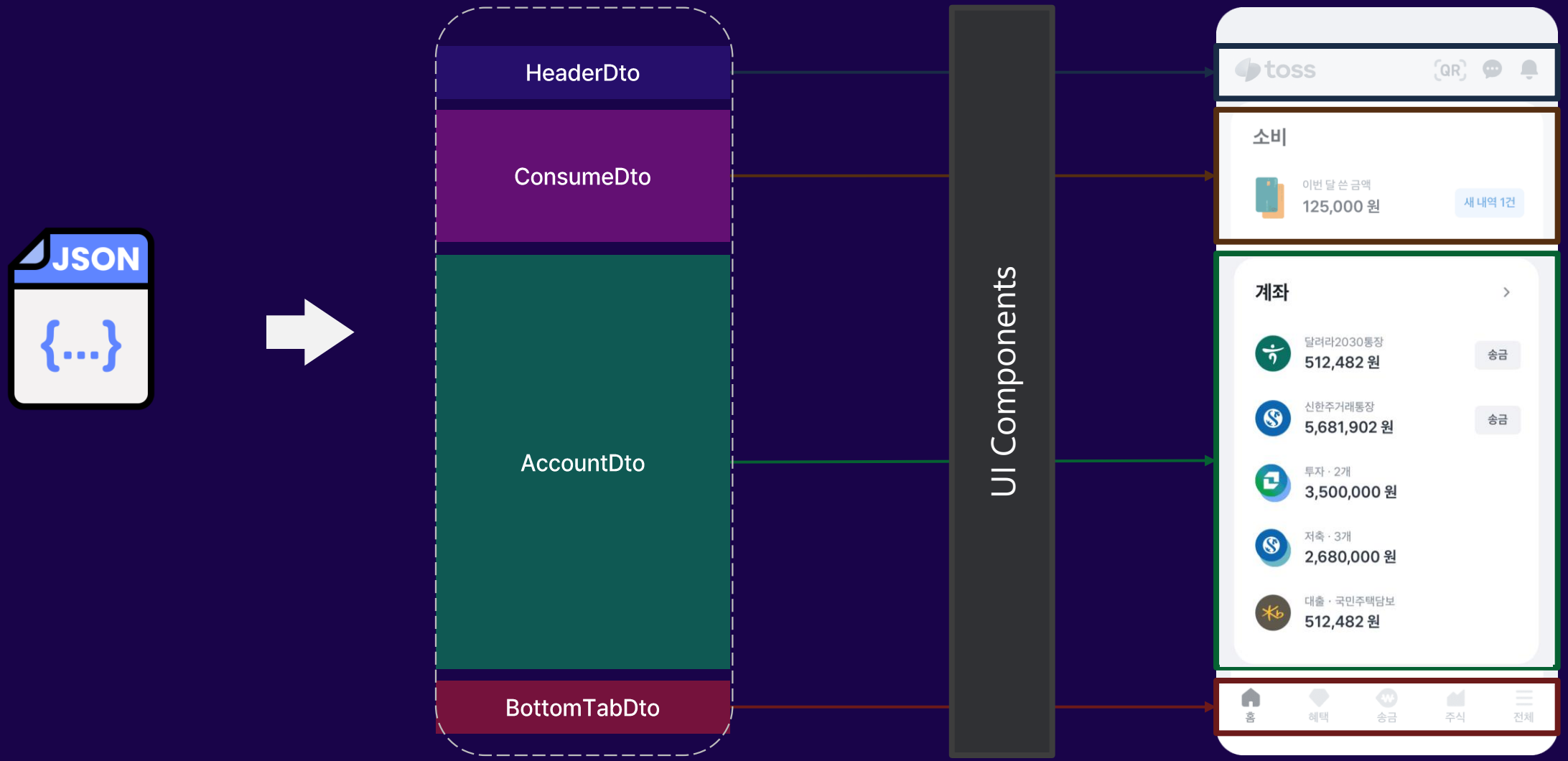


04

APP UPDATE

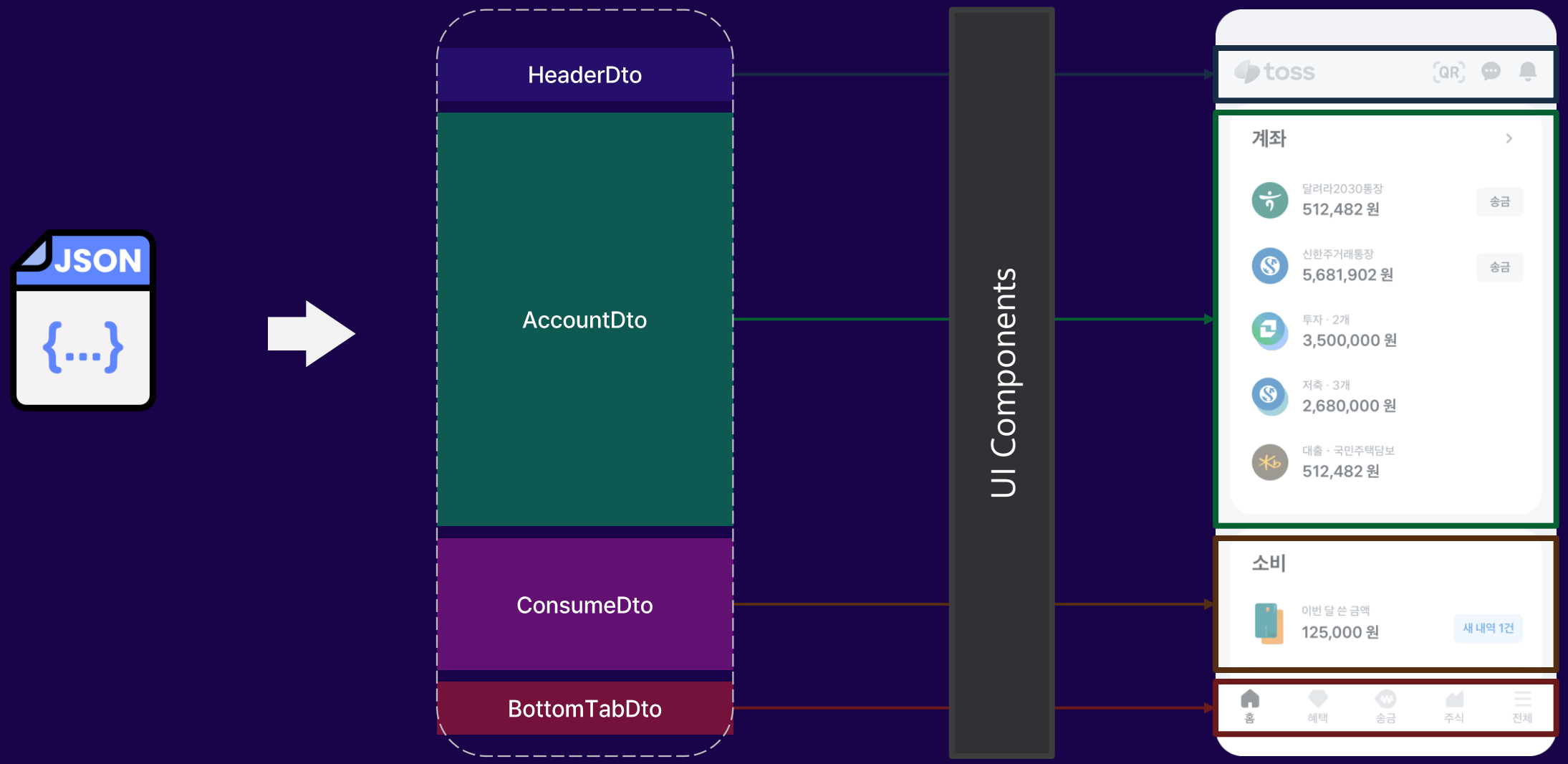
SDUI

App Update



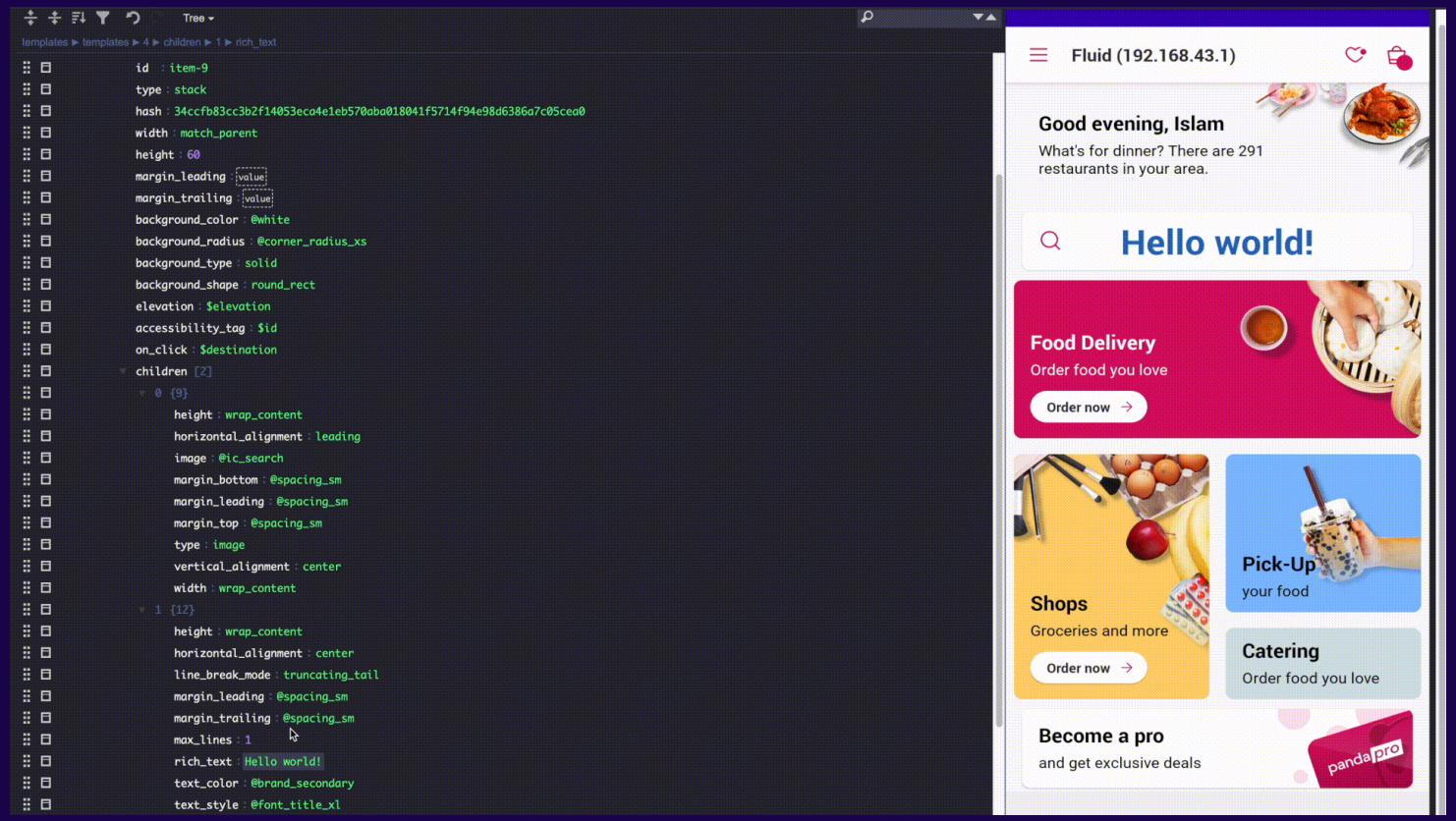
SDUI

App Update



SDUI

App Update



즉각적인 업데이트

일관 된 사용자 경험

서버 중심 로직 구성

유동적인 피처 테스팅

리소스 최적화

UI:

Old response:


```
{
  "nodes": [
    {
      "type": "AnnouncementCollection",
      "announcements": [
        {
          "title": "API is free for mod bots",
          "description": "Moderation Bots..."
        }
      ]
    },
    {
      "type": "SubredditPost",
      "title": "Pakkenwinkel of niet?",
      "description": "Hallo Nederland! Ik vraag...",
      "content": null,
      "topAwardedType": null,
      "domain": "i.redd.it",
      "isSpoiler": false,
      "isNsfw": false,
      "isLocked": false,
      "isSaved": false,
      "isHidden": false,
      "isCrosspostable": true,
      "isScoreHidden": false,
      "isTopAwarded": false,
      "removedByCategory": null,
      "isArchived": false,
      "isStickied": false,
      "isOriginalContent": false,
      "isCreatedFromAdsUi": false,
      "awardings": [],
      "isGildable": true,
      "isContestMode": false,
      "distinguishedAs": null,
      "gallery": null,
      "location": null,
      "voteState": "NONE",
      "score": 225,
      "upvoteRatio": 0.9828326180257511,
      "commentCount": 3,
      "viewCount": null,
      "authorFlair": null,
      "flair": null
      // ... and dozens of other unrelated fields
```


New response:


```
{
  "nodes": [
    {
      "type": "Group",
      "cells": [
        {
          "cellType": "AnnouncementHeaderCell",
          "title": "Updates from Reddit"
        },
        {
          "cellType":"AnnouncementCell",
          "announcements": [
            {
              "title": "API is free for mod bots",
              "description": "Moderation Bots..."
            }
          ]
        }
      ]
    },
    {
      "type": "Group",
      "cells": [
        {
          "cellType": "MetadataCell",
          "title": "r/nederlands"
        },
        {
          "cellType":"TitleCell",
          "title": "Pakkenwinkel of niet?"
        },
        {
          "cellType":"DescriptionCell",
          "title": "Hallo Nederland! Ik vraag..."
        },
        {
          "cellType": "ActionBarCell",
          "voteState": "NONE"
        }
      ]
    }
  ]
}
```

SDUI

App Update

Maui.ServerDrivenUI

Server Driven UI library for dotnet MAUI. New features to be deployed on all platforms simultaneously via a backend change, without releasing new versions of the native apps.

nuget v8.0.41-pre

Build and publish packages passing

Sample Api Response: https://serverdrivenui.azurewebsites.net/ServerDrivenUI?key=MyView

Getting started

- Install the ServerDrivenUI.Maui package

```
Install-Package ServerDrivenUI.Maui -Version 8.0.31-pre
```

- Add UseServerDrivenUI declaration to your `MauiAppBuilder` and configure it to connect to your API

```
public static class MauiProgram
{
        public static MauiApp CreateMauiApp()
        {
            var builder = MauiApp.CreateBuilder();
            builder
            .UseMauiApp<App>()
            .ConfigureFonts(fonts =>
            {
                fonts.AddFont("OpenSans-Regular.ttf", "OpenSansRegular");
                fonts.AddFont("OpenSans-Semibold.ttf", "OpenSansSemibold");
            })
            .ConfigureServerDrivenUI(s =>
            {
                s.RegisterElementGetter((key, provider) => provider.GetService<IYourApiService>().YourApiMethod(key));

                s.AddServerElement("MyView"); // Register the visual elements that will follow server driven ui
            });

                return builder.Build();
        }
}
```


```
JSON
  key : "MyView"
  type : "ServerDrivenView"
  class : ""
  properties
    Padding : "0"
    x:Name : "sduiView"
  customNamespaces
  content
    0
      key : ""
      type : "ScrollView"
      class : ""
      properties
      customNamespaces
      content
        0
          key : ""
          type : "VerticalStackLayout"
          class : ""
          properties
            Padding : "32,16"
            Spacing : "16"
          customNamespaces
          content
            0
              key : ""
              type : "Image"
              class : ""
              properties
                Aspect : "AspectFit"
                HeightRequest : "185"
                Source : "dotnet_bot.png"
              customNamespaces
              content
```


https://serverdrivenui.azurewebsites.net/ServerDrivenUI?key=MyView

https://github.com/felipebaltazar/Maui.ServerDrivenUI

C# Copy

```
string navigationButtonXAML = "<Button Text=\"Navigate\" />";
Button navigationButton = new Button().LoadFromXaml(navigationButtonXAML);
...
stackLayout.Add(navigationButton);
```

In this example, a Button instance is created, with its `Text` property value being set from the XAML defined in the `string`. The Button is then added to a StackLayout that has been defined in the XAML for the page.

> **Note**
>
> The `LoadFromXaml` extension methods allow a generic type argument to be specified. However, it's rarely necessary to specify the type argument, as it will be inferred from the type of the instance it's operating on.

The `LoadFromXaml` method can be used to inflate any XAML, with the following example inflating a ContentPage and then navigating to it:

C# Copy

```
// See the sample for the full XAML string
string pageXAML = "<?xml version=\"1.0\" encoding=\"utf-8\"?>\r\n<ContentPage xmlns=\"http://schemas.microso

ContentPage page = new ContentPage().LoadFromXaml(pageXAML);
await Navigation.PushAsync(page);
```

Microsoft.Maui.Controls.Xaml.Extensions.LoadFromXaml

https://learn.microsoft.com/ko-kr/dotnet/maui/xaml/runtime-load?view=net-maui-8.0

https://github.com/felipebaltazar/Maui.ServerDrivenUI/blob/8c881094bee0a4104468757a5ef76bf6ca64f8cd/Maui.ServerDrivenUI/Services/XamlConverterService.cs#L15

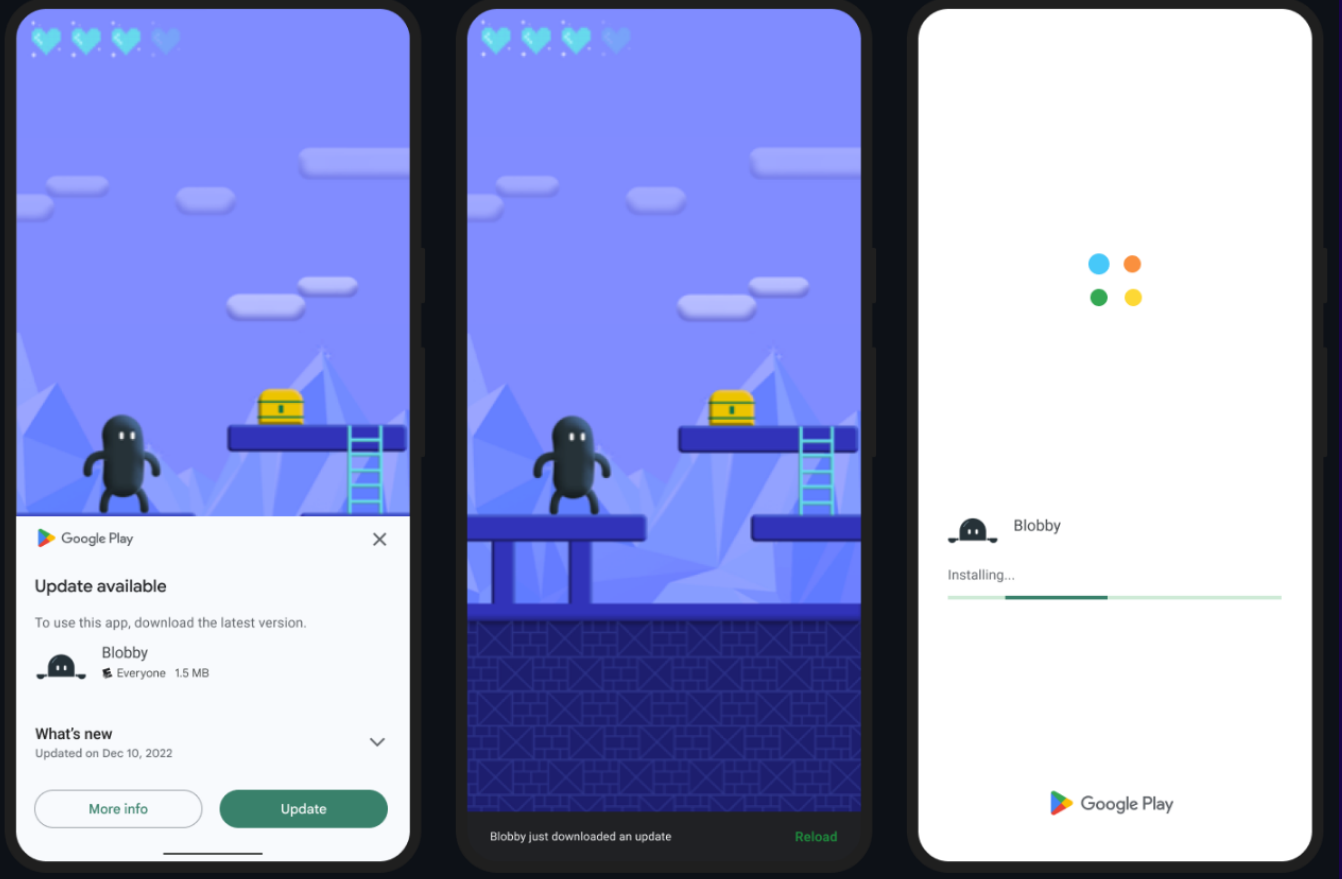
In App Update

App Update

Maui.Android.InAppUpdates

nuget v1.2.0 | Build, test and publish passing | license MIT

NuGet package that implementing Android In-App Updates for MAUI with debugging capabilities.



업데이트 우선순위 확인

Google Play Developer API를 사용하여 각 업데이트의 우선순위를 설정할 수 있습니다. 이렇게 하면 앱에서 사용자에게 업데이트를 얼마나 강력하게 권장할지 결정할 수 있습니다. 업데이트 우선순위를 설정하는 다음 전략을 예로 들어 보겠습니다.

- 사소한 UI 개선: **낮은 우선순위** 업데이트. 유연한 업데이트와 즉시 업데이트 중 어떤 것도 요청하지 않습니다. 사용자가 앱과 상호작용하지 않을 때만 업데이트합니다.
- 성능 개선: **중간 우선순위** 업데이트. 유연한 업데이트를 요청합니다.
- 중요 보안 업데이트: **높은 우선순위** 업데이트. 즉시 업데이트를 요청합니다.

우선순위를 결정하기 위해 Google Play는 0에서 5 사이의 정숫값을 사용하며 0은 기본값, 5는 가장 높은 우선순위를 나타냅니다. 업데이트 우선순위를 설정하려면 Google Play Developer API의 `Edits.tracks.releases` 아래에 있는 `inAppUpdatePriority` 필드를 사용하세요. 출시에 새로 추가된 모든 버전은 출시와 동일한 우선순위로 간주됩니다. 우선순위는 새 버전을 출시할 때만 설정할 수 있으며 나중에 변경할 수 없습니다.

Play Developer API 문서에 설명된 대로 Google Play Developer API를 사용하여 우선순위를 설정합니다. 인앱 업데이트 우선순위는 `Edit.tracks: update` 메서드에 전달된 `Edit.tracks` 리소스에서 지정해야 합니다. 다음 예는 버전 코드가 88이고 `inAppUpdatePriority`가 5인 APK 출시를 보여 줍니다.

```
{
  "releases": [{
      "versionCodes": ["88"],
      "inAppUpdatePriority": 5,
      "status": "completed"
  }]
}
```

- 우선순위 1~3이 지정되면 유연한 업데이트가 제공됩니다.
- 우선순위 4-5가 지정되면 즉시 업데이트가 제공됩니다.

Custom Update

App Update

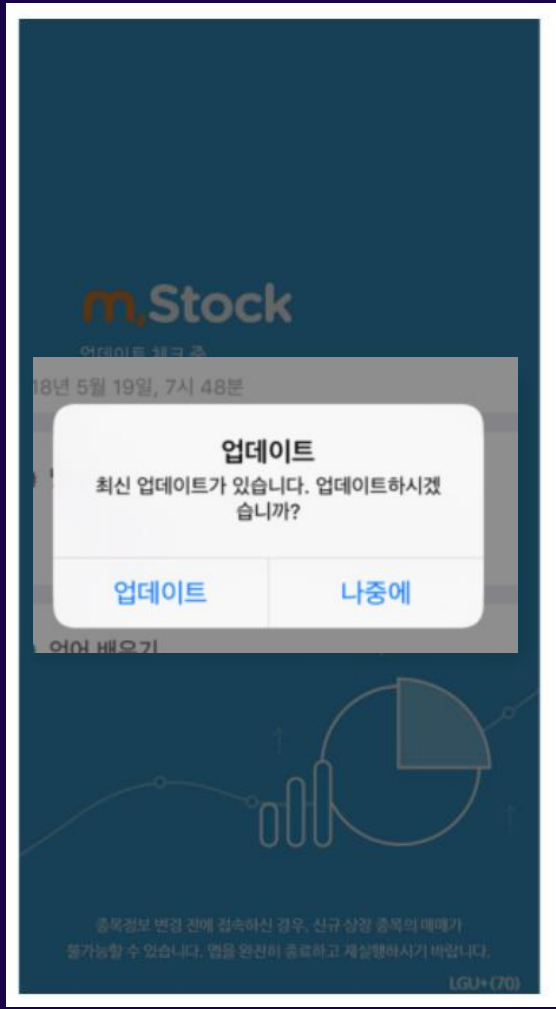
BunnyCDN

/Version.json

Get / API

API

```
{
  "isUpdateMode" : true,
  "android": {
    "latestVersion": "3.2.2",
    "priority": "immediate",
    "releaseNotes": "안드로이드용 새 기능과 성능 개선 업데이트입니다.",
    "updateUrl": "{AOS 스토어 주소}"
  },
  "ios": {
    "latestVersion": "3.2.2",
    "priority": "flexible",
    "releaseNotes": "iOS에서 더 나은 사용자 경험을 위한 업데이트입니다.",
    "updateUrl": "{iOS 스토어 주소}"
  },
  "popupContent": {
    "title": {
      "ko": "새로운 업데이트가 필요합니다! 🚀",
      "en": "A New Update is Required! 🚀"
    },
    "message": {
      "ko": "최신 버전으로 업데이트하고 새롭게 개선된 기능과 안정성을 경험하세요.",
      "en": "Update to the latest version for improved features and stability."
    },
    "optionalMessage": {
      "ko": "지금 업데이트하시면 더 나은 사용자 경험을 누릴 수 있습니다!",
      "en": "Enjoy a better user experience by updating now!"
    }
  },
  "buttons": {
    "updateNow": {
      "ko": "지금 업데이트",
      "en": "Update Now"
    },
    "remindLater": {
      "ko": "나중에 하기",
      "en": "Remind Me Later"
    }
  }
}
```



05

PUSH NOTIFICATION

OneSignal

New OneSignal App

Set up a new OneSignal app to get sending. Need help? Read our getting started docs

OneSignal App Name

What organization should it belong to?

Default Organization

Set up your first channel

Apple iOS (APNs)

Google Android (FCM)

Web

Email

SMS

More Options

Default Organization > dotnetconf > Settings

Google Android (FCM) Configuration

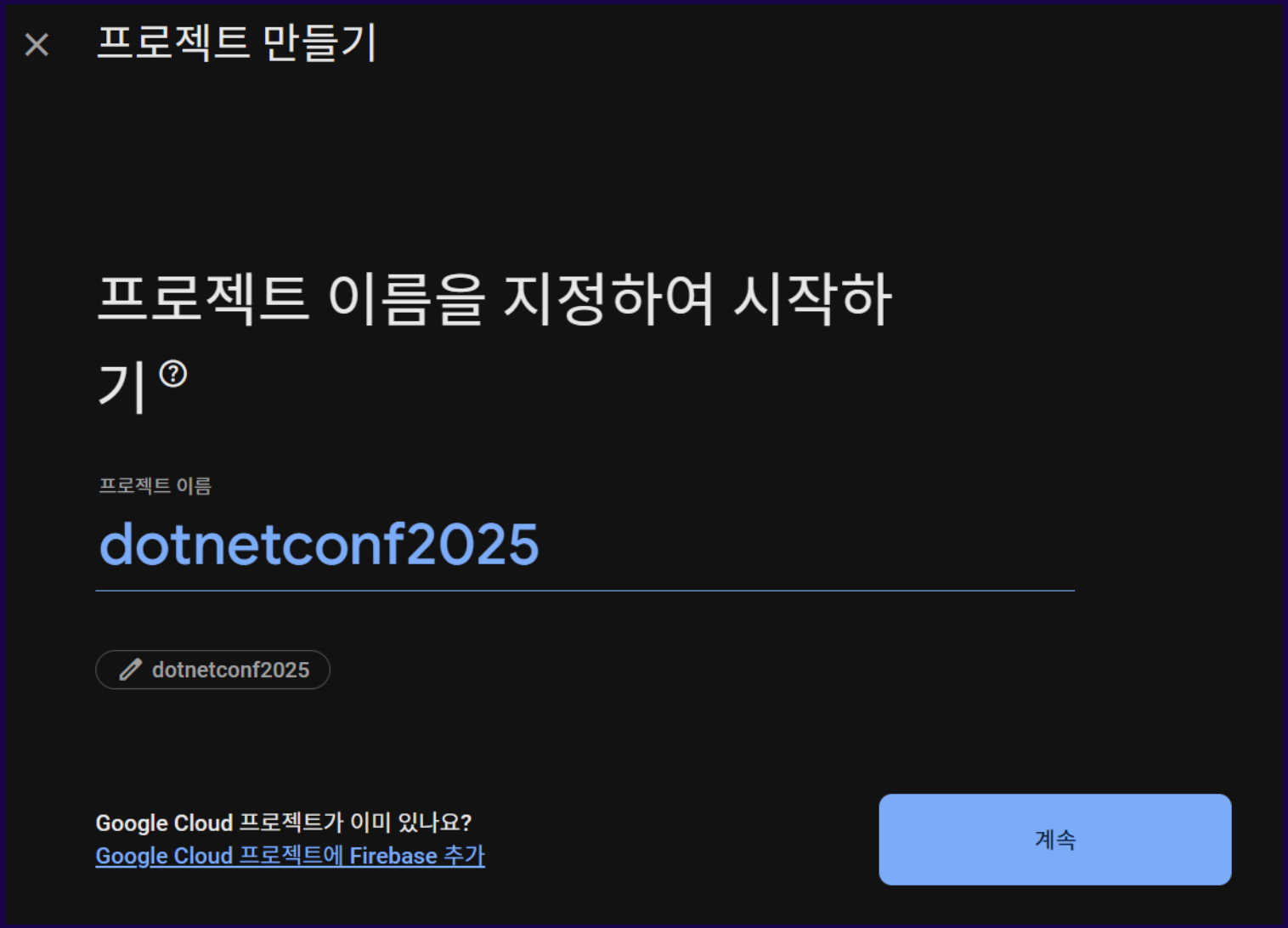
Now, let's configure your app. Read our Android documentation to learn how to complete the fields below.

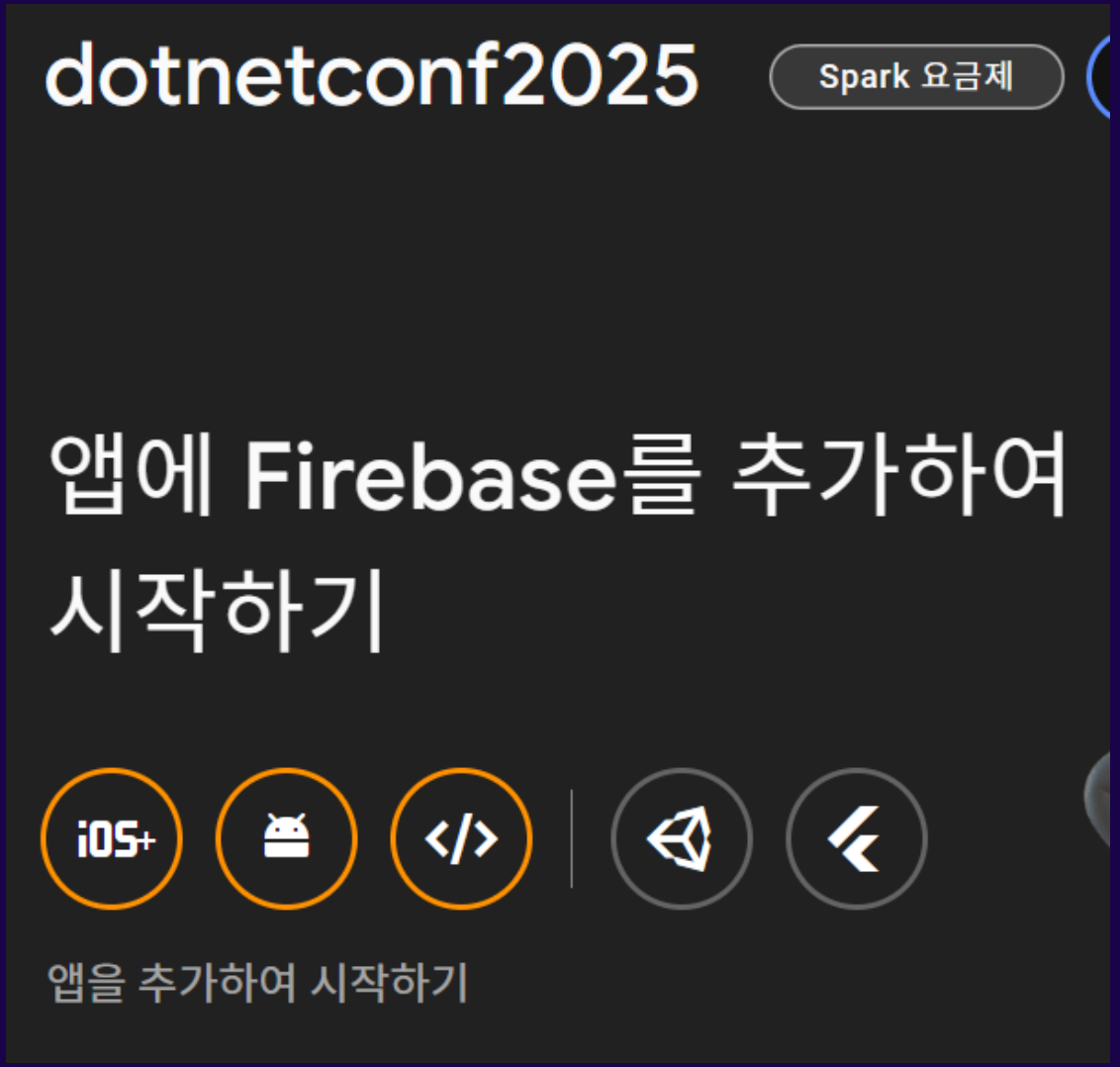
Service Account JSON *

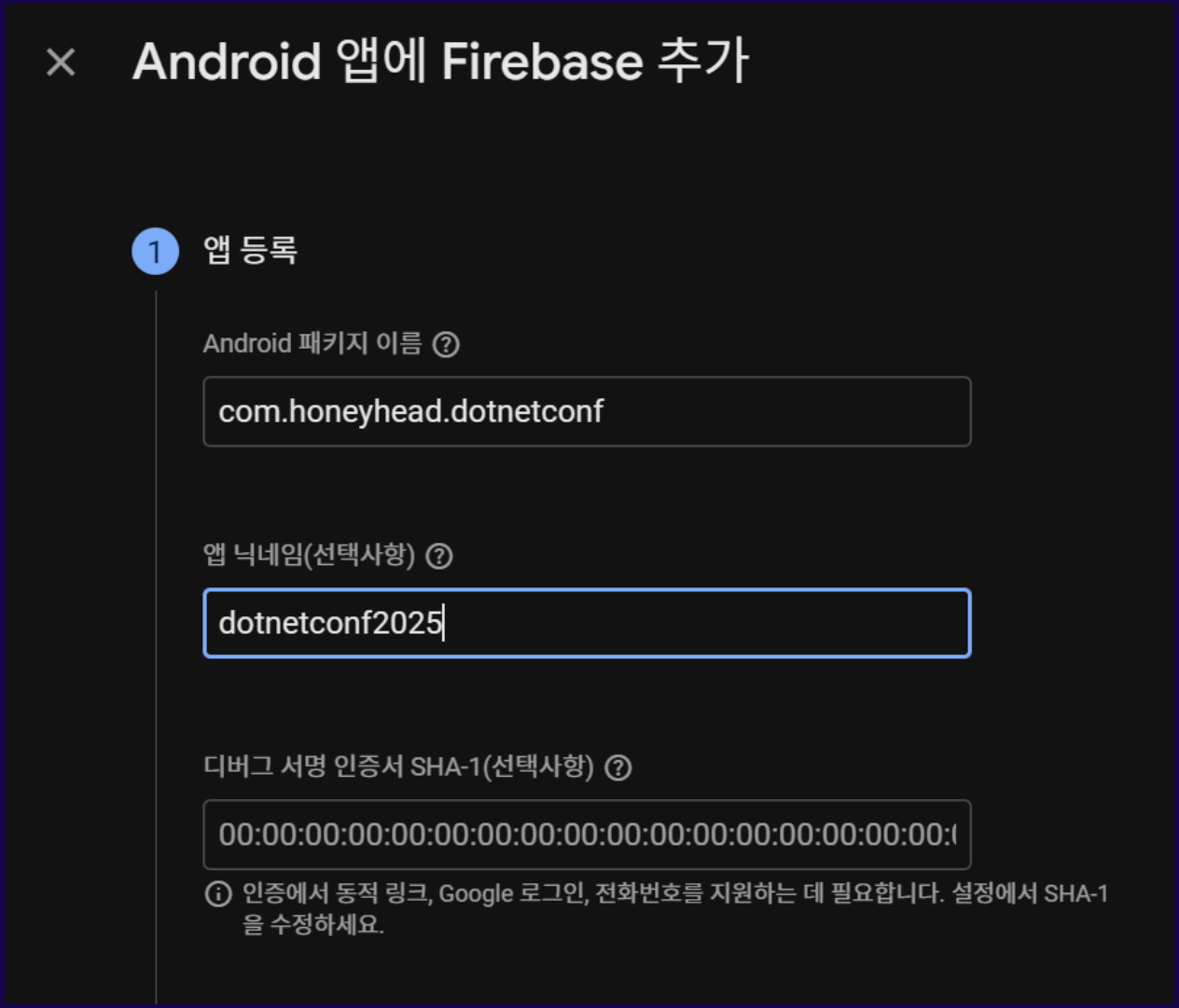
Upload from your computer

Retrieve from your Firebase Console

Need help? Invite a team member to complete this step.


프로젝트 만들기
프로젝트 이름을 지정하여 시작하기
프로젝트 이름
dotnetconf2025
dotnetconf2025
Google Cloud 프로젝트가 이미 있나요?
Google Cloud 프로젝트에 Firebase 추가
계속


dotnetconf2025
Spark 요금제
앱에 Firebase를 추가하여 시작하기
iOS+
앱을 추가하여 시작하기


Android 앱에 Firebase 추가
1
앱 등록
Android 패키지 이름
com.honeyhead.dotnetconf
앱 닉네임(선택사항)
dotnetconf2025
디버그 서명 인증서 SHA-1(선택사항)
00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:0
인증에서 동적 링크, Google 로그인, 전화번호를 지원하는 데 필요합니다. 설정에서 SHA-1을 수정하세요.

dotnetconf2025

프로젝트 설정

일반 | 클라우드 메시징 | 통합 | 서비스 계정

Firebase Admin SDK

이전 사용자 인증 정보

데이터베이스 비밀번호

모든 서비스 계정

서비스 계정 2개

Firebase Admin SDK

Firebase 서비스 계정을 사용하여 통합 Admin SDK를 통해 데이터베이스, Storage, 인증 등의 다양한 Firebase 기능을 프로그래매틱 방식으로 인증할 수 있습니다. 자세히 알아보기

Firebase 서비스 계정

firebase-adminsdk-oev7n@dotnetconf2025.iam.gserviceaccount.com

Admin SDK 구성 스니펫

Node.js | 자바 | Python | Go

```
var admin = require("firebase-admin");

var serviceAccount = require("path/to/serviceAccountKey.json");

admin.initializeApp({
  credential: admin.credential.cert(serviceAccount)
});
```

새 비공개 키 생성

Default Organization > dotnetconf > Settings

Google Android (FCM) Configuration

Now, let's configure your app. Read our Android documentation to learn how

Service Account JSON *

dotnetconf2025-firebase-adminsdk.json

Retrieve from your Firebase Console

Need help? Invite a team member to complete this step.

Save & Continue

Go Back

Default Organization > dotnetconf > Settings

Google Android (FCM) Configuration

Select your target SDK. We'll take you through the steps to get your first user & send your first test notification.

Native Android

Cordova

React Native / Expo

Unity

Xamarin

Need help? Invite a team member to complete this step.

Save & Continue

Go Back

Default Organization > dotnetconf > Settings

Google Android (FCM) Configuration

1. Install the SDK

Read our Xamarin documentation to learn how to install the SDK.

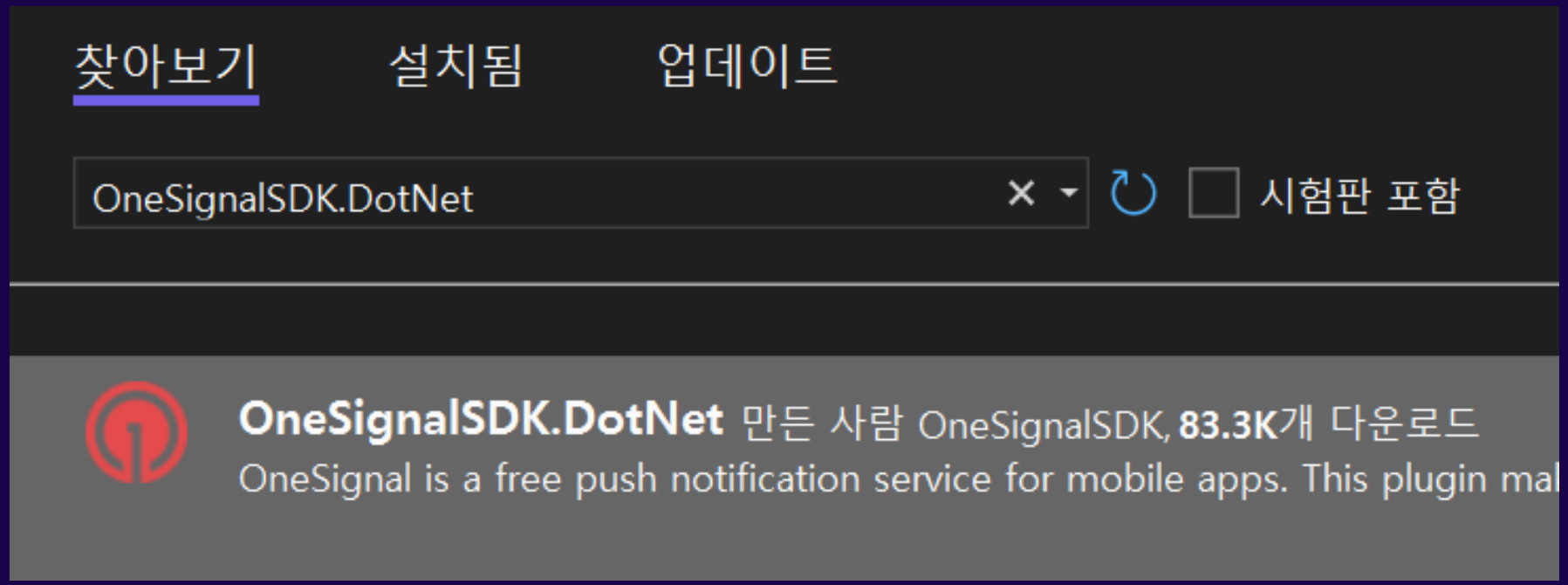
Your App ID: `0a10de18-bd70-4c2f-aefe-0f7238c4afe3`

2. Test for Subscribed Users

Build and run your app. The OneSignal SDK, once set up correctly,

Check Subscribed Users

Need help? Invite a team member to complete this step.




```
OneSignal.Debug.LogLevel = OneSignalSDK.DotNet.Core.Debug.LogLevel.VERBOSE;
OneSignal.Initialize("0a10de18-bd70-4c2f-aefe-0f7238c4afe3");
```

위에 키는 발급 받으신 키를 입력하셔야 합니다

기본 조직 > dotnetconf > 설정

Google Android(FCM) 구성

1. SDK 설치

우리의 것을 읽어보세요Xamarin 설명서SDK 설치 방법을 알아보세요. 플랫폼에 맞는 올바른 가이드를 따르세요!

귀하의 앱 ID: `0a10de18-bd70-4c2f-aefe-0f7238c4afe3`

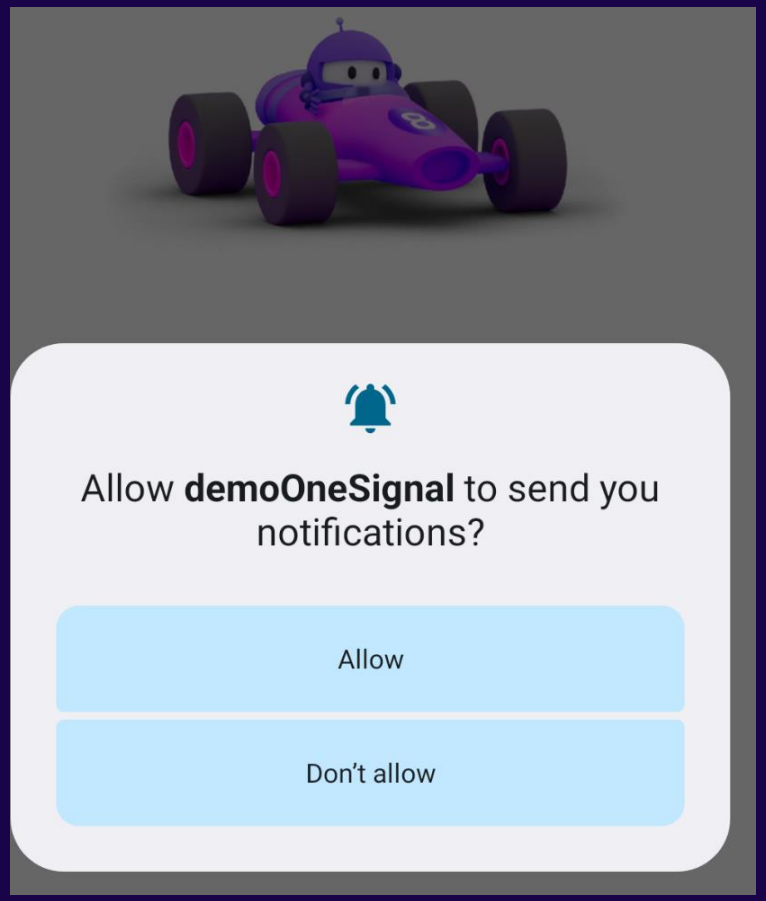
2. 구독 사용자 테스트

앱을 빌드하고 실행하세요. OneSignal SDK는 올바르게 설정되면 자동으로 기기의 알림 구독을 처리합니다.

구독 사용자 확인

이 앱에 대한 총 구독 사용자가 1명 있지만 , **Google Android(FCM)** 플랫폼에 대한 구독 사용자는 없습니다. 계속하려면 **Google Android(FCM)** 에 대한 구독 사용자가 최소 1명 필요합니다 .

도움이 필요하세요? 팀원을 초대하여 이 단계를 완료하세요.



2. 구독 사용자 테스트

앱을 빌드하고 실행하세요. OneSignal SDK는 올바르게 설정되면 자동으로 기기의 알림 구독을 처리합니다.

구독 사용자 확인

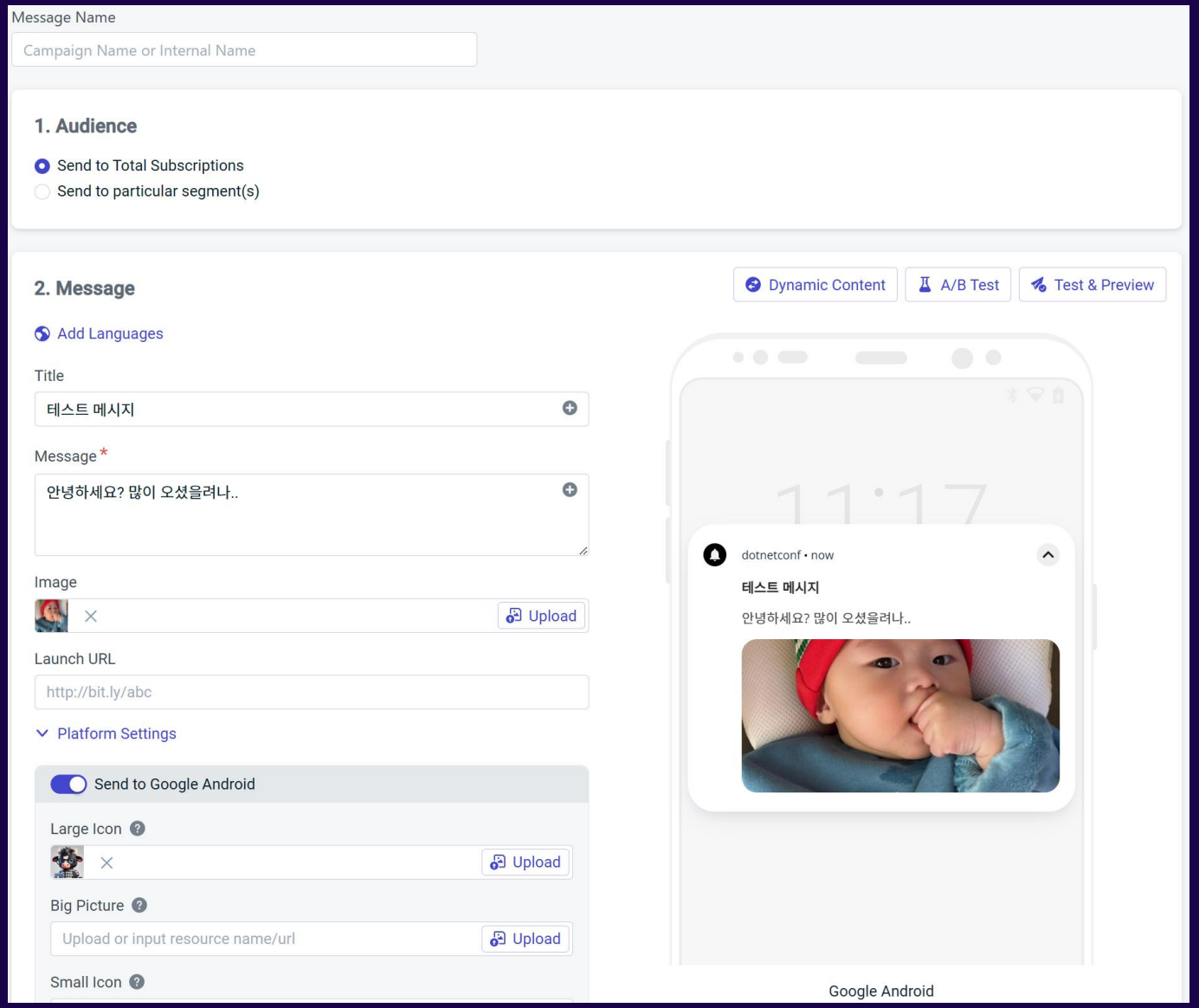
축하합니다! **Sdk_gphone_x86_64** 에서 실행되는 첫 번째 사용자 가 성공적으로 구독되었습니다!

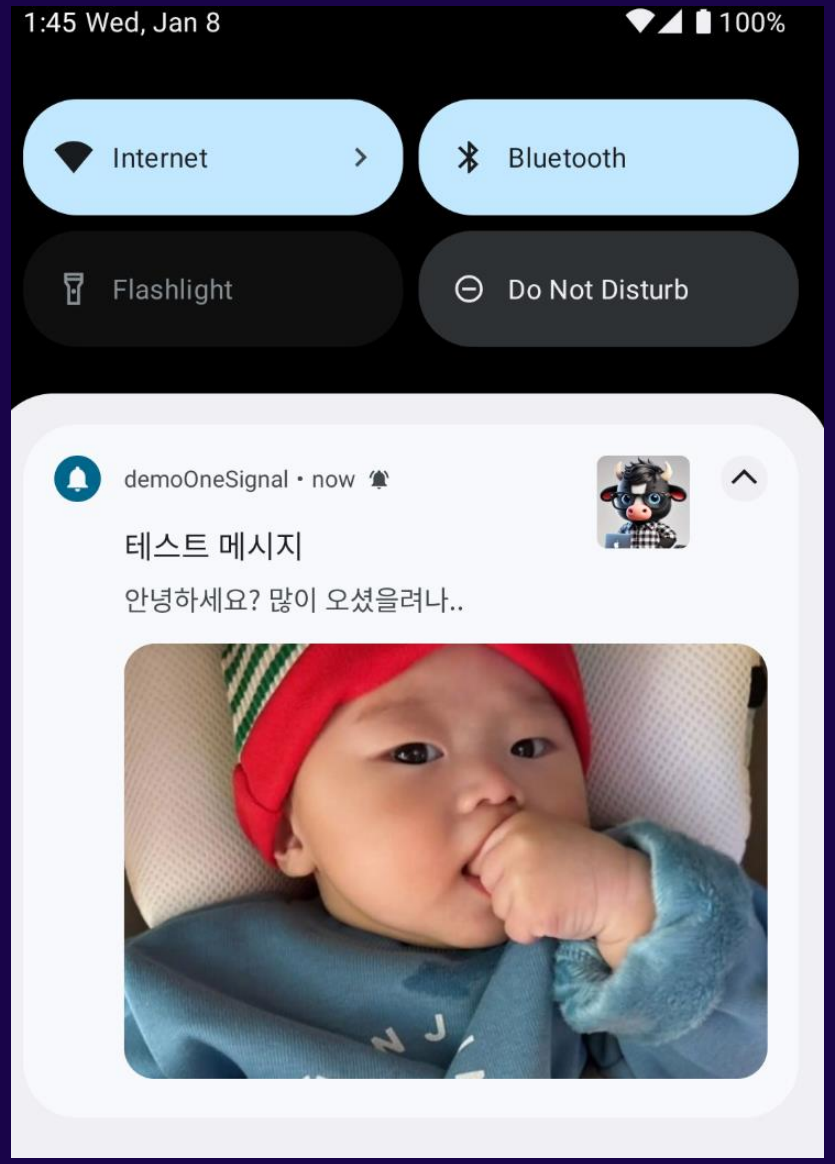
SDK의 문제 해결이나 기능 사용자 정의에 대한 설명서를 꼭 확인하세요 !

메시지 보내기

OneSignal

App Update





Default Organization > dotnetconf > Settings

Keys & IDs

App ID

OneSignal App ID

0a10de18-bd70-4c2f-aefe-0f7238c4afe3

API Keys

Key Security

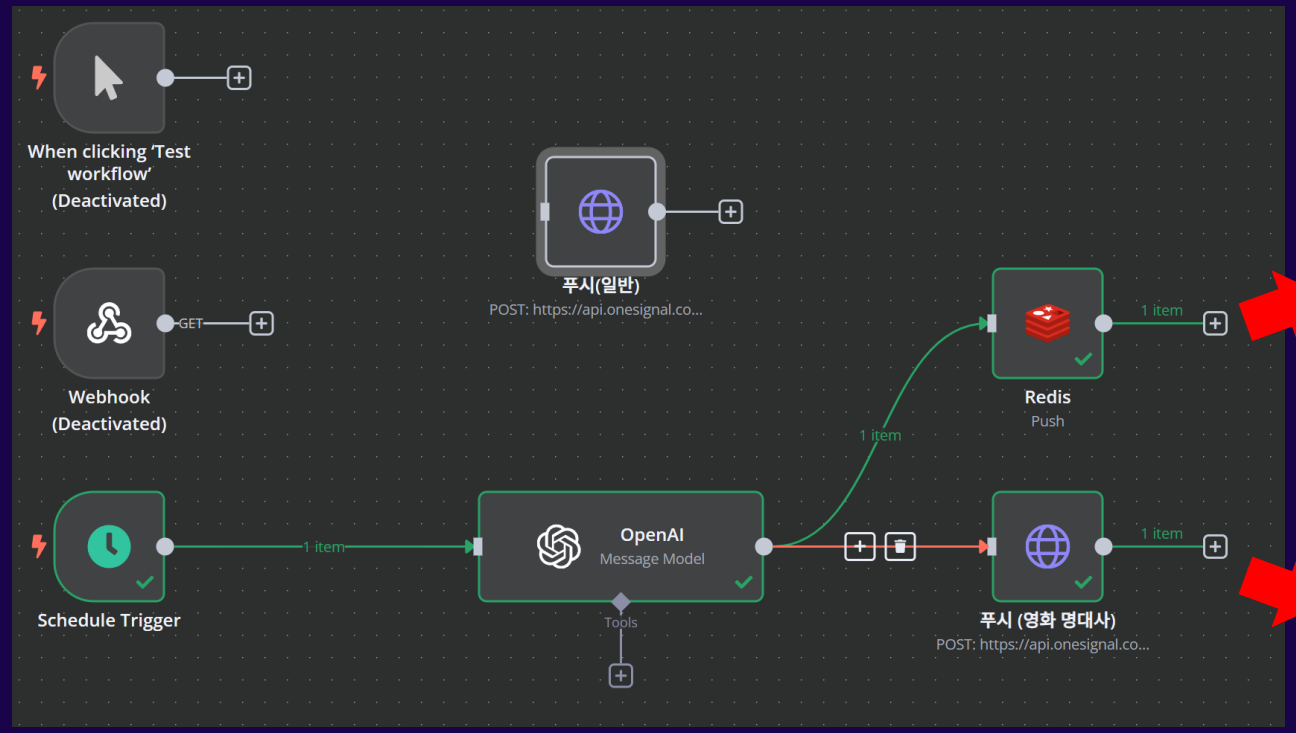
Treat your API key like a password.

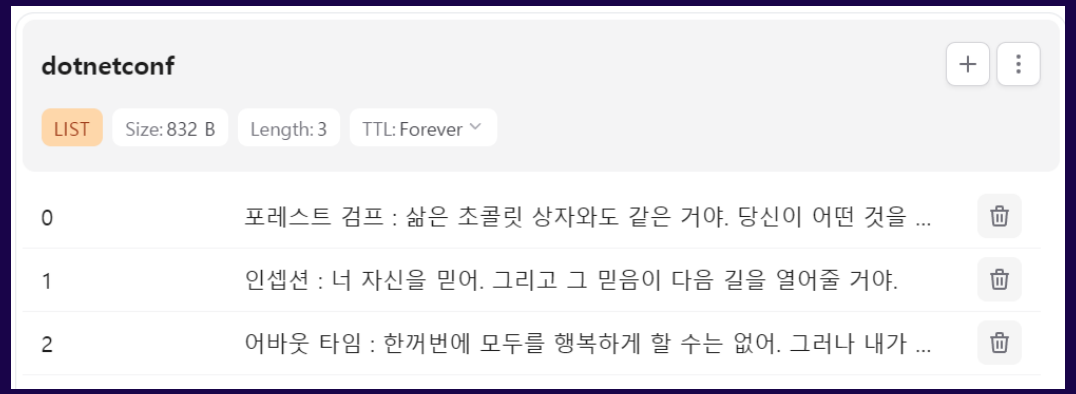
Anyone who has your REST API key will be able to send notifications from you application code. Do not share it on GitHub or anywhere else online. Rest API

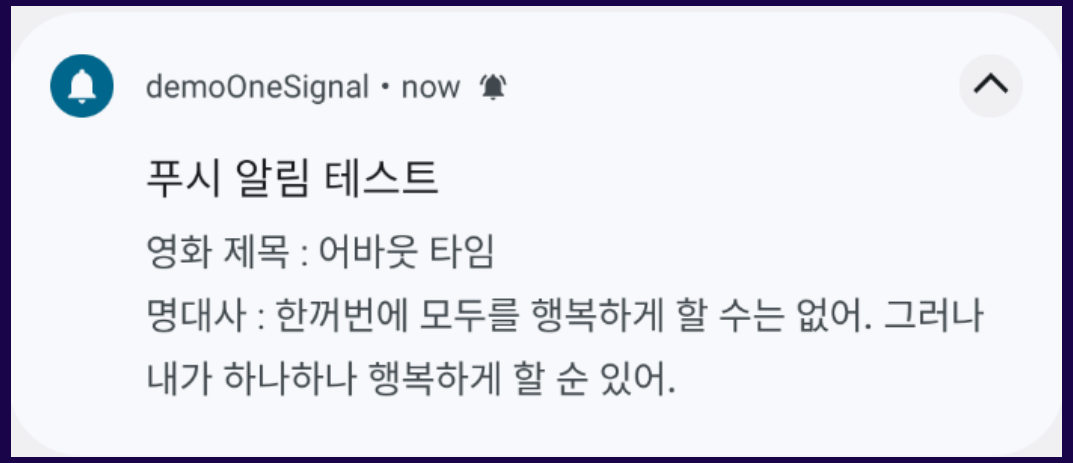
Key ID	Name	Created A
	Legacy API Key	
hobr7tddquhjmtwdywg2deyhz	apikey	2025. 1. 8

OneSignal + N8N

App Update







로우코드 자동화를 통한 푸시 발송 구현

06

AUTHENTICATION

Supabase (kakao)

Authentication

설정

항목	상태	상세 정보
카카오 로그인	ON	설정하기 가이드 확인하기 →
동의항목	설정함	설정하기 가이드 확인하기 →
간편가입	OFF	설정하기 가이드 확인하기 →
Redirect URI	설정함	설정하기 가이드 확인하기 →
연결 끊기	설정 안 함	설정하기 가이드 확인하기 →
보안 이벤트	설정 안 함	설정하기 가이드 확인하기 →

활성화 설정

상태 ON

카카오 로그인 API를 활용하면 사용자들이 번거로운 회원 가입 절차 대신, 카카오톡으로 서비스를 시작할 수 있습니다.
상태가 OFF일 때도 카카오 로그인 설정 항목을 변경하고 서버에 저장할 수 있습니다.
상태가 ON일 때만 실제 서비스에서 카카오 로그인 화면이 연결됩니다.

OpenID Connect 활성화 설정

상태 ON

카카오 로그인의 확장 기능인 OpenID Connect를 활성화합니다.
이 설정을 활성화하면 카카오 로그인 시 사용자 인증 정보가 담긴 ID 토큰을 액세스 토큰과 함께 발급받을 수 있습니다.

Redirect URI

삭제 수정

Redirect URI	https://unypozjstlkzhakzqyrt.supabase.co/auth/v1/callback

- 카카오 로그인에서 사용할 OAuth Redirect URI를 설정합니다. (최대 10개)
- REST API로 개발하는 경우 필수로 설정해야 합니다.

앱 키

플랫폼	앱 키
네이티브 앱 키	77d54e4dc0e9fe0fff7efb
REST API 키	a423263d020771
JavaScript 키	50dc5a19bab8
Admin 키	db6f3db1b882

- 네이티브 앱 키: Android, iOS SDK에서 API를 호출할 때 사용합니다.
- JavaScript 키: JavaScript SDK에서 API를 호출할 때 사용합니다.
- REST API 키: REST API를 호출할 때 사용합니다.
- Admin 키: 모든 권한을 갖고 있는 키입니다. 노출이 되지 않도록 주의가 필요합니다.

비즈니스 인증

Client Secret

토큰 발급 시, 보안을 강화하기 위해 Client Secret을 사용할 수 있습니다. (REST API인

코드	UnyBZGZ7bfYqYb	재발급
활성화 상태	사용함	설정

Supabase (kakao)

Authentication

CONFIGURATION

- Policies
- **Providers**
- Rate Limits
- Email Templates
- URL Configuration
- Hooks BETA

Google — Disabled

Kakao — Enabled

Kakao enabled

REST API Key

a423263d0207711

Client Secret Code

••••••••••••••••••••••••••••••••

Callback URL (for OAuth)

https://unypozjstlkzhakzqyrt.supabase.co/auth/v1/callback Copy

Docs Cancel Save

Supabase (kakao)

Authentication

Authentication

dev Free / dotnetconf Connect Enable branching Feedback

MANAGE

Users

CONFIGURATION

Policies

Providers

Rate Limits

Email Templates

URL Configuration

Hooks BETA

Site URL

Configure the default redirect URL used when a redirect URL is not specified or doesn't match one from the allow list. This value is also exposed as a template variable in the email templates section. Wildcards cannot be used here.

Site URL

dotnetconf://auth/callback

Cancel Save

Redirect URLs

URLs that auth providers are permitted to redirect to post authentication. Wildcards are allowed, for example, https://*.domain.com

Docs

Add URL

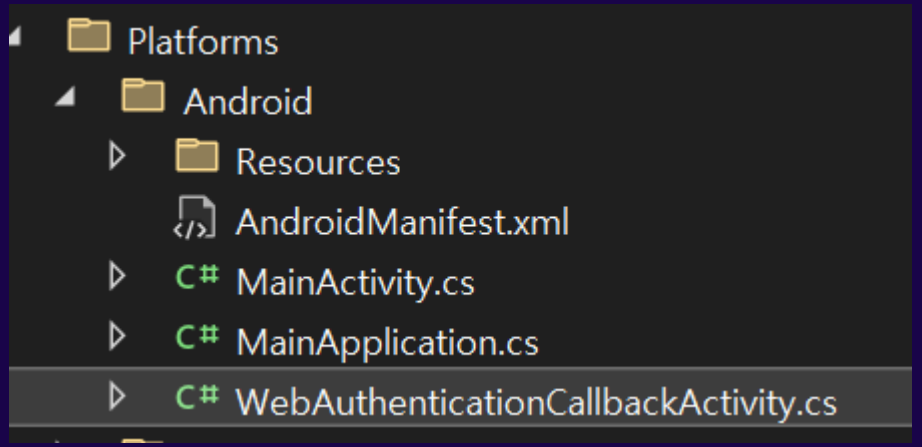
No Redirect URLs

Supabase (kakao)


```
Supabase.Client client =
    new Supabase.Client("https://unypozjstlkzhakzqyrt.supabase.co",
    "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInJlZiI6InVueXBvempzdGxremhha3pxe

var authUrl = new Uri("https://unypozjstlkzhakzqyrt.supabase.co/auth/v1/authorize?provider=kakao");
var callbackUrl = new Uri("dotnetconf://auth/callback");

try
{
    var authResult = await WebAuthenticator.AuthenticateAsync(authUrl, callbackUrl);
    if (authResult != null)
    {
        // 성공 처리
        Supabase.Gotrue.Session session = new Supabase.Gotrue.Session
        {
            AccessToken = authResult.AccessToken,
            CreatedAt = authResult.Timestamp.DateTime,
            ExpiresIn = authResult.ExpiresIn.Value.ToUnixTimeMilliseconds(),
            RefreshToken = authResult.RefreshToken,
            ProviderToken = authResult.Properties["provider_token"],
            TokenType = authResult.Properties["token_type"]
        };
        await client.Auth.SetSession(session.AccessToken, session.RefreshToken);
    }
}
```

Supabase (kakao)

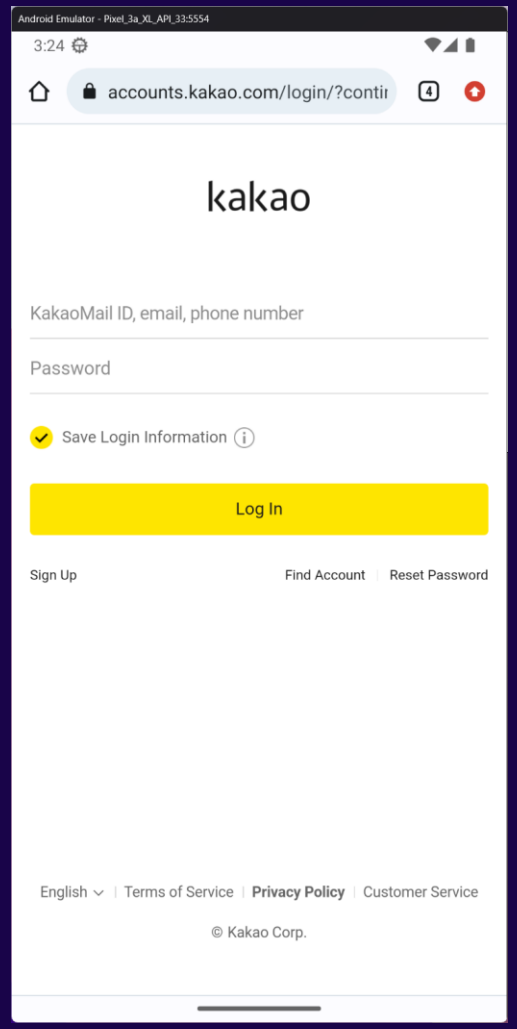
Authentication

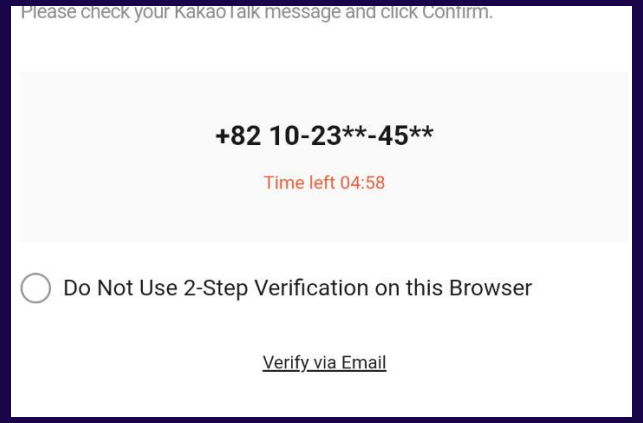


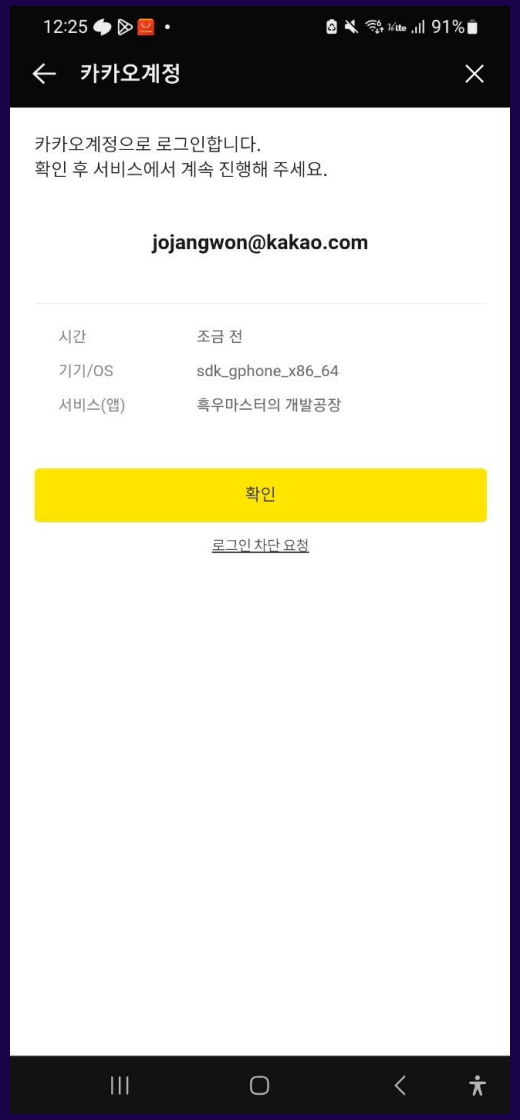

```
[Activity(NoHistory = true, LaunchMode = LaunchMode.SingleTop, Exported = true)]
[IntentFilter(new[] { Intent.ActionView },
    Categories = new[] {
    Intent.CategoryDefault,
    Intent.CategoryBrowsable
    },
    DataScheme = "dotnetconf",
    DataHost = "auth",
    DataPath = "/callback")]
참조 0개
public class WebAuthenticationCallbackActivity :
    Microsoft.Maui.Authentication.WebAuthenticatorCallbackActivity
{

}
```

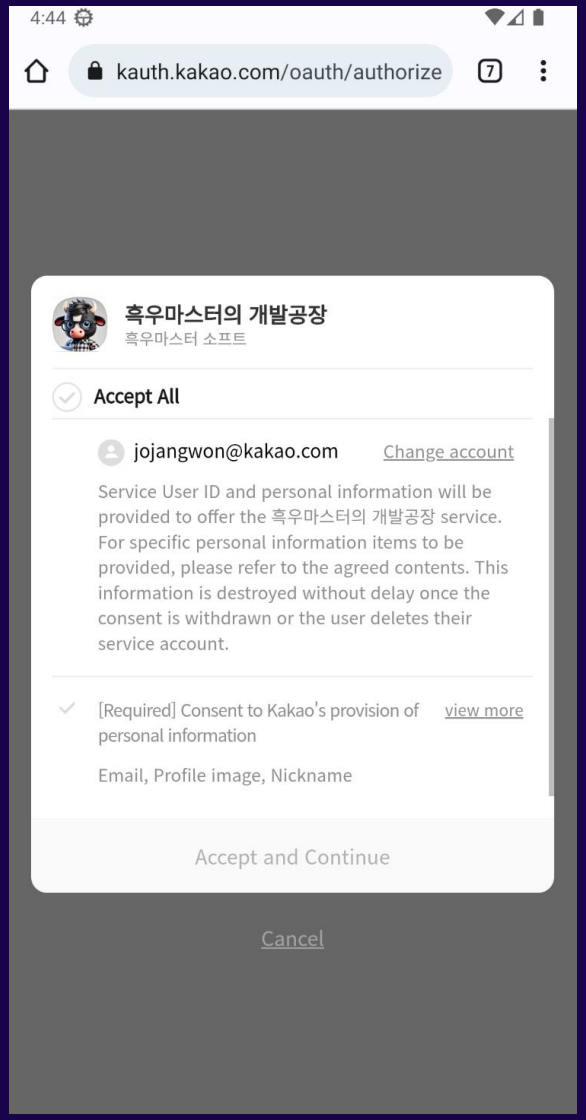
Supabase (kakao)

Authentication









Supabase (kakao)

Authentication

client	{Supabase.Client}
Auth	{Supabase.Gotrue.Client}
CurrentSession	{Supabase.Gotrue.Session}
CurrentUser	{Supabase.Gotrue.User}
AppMetadata	Count = 2
Aud	"authenticated"
BannedUntil	null
ConfirmationSentAt	null
ConfirmedAt	{System.DateTime}
CreatedAt	{System.DateTime}
Email	"jojangwon@kakao.com"
EmailConfirmedAt	{System.DateTime}
Factors	Count = 0
Id	"ecbbd7ad-8e85-4cb5-916c-0dfd9d0f3f6f"
Identities	Count = 1
InvitedAt	null
IsAnonymous	false

Supabase (kakao)

Authentication

dev Free / dotnetconf Connect Enable branching

Users

Search email, phone or UID | All users | Provider | All columns | Sorted by created at

	UID	Display name	Email	Phone	Providers
	ecbbd7ad-8e85-4cb5-916c-0dfd9d0f3f6f	조장원	jojangwon@kakao.com	-	Kakao

Auth Hooks

Use Postgres functions or HTTP endpoints to customize the behavior of Supabase Auth to meet your needs.

Add hook

No hooks configured yet

Add a new hook

- Send SMS hook
- Send Email hook
- Customize Access Token (JWT) Claims hook

Team or Enterprise Plan required

The following hooks are not available on your plan.

- MFA Verification Attempt hook
- Password Verification Attempt hook

Add Customize Access Token (JWT) Claims hook

Enable Customize Access Token (JWT) Claims hook

Hook type

Postgres
Used to call a Postgres function.

HTTPS
Used to call any HTTPS endpoint.

URL

Supabase Auth will send a HTTPS POST request to this URL each time the hook is triggered.

Secret

It should be a base64 encoded hook secret with a prefix `v1,whsec_`. `v1` denotes the signature

Auth Hook 호출을 100% 믿으면 위험!

Database Triggers

Execute a set of actions automatically on specified table events

schema: auth

Search for a trigger

Currently viewing triggers from a protected schema

The auth schema is managed by Supabase and is read-only through the dashboard.

Learn more

Name	Table	Function	Events
on_auth_user_created	users	handle_new_user	AFTER INSERT

Edit 'handle_new_user' function

Name of function: handle_new_user

Name will also be used for the function

Schema: schema: public

Tables made in the table editor will be in

Arguments

Arguments can be referenced in the function body using either names or numbe

No argument for this function

Definition

The language below should be written in plpgsql.

```
1
2   begin
3     insert into public.users (id,email)
4     -- Reference the newly created `id`
5     values (new.id,new.email);
6     return new;
7   end;
8
9   -- Return the function
10
```

auth.user는 제한적인 테이블

public.user을 만들어서 Trigger+Function으로

가입과 동시에 입력되게 만드는 것을 추천

07

CACHE & STORAGE

Secure Storage

csharp 코드 복사

```csharp
dotnet add package Microsoft.Maui.Essentials
```

csharp 코드 복사

```csharp
await SecureStorage.SetAsync("key", "value");
```

csharp 코드 복사

```csharp
var value = await SecureStorage.GetAsync("key");
```

- 안드로이드 내부저장소 Shared Preferences / iOS KeyChain
- 큰 데이터 저장하는데 부적합 1.42MB 초과 시 메모리 예외 오류
- 앱 업데이트 시 데이터 유지
- Preferences와 달리 단일 타입
- SecureStorage 보안 데이터 용도

Preferences 비 민감 단순 데이터 용도

MonkeyCache

Cache

> 2.0-beta -> This is a WIP and targets only .NET 6 using source generators instead of Newtonsoft.Json. This is cutting edge, please don't update yet unless you want to try out some crazy awesome stuff ;)

Easily cache any data structure for a specific amount of time in any .NET application.

Monkey Cache is comprised of one core package (MonkeyCache) and three providers which reference the core package as a dependency. At least one provider must be installed for Monkey Cache to work and each offer the same API (IBarrel). Depending on your existing application you may already have SQLite or LiteDB installed so these would be your natural choice. A lightweight file based Monkey Cache is also provided if you aren't already using one of these options.

Listen to our podcast Merge Conflict: Episode 76 for an overview of Monkey Cache and it's creation.

A full breakdown of performance can be found in the performance.xlsx. When dealing with a small amount of records such as inserting under 50 records, the performance difference between each provider is negligible and it is only when dealing with a large amount of records at a single time that you should have to worry about the provider type.

Azure DevOps

You can follow the full project here: https://dev.azure.com/jamesmontemagno/MonkeyCache

Build Status: Azure Pipelines succeeded

NuGets

Name	Description	NuGet
MonkeyCache	Contains base interfaces and helpers	NuGet v2.1.1
MonkeyCache.SQLite	A SQLite backing for Monkey Cache	NuGet v2.1.1
MonkeyCache.LiteDB	A LiteDB backing for Monkey Cache	NuGet v2.1.1
MonkeyCache.FileStore	A local file based backing for Monkey Cache	NuGet v2.1.1
Development Feed		MyGet

```csharp
// 파일 시스템 저장소 사용
Barrel.ApplicationId = "your_app_id";
var cachedData = Barrel.Current.Get<string>("cache_key");
if (cachedData == null)
{
    // 데이터가 없으면 새로 가져와서 캐시
    cachedData = FetchDataFromNetwork();
    Barrel.Current.Add("cache_key", cachedData, TimeSpan.FromDays(1)); // 1일 동안 캐시
}
```

```csharp
private void OnCounterClicked(object sender, EventArgs e)
{
    count++;

    if (count == 1)
        CounterBtn.Text = $"Clicked {count} time";
    else
        CounterBtn.Text = $"Clicked {count} times";


    if (MonkeyCache.SQLite.Barrel.Current.Exists("count"))
    {
        var cacheResult = MonkeyCache.SQLite.Barrel.Current.Get<int>("count");
        if (cacheResult >= 10)
        {
            App.Current.MainPage.DisplayAlert("안내", "어엇! 선생님 10개 넘었는데요?", "확인");
            return;
        }
    }
    MonkeyCache.SQLite.Barrel.Current.Add("count", count, TimeSpan.FromDays(1));
}
```

MonkeyCache


```
참조 0개
public static class CustomCache
{
    참조 0개
    public static T Get_Cache<T>(string key)
    {
        var cache = Barrel.Current.Get<T>(key);
        return cache;
    }

    참조 0개
    public static void Set_Cache<T>(string key, T value)
    {
        Barrel.Current.Add(key, value, TimeSpan.FromDays(1));
    }

    참조 0개
    public static void Remove_Cache(string key)
    {
        Barrel.Current.Empty(key);
    }
```

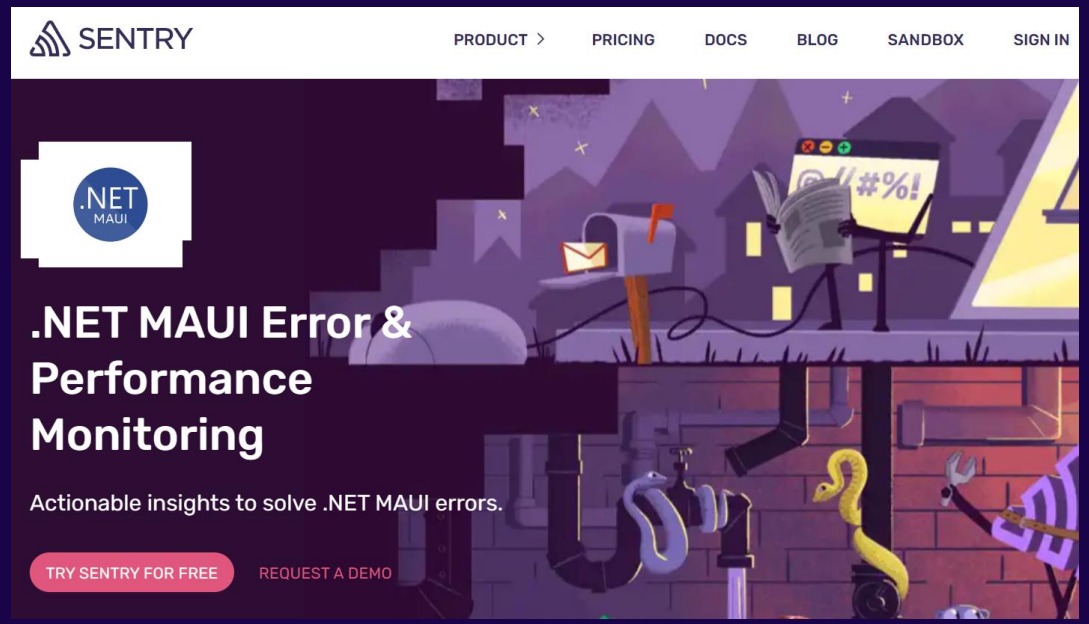
```
    참조 0개
    public static void Remove_Cache(string key)
    {
        Barrel.Current.Empty(key);
    }

    참조 0개
    public static bool Exists_Cache(string key)
    {
        return Barrel.Current.Exists(key);
    }
    - 참조
    public static void Clear_AllCache()
    {
        Barrel.Current.EmptyAll();
    }

    참조 0개
    public static void Clear_Cache(string key)
    {
        Barrel.Current.Empty(key);
    }
}
```

08

MONITORING

Sentry

Monitoring




```
dotnet add package Sentry.Maui
```

```
public static MauiApp CreateMauiApp()
{
    var builder = MauiApp.CreateBuilder();
    builder
        .UseMauiApp<App>()

        // Add this section anywhere on the builder:
        .UseSentry(options =>
        {
            // The DSN is the only required setting.
            options.Dsn = "https://<key>@sentry.io/<project>";

            // Other Sentry options can be set here.
        })

        // ... the remainder of your MAUI app setup

    return builder.Build();
}
```

Sentry

Monitoring

BREADCRUMBS

Filter By

Search breadcrumbs

TYPE	CATEGORY	DESCRIPTION	LEVEL	TIME

	navigation	App.PageAppearing
		{
		Page : MainPage
		}
			Info	15:30:42

	ui.lifecycle	Window.Created	Info	15:30:42
	ui.lifecycle	Window.Activated	Info	15:30:42

	ui.useraction	Button.Clicked
		{
		Button.Name : HelloButton
		}
			Info	15:30:54

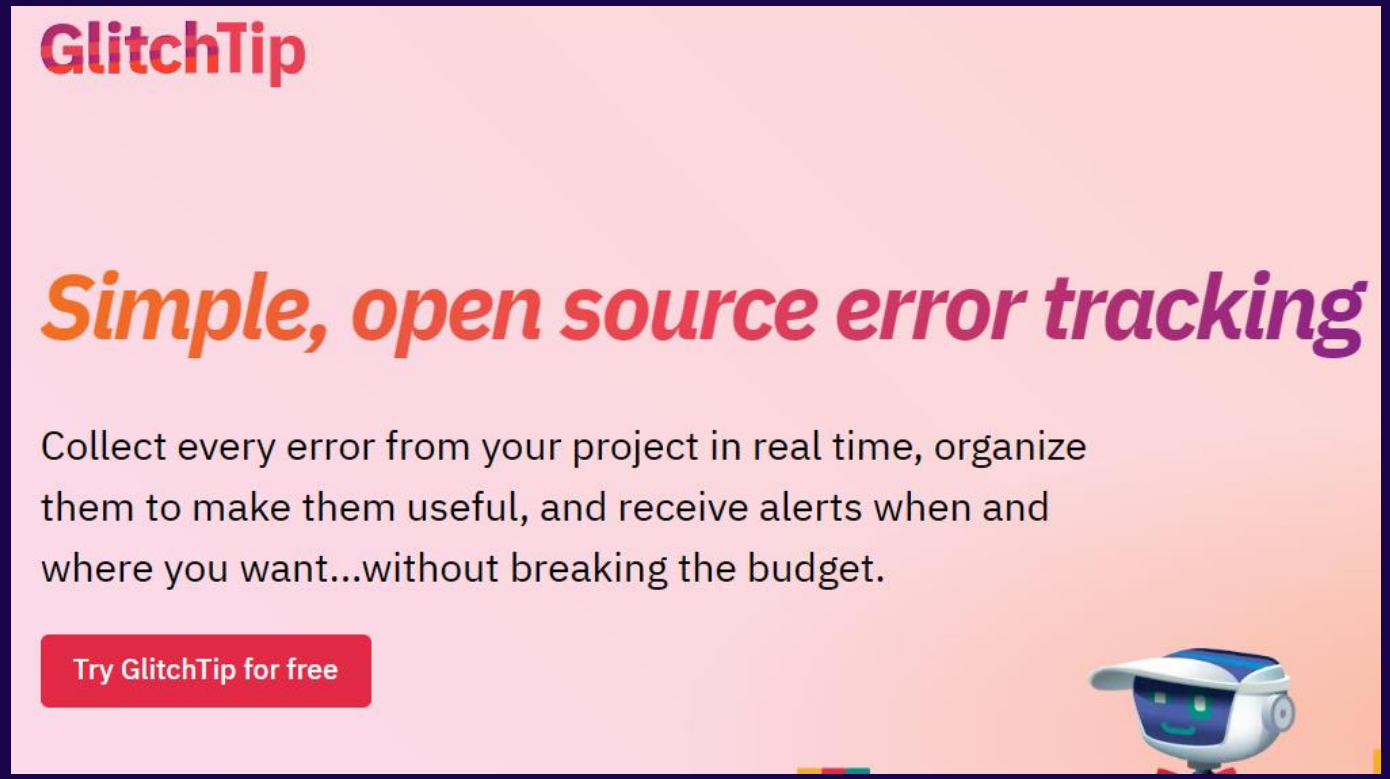
	exception	System.NullReferenceException: Object reference not set to an instance of an object.	Error	15:30:54

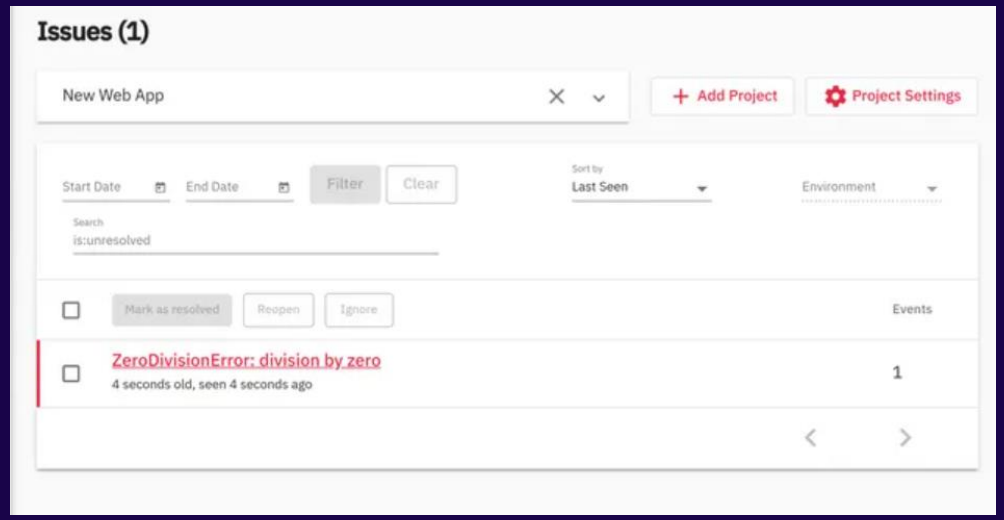
.NET
Version: 6.0.6

Android
Version: 12

SAMSUNG
SM-G998U
Arch: arm64-v8a

device	SM-G998U	device.family	SM-G998U	environment	production	handled	no
mechanism	UnhandledExceptionRaiser	os	Android 12	os.name	Android	os.root	





Sentry 대체 오픈소스

(Sentry SDK 그대로 사용 가능)

.NET

Analytics for .NET MAUI Apps

Privacy-First. Simple. Real-Time.

Aptabase is an Open Source alternative to Google Firebase Analytics for .NET MAUI Apps

1. Register

Signing up takes less than a minute.

Dashboard →

2. Instrument

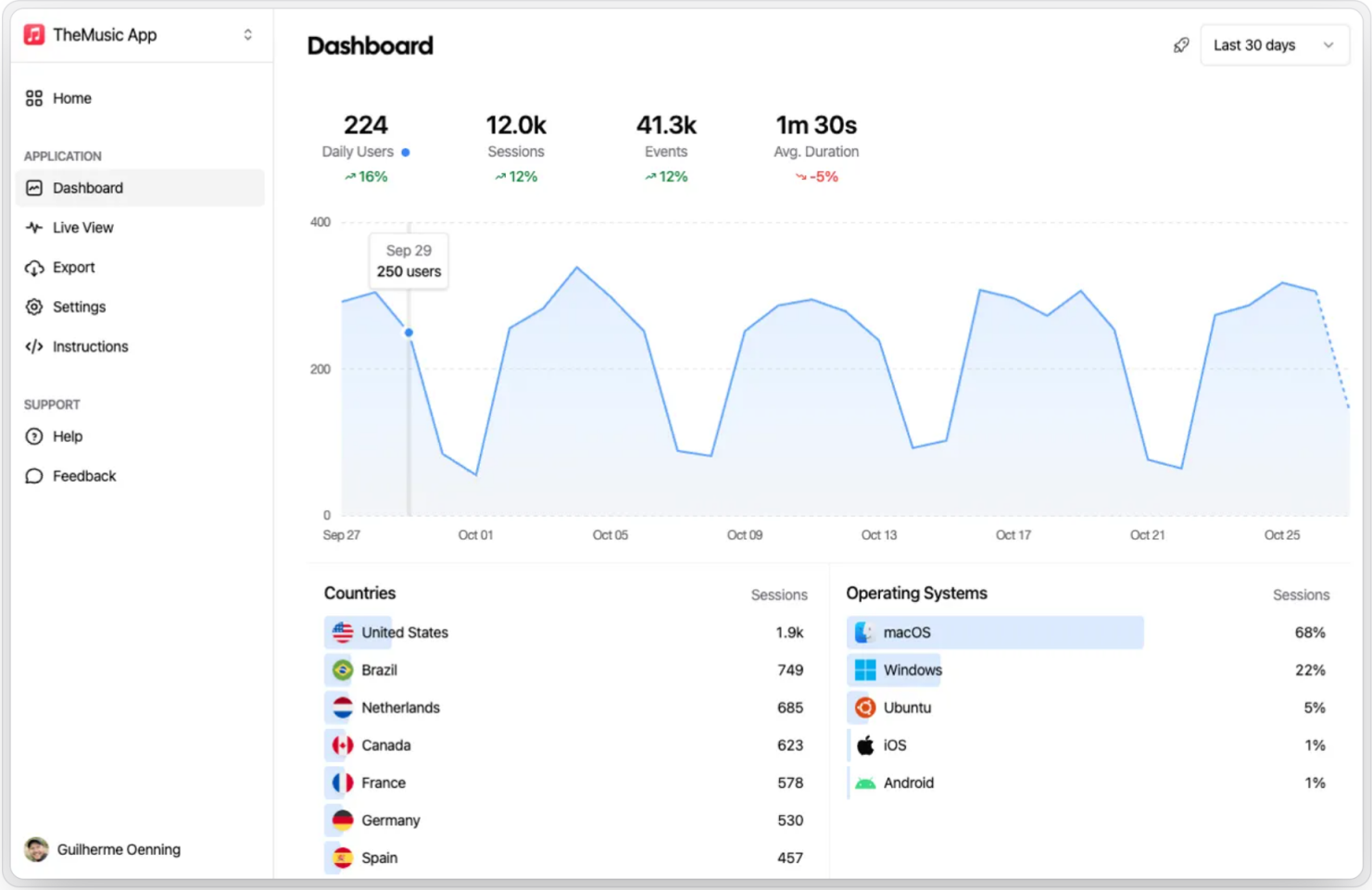
Use the .NET MAUI SDK for Aptabase to instrument your app. You're in total control of what events to track.

Here's an example 👇

```
_aptabase.TrackEvent("play_music", new() {
    { "name", "Here comes the sun" }
});
```

3. Explore

That's it! You'll now see the data on your dashboard as the events come through from your app.



Aptabase

Monitoring

Integrations

Here is a list of frameworks that you can use with Aptabase

Electron	Tauri	Native iOS (Swift)
Native Android (Kotlin)	Flutter	NativeScript
.NET MAUI	Unreal Engine	Unity
React Native	Web App	Next.js
Angular	Browser Extensions	

```
_aptabase.TrackEvent("play_music", new() {
    { "name", "Here comes the sun" }
});
```

Aptabase Self-Hosting Guide

This guide will help you get started with self hosting Aptabase. You can alternatively use our Cloud Hosting for a zero maintenance and fully managed solution.

Prerequisites

All you need is to have a recent version of Docker as this guide uses Docker Compose.

How-to

1. Clone this repository

```
git clone https://github.com/aptabase/self-hosting
```

2. Modify the `docker-compose.yml` file to your liking, but especially the sections with comments as the default values are just dummy examples.
3. Run the following command to start the services:

```
docker-compose up -d
```

4. That's it! You can now access Aptabase at `http://localhost:8000` 🎉

Note: By default, Aptabase is not configured to send email via SMTP . You can either see the emails in the logs or add the relevant SMTP variables in the `docker-compose.yml` file. A list of all Environment Variables is available here.

> If that's not working, please check the logs with `docker-compose logs -f` and make sure there are no errors.

First Login / Account

There is no default login. Once you are able to access your self-hosted instance, you need to create/register a new account. After registering, the activation link that is required to activate it will be in the container logs. Once you follow that link, your new account will be active, and you will be able to access Aptabase in its entirity.

One-Click Deployment

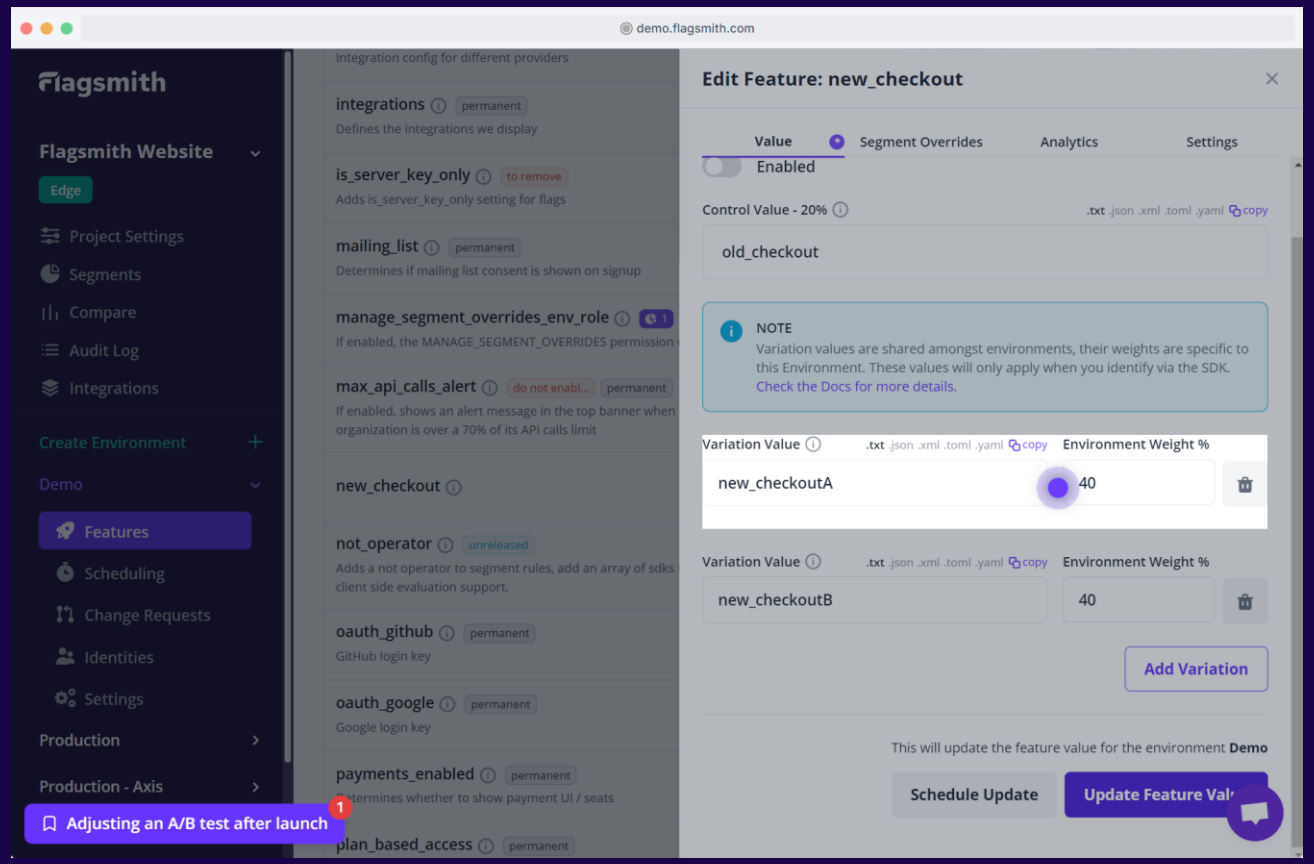
Deploy on RepoCloud

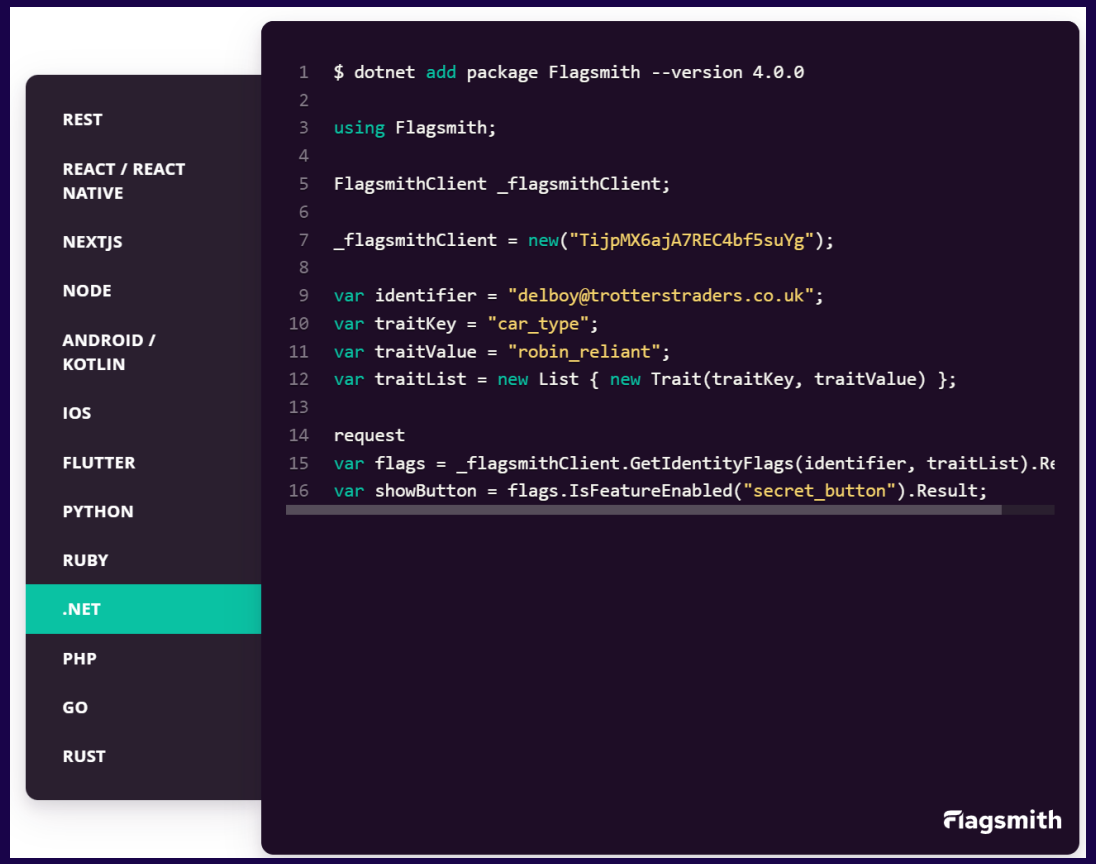
09

FEATURE FLAGS

Flagsmith

Feature Flags





10

DEEPLINK

Dynamic Link Deprecation

DeepLink

Firebase > Support

Was this helpful?

Dynamic Links Deprecation FAQ

Send feedback

Note: Firebase Dynamic Links is *deprecated* and should not be used in new projects. The service will shut down on August 25, 2025.

We launched Firebase Dynamic Links over 7 years ago to make URLs more powerful - for example, dynamically changing the destination of a link based on run-time conditions.

Over the years, the web and mobile ecosystems have evolved with technologies such as A
Links, and App Clips and have made user journeys across apps and the web more s
believe you and your users will benefit more from adopting these technologies directly an
However, alongside these new changes, the original native APIs that Firebase Dynamic Li
challenges.

For example, ecosystem changes impacted our ability to provide a consistently stable exp
features - giving app users a smooth transition into the app post-install, regardless of plat

Instead of continuing to support a less than ideal experience, we have decided to sunset
resources on solving other developer pain points.

We understand that this change will require your time to evaluate and adopt alternative so

We are continuing to evolve Firebase to meet your needs as the ecosystem continues to e
features and updates across Firebase products, and are committed to helping you exc

5만 MAU 미만

연 320만원

268,650원/월

도입 문의하기

사용할 수 있는 주요 기능

- 트래킹 링크 생성 최대 5,000개 제공
- QR 코드 생성 최대 5,000개 제공
- 링크 브랜딩 제공 ✅
- 링크 클릭 제공 무제한 ✅

10만 MAU 미만

연 480만원

403,650원/월

도입 문의하기

사용할 수 있는 주요 기능

- 트래킹 링크 생성 최대 10,000개 제공
- QR 코드 생성 최대 10,000개 제공
- 링크 브랜딩 제공 ✅
- 링크 클릭 제공 무제한 ✅

10만 MAU 이상

별도 문의

에어브릿지와 이야기 나누어 보세요.

플랜 문의하기

사용할 수 있는 주요 기능

- 트래킹 링크 생성 10,000개 이상
- QR 코드 생성 10,000개 이상
- 링크 브랜딩 제공 ✅
- 링크 클릭 제공 무제한 ✅

2025년 8월 25일 서비스 종료

Open Source

DeepLink

Deep Link Generator

Email Address *

your@email.com

Email verification is required before short link generation

Default Fallback URL *

https://myapp.com

Users will be redirected here if no platform-specific deep link is available

Hide Advanced Settings

iOS Configuration

iOS Deep Link

myapp://path

iOS Fallback URL

https://apps.apple.com/app/myapp

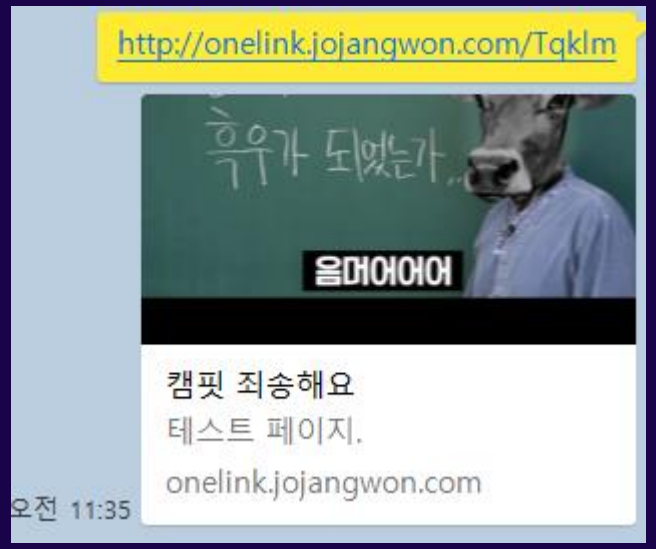
Android Configuration

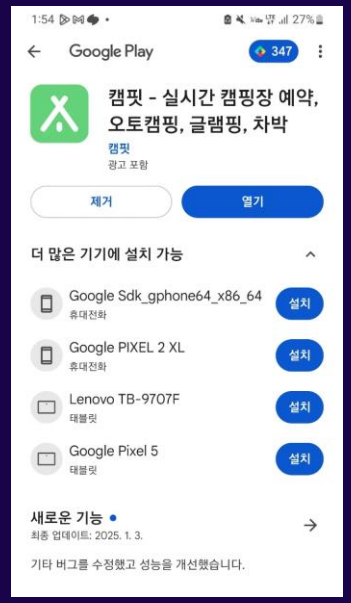
Android Deep Link

myapp://path

Android Fallback URL

https://play.google.com/store/apps/details?id=com.myapp



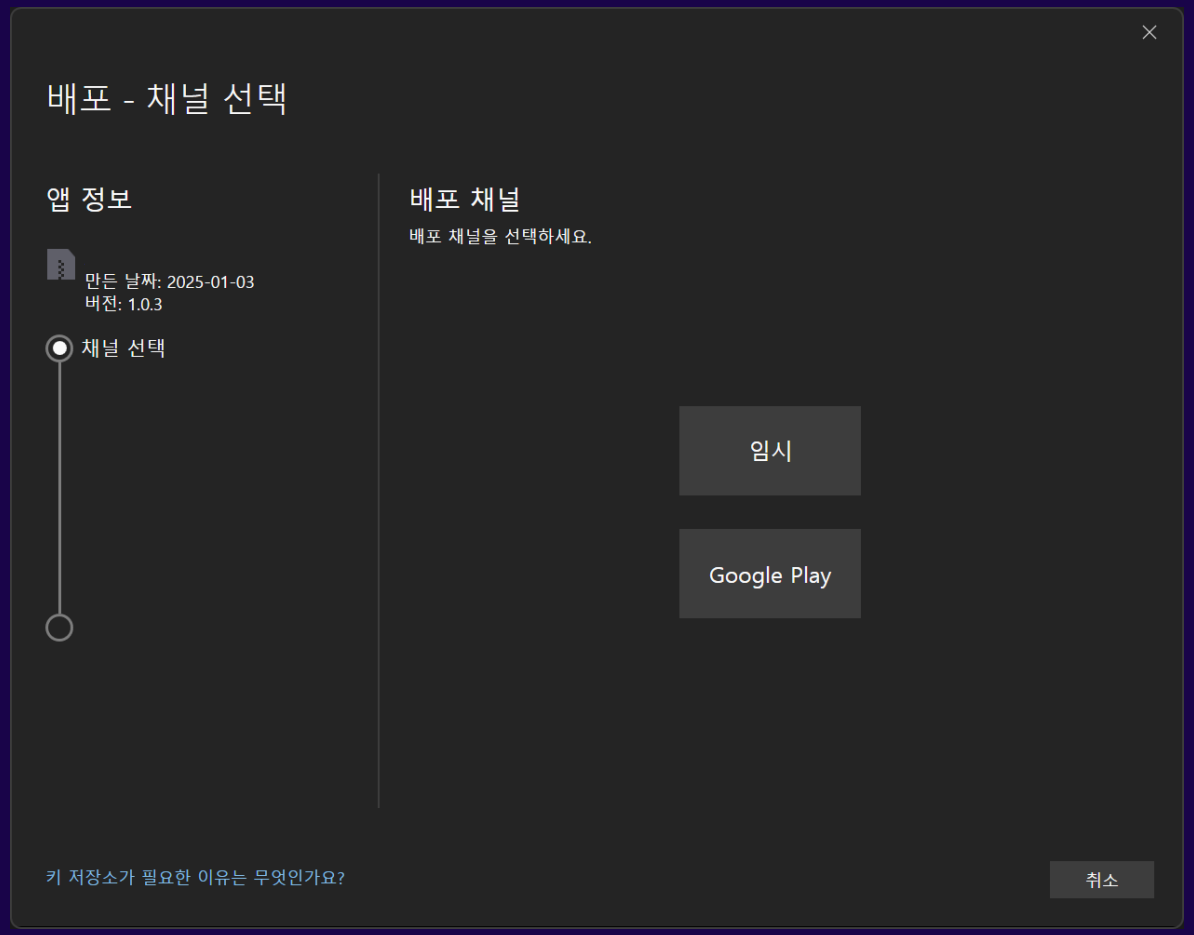


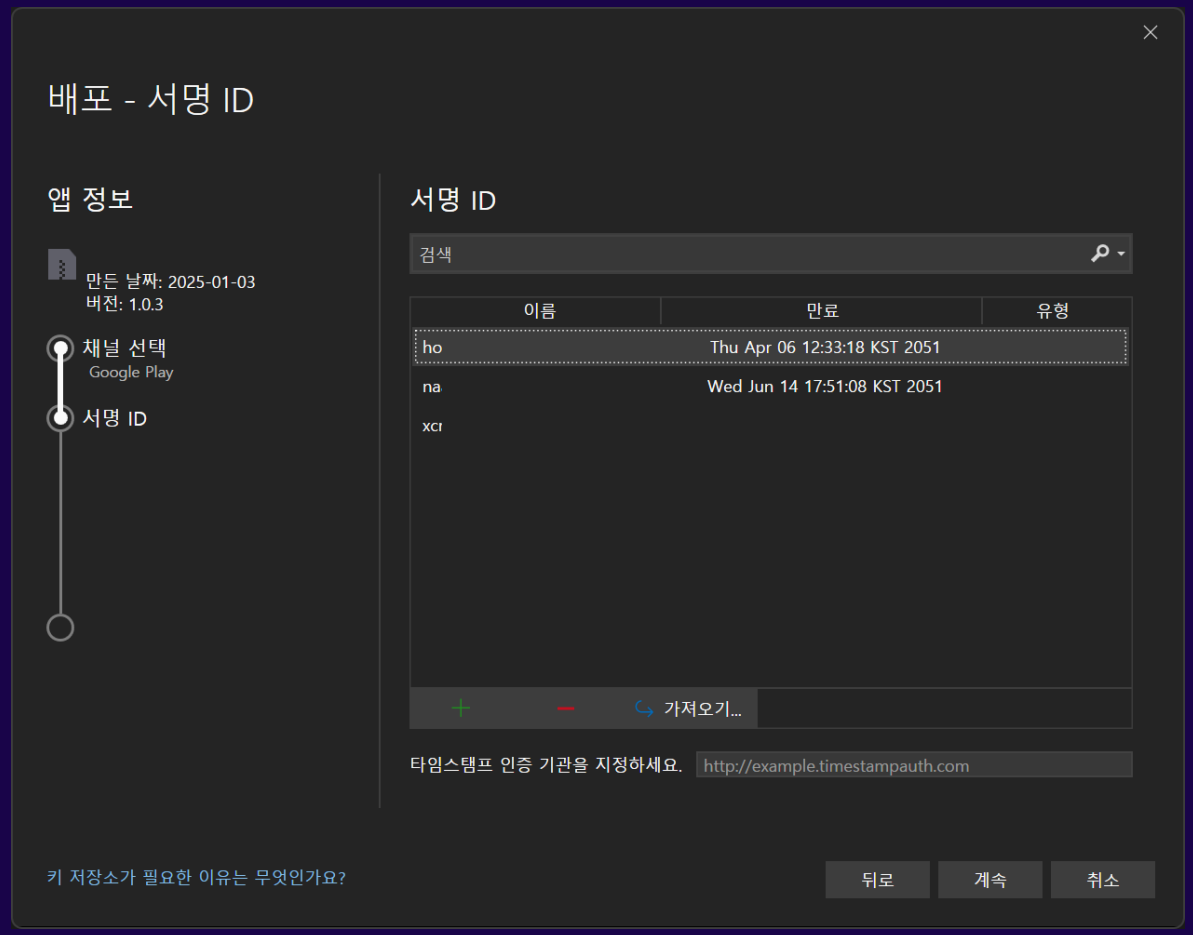
11

DEPLOY

Google Developer API

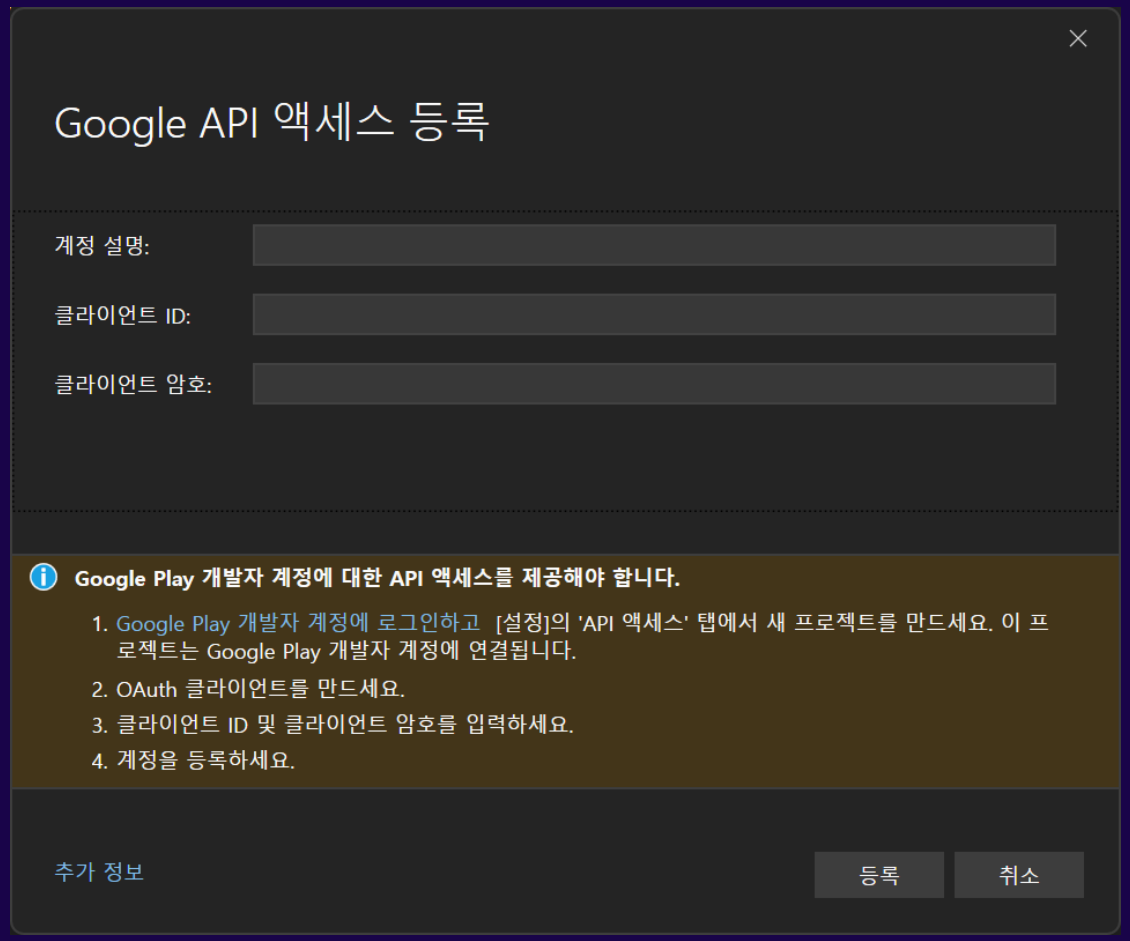
Deloy

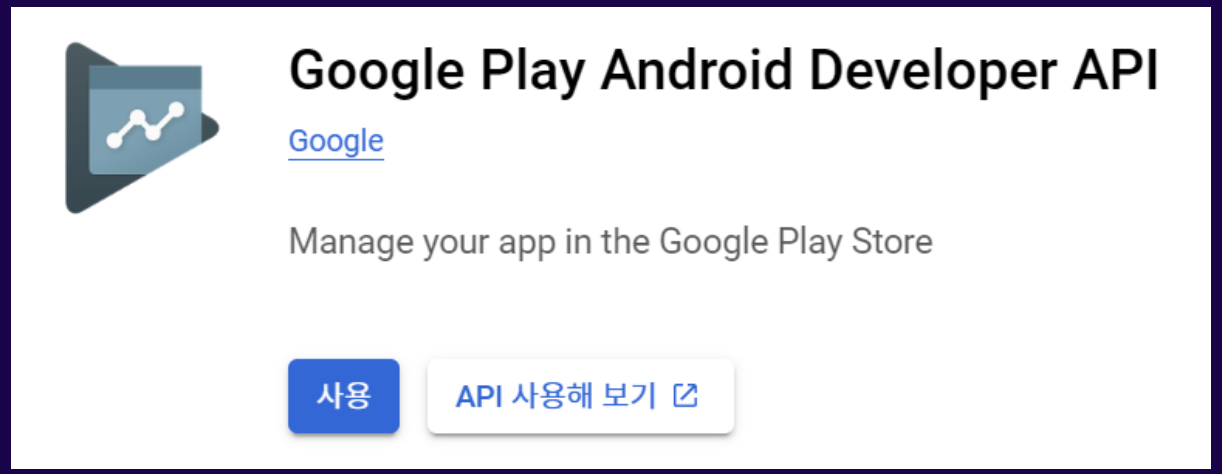


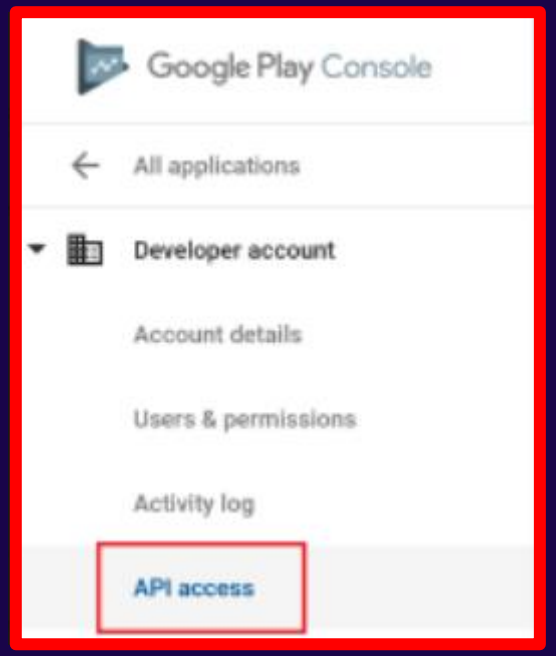


Google Developer API

Deloy



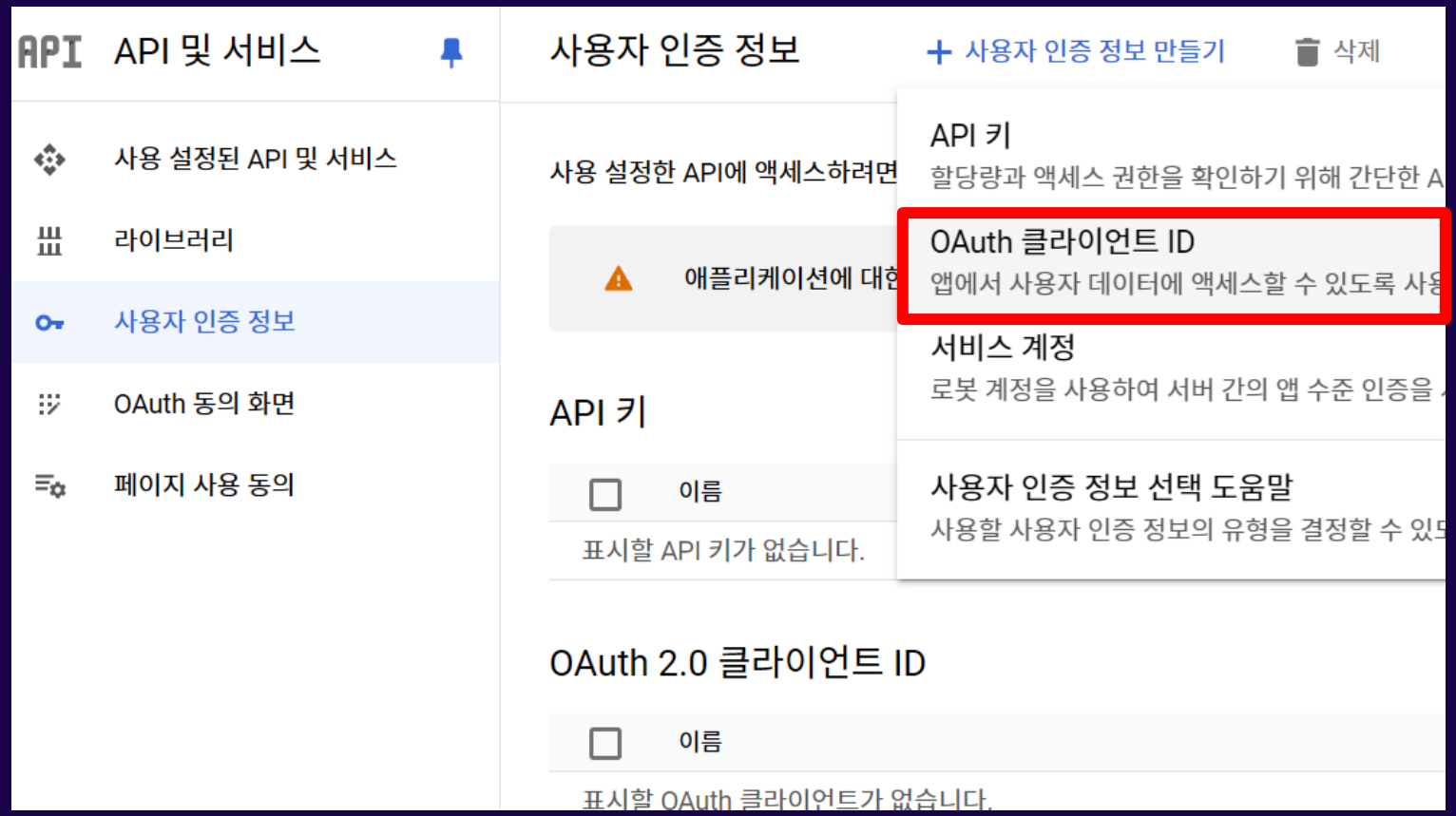




구글 개발자 콘솔에 없음 !

Google Developer API

Deloy



OAuth 클라이언트 ID 만들기
클라이언트 ID는 Google OAuth 서버에서 단일 앱을 식별하는 데 사용됩니다. 앱이 여러 플랫폼에서 실행되는 경우 각각 자체 클라이언트 ID가 있어야 합니다. 자세한 내용은 OAuth 2.0 설정을 참조하세요. OAuth 클라이언트 유형을 자세히 알아보세요.
애플리케이션 유형 *
웹 애플리케이션
Android
Chrome 확장 프로그램
iOS
TV 및 입력 제한 기기
데스크톱 앱
Universal Windows Platform(UWP)

Google Developer API

Deloy

데스크톱의 클라이언트 ID

삭제

이름 *
데스크톱 클라이언트 1

OAuth 2.0 클라이언트의 이름입니다. 이 이름은 콘솔에서 클라이언트를 식별하는 용도로만 사용되며 최종 사용자에게 표시되지 않습니다.

참고: 설정이 적용되는 데 5분에서 몇 시간이 걸릴 수 있습니다.

저장 취소

Additional information

클라이언트 ID	412424242424
생성일	2025년 1월 3일 PM 10시 27분 24초 GMT+9

클라이언트 보안 비밀번호

클라이언트 보안 비밀번호를 변경하는 중인 경우 다운타임 없이 수동으로 순환할 수 있습니다. 자세히 알아보기

클라이언트 보안 비밀번호	41242424244
생성일	2025년 1월 3일 PM 10시 27분 24초 GMT+9
상태	사용 설정됨

+ ADD SECRET

Google 계정으로 로그인

daroom manager에서 Google 계정에 대한 액세스를 요청합니다.

이 액세스를 허용하면 daroom manager 앱에서 다음 작업을 할 수 있습니다.

- Google Play 개발자 계정 보기 및 관리. 자세히 알아보기

daroom manager 앱을 신뢰할 수 있는지 확인

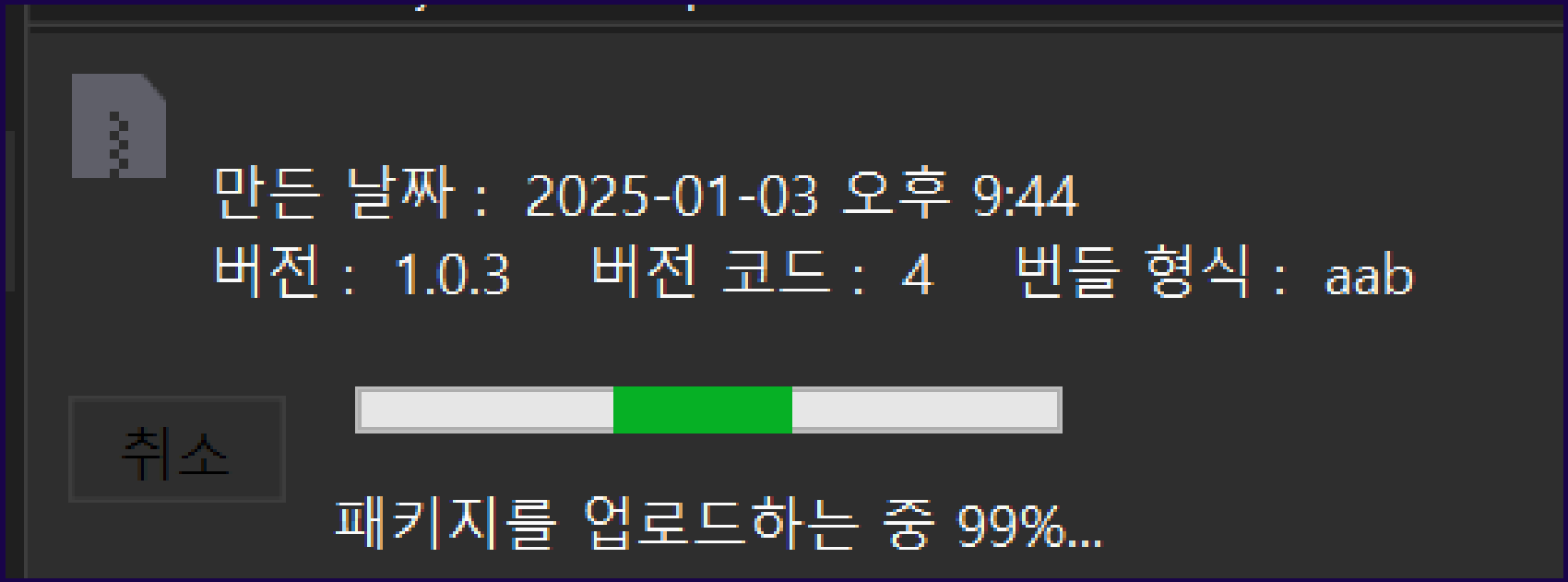
daroom manager의 개인정보처리방침 또는 서비스 약관 링크가 표시되지 않는 이유 알아보기

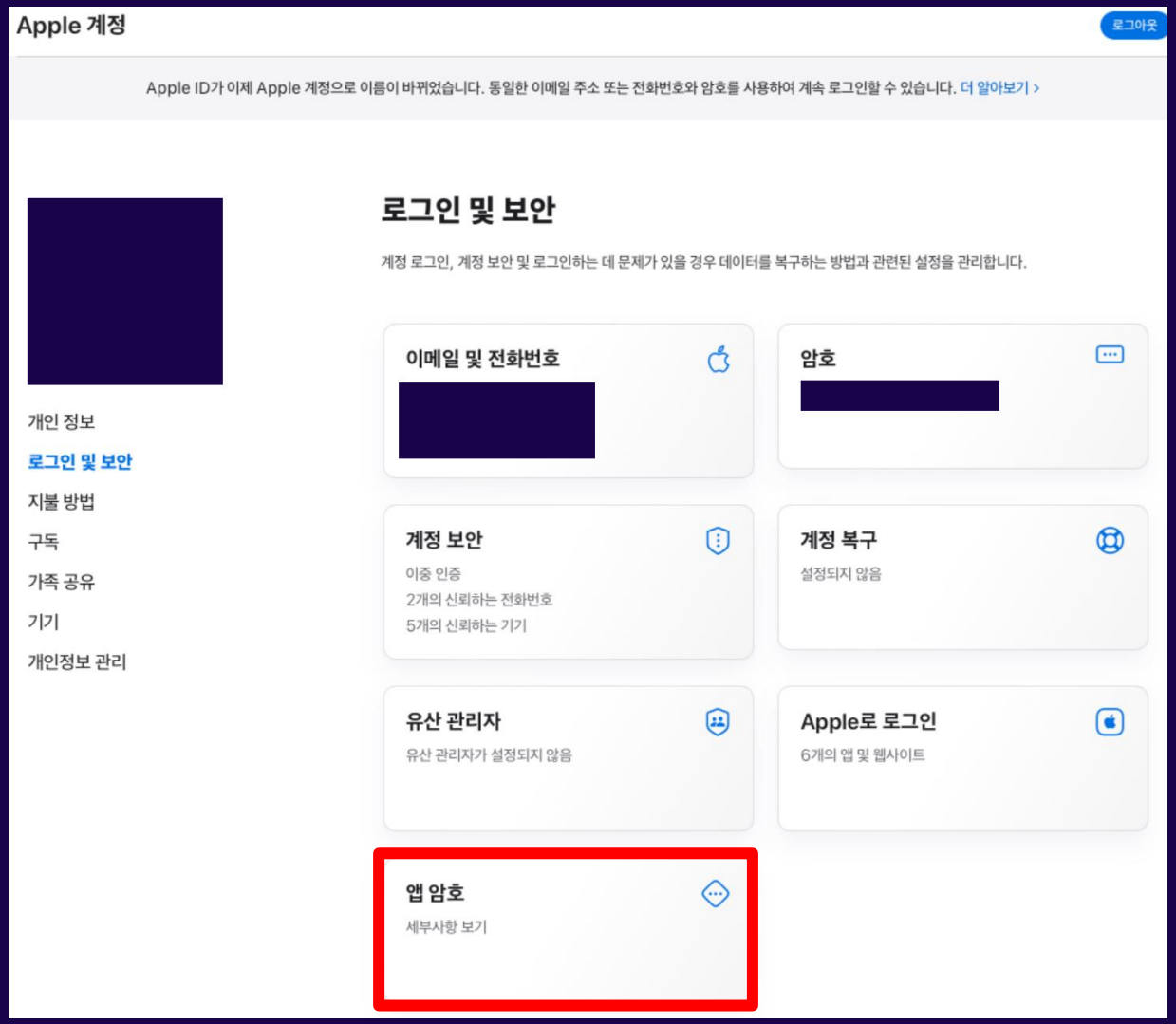
daroom manager의 개인정보처리방침 및 서비스 약관을 검토하여 daroom manager에서 내 데이터를 처리하고 보호하는 방법을 알아보세요.

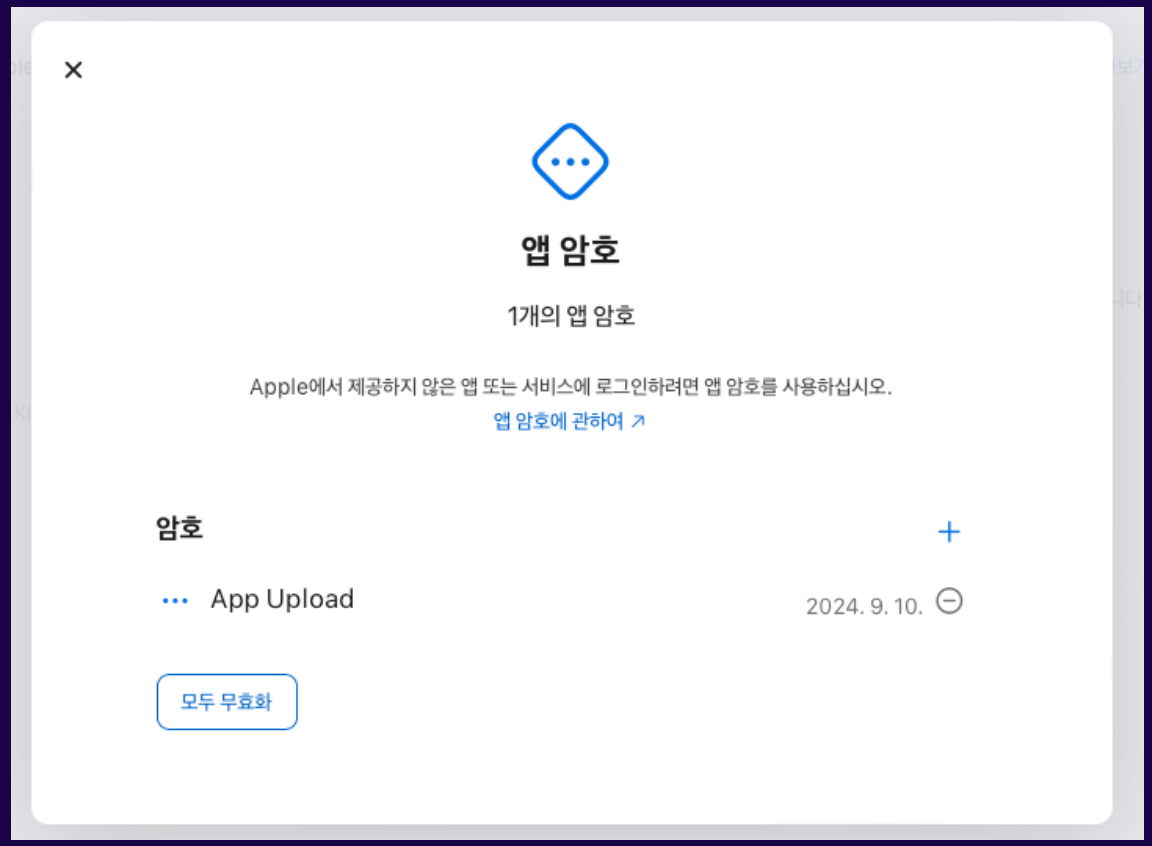
Google 계정에서 언제든지 변경할 수 있습니다.

Google이 데이터를 안전하게 공유하는 방법을 알아보세요.

취소 계속


만든 날짜 : 2025-01-03 오후 9:44
버전 : 1.0.3 버전 코드 : 4 번들 형식 : aab
취소
패키지를 업로드하는 중 99%...





https://appleid.apple.com/account/manage

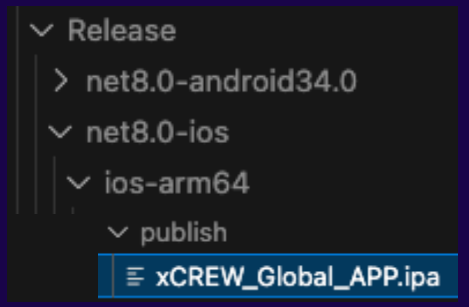
Visual Studio Code iOS CLI

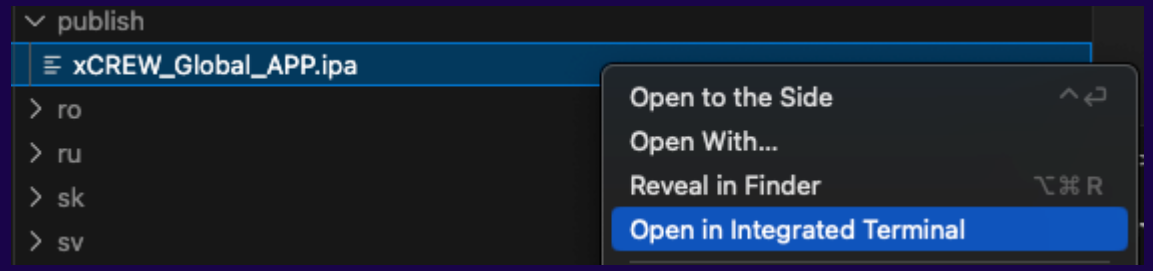
Deloy

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

honeyhead@jojang-won-ui-Macmini xCREW_Global_APP %
```

dotnet publish "프로젝트 경로/프로젝트명.csproj" -f net8.0-ios -c Release -p:ArchiveOnBuild=true -p:RuntimeIdentifier=ios-arm64 -p:CodesignKey="iPhone Distribution 키 이름" -p:CodesignProvision="코드사인 프로비전 이름"

1. Visual Studio Code에서 하단 작업 환경 상태를 Debug->Release로 변경
2. 터미널을 열고 다음과 같이 입력 (버전이 8.0이 아닌 경우 교체)






```
xcrun altool --upload-app --type ios --file "ipa 파일명.ipa" -u "배포 이메일" -p "배포 패스워드"
```

1. Release에 파일이 정상적으로 생겼다면 해당 폴더로 가서 새로운 터미널 열기
2. 위 코드 입력

   (사실 터미널 안 열고 해도 되는데 경로 쓰기 귀찮으니까 ..ㅎㅎ)

여러분은 혼자가 아닙니다

.NET MAUI 단톡방

https://open.kakao.com/o/g8IzoOm

닷넷데브

https://forum.dotnetdev.kr

유튜브

흑우마스터의 야매코딩

Q&A

Thank you