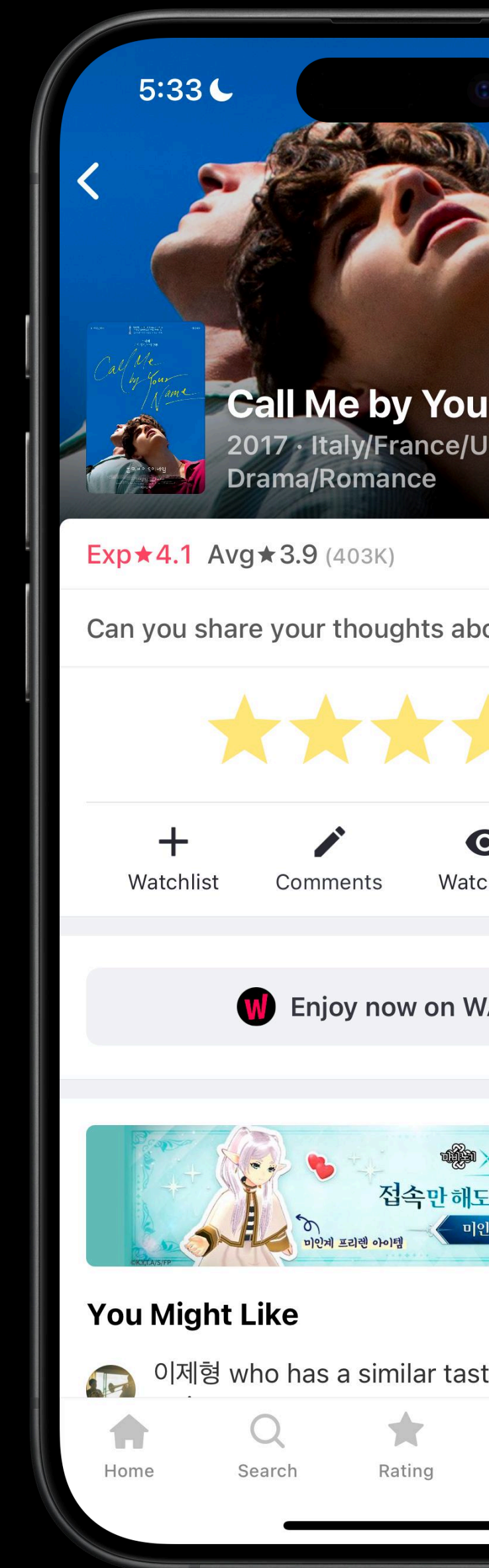
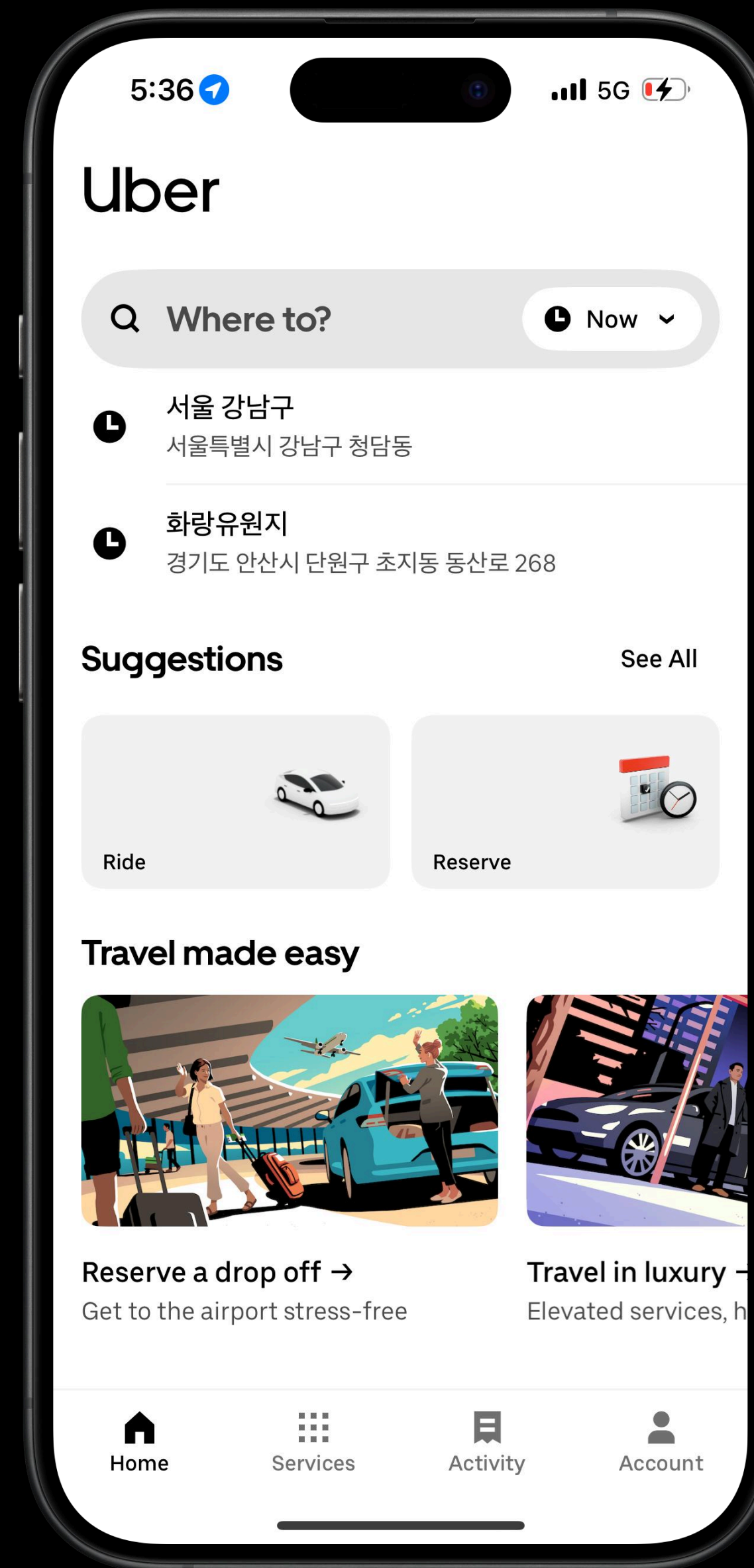
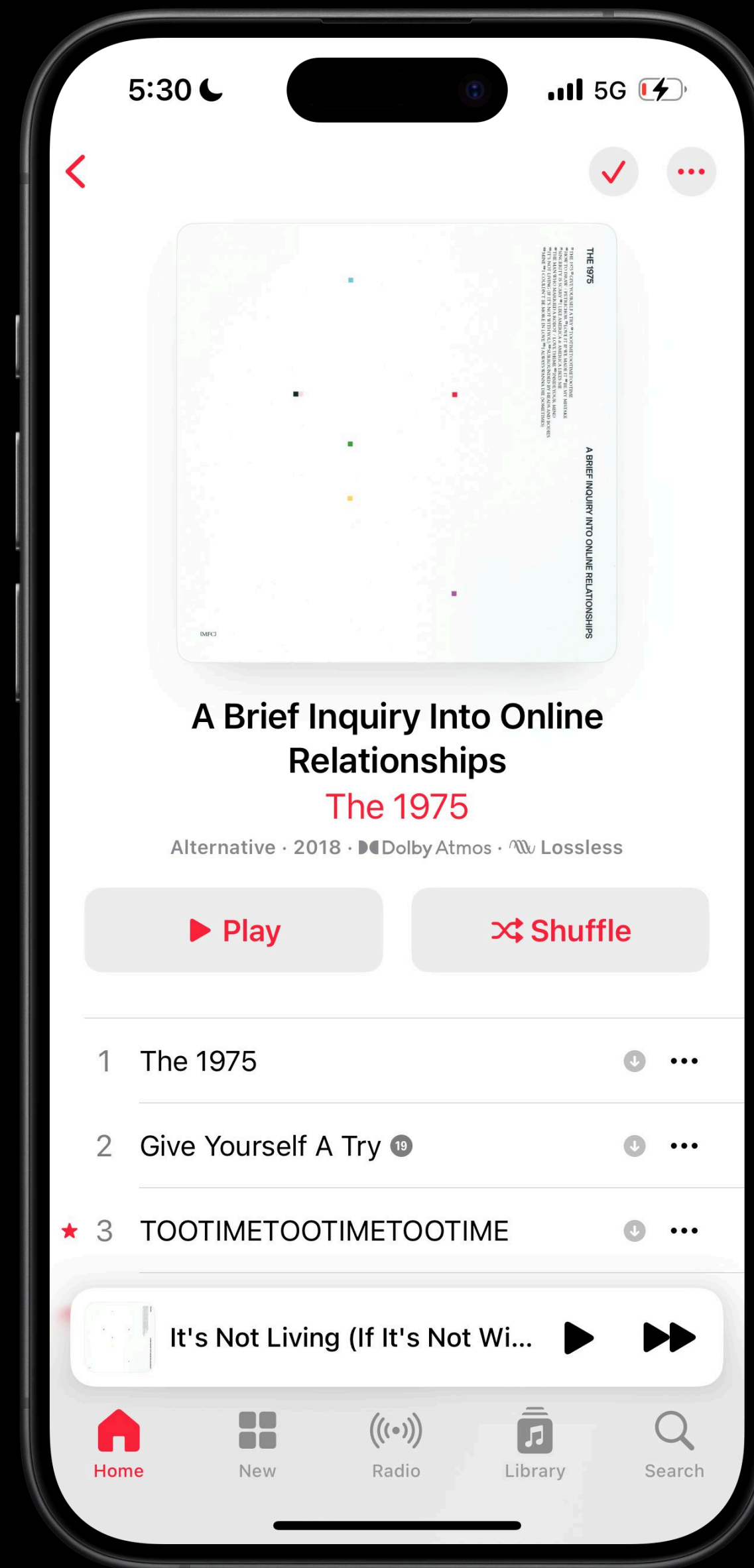
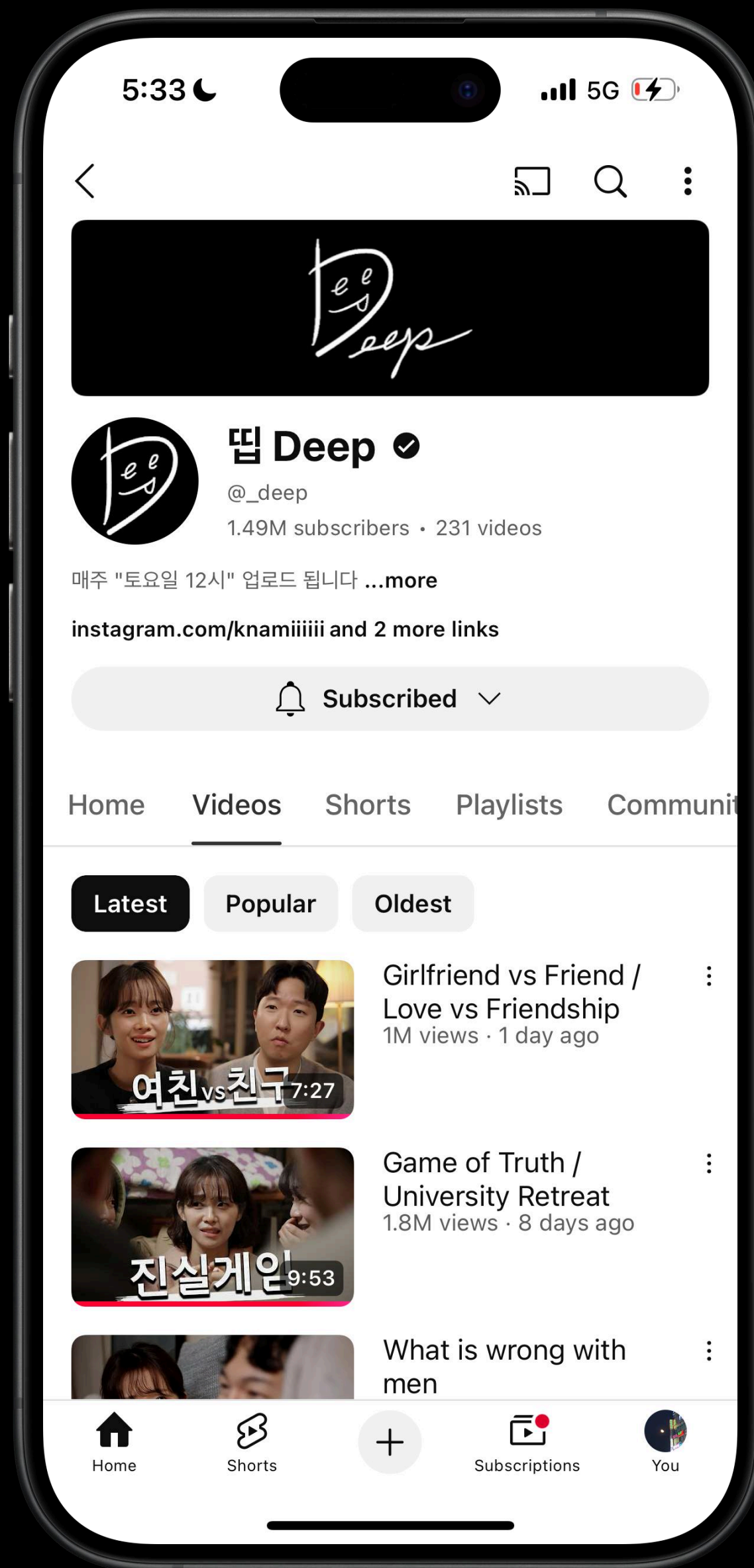
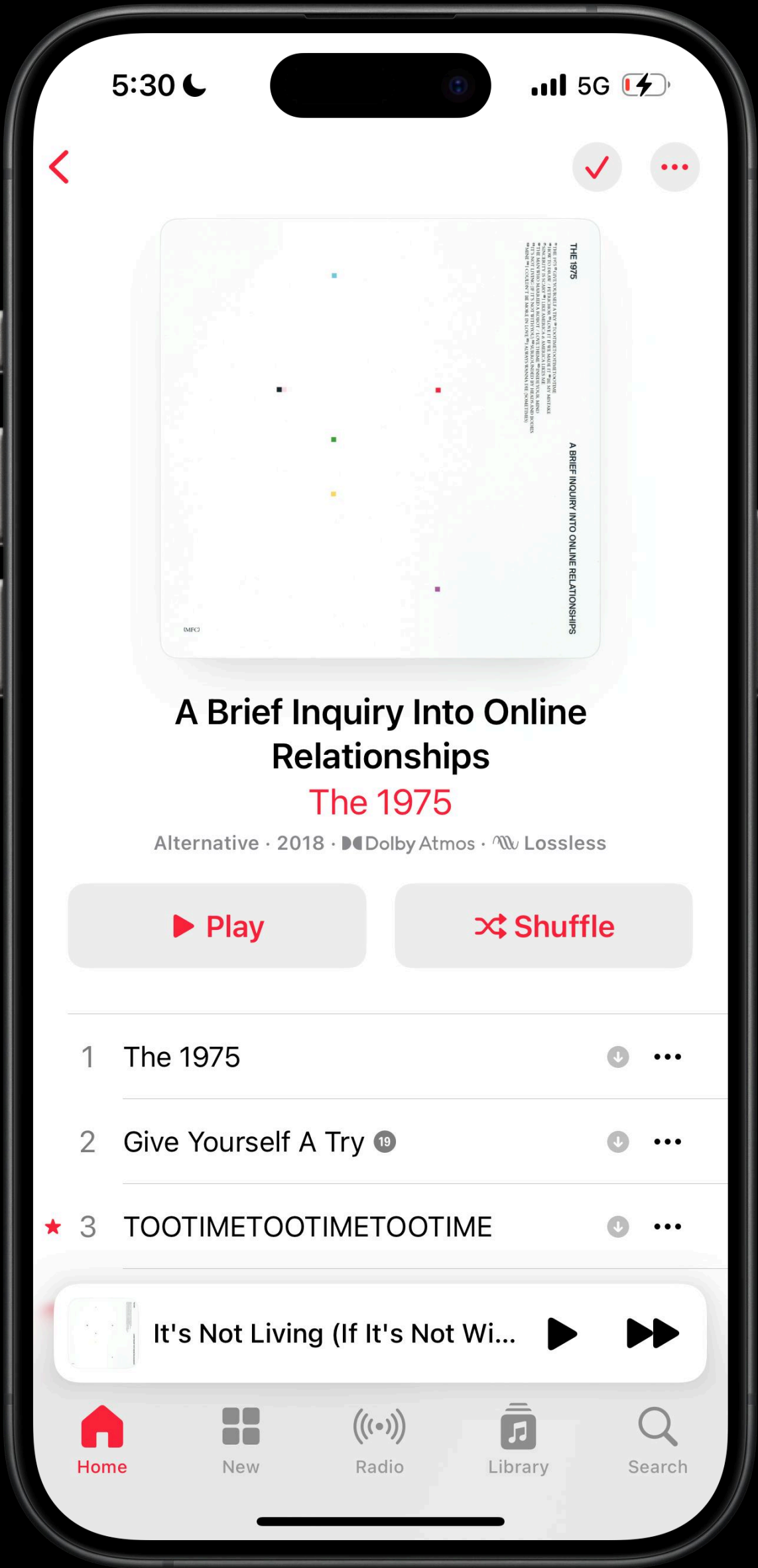
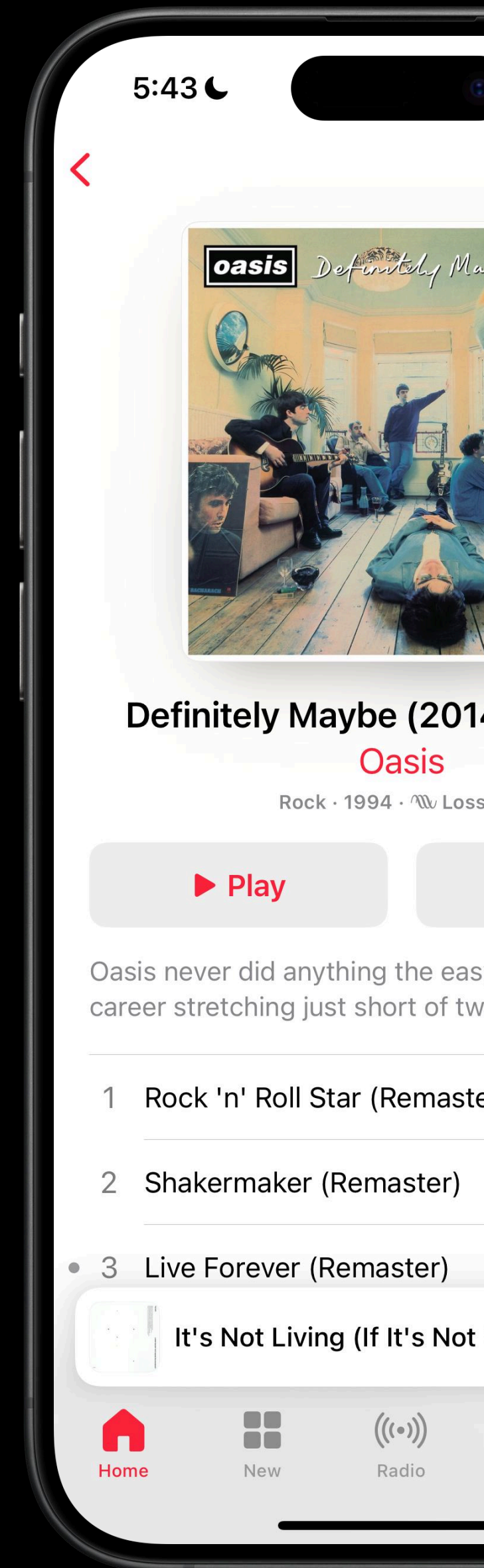
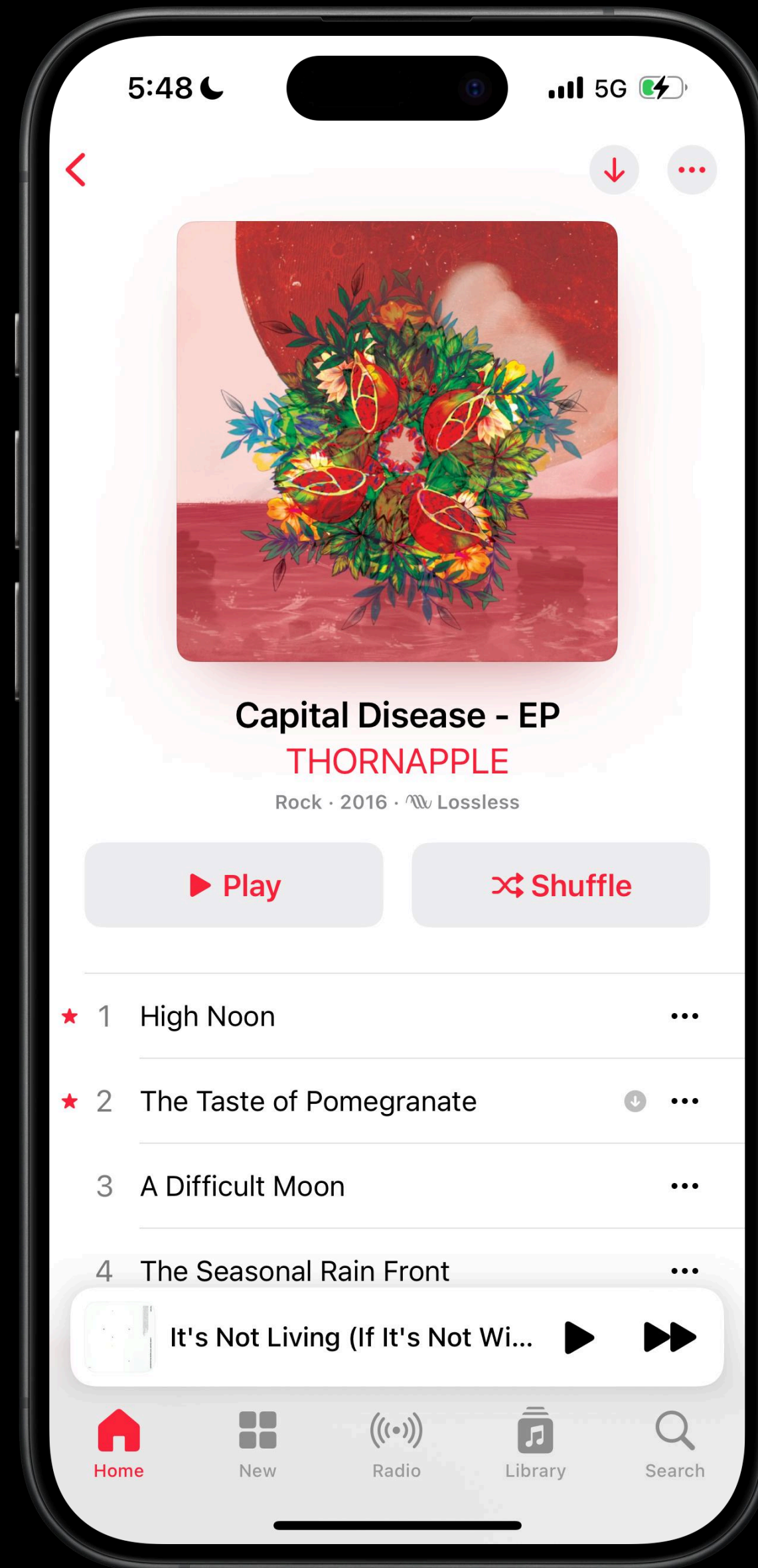
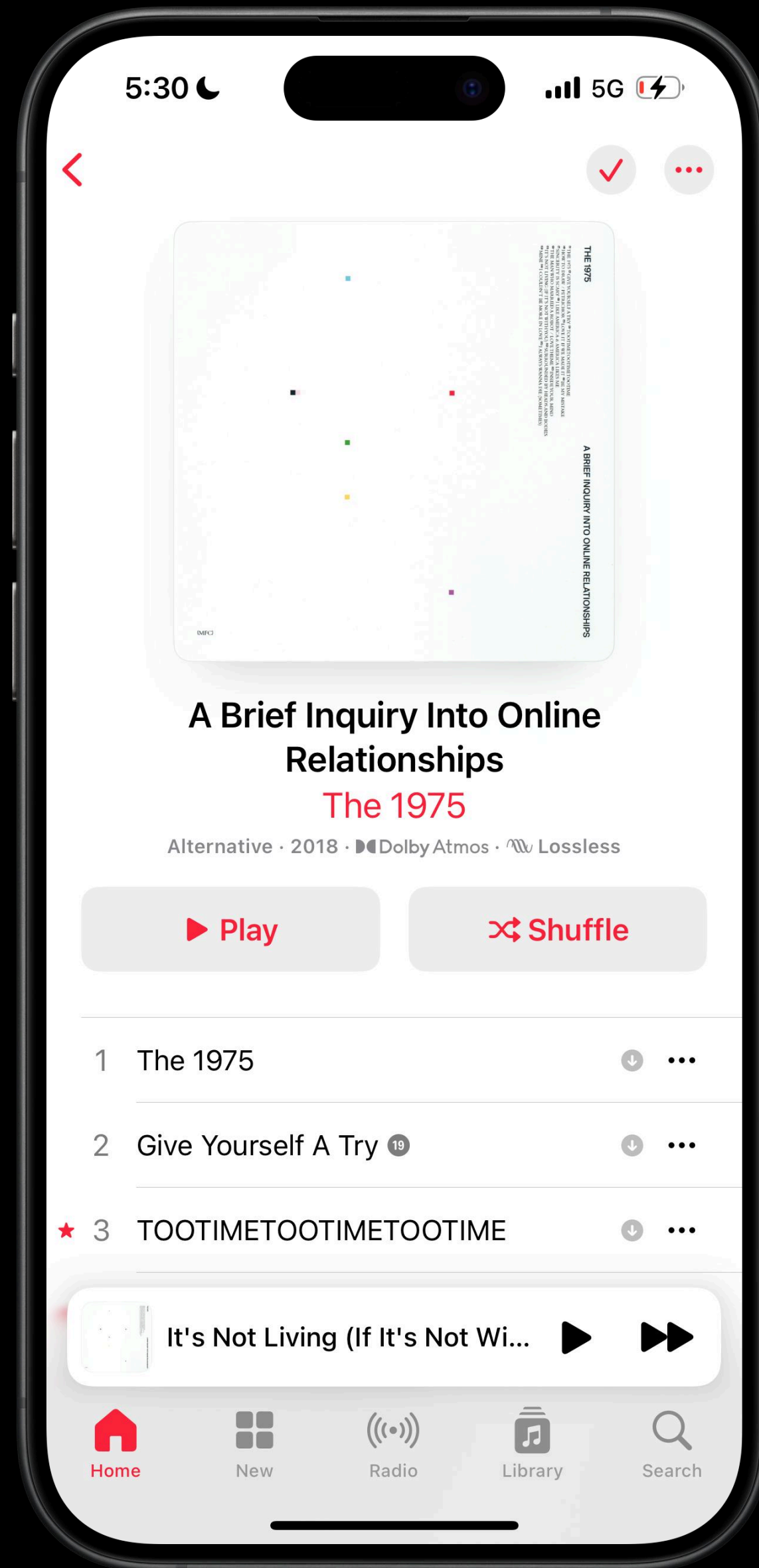
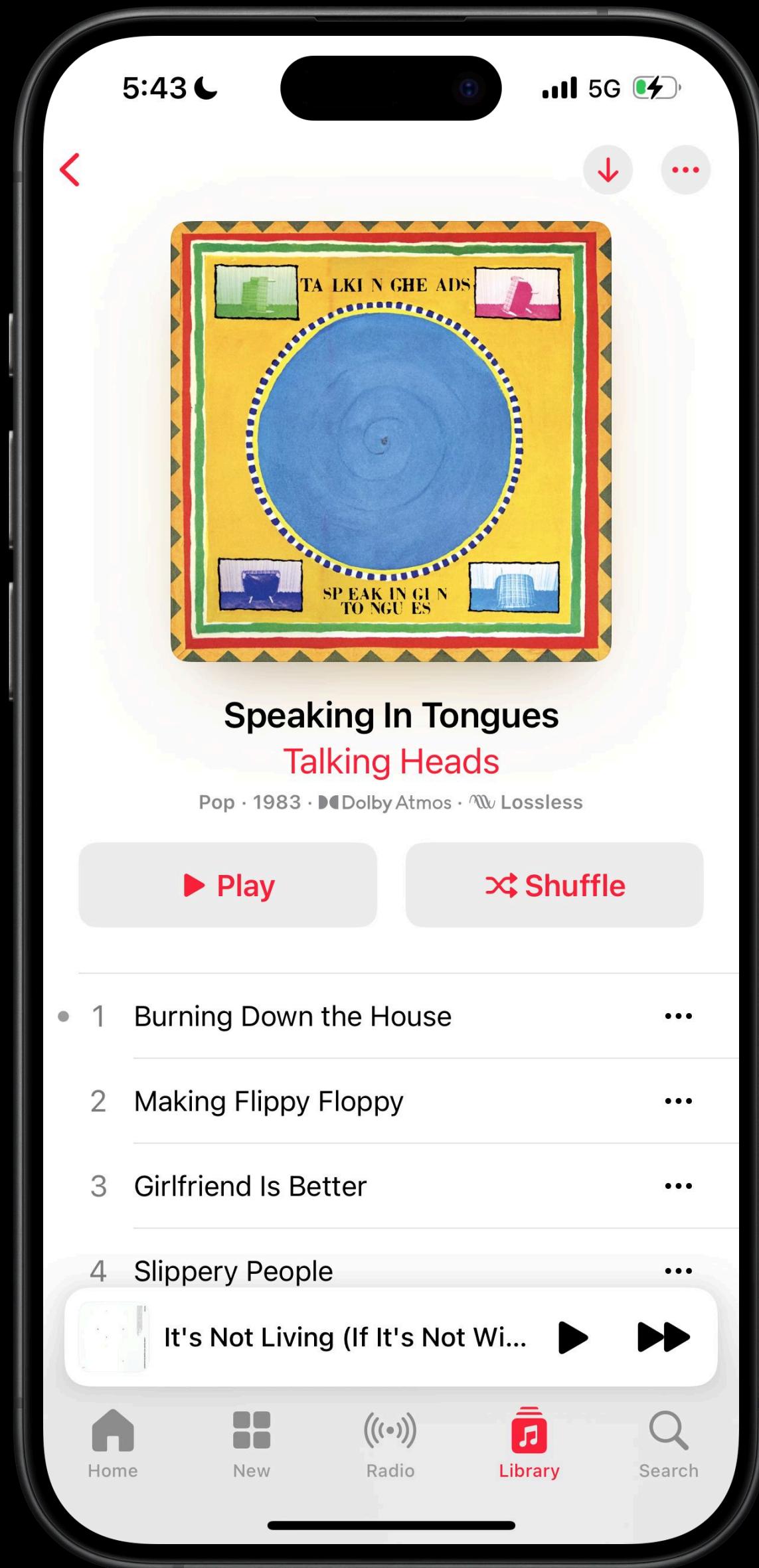
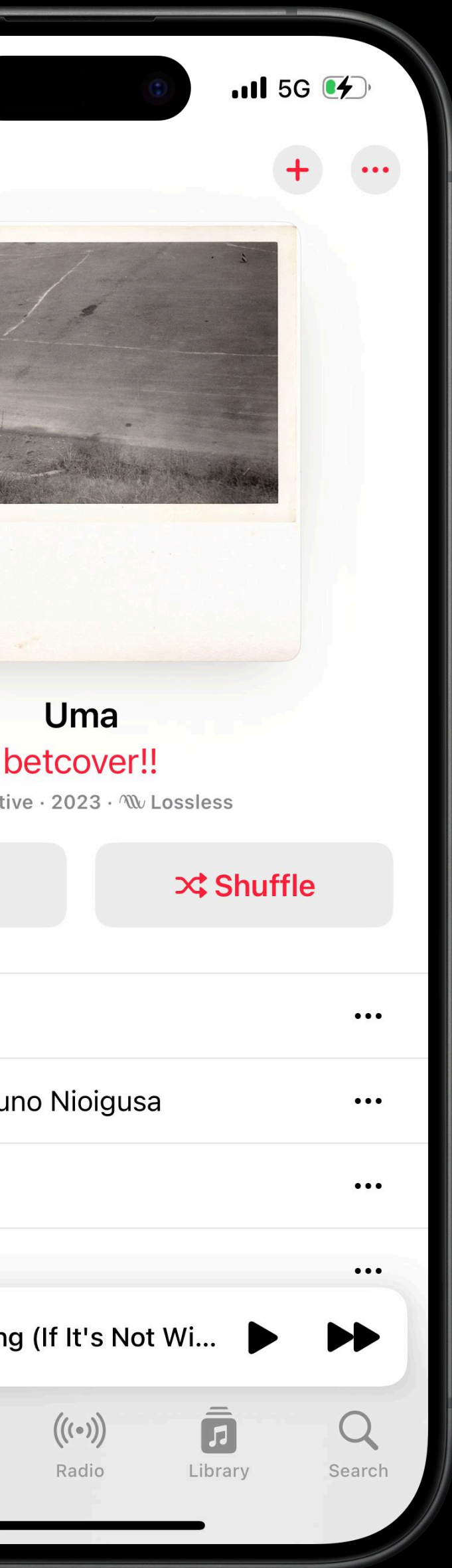


왜 디자인시스템을 쓰나요?

디미고 인트라넷 개발팀, DIN 조현우

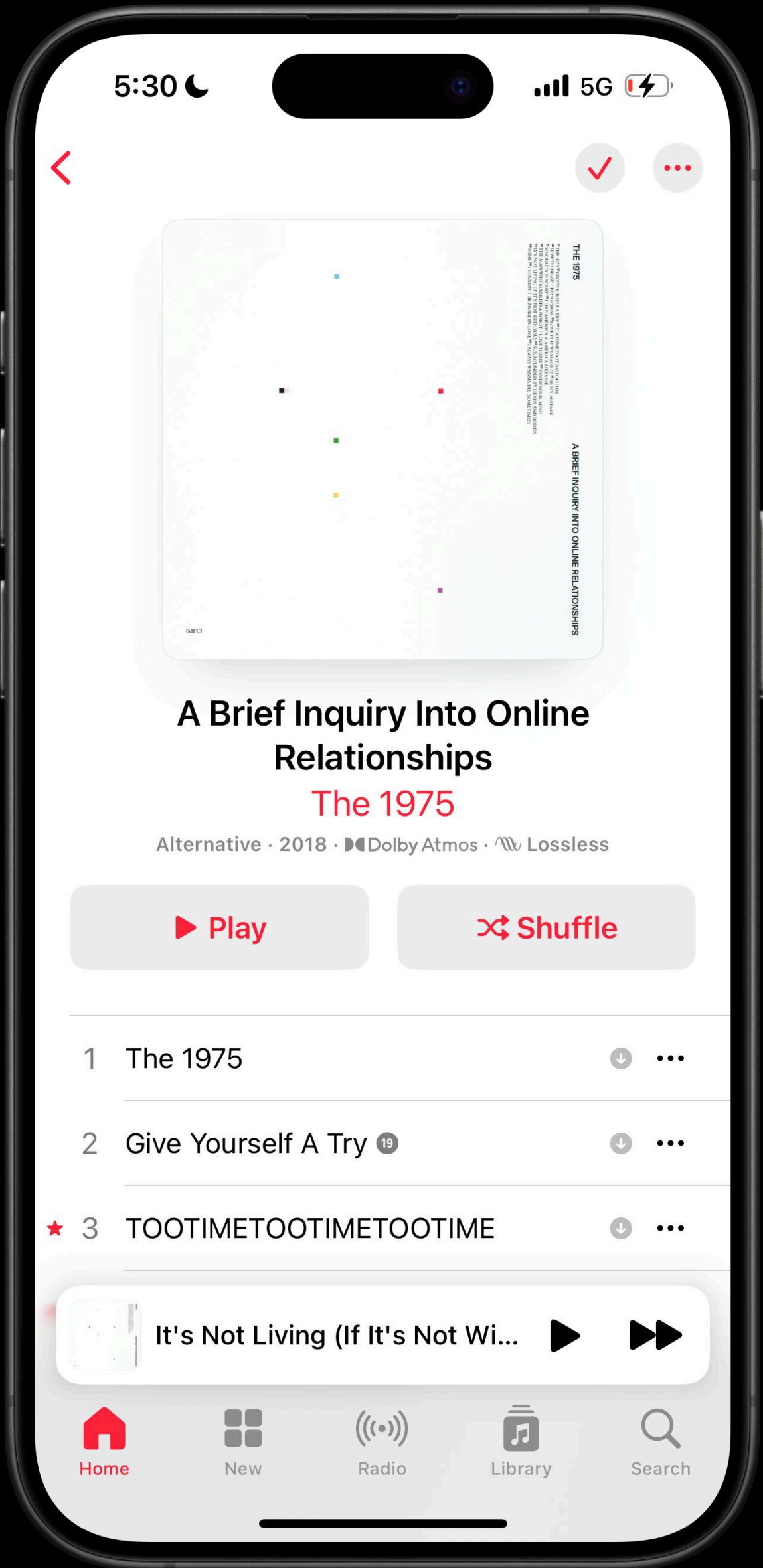






보여주고자 하는 내용이 다를 때마다,
코드 혹은 레이아웃을 일일이 재작성해야할까?


```
<Scroll navbar>
<VStack>
  <ImageView url="https://
img_to_abiior" />
  <Header album="A Brief Inquiry
Into Online Relationships"
artist="The 1975" />
  <HStack>
    <Button text="Play" />
    <Button text="Shuffle" />
  </HStack>
  <TrackList name="The 1975" />
  <TrackList name="Give Yourself A
Try" explicit=true />
  ...
</VStack>
```



AppNavigationBar

ImageView

url: https://img_to_abiior

Header

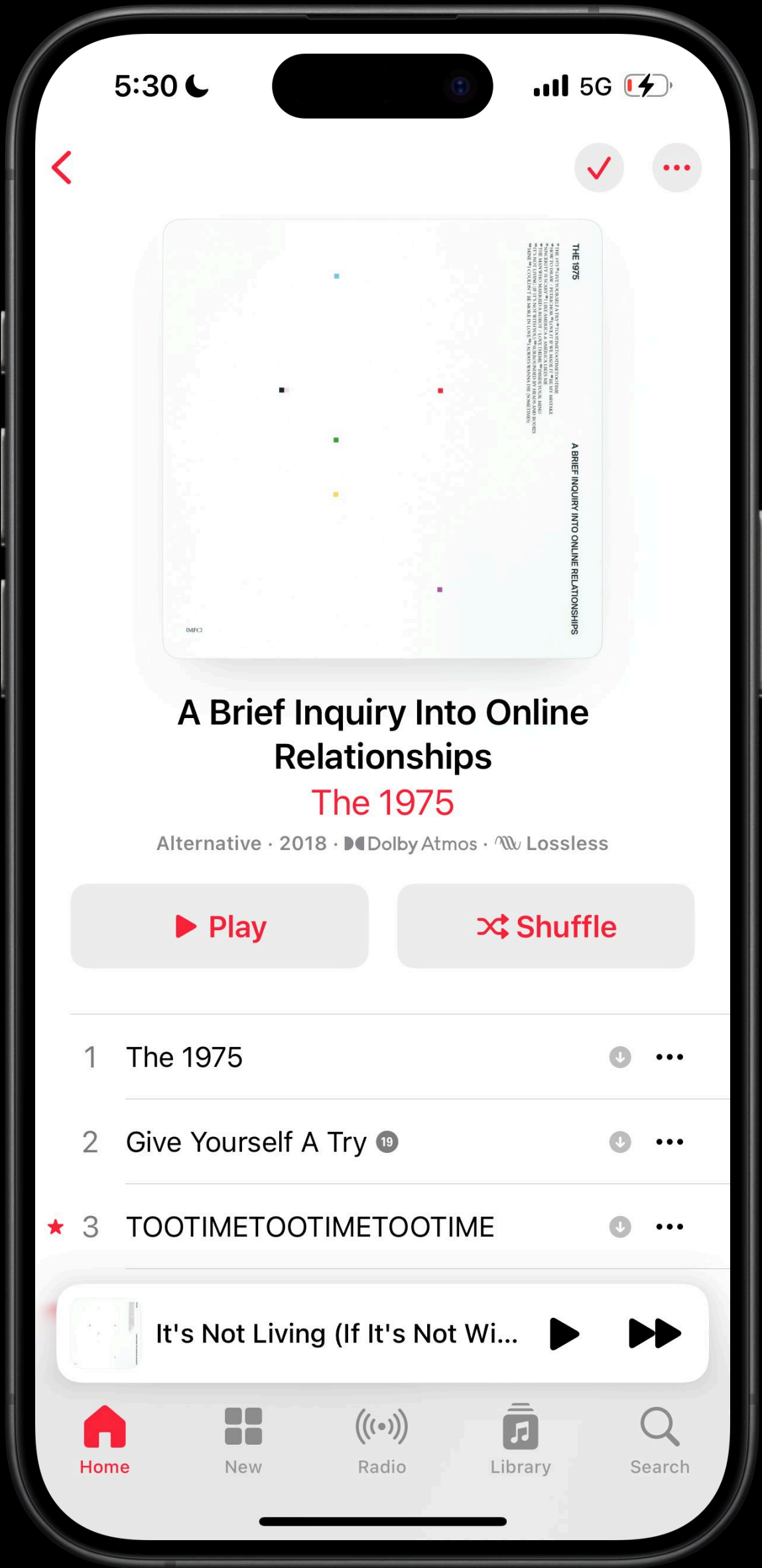
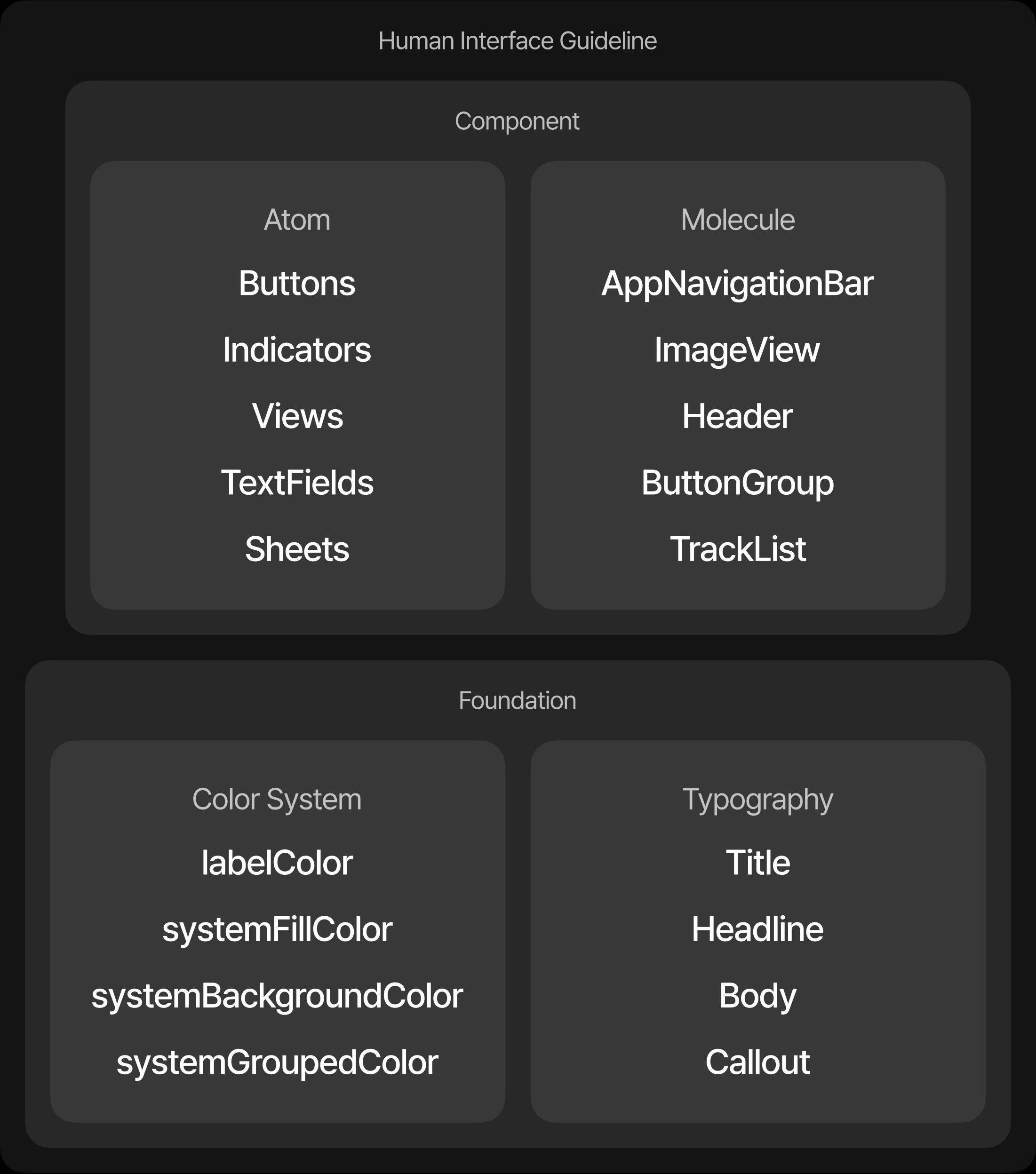
album: A Brief Inquiry Into Online Relationships, artist: The 1975

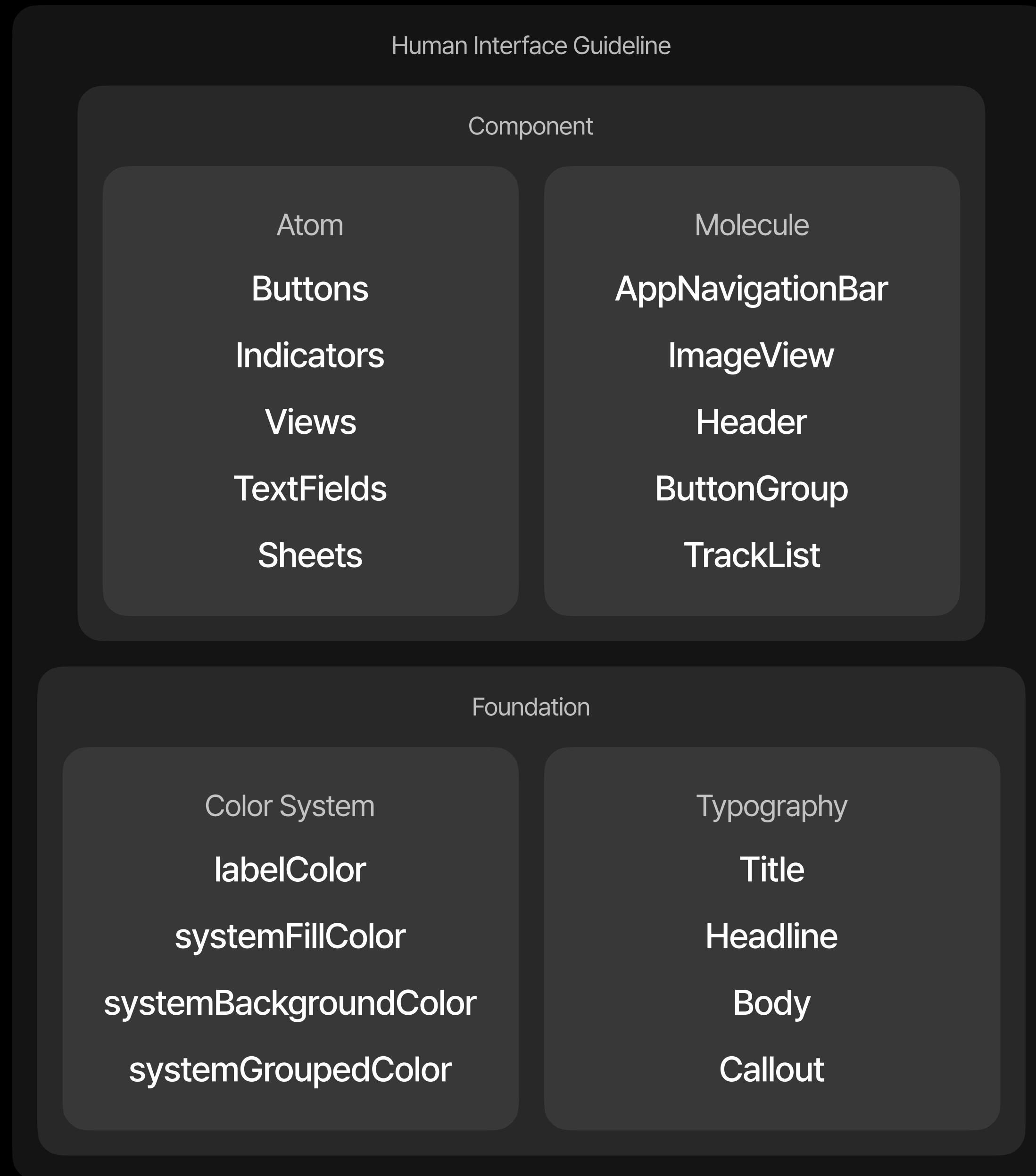
ButtonGroup

TrackList

재사용되는 규칙들을 **정형화**

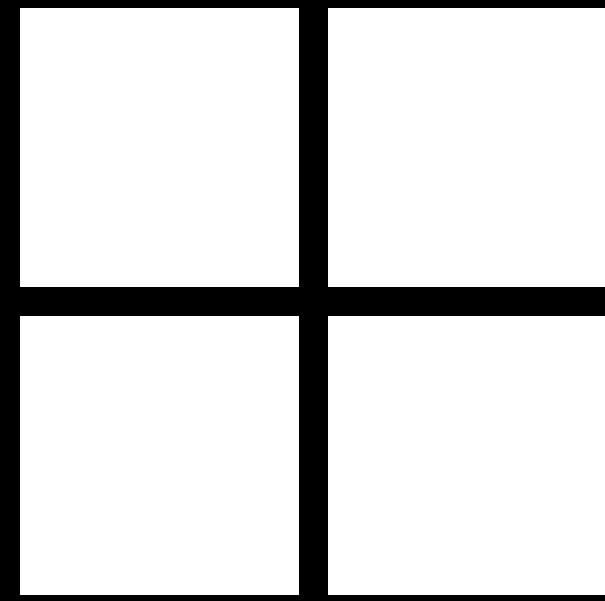
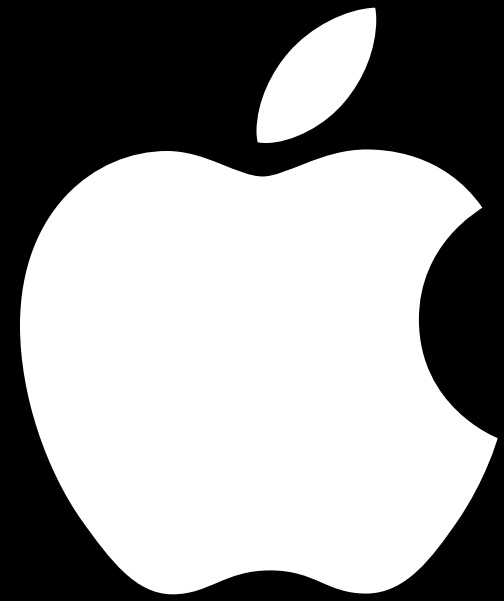
서비스가 제공하고자 하는 주요 기능을 **일관성 있게 적용**





단순히 재사용되는
컴포넌트만 정의하는 것이 아닌

**서비스 전체에서 사용되는
타이포, 컬러까지 전부 정의**



Human Interface Guideline

Fluent 2

Material Design 3



우리 서비스 정상영업 합니다

디자인 시스템이요?
전 그냥 작은 동네 밥집 운영할건데
굳이 필요한가요?

김** (18세) / 소규모 스타트업 재직

디자이너님... 같은 컴포넌트인데
여기는 4px이고 여기는 8px이고
여기는 fffffff인데 여긴 f0f0f0이고
제발 좀 통일해주세요





















이히힝 ㅎㅎ이힝
에헤헤헝 ㅎㅎ헝
무지성 컬러 난사
(규칙과 대책 없이 디자인하기)



소규모 팀에서도 프로덕트의 **퀄리티**와 **통일성**을 위해
최소한의 스타일 가이드는 정립하는 것이 좋다

물론 제대로 된 디자인 시스템을 만드는 일은 **n년 이상** 필요!

Design Token

All variables47		Name	Light	Dark
<div>Background</div> <div>Root</div> <div>Fill</div> <div>Group</div> <div>State</div> <div>Border</div> <div>Content</div> <div>Neutral</div> <div>Semantic</div> <div>Status</div> <div>Calendar</div> <div>Dim</div>	Background / Root			
		 Regular	 System/White/100	 Theme/Contrast/2
		 Strong	 Theme/Contrast/5	 Theme/Contrast/1
	Background / Fill			
		 Regular	 Theme/Brand Black/4	 Theme/Vibrant White/2
		 Strong	 Theme/Brand Black/8	 Theme/Vibrant White/4
		 Inverted	 Theme/Contrast/3	 Theme/Contrast/5
	Background / Group			
		 Regular	 System/White/100	 Theme/Contrast/2

디자인 토큰... 디자인이니까
디자이너가 만드는데로 하면 되겠지?



토큰...? 뭔가 개발자가 알아서 할 거 같은데?

디자인 토큰은 단순한 이름 붙이기가 아니다

플랫폼, 개발 환경, 디자인 성향을 모두 고려하여
요소 하나하나 신중하게 결정

* The 3102

design system

Primitives (Brand)

Theme (Color)

Raw Values

 `hsba(340, 72%, 94%, 1)`

 `hsba(349, 86%, 100%, 1)`

Brand Color

브랜드 Primary의 HSB 값을 기준으로 보정되는 컬러

 `prim.brand.primary.100`

 `prim.system.red.100`

System Color

브랜드에 관계 없이 유지되는 시스템 계열 컬러

 `color.semantic.brand`

 `color.semantic.status.danger`

Primitives to Theme, 2계층 구조

Primitives (Brand)

 prim.brand.black.80

 prim.brand.white.80

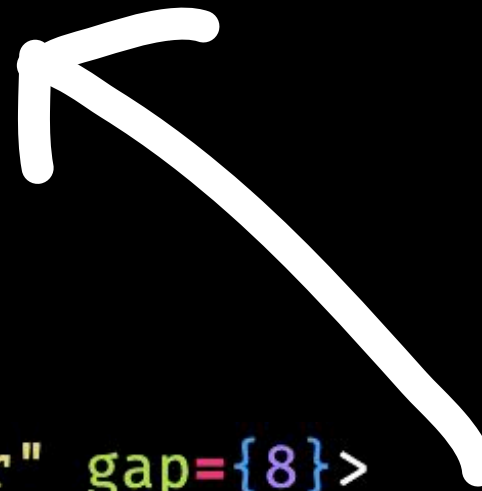
 prim.brand.white.80

Theme (Color)

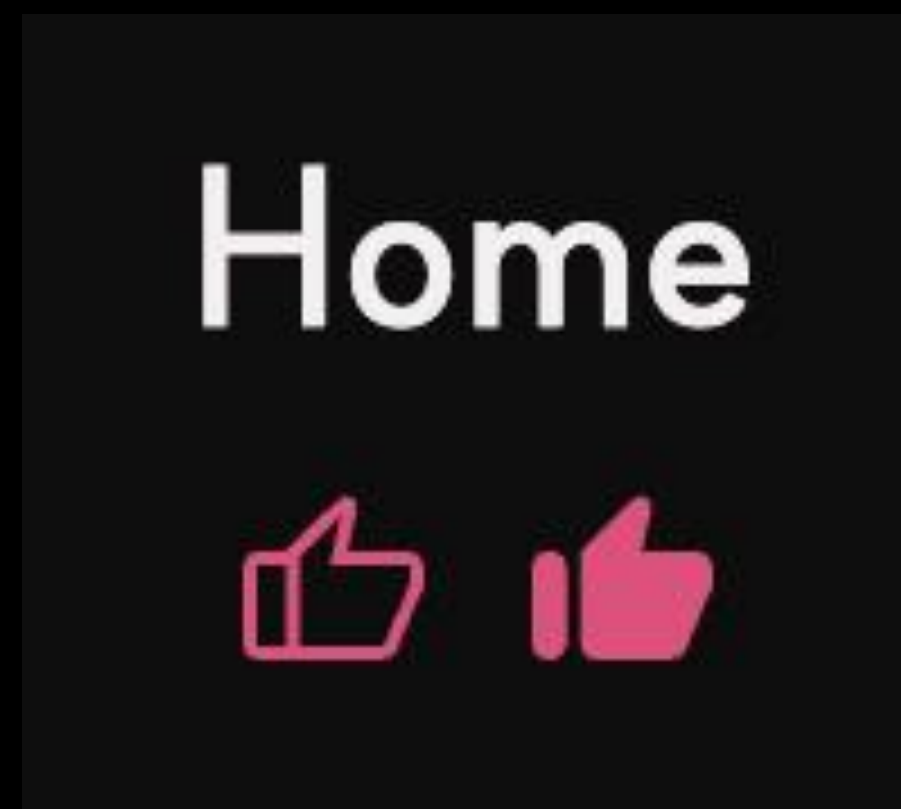
 color.content.neutral.secondary

Light, Dark, Elevated

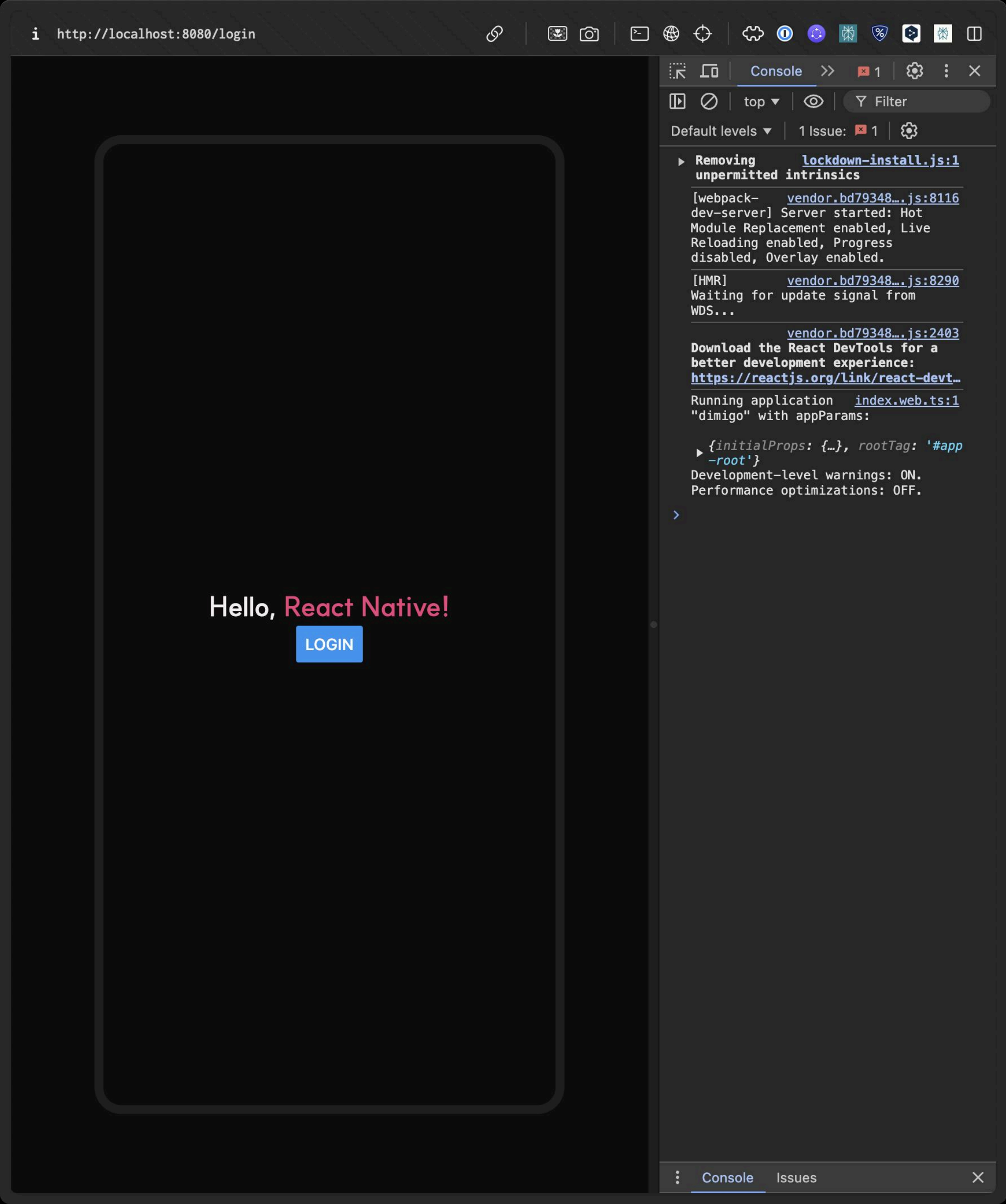
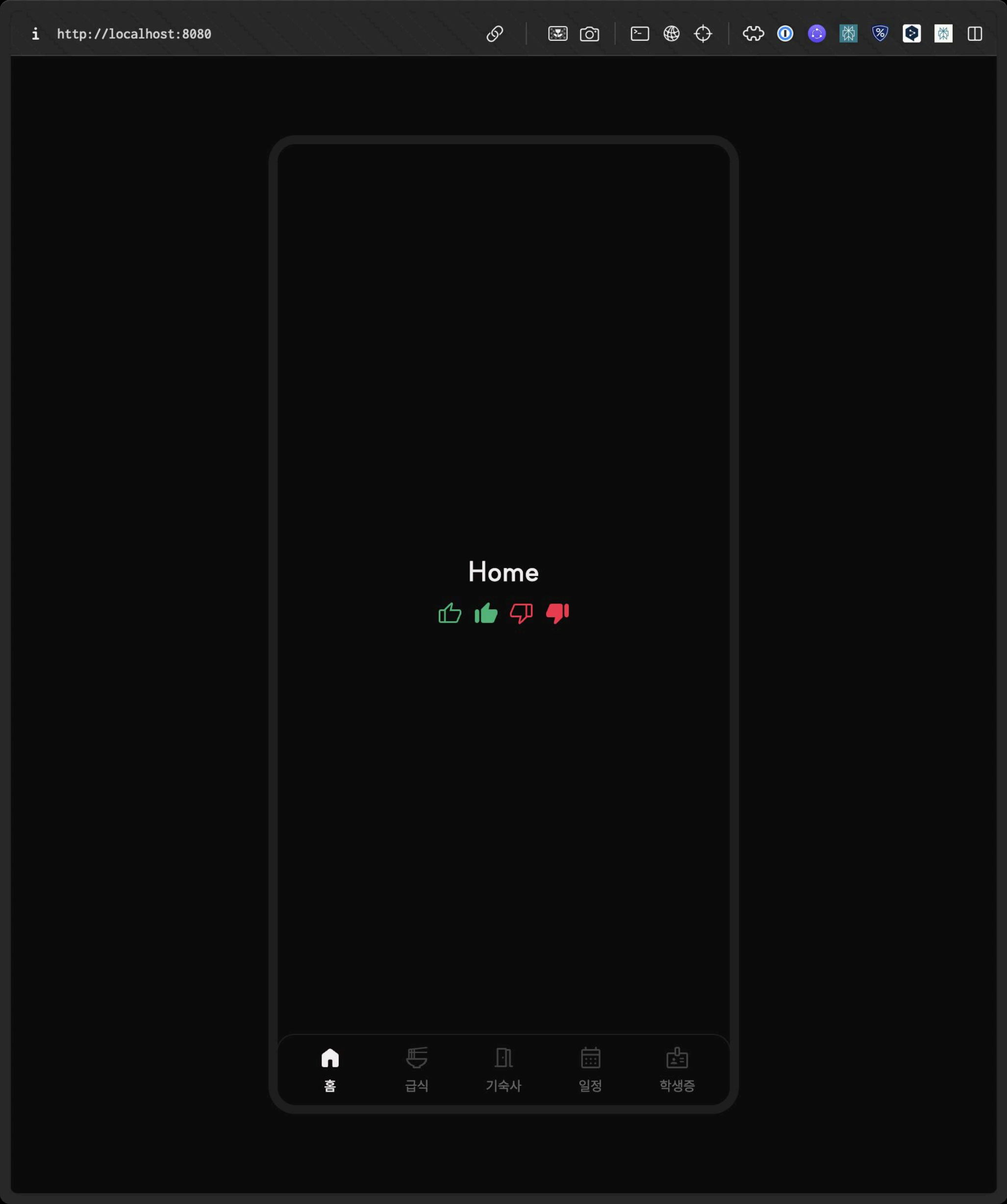
이렇게 토큰을 사용!

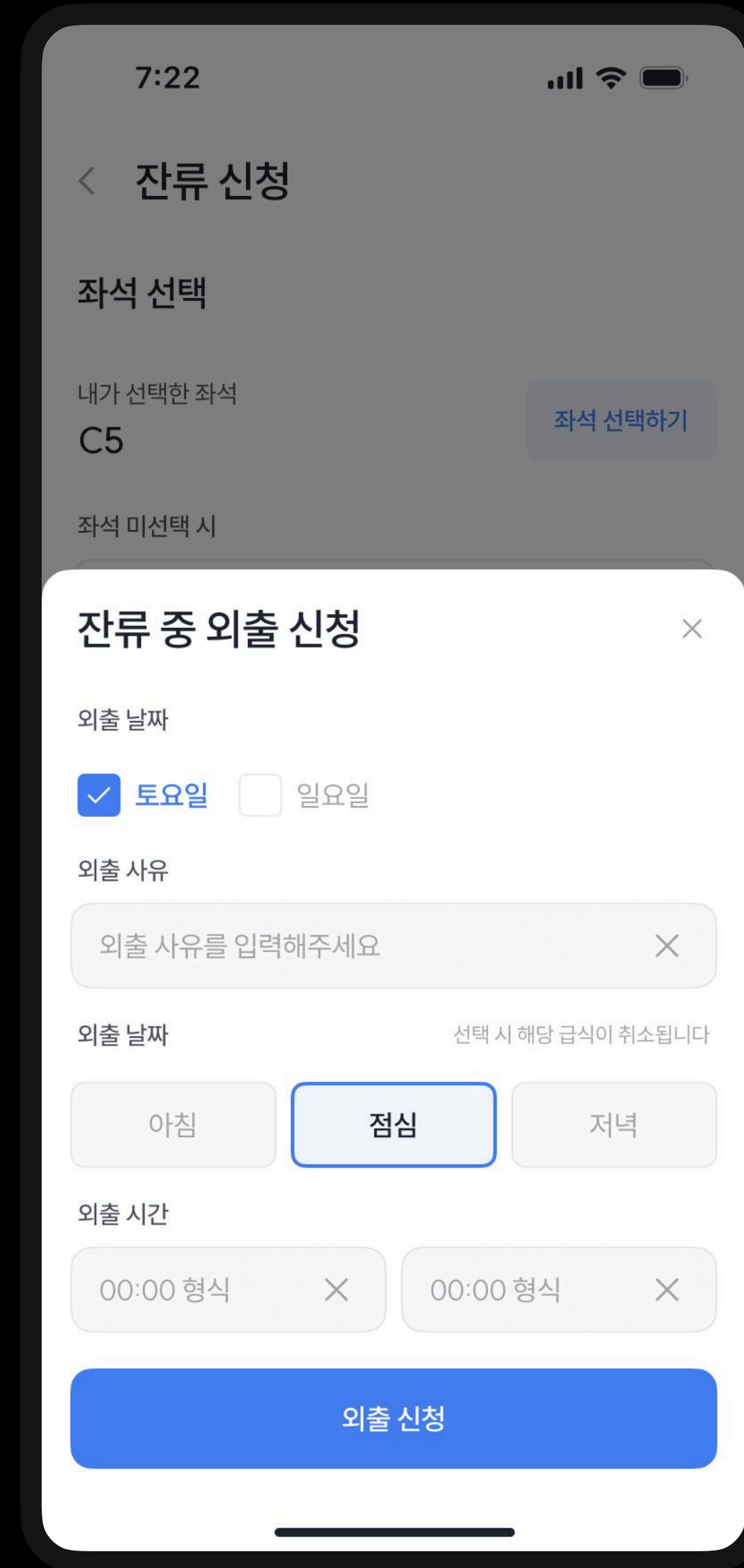
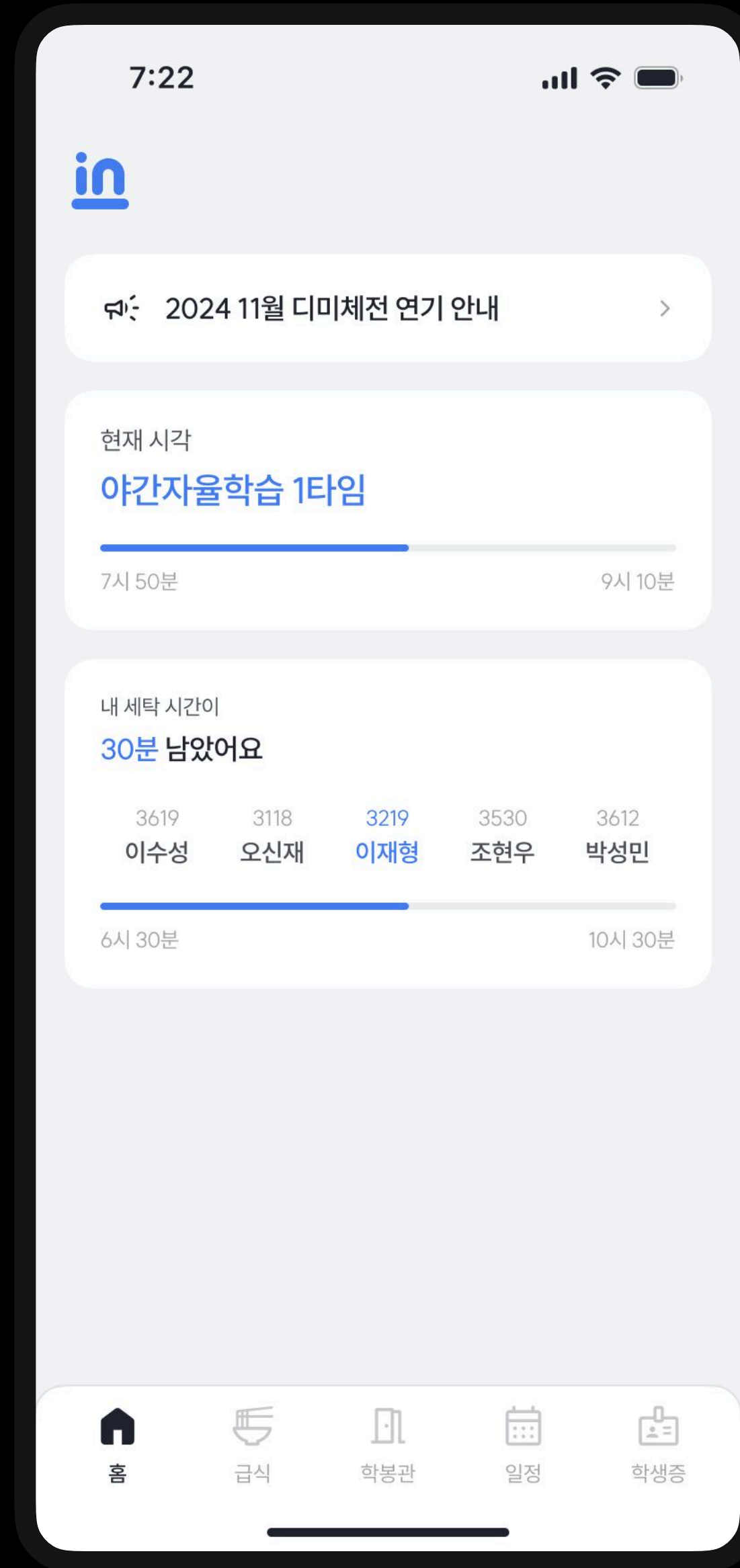
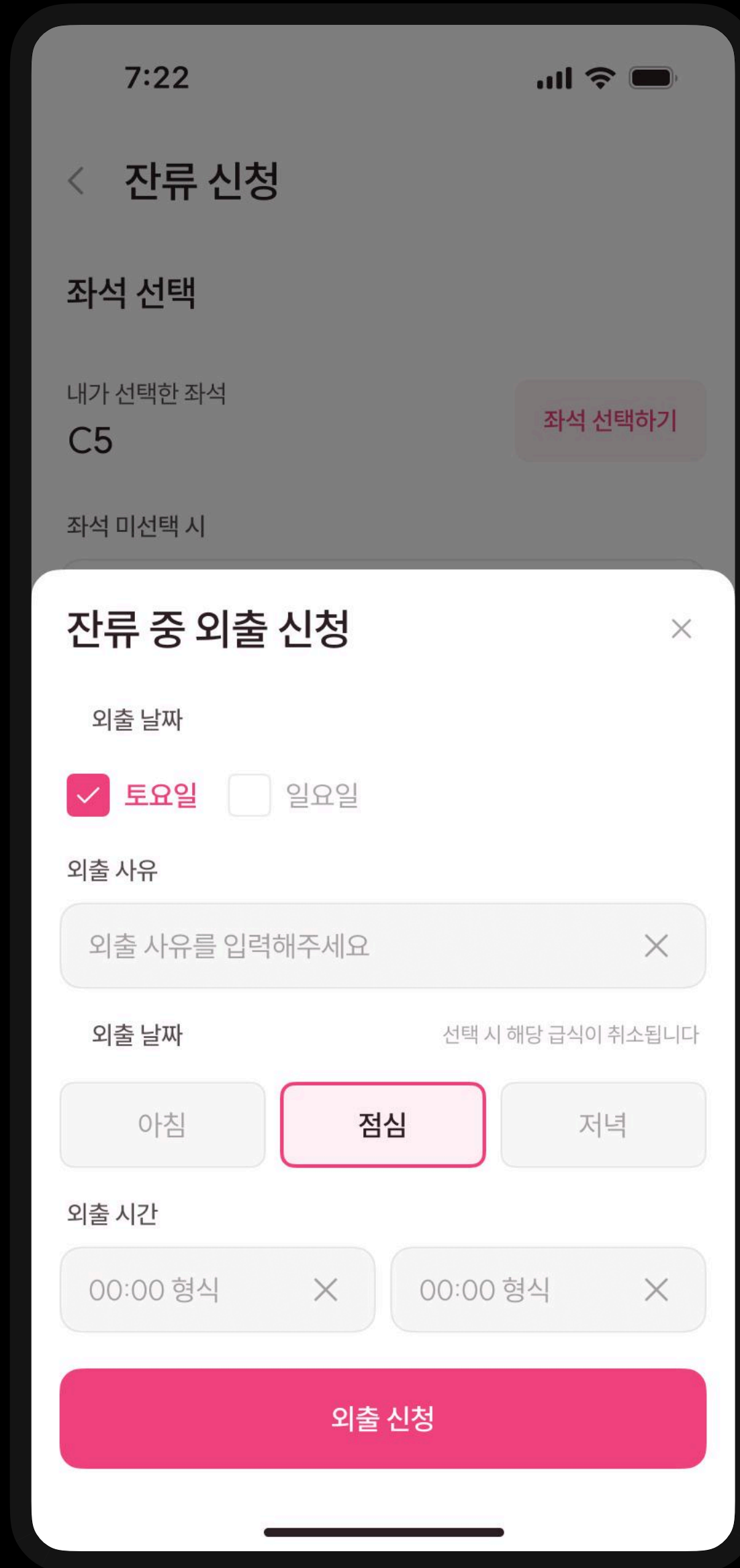
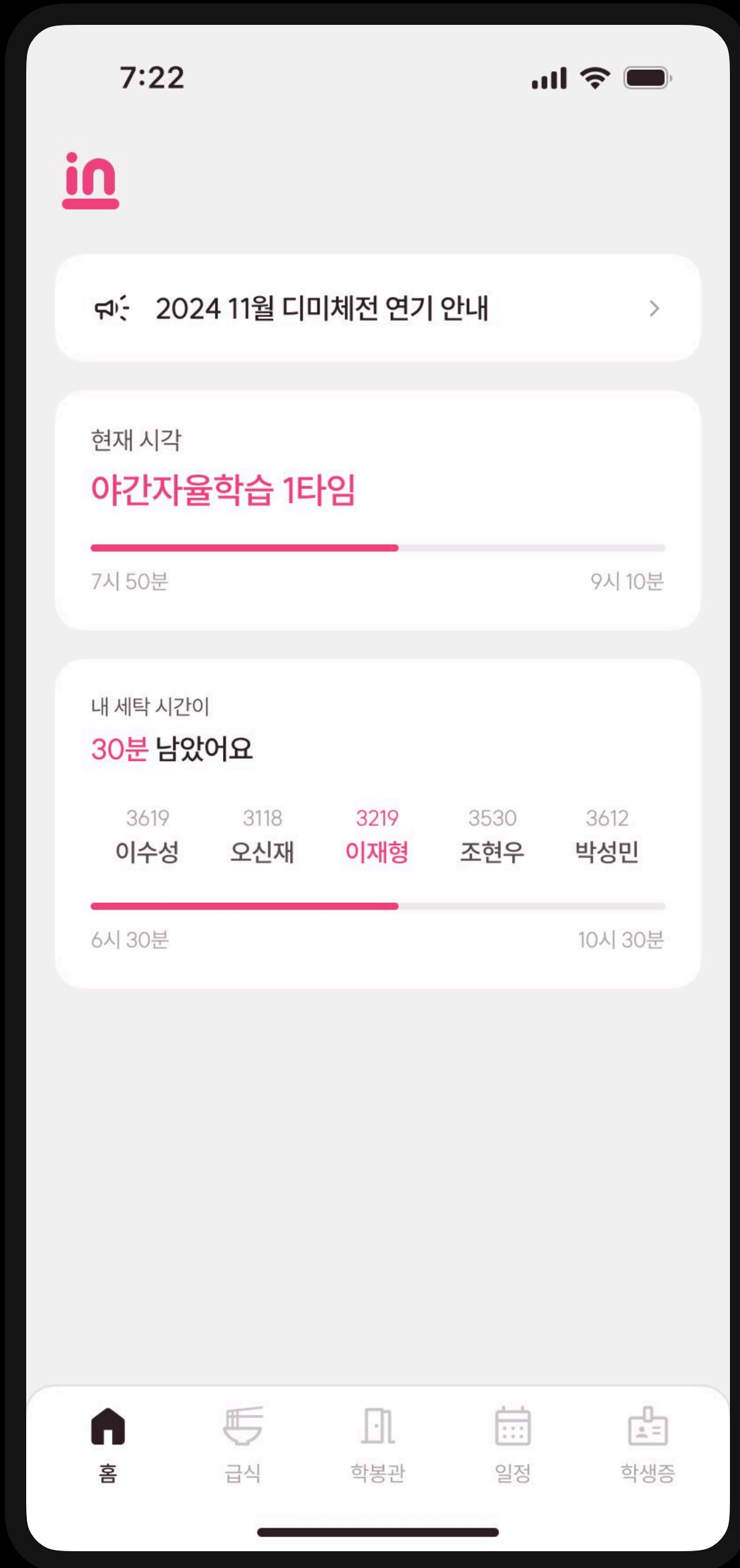


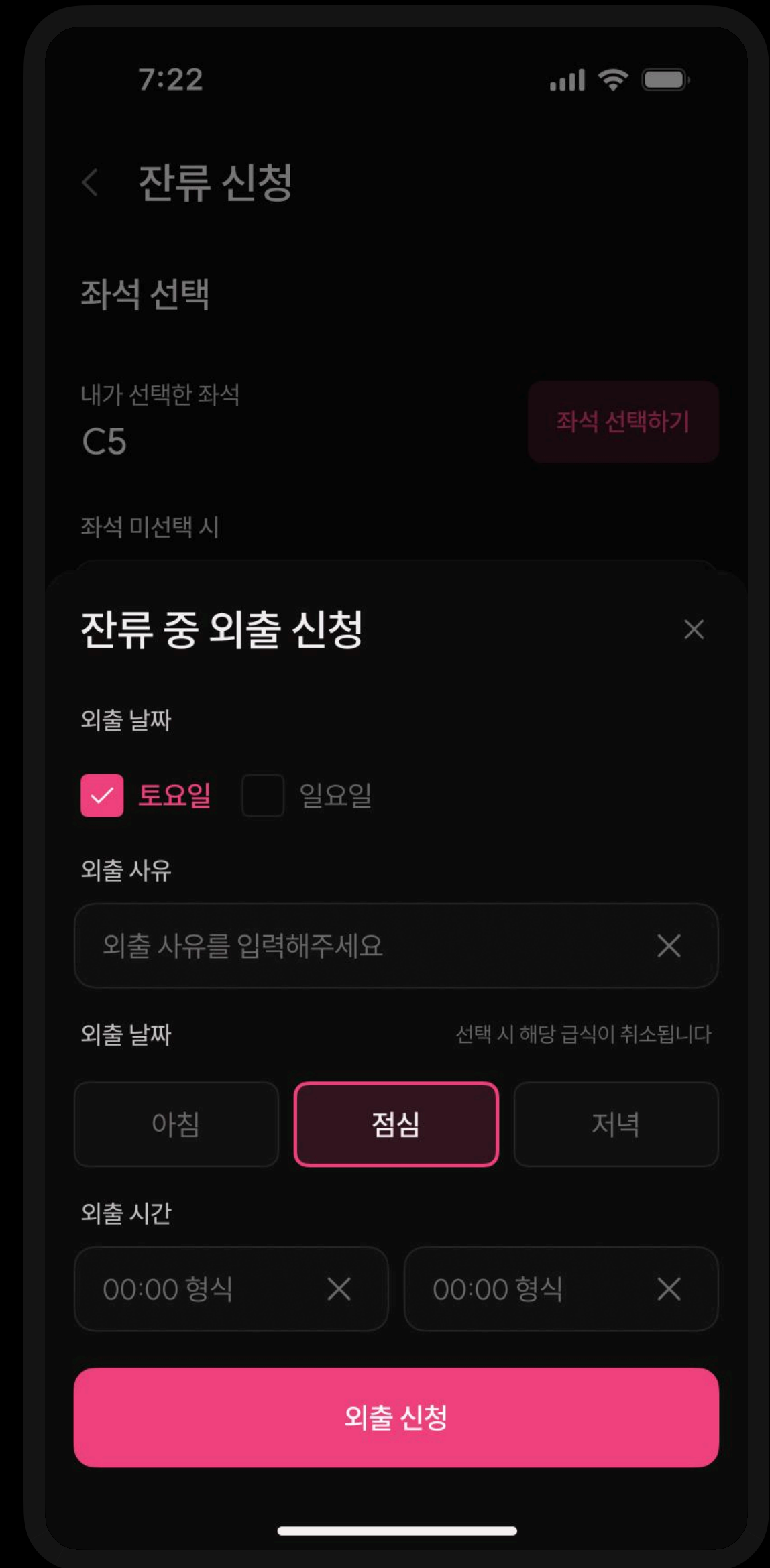
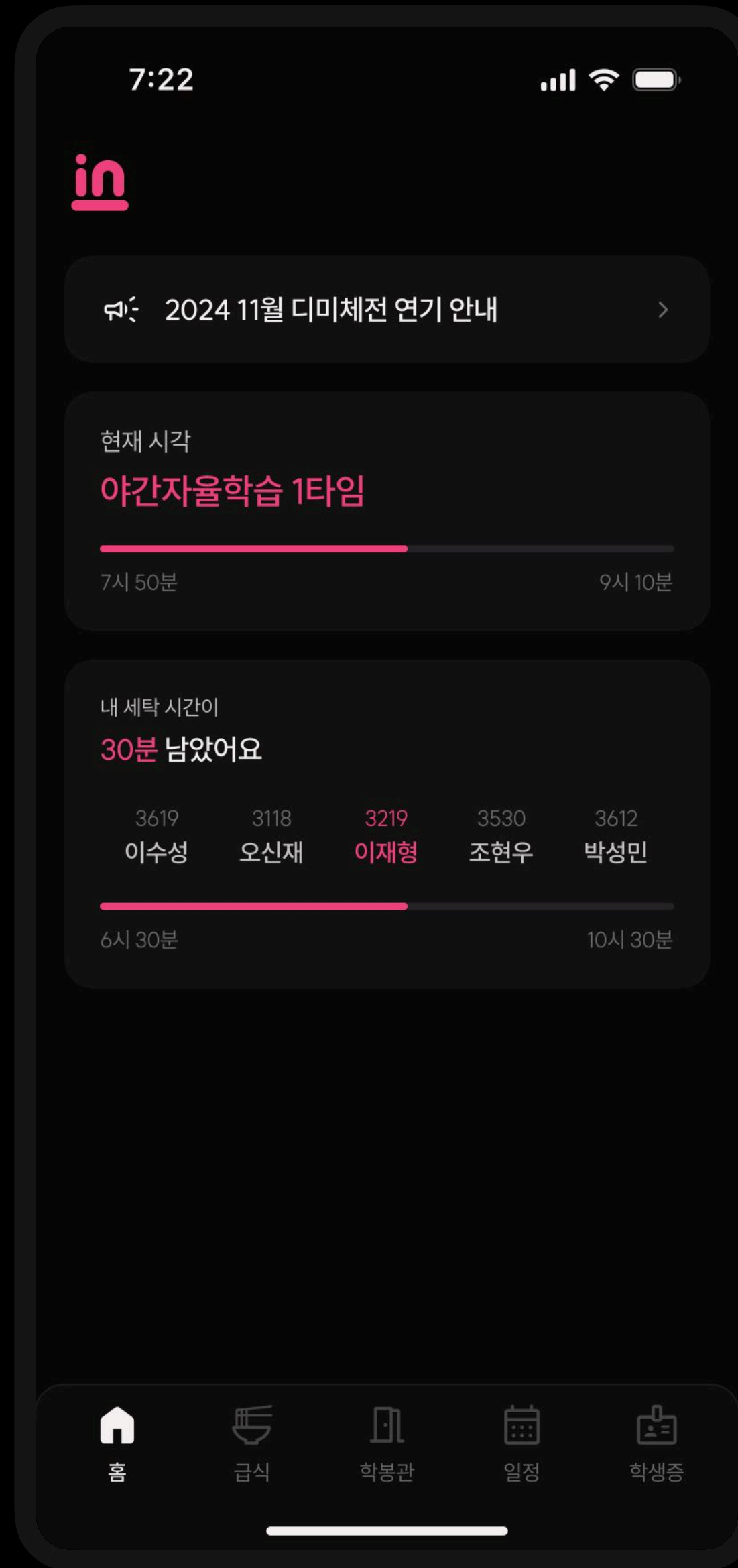
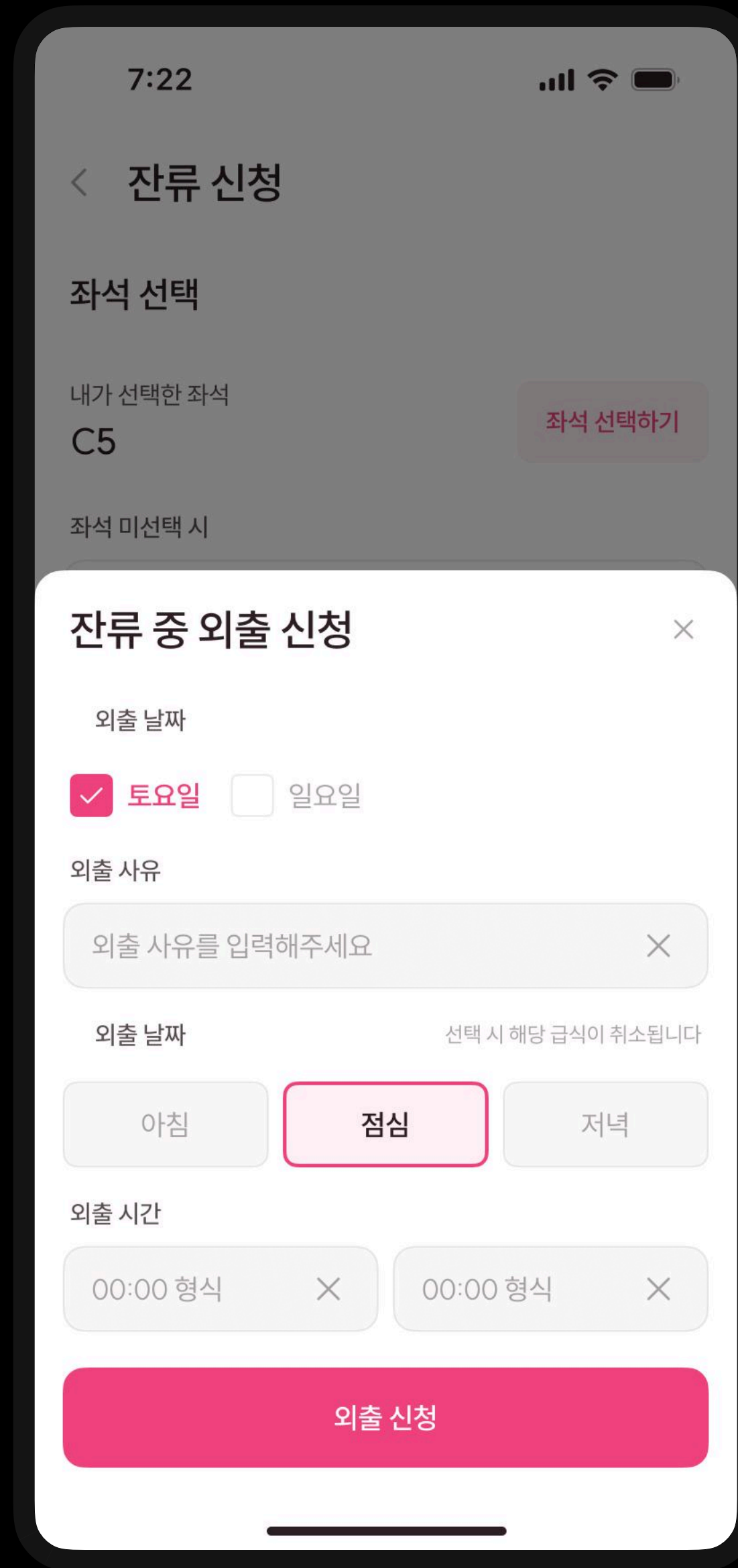
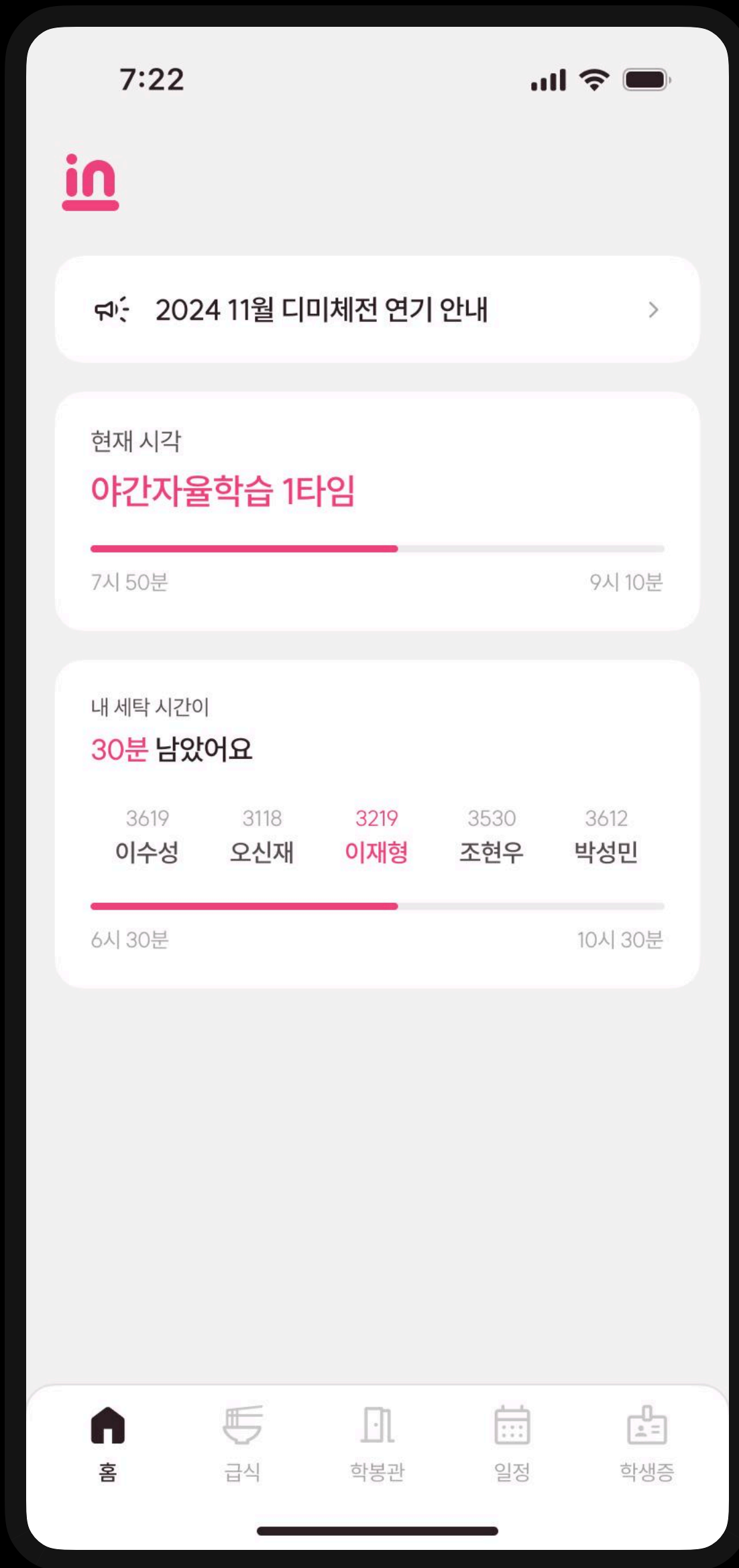
```
<VStack fill align="center" justify="center" gap={8}>
  <Text type="title.strong" color="content.neutral.primary">
    Home
  </Text>
  <HStack gap={8}>
    <Icon name="ThumbUp" color="semantic.brand" />
    <Icon name="ThumbUp" color="semantic.brand" fill />
  </HStack>
</VStack>
```



RN 0.76 기준







멀티 브랜딩, 컬러 모드 등

디자인 토크을 잘 정리하면
다양한 상황에 **유연하게 대응**할 수 있다



마로 Today at 2:25 PM

배경은 어떻게 할까?

완전 밖에 부분

html

태두리밖 색



avanturation Today at 2:25 PM

그거

background.root.regular

하자

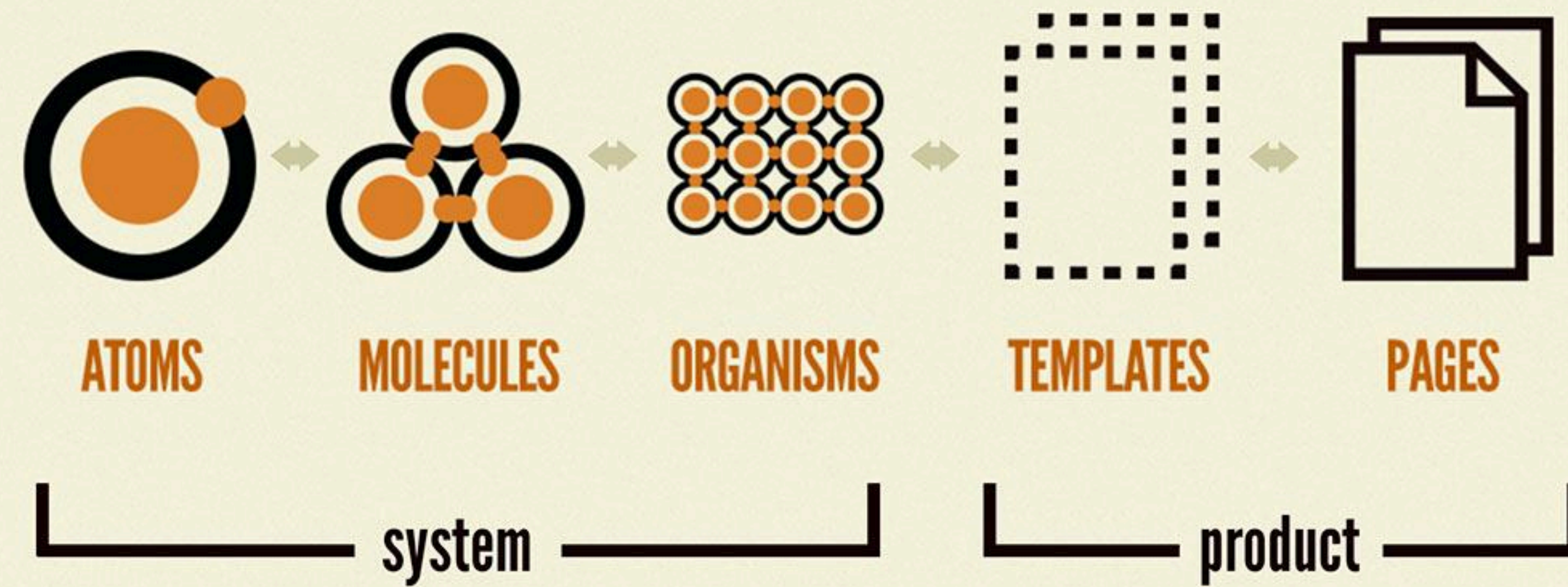


마로 Today at 2:25 PM

ㅇㅋㅇㅋ

여담으로 개발자 - 디자이너 간 커뮤니케이션이 편해집니다

Atomic Design



<https://atomicdesign.bradfrost.com/>

< * The 3102



② Assets



③ Patterns



④ Atoms

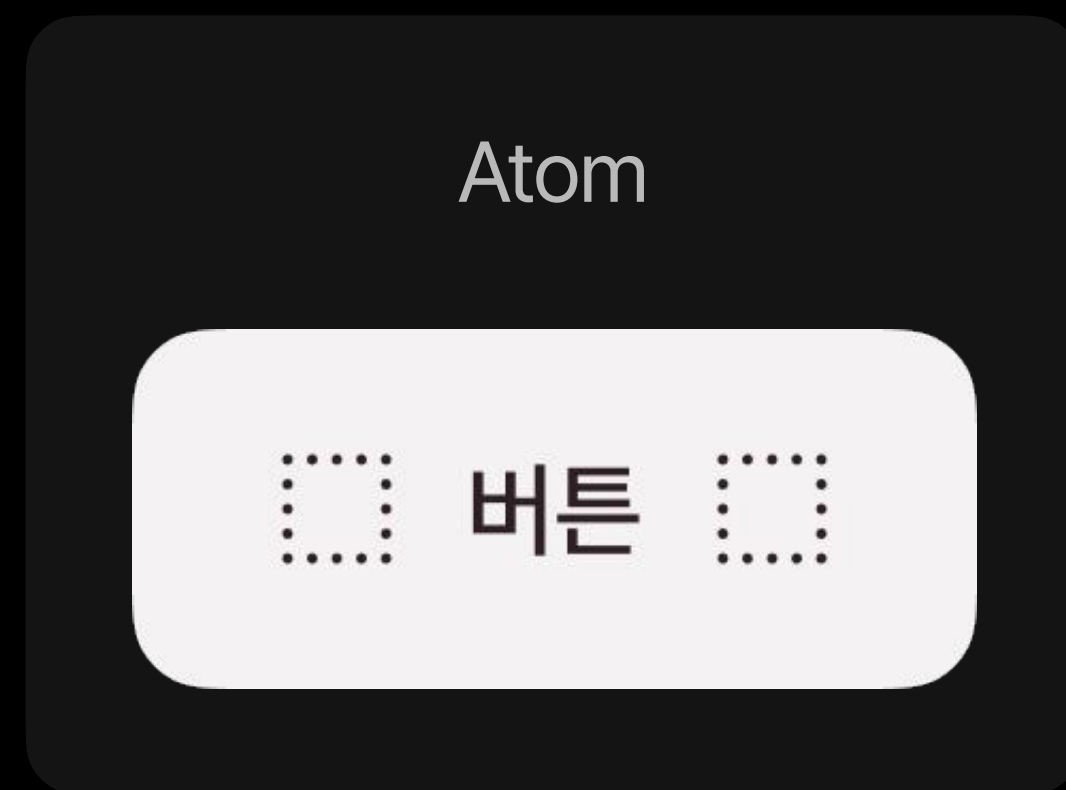


⑤ Molecule

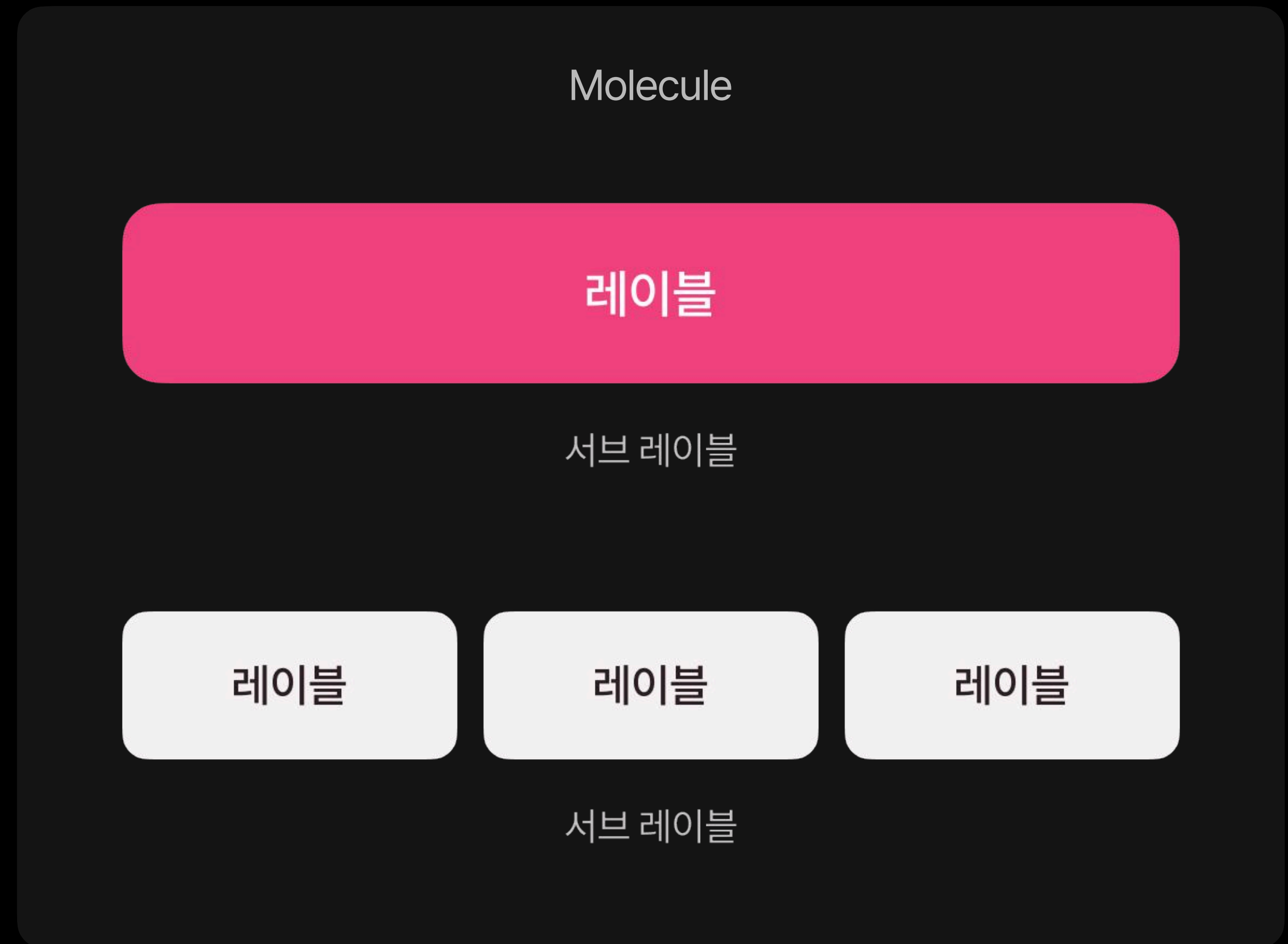


Miscellaneous





더 이상 쪼갤 수 없는 최소 단위



Atom을 조합하여 자주 재사용되는 컴포넌트들



Button | ActionButton

페이지의 핵심 액션을 버튼 형태로 유도할 때 사용하는 컴포넌트입니다.

Share

Design Prototype 84%

ActionButton

Local instance

Type Single

Sub-Label ☒

Sub-Label 서버 레이블

Primary

Size Large

Theme Brand

Style Primary

Disabled ☐

Leading Area ☐

Trailing Area ☐

Label ☒

Content 레이블

Position

|=

≡

=|

|

+

||

...

X 0 Y 176

Atom을 조합하여 Nested Component 형태로 관리

하늘
고사
서어
독화

백문불여일견

百問不若一見

백번 듣기보다는
한번 보는 게 낫다

글 오주영 | 그림 서광진



Q&A