

# Apple Vision Pro の UI

---

宇佐見公輔

2024-07-31

株式会社ゆめみ

# 自己紹介

- 宇佐見公輔
- 株式会社ゆめみ
  - ▶ iOS テックリード
- iOSDC Japan 2024
  - ▶ パンフレット記事執筆
  - ▶ ポスターセッション



# 今日の内容

1. Apple Vision Pro の操作方法
2. 視線の動きを調べる
3. UI のガイドライン
4. アラートの UI
5. まとめ

# Apple Vision Pro の操作方法

---

# Apple Vision Pro 購入

- Apple Vision Pro 日本発売
  - 2024 年 6 月 28 日
- 日本での発売を待っていた
  - 発売日に購入



# 操作方法

次の方法が基本。

- 視線によるフォーカス移動
- 手のジェスチャによるタップ

それ以外に次の方法もある。

- 頭、手首、人差し指
- Magic Trackpad (visionOS 2 からマウスも使用可能)
- Mac のトラックパッド (Mac 仮想ディスプレイ使用時)
- ゲームコントローラ (対応アプリのみ)

# 視線と注視による操作

## Apple Vision Pro ユーザガイド

丁寧なユーザガイドだが、視線や注視についてはあまり詳しく書かれていない。

### 手と視線のジェスチャを使用する

アクション	このジェスチャで行えること	説明
指をタップする(または単に「タップ」)	オプションを選択する、アプリを開く	コンテンツ(アプリアイコンやボタンなど)を注視してから、人差し指と親指を <a href="#">タップ</a> して、そのコンテンツを選択します。

# 理解はできるが実践が難しい

実際には次のようなことが起こる。

- ・ 視線を移動しても、フォーカスがうまく移動しない
- ・ 対象を注視していても、フォーカスが移動してしまう

このため、次のような問題が発生する。

- ・ 選択したい項目に、フォーカスを当てられない
- ・ 選択するときに、意図と異なる項目をタップしてしまう

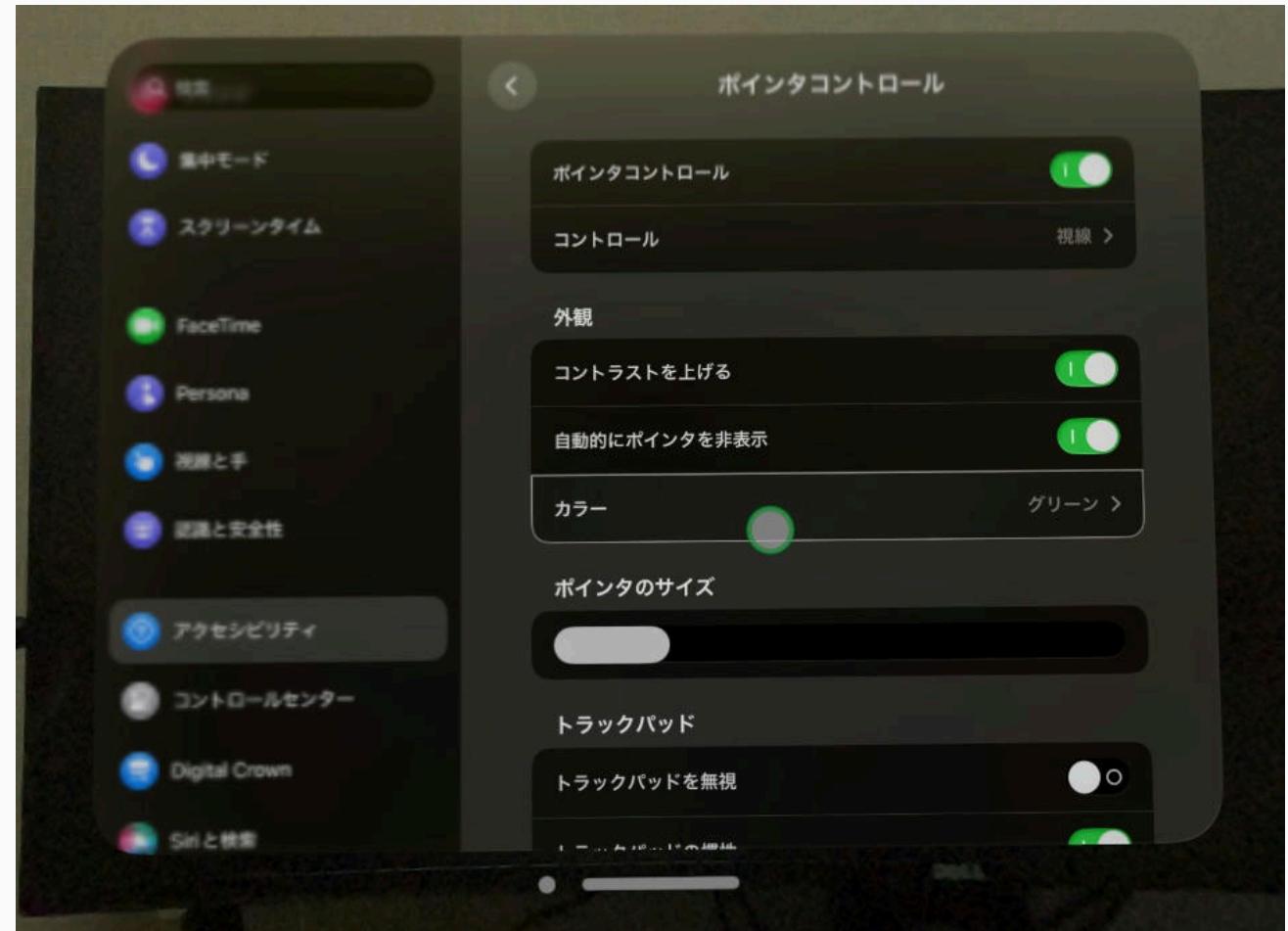
# 視線の動きを調べる

---

# ポインタ表示

アクセシビリティ  
設定でポインタを  
表示できる。

画像の中央あたり  
の緑丸がポインタ



# 補足：フォーカス項目を分かりやすくする

フォーカスのコン  
トラストを上げる  
設定がある。



# 視線は動き回る

ポインタ表示すると、視線の動きがわかる。

- ・何もしていないときに、視線はふらふらと動き回る
- ・視線を移動させたときに、意識と視線は意外と一致しない
- ・注視したときに、視線は固定せずに動いている

視線はじっとしていないで、常に動き回っているもの。

# UI のガイドライン

---

# 参考セッション・ガイドライン

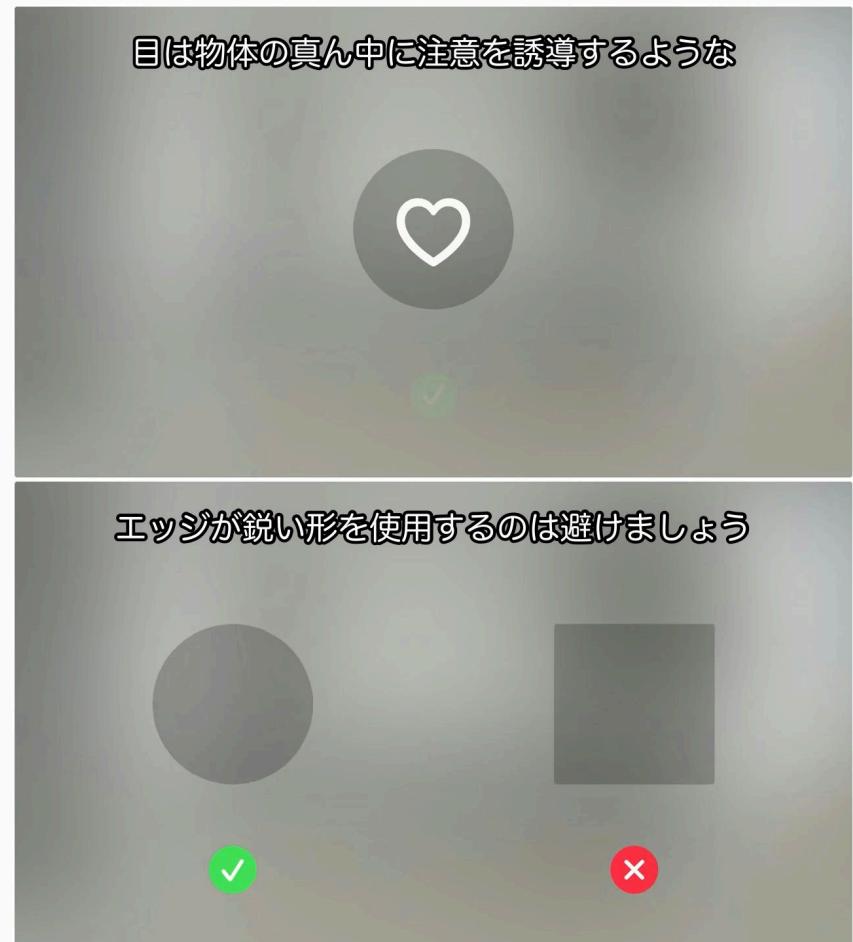
- [空間入力のためのデザイン - WWDC23](#)
- [空間ユーザーインターフェイスのためのデザイン - WWDC23](#)
- [優れた visionOS アプリのデザイン - WWDC24](#)
- [視線 - ヒューマンインターフェースガイドライン](#)

これ以外にも、visionOS のデザインに関するセッションやガイドラインがある。

# 項目の中央に視線を誘導する

視線の移動や注視が難しいことを考慮した注意点がある。

- ・丸い形で中央に視線を誘導
- ・鋭い形は外側に注意が逃げる
- ・外側に縁取りを入れるのも同様によくない



# ボタンの大きさ・間隔

- ボタンのタップ領域は 60pt
- ボタンの表示領域は 44pt
- 表示領域の間隔は 16pt

ポインタが見た目で正確に  
あっていなくても、タップが  
可能になっている。



# アラートの UI

---

# アラートの UI に対する不満

アラートは選択しづらいと感じる。

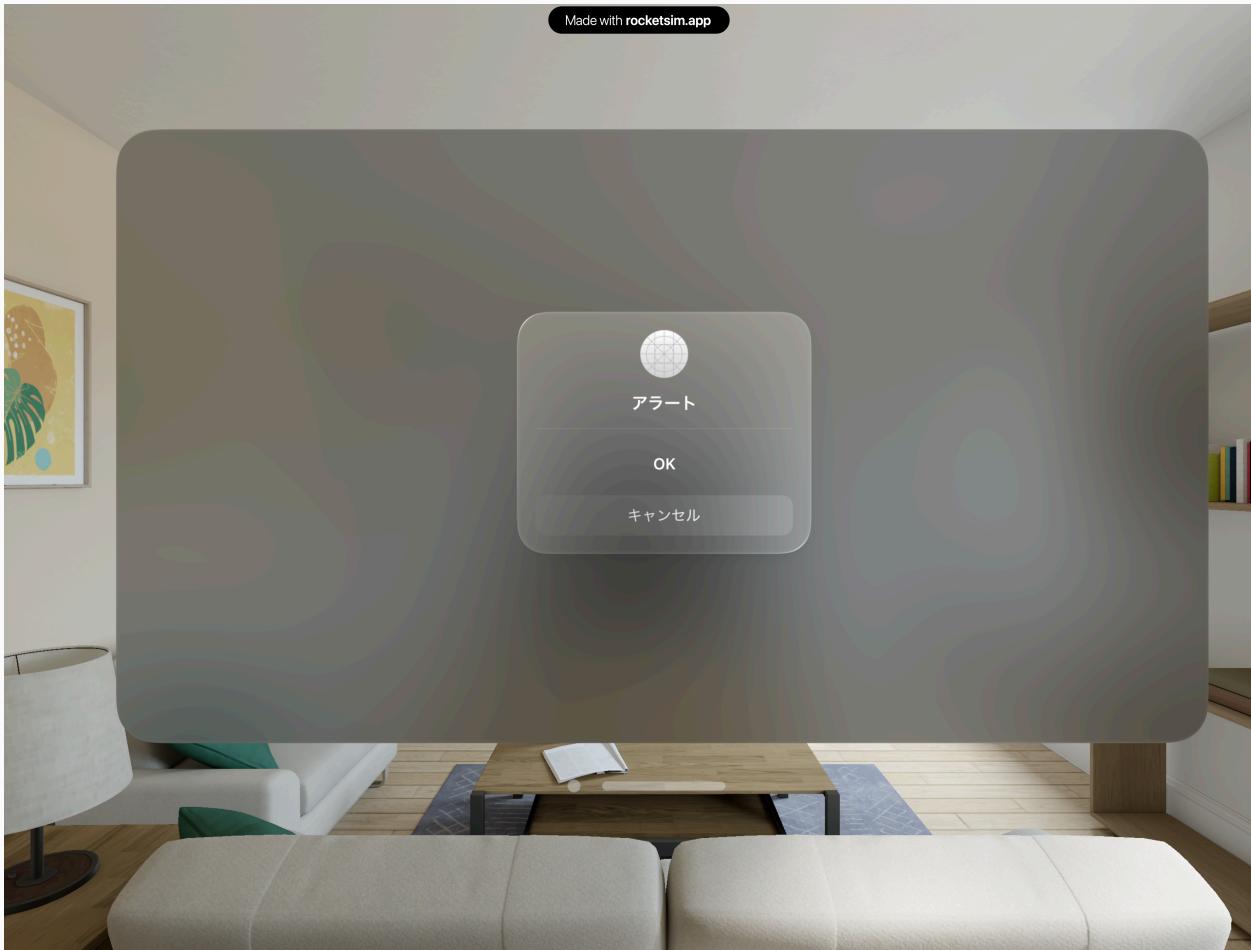
意図と異なる項目をタップしてしまいがち。わりと厄介な結果をもたらす。



# アラートの標準実装

```
Button("アラートを表示") {  
    showAlert = true  
}  
.alert("アラート", isPresented: $showAlert) {  
    Button("キャンセル", role: .cancel) {  
        message = "Cancel tapped"  
    }  
    Button("OK") {  
        message = "OK tapped"  
    }  
}
```

# 標準実装の結果

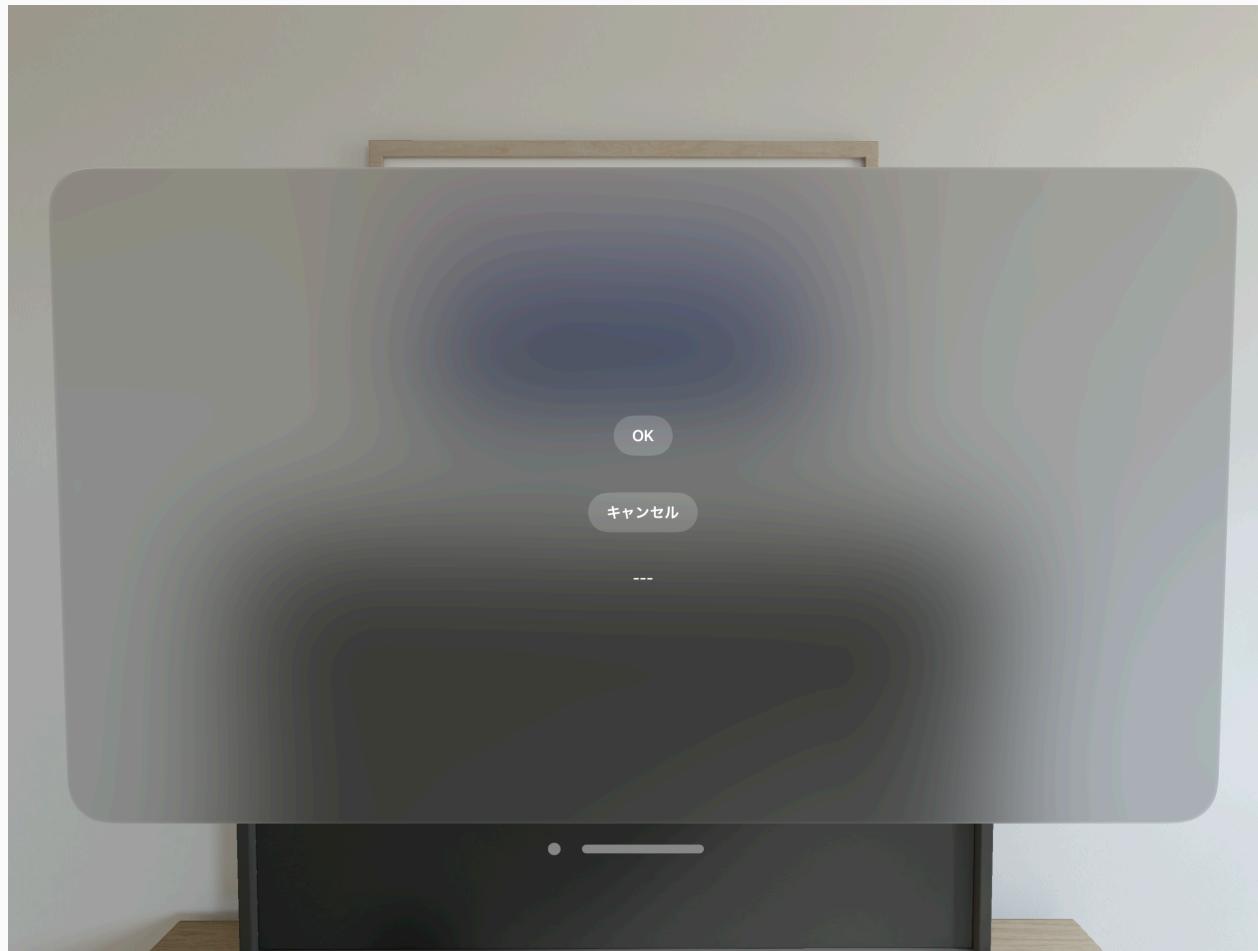


UI ガイドラインの  
知見が活かされてな  
い感じがする。

# 選択しやすい UI を考えてみる

```
 VStack {  
     Button("OK") {  
     }  
     .padding()  
     Button("キャンセル", role: .cancel) {  
     }  
     .padding()  
 }
```

# VStack の場合

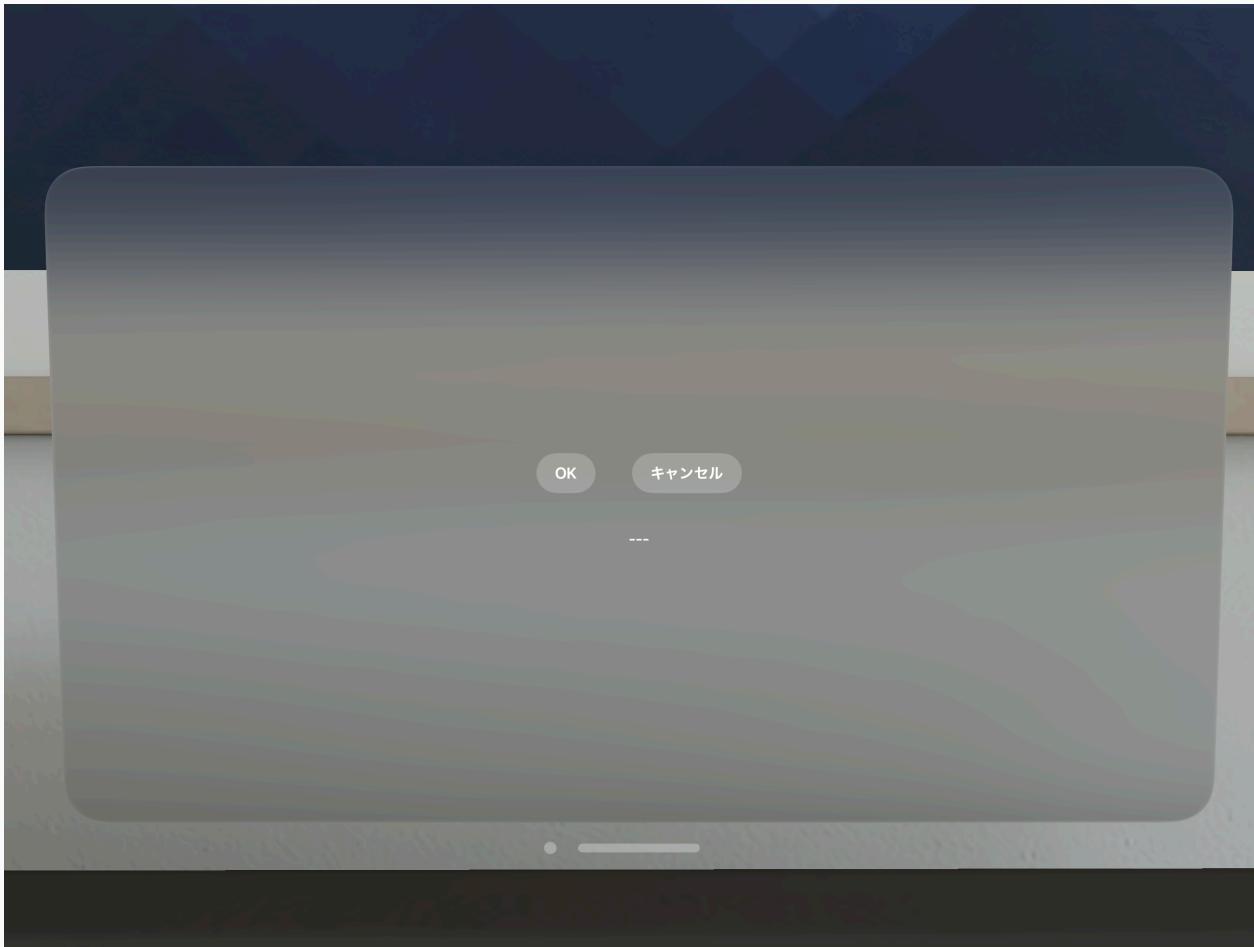


padding があると選  
択ミスしにくい

ボタンの形が見える  
ほうが押しやすい

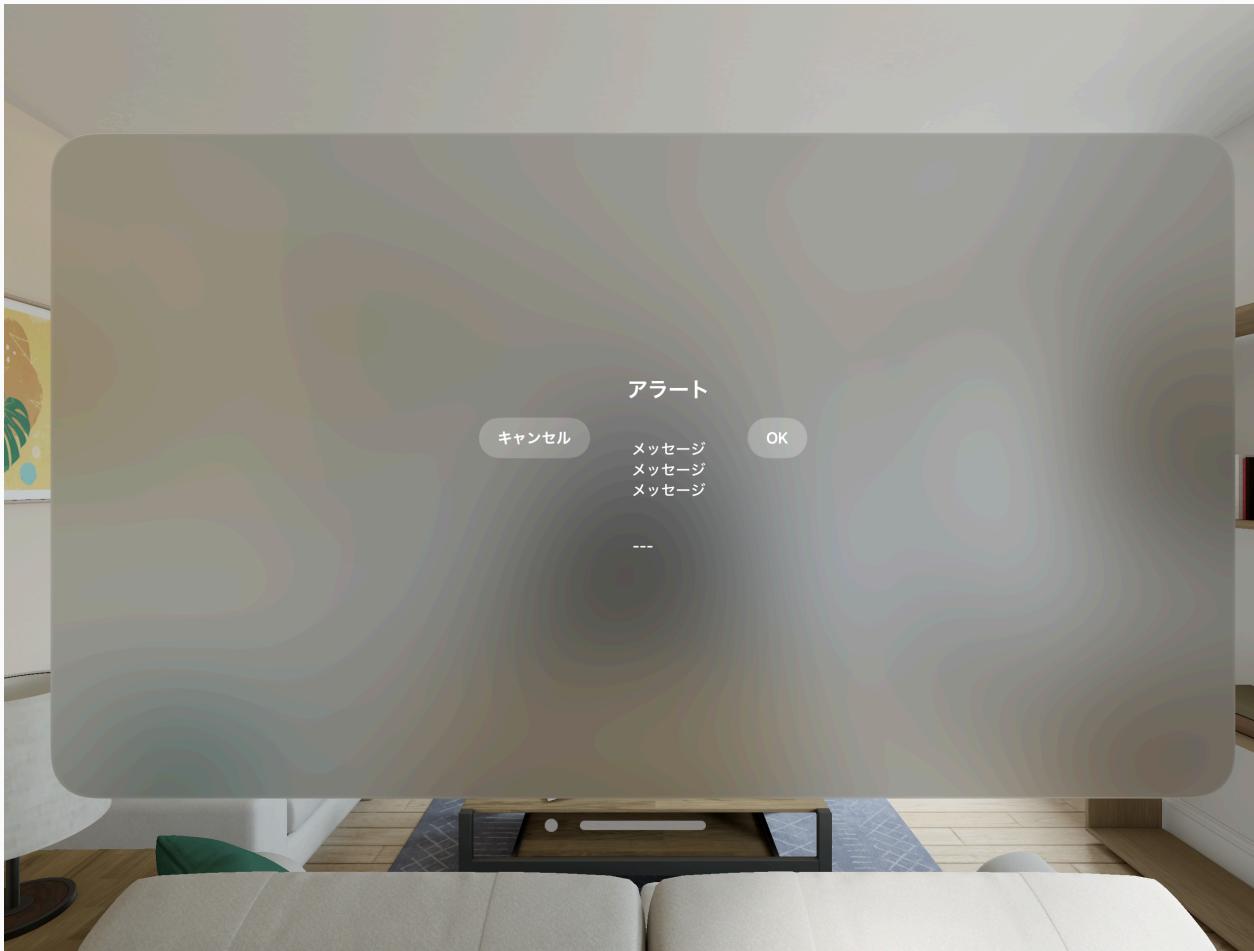
上下の選択は難しい  
(個人の感想)

# HStack の場合



左右の選択は上下より  
楽（個人の感想）

# いっそのことこんな感じで



アラートの左右にボタンを置く  
選択を間違えることはない  
UI がダサい感じはある（ボタン増えたらどうするの）

# まとめ



# まとめ

- 視線と注視によるフォーカス操作は意外と難しい
- どんな UI が良いのかはまだ検討の余地がありそう