# " FLUTTER ANIMATIONS "

FROM BASIC MOTION TO SMOOTH UX

Bahodirov A.A.
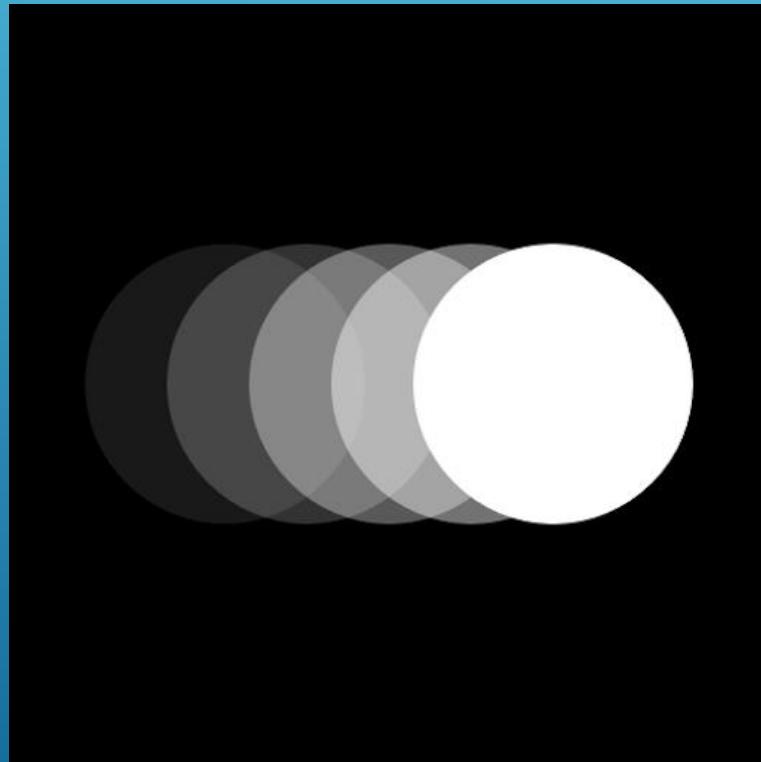
ANIMATSIYALAR NIMA UCHUN MAVJUD:
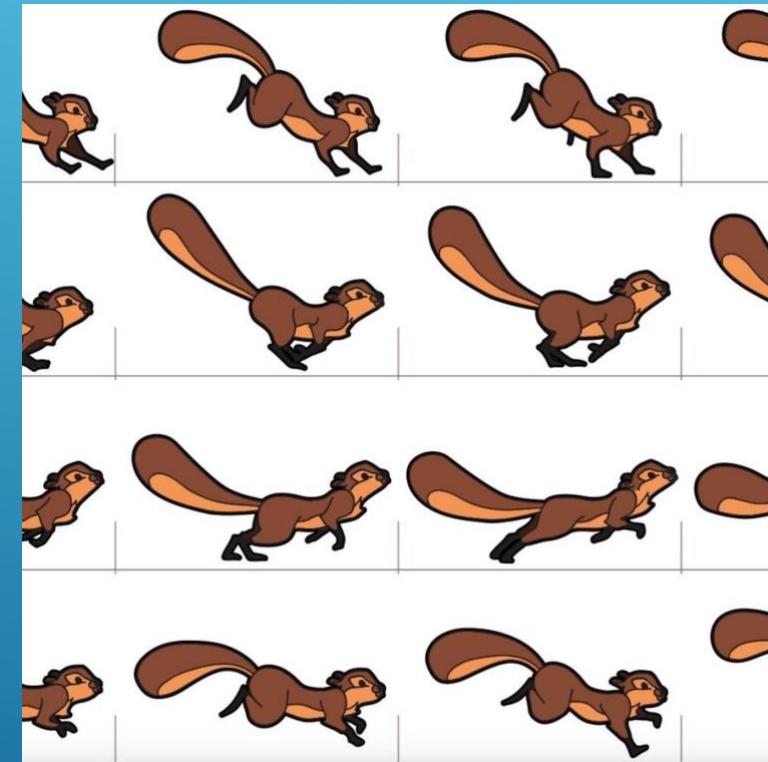
- Dekoratsiya emas
- Holat o'zgarganini ko'rsatish uchun
- Samaradorlikni yaxshilash

Animatsiyalar vizual o'yinchoqlar emas, balki UX vositalaridir.

EXPLICIT VS IMPLICIT ANIMATION

Siz aniq hamma narsani boshqarasiz

Siz nima qilishni flutterga aytasiz, u nima hohlasa o'sha..

IMPLICIT (YAXSHIRIN) ANIMATSIYALAR NIMA UCHUN YAXSHI

- Oddiy transitionlar
- Kichkina UI o'zgarishlar

EXPLICIT (ANIQ) ANIMATSIYALAR NIMA UCHUN YAXSHI
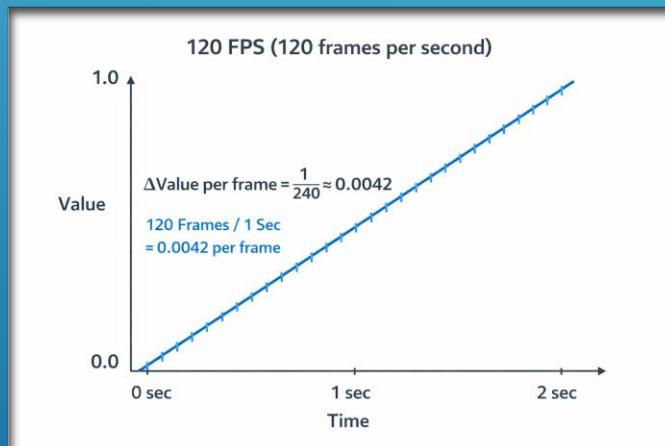
- To'liq nazorat
- Aniqlik va tahmin qilib bo'ladigan

CHEKLOVLAR

- Pauza yo'q
- Teskari logika yo'q
- Bir nechta animatsiyalarni sinxronlashtirish qiyin

- Moslashuvchan emas
- Yasash qiyinroq
- Kamroq realistik harakat

ANIMATIONCONTROLLER NIMA?

120 FPS (120 frames per second)

- Vaqtni Boshqaradi
- 0 → 1 Dan Qiymatlarni Hosil Qiladi

Pult deb hisoblash mumkin, chunki...

```
[CREATED] (value=0.0)

   |

   v

[FORWARD] ---> [COMPLETED] ---> [REVERSE] ---> [DISMISSED]

   |              |                |

   |              |                v

   |              |           [STOP/PAUSE]

   |              |                |

   --------------<--------------
```

```dart
class MyHomePage extends State<MyHomePage> with SingleTickerProviderStateMixin
{ late final AnimationController controller;

  @override
  void initState() {
    super.initState();
    controller = AnimationController(
      duration: Duration(seconds: 2),
      vsync: this,
    );
  }

  @override
  void dispose() {
    controller.dispose();
    super.dispose();
  }
}
```

AnimationController Methods

.forward() => ishga tushirish

.reverse() => tugagandan keyin ortga qaytarish

.reset(), .repeat(), .stop()

.addListener(() {})

.addStatusListener((status) {})

.removeListener(() {})

.removeStatusListener((status) {})

```
enum AnimationStatus {
  dismissed,
  forward,
  reverse,
  completed,
}
```

CURVES & PERCEPTION

PERFORMANCE & ANIMATION SAFETY
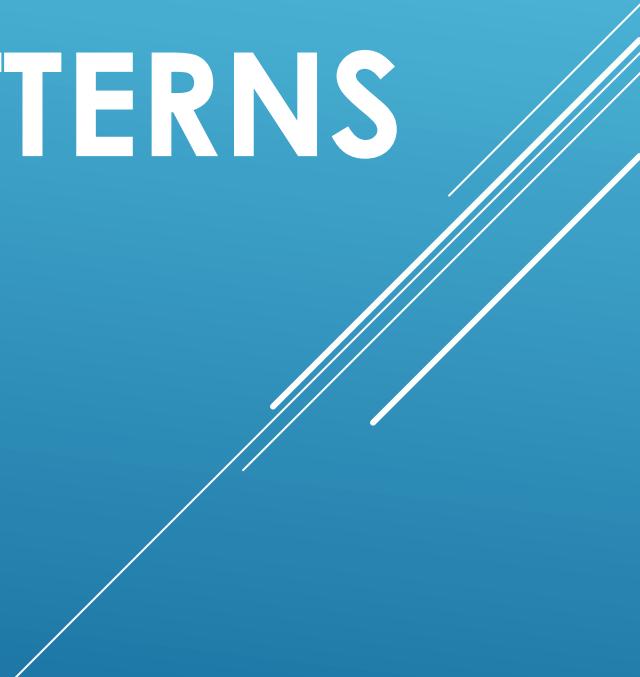
Cheap Animations:

- Opacity
- Transform

(Moving, scaling, rotating, or skewing)

Expensive Animations:

- Layout changes

(position, constraints, padding, align)

- Size changes in list
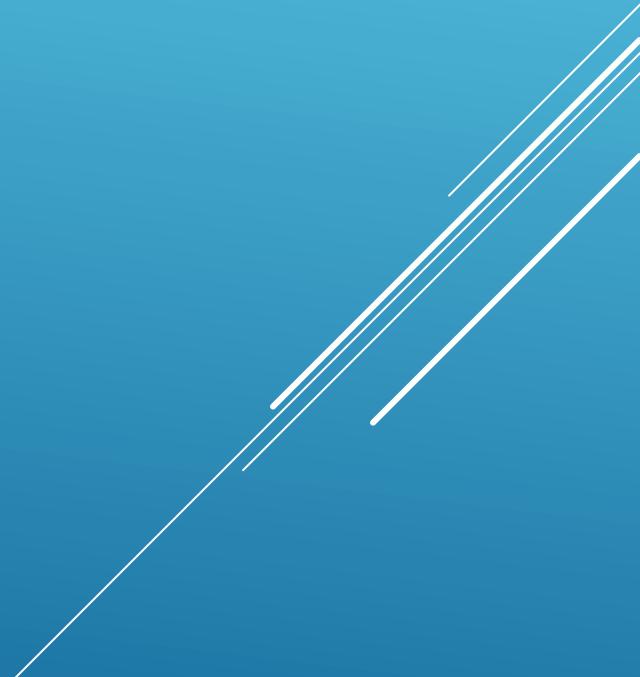
# REAL-WORLD USAGE PATTERNS

PAGE TRANSITIONS
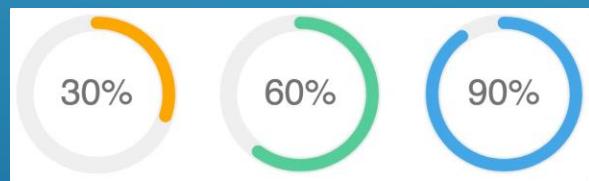
```
pageBuilder: (context, state) => CustomTransitionPage(
    key: state.pageKey,
    child: const SearchScreen(),
    transitionsBuilder: (context, animation, secondaryAnimation, child)
{
        const begin = Offset(1.0, 0.0);
        const end = Offset.zero;
        final tween = Tween(begin: begin, end: end);
        final offsetAnimation = animation.drive(tween);

        return SlideTransition(
            position: offsetAnimation,
            child: child,
        );
    },
),
```

Hero

LOADING INDICATORS

Flutter default
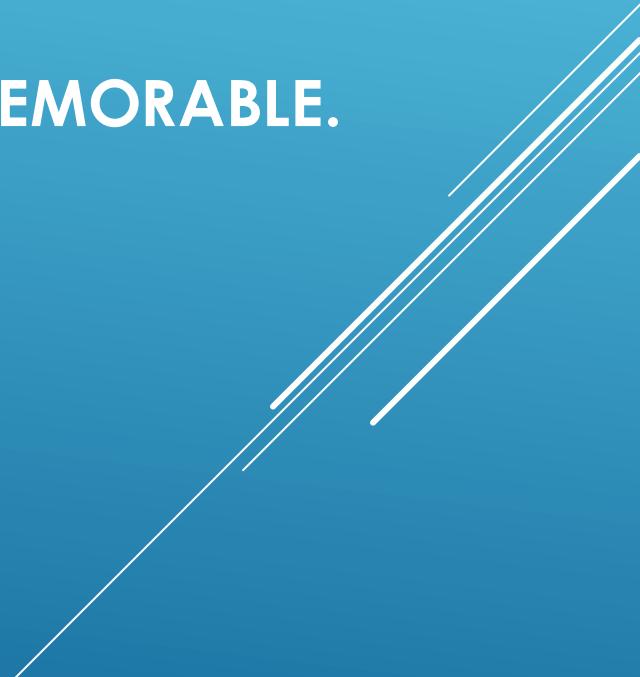
Flutter packages:

percent_indicator

simple_circular_progress_bar

Lottie

COMMON MISTAKES

- Animating everything

- Long durations

- Ignoring performance

- Choose the right animation type

- Control time when needed

- Animate with purpose

GOOD ANIMATIONS ARE INVISIBLE. BAD ONES ARE MEMORABLE.

END OF PRESENTATION