Technical Requirements – Haunted Theme Switcher

Challenge is accessible on Memberspot: https://pl-coding.mymemberspot.io/library/jx3b7Qik9ip5qpNI8IF2/FgYR5e2HfPtserJY1udz/PtYdgGS0zpL3RW2e8XB0/details

🎭 Scenario

This challenge simulates a day/night theme switcher in a Halloween-inspired style. Tapping the custom pumpkin toggle triggers a sequence of animations that transition the scene between day and night.

🎨 Figma Mockups

https://www.figma.com/design/jBfIfs7N6IphZE3Ue03Lot/Android-Halloween-Lab?node-id=2010-1429

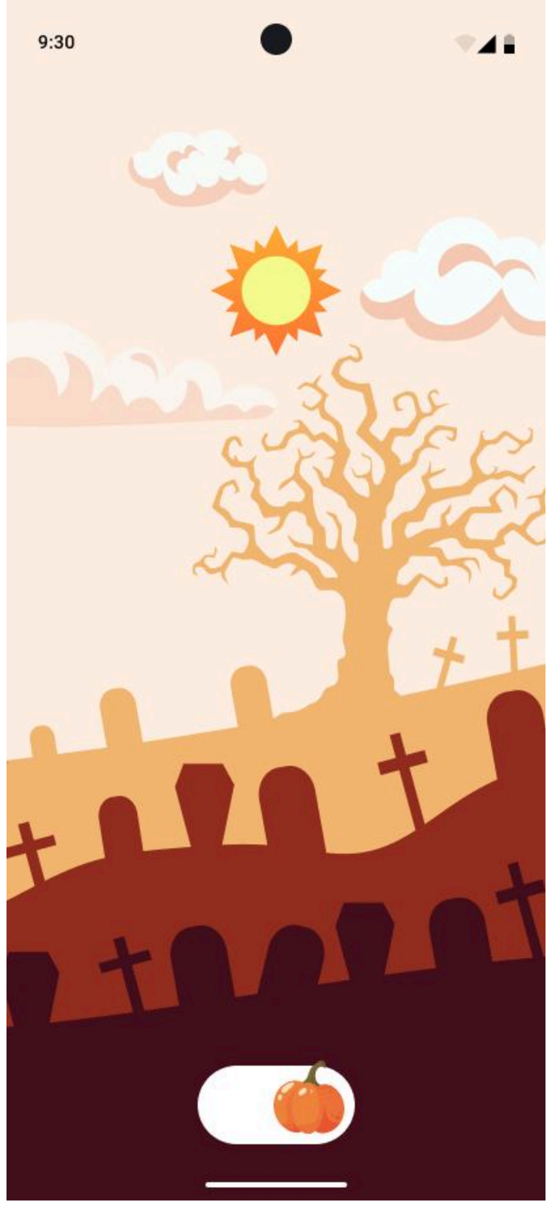
🎯 Feature Goal

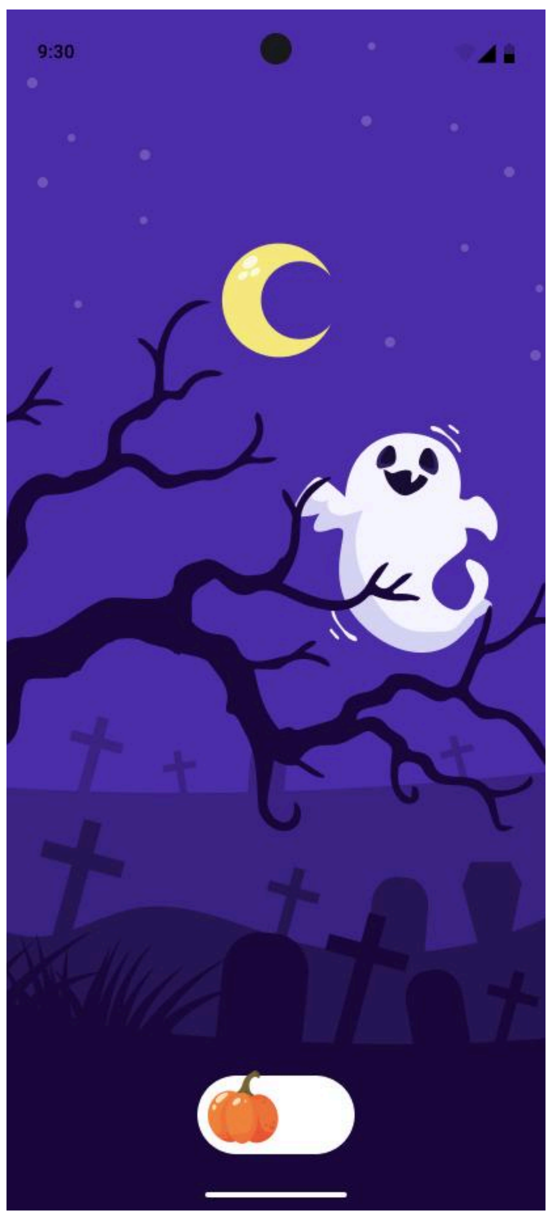
Implement a smooth animated transition between two states — "Day" and "Night" — according to the following animation rules.

📌 Requirements

Transition Day → Night

- The **sun** moves downward to the right along a curved path and disappears.
- Simultaneously, the **moon** moves upward from the left along a mirrored curved path and replaces the sun.
- **Clouds** slide outward in opposite directions (left and right) while gradually fading out.
- The **graveyard background** slides to the right, and a new night version of the graveyard (with a different tree and scenery) appears.
- A **semi-transparent dark overlay** fades in. Overlay must cover only the graveyard, while all other elements (sun, moon, clouds, stars, ghost) must appear above it.
- The overlay color is **black (#000000)** with 50% transparency.
- **Stars** fade in across the sky.
- The **ghost** appears statically at a fixed point with a scale-in effect.
- The ghost should start at 50% scale and smoothly grow to 100% when appearing.



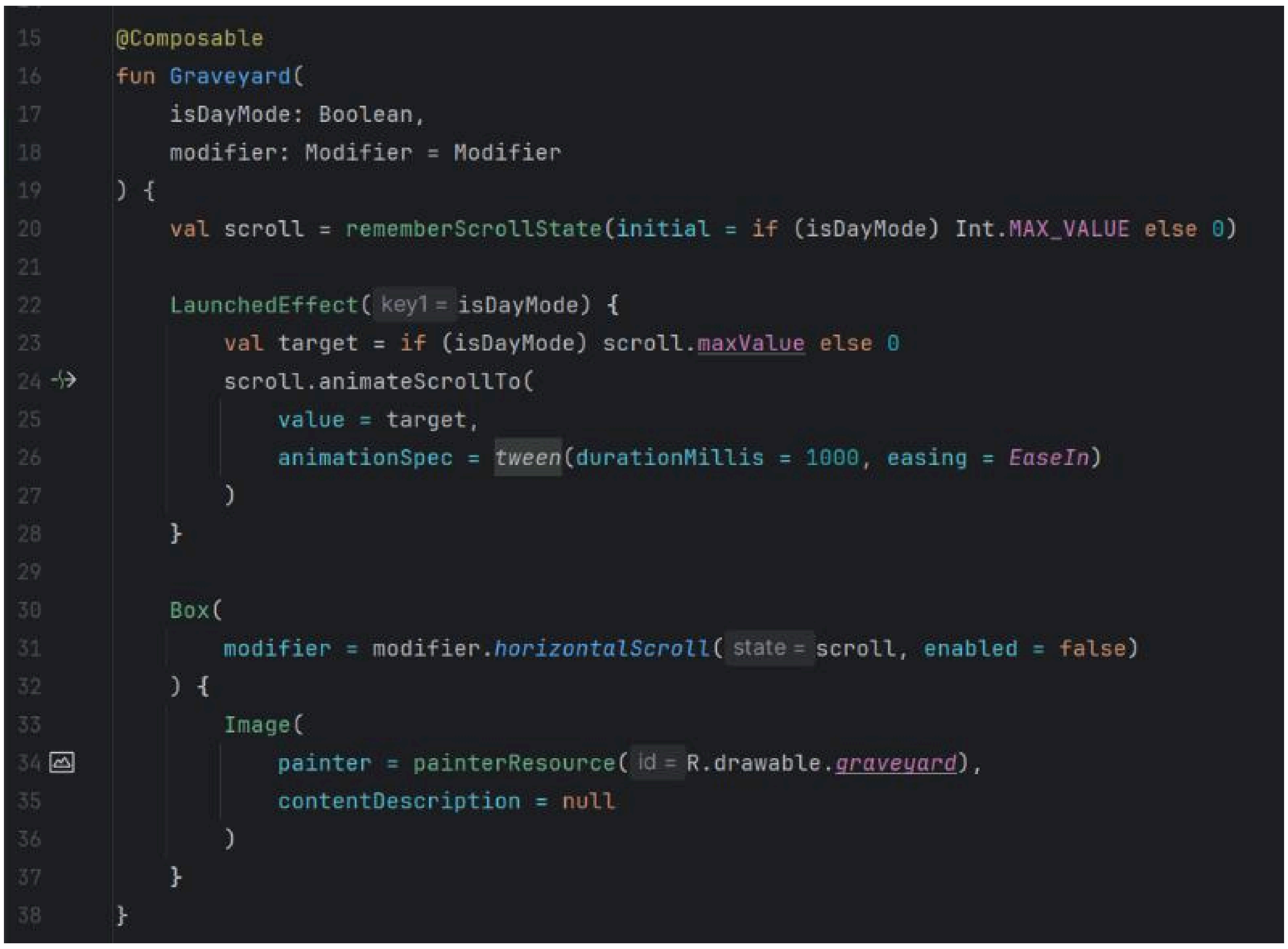


Transition Night → Day

- The **moon** moves downward to the left along its curved path and disappears.
- Simultaneously, the **sun** moves upward from the right along its original curved path to reclaim its position.
- **Stars** fade out.
- The ghost fades out by shrinking from 100% to 50%.
- The **graveyard background** slides to the left, and the day version of the scenery reappears.
- **Clouds** slide back into the scene from the sides while gradually fading in.
- The **dark overlay** fades out.

Additional Animation Notes

- All animations must complete within **1 second**.
- Elements should animate smoothly and synchronously.
- All elements that disappear off-screen (e.g., sun, moon, clouds) must move by a distance equal to the **screen width**. The screen width can be obtained from LocalConfiguration.current.
- The semi-transparent dark overlay must cover only **the graveyard**, while all other elements (sun, moon, clouds, stars, ghost) must appear above it.
- For the Graveyard Background, wrap the element in a **horizontal scrollable container** (e.g., a Box with scrolling) to implement smooth sliding of the scenery during transitions. Example:



```
@Composable
fun Graveyard(
    isDayMode: Boolean,
    modifier: Modifier = Modifier
) {
    val scroll = rememberScrollState(initial = if (isDayMode) Int.MAX_VALUE else 0)

    LaunchedEffect( key1 = isDayMode) {
        val target = if (isDayMode) scroll.maxValue else 0
        scroll.animateScrollTo(
            value = target,
            animationSpec = tween(durationMillis = 1000, easing = EaseIn)
        )
    }

    Box(
        modifier = modifier.horizontalScroll( state = scroll, enabled = false)
    ) {
        Image(
            painter = painterResource( id = R.drawable.graveyard),
            contentDescription = null
        )
    }
}
```

🤔 What's Allowed?

- Standard Android/Jetpack libraries
- No 3rd party libraries are allowed or would be required to complete this challenge
- Use Jetpack Compose Animation APIs (animateFloat, fadeIn/fadeOut, scaleIn/scaleOut, AnimatedVisibility).

⚠️ What's not important

- Responsiveness across every device size or orientation is not mandatory.
- Exact motion curves (smooth interpolation is enough).
- Additional effects like sound or parallax.

🔗 Useful Links for This Challenge

- Full Guide to Jetpack Compose Animations
- Quick Guide to Animations in Compose
- Value-based Animations
- Jetpack Compose Animation Codelab
- Choosing the Right Animation API in Compose
- Vector drawables overview
- How You Use an AI Coding Agent

🏆 Submission & Rewards

- Successfully submitting this challenge via the /submit-challenge command on Discord grants you **100 XP**.
- Your submission must include:
    a. A link to a Gist with your implementation
    b. A screen recording (max 20 seconds) showing:
        - Initial daytime state.
        - Transition Day → Night with all animations.
        - Transition Night → Day back to daytime.