🎃 Spooktacular Flutter Challenge 👻

Animated Halloween Storybook — Group Challenge #1 (V.4)

Due Date: October 6, 2025 — 7:30 PM

Teams: Collaborate with your Project Group Members via GitHub

Prizes: Top 2 on the grid and Top Graduate will be awarded!

Objective

The primary objective of this challenge is to design, build, and deploy a graphical, interactive, and multi-page Halloween storybook experience using Flutter and Dart. This project requires leveraging Flutter's widget-based architecture, state management, and animation capabilities to create a seamless and engaging user experience. Your team must harness the power of custom widgets, animated transitions, and responsive layouts to deliver a fully functional, visually compelling, and spooky narrative that feels alive across multiple devices.

Key Expectations

- Multi-Page Navigation: Implement smooth navigation using Flutter's Navigator and routing system.
- Interactive Graphics: Use Flutter's CustomPainter, Canvas, and animation libraries to create dynamic, spooky visuals.
- Animations & Transitions: Apply AnimatedContainer, Hero animations, and Implicit/Explicit animations for immersive storytelling.
- Responsive Design: Ensure the storybook adapts to different screen sizes and orientations.
- Performance Optimization: Optimize rendering for a fluid experience, even with complex graphics.

Instructions & Project Overview

Develop a fun, tricky Halloween storybook with interactive elements that keep the user on their toes! Players will need to find the "correct" item amidst spooky Halloween-themed objects moving around the screen. You can add some selections, create some spooky reactions with sound effects, etc.

Features to Include

- Animated Spooky Characters: Add Halloween characters or objects (like ghosts, pumpkins, bats, etc.) that move around the screen.
- Interactive Traps: Some items should be "traps" that, when clicked, play a spooky or jump scare sound effect to surprise the player.
- Background Music: Loop a spooky Halloween background soundtrack.
- Winning Element: Hide the "correct" item among the moving objects. When clicked, play a festive Halloween sound and display a "You Found It!" message.

Development Steps

1. Set Up: Start a new Flutter project.
2. Design the Layout: Place spooky elements on the screen in a way that engages the player. Ensure some elements are marked as traps.
3. Implement Animations: Use animations to make objects appear as if they're floating, glowing, or moving randomly.
4. Sound Integration: Add background music, jump scare sounds for traps, and a success sound for the correct item.
5. Testing: Test gameplay to make sure that sounds, animations, and interactions work seamlessly.
6. Customization & Creativity: Make it spooky, fun, and visually appealing. Add more challenging elements if time permits, such as items that change position or additional jump scare effects.

Submission

- The GitHub repository link with your code.
- The APK file of your completed app.

Grading Rubric (100 Points Total)

Criteria	Points
Theme & Visual Design	20
Animation & Interactivity	20
Code Quality & GitHub Collaboration	20
Functionality & User Experience	20
Creativity & Originality	10
Presentation & Submission	10

Tips to Make It Top-Notch

- Stick to a cohesive Halloween theme throughout.
- Use smooth animations and transitions.
- Organize your code and collaborate effectively on GitHub.
- Add immersive sound effects and music.
- Ensure responsiveness across devices.
- Test thoroughly for seamless interactions.
- Include creative twists like hidden surprises or spooky traps.

Design Theme & Organization

Projects should be well-organized, visually engaging, and thematically consistent. The top designs will be those that feel like they could be brought to life — think spooky, creative, and interactive!

Helpful Hints & Resources

- Flutter Documentation: https://docs.flutter.dev
- Dart Language Guide: https://dart.dev/guides
- Flutter Animation Samples: https://flutter.github.io/samples/#animations
- GitHub Collaboration Tips: https://docs.github.com/en/get-started/quickstart/github-flow