```
#include <FastLED.h>

#define LED_PIN 6
#define NUM_LEDS 60
#define BRIGHTNESS 64
#define FSR_PIN A0
#define BUTTON_PIN 2  // A button to start the game
#define BUTTON_LED 13  // Built-in LED to indicate the game state
#define HIT_THRESHOLD 600
#define HIT_TIMEOUT 3000  // 3 seconds

CRGB leds[NUM_LEDS];
boolean gameRunning = false;
unsigned long gameStartTime = 0;
unsigned long lastHitTime = 0;
boolean buttonState = false;
boolean lastButtonState = false;
```

```
void setup() {
  FastLED.addLeds<WS2812, LED_PIN, GRB>(leds, NUM_LEDS);
  FastLED.setBrightness(BRIGHTNESS);
  pinMode(FSR_PIN, INPUT);
  pinMode(BUTTON_PIN, INPUT_PULLUP);
  pinMode(BUTTON_LED, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  buttonState = digitalRead(BUTTON_PIN);

  if (!gameRunning) {
    if (buttonState && !lastButtonState) {
      startGame();
    }
  } else {
    unsigned long currentTime = millis();
    int fsrValue = analogRead(FSR_PIN);

    if (fsrValue >= HIT_THRESHOLD) {
      setLedColor(CRGB::Green, 100); // Successful hit (blinking green)
      lastHitTime = currentTime;
```

```
    } else {
      if (currentTime - lastHitTime > HIT_TIMEOUT) {
        // No hit within the timeout (blinking red)
        setLedColor(CRGB::Red, 100);
      } else {
        // Waiting to hit (solid white)
        setLedColor(CRGB::White);
      }
    }

    if (currentTime - gameStartTime >= 180000) { // 3 minutes
      endGame();
    }
  }

  lastButtonState = buttonState;
}

void startGame() {
  gameRunning = true;
  gameStartTime = millis();
  lastHitTime = gameStartTime;
  digitalWrite(BUTTON_LED, HIGH); // Turn on the built-in LED
```

```
}

void endGame() {
  gameRunning = false;
  digitalWrite(BUTTON_LED, LOW); // Turn off the built-in LED
  fill_solid(leds, NUM_LEDS, CRGB::Black); // Turn off all LEDs
  FastLED.show();
}

void setLedColor(CRGB color, int delayMs = 0) {
  fill_solid(leds, NUM_LEDS, color);
  FastLED.show();
  if (delayMs > 0) {
    delay(delayMs);
    fill_solid(leds, NUM_LEDS, CRGB::Black);
    FastLED.show();
  }
}
```