```
import React, {ReactNode} from 'react';
import {StyleSheet} from 'react-native';
import {Gesture, GestureDetector} from 'react-native-gesture-handler';
import Animated, {
  useAnimatedStyle,
  useSharedValue,
} from 'react-native-reanimated';

interface IPropsDraggableView {
  children: ReactNode;
}

const DraggableView: React.FC<IPropsDraggableView> = ({children}) => {
  const x = useSharedValue(0);
  const y = useSharedValue(0);
  const context = useSharedValue({x: 0, y: 0});

  const panGesture = Gesture.Pan()
    .onStart(() => {
      context.value = {x: x.value, y: y.value};
    })
    .onUpdate(event => {
      x.value = event.translationX + context.value.x;
      y.value = event.translationY + context.value.y;
    })
    .onEnd(() => {});

  const panStyle = useAnimatedStyle(() => {
    return {
      transform: [{translateX: x.value}, {translateY: y.value}],
    };
  }, [x.value, y.value]);

  return (
    <GestureDetector gesture={panGesture}>
      <Animated.View style={[styles.draggableView, panStyle]}>
        {children}
      </Animated.View>
    </GestureDetector>
  );
};
const styles = StyleSheet.create({
  draggableView: {position: 'absolute'},
});

export default DraggableView;
```