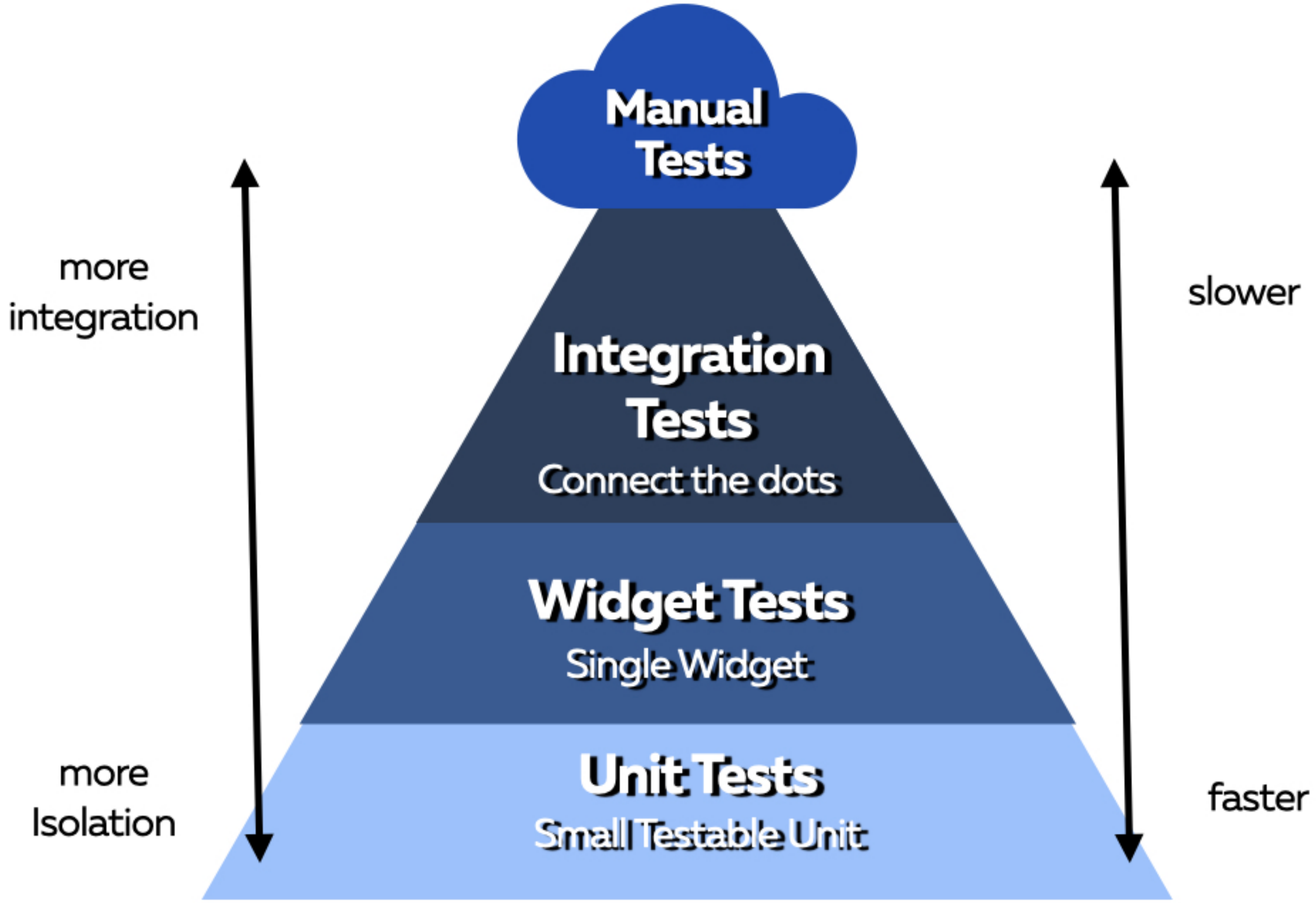
Flutter UITest

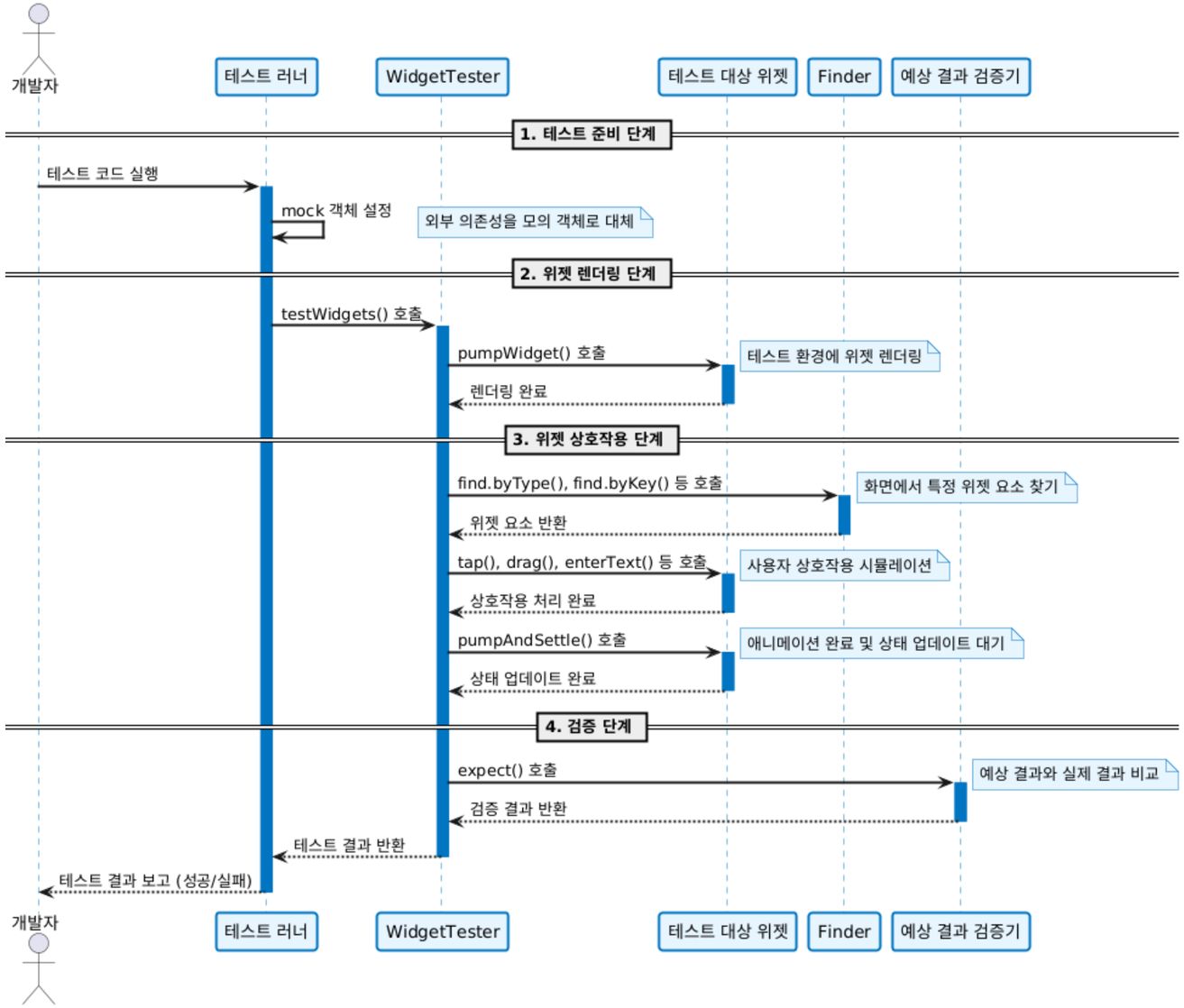
What to Test and How in UI Testing

2025.04.08 suojae

What is UITest?



How To UITest In Flutter

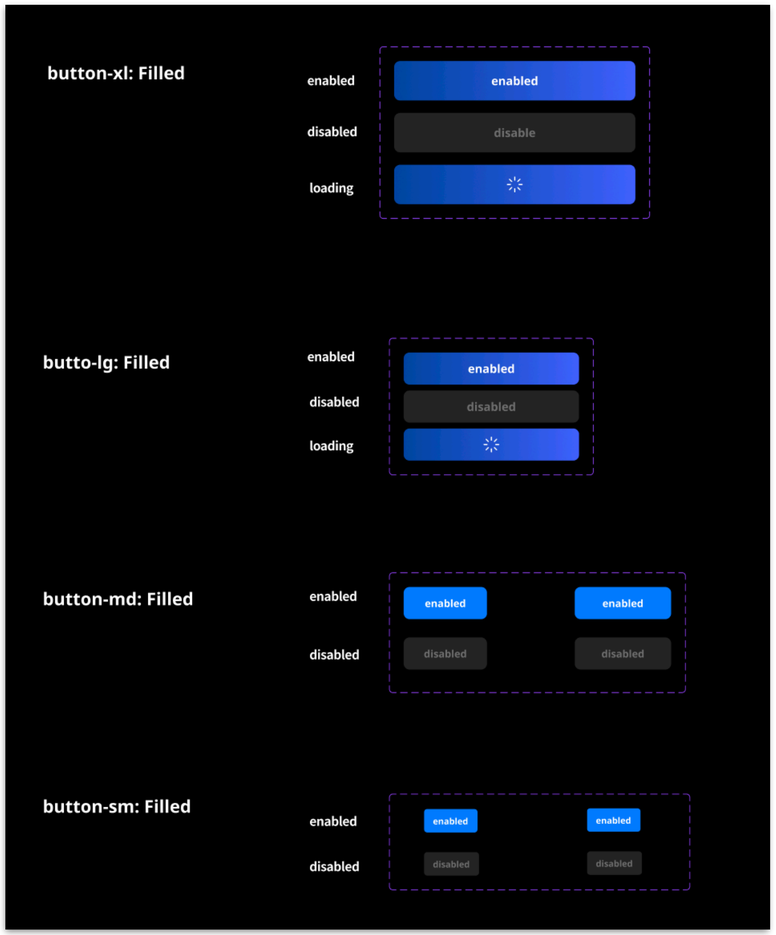


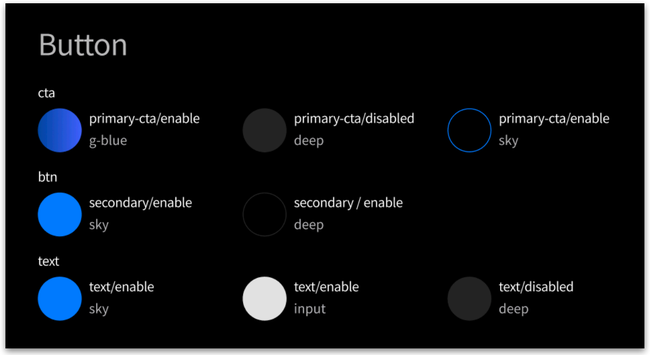
How To UITest In Flutter

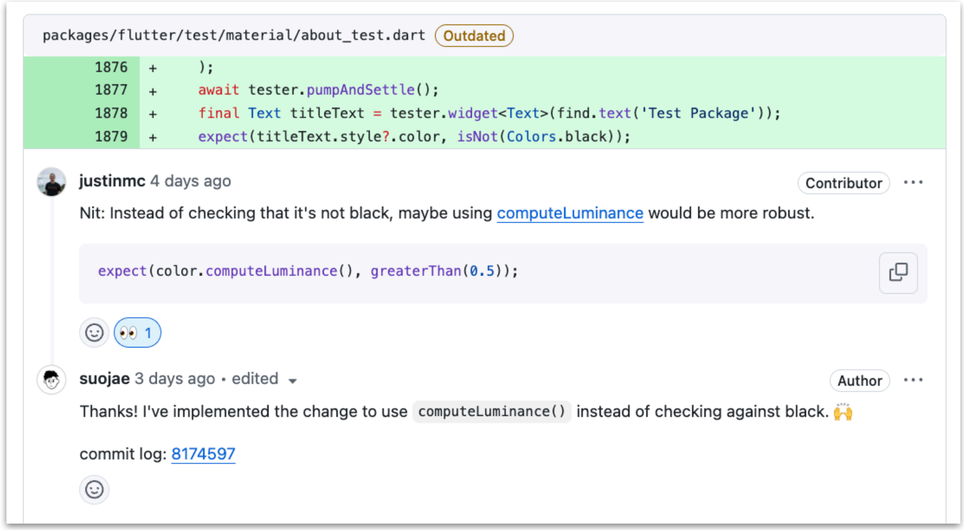
```dart
class CounterWidget extends StatefulWidget {
  final String title;

  const CounterWidget({super.key, required this.title});

  @override
  _CounterWidgetState createState() => _CounterWidgetState();
}

class _CounterWidgetState extends State<CounterWidget> {
  int _counter = 0;

  void _incrementCounter() => setState(() => _counter++);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(widget.title),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            const Text('버튼을 눌러 숫자를 증가시키세요:'),
            Text('$_counter',
              style: Theme.of(context).textTheme.headlineLarge,
              key: const Key('counter_value'),
            ),
          ],
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: _incrementCounter,
        tooltip: '증가',
        child: const Icon(Icons.add),
        key: const Key('increment_button'),
      ),
    );
  }
}
```

```dart
void main() {
  // 1. 테스트 준비 단계
  group('CounterWidget 테스트', () {
    testWidgets('버튼을 누르면 카운터가 증가해야 함', (WidgetTester tester) async {
      // 2. 위젯 렌더링 단계
      await tester.pumpWidget(const MaterialApp(
        home: CounterWidget(title: '카운터 앱 테스트'),
      ));

      // 초기 상태 확인
      expect(find.text('0'), findsOneWidget);
      expect(find.text('1'), findsNothing);

      // 3. 위젯 상호작용 단계
      // 특정 키를 가진 버튼 찾기
      final incrementButton = find.byKey(const Key('increment_button'));

      // 버튼 탭 동작 수행
      await tester.tap(incrementButton);

      // 애니메이션과 상태 업데이트 처리 대기
      await tester.pumpAndSettle();

      // 4. 검증 단계
      // 카운터 값이 증가했는지 확인
      expect(find.text('0'), findsNothing);
      expect(find.text('1'), findsOneWidget);
    });

    testWidgets('카운터 값이 올바르게 표시되어야 함', (WidgetTester tester) async {
      // 2. 위젯 렌더링 단계
      await tester.pumpWidget(const MaterialApp(
        home: CounterWidget(title: '카운터 앱 테스트'),
      ));

      // 3. 위젯 상호작용 단계
      // 카운터 값 요소 찾기
      final counterTextWidget = find.byKey(const Key('counter_value'));

      // 현재 텍스트 확인
      expect((tester.widget(counterTextWidget) as Text).data, '0');

      // 버튼 세 번 탭하기
      final incrementButton = find.byKey(const Key('increment_button'));
      await tester.tap(incrementButton);
      await tester.pump();
      await tester.tap(incrementButton);
      await tester.pump();
      await tester.tap(incrementButton);
      await tester.pumpAndSettle();

      // 4. 검증 단계
      // 카운터 값이 3이 되었는지 확인
      expect((tester.widget(counterTextWidget) as Text).data, '3');
    });
  });
}
```

Personal Widget Test Tip







Thanks!

2025.04.08 suojae