

Model checking Flutter GUI Application

202173154 송승현 Dependable System Lab

Index

- 1. Issue**
- 2. Problem**
- 3. Related Work**
- 4. Suggested Solution**
- 5. Demonstration**

1.Issue

Flutter

2018년 Google에서 발표한 Multi-Platform GUI Framework

- 하나의 소스코드로 Window, Mac, Android, iOS, Web, Linux 6가지의 운영체제에서 동일한 GUI 어플리케이션을 작성 가능함
- **Widget**이라는 UI component로 GUI 어플리케이션을 구성함
- State를 사용하여 GUI의 변화를 관리함

1.Issue

Flutter & Testing

Flutter는 Unit Testing, Widget Testing, Integration Testing을 통해 GUI 어플리케이션에 대한 테스트를 지원하고 있음



Unit Testing

- 단수의 함수 혹은 클래스에 대한 테스트



Widget Testing

- Widget 단위에 대한 테스트



Integration Testing

- 전체 App 혹은 App에서 커다란 기능, 부분에 대한 테스트

1.Issue

Flutter & Testing

	Unit	Widget	Integration
Confidence	Low	Higher	Highest
Maintenance	Low	Higher	Highest
Dependencies	Few	More	Most
Execution speed	Quick	Quick	Slow

표1 .Flutter Framework 테스트 별 특징

2.Problem

GUI Testing

GUI Testing이 어려운 이유



Size

- 메모장, Stick Note와 같은 작은 프로그램도 300개 이상의 작업과 100여개 이상의 화면 상태를 가짐. 이는 어플리케이션의 규모가 커질수록 기하급수적으로 늘어남



Sequence

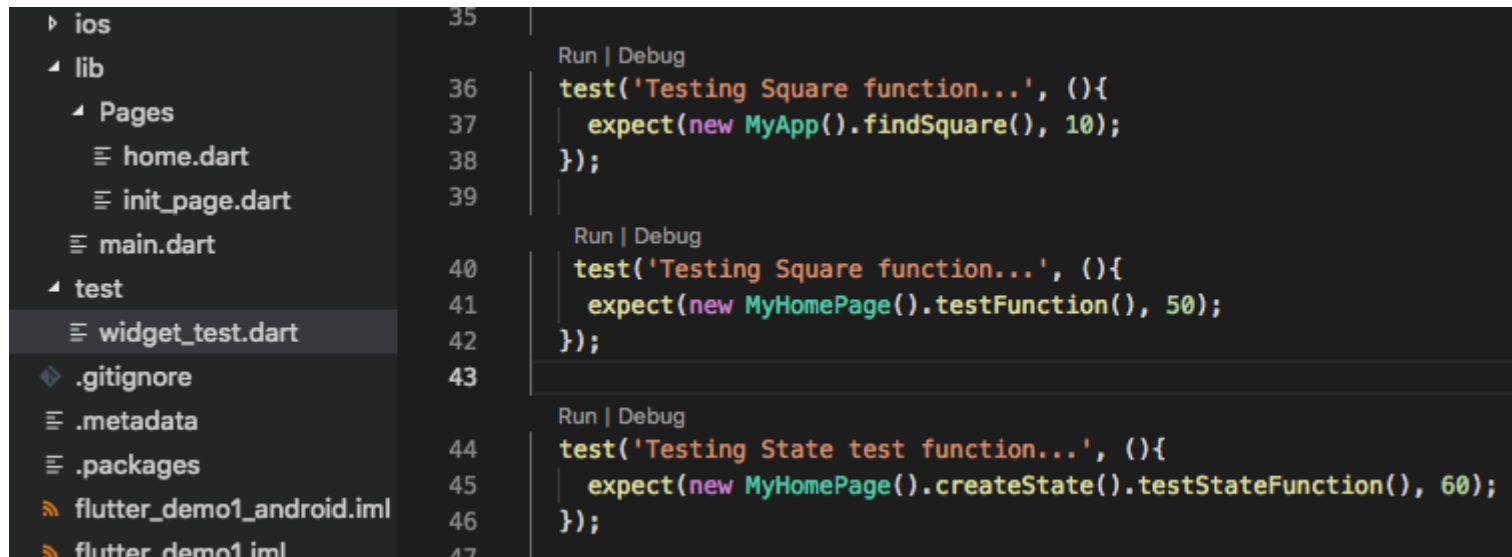
- GUI의 특징 중 하나로, 특정 GUI에 도달하려면 앞에서 일련의 작업을 진행시켜야만 도달 가능한 상태가 존재함.

2.Problem

Flutter GUI Testing

개발자가 작성하는 user scenario에 의존적인 Testing

Flutter의 테스트 또한 일반적인 GUI Testing과 마찬가지로, 개발자가 설계한 user scenario에 의존하여 test case를 작성함에 따라 GUI 어플리케이션이 가지는 다양한 상태를 모두 커버하지 못함.



```
35
36
37
38
39
40
41
42
43
44
45
46
47

Run | Debug
test('Testing Square function...', (){
  expect(new MyApp().findSquare(), 10);
});

Run | Debug
test('Testing Square function...', (){
  expect(new MyHomePage().testFunction(), 50);
});

Run | Debug
test('Testing State test function...', (){
  expect(new MyHomePage().createState().testStateFunction(), 60);
});
```

그림1 . Flutter Widget Testing

3. Related Work

JPF-AWT: Model checking GUI applications

2011 26th IEEE/ACM International Conference on Automated Software Engineering (ASE 2011)

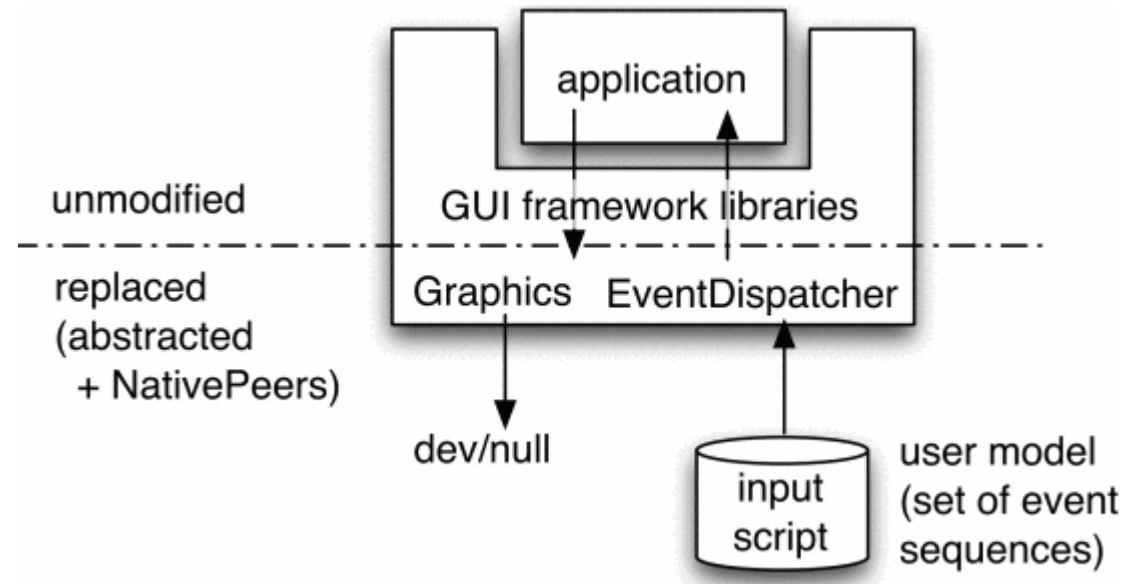


그림2 . JPF-AWT Architecture

3. Related Work

JPF-AWT: Model checking GUI applications

2011 26th IEEE/ACM International Conference on Automated Software Engineering (ASE 2011)

User Input Modeling

ANY

모델 검사기가 탐색할 non-deterministic input choice

REPEAT

Event Sequence에 대한 loop를 특정하기 위한 Constructs

NONE

다음 이벤트로 넘어가는 간단한 이벤트

```
// start the data acquisition thread
$acquire_status.doClick()

// enter command into "Command" text field
$Command:input.setText("turn")

// select options checkbox combinations
REPEAT 3 {
  ANY { NONE, $<FORCED|QUEUED|SINGLE_STEP>.doClick() }
}

// select a target robot (list selection)
ANY { $Robots:list.setSelectedIndex([0-3]) }

// click on "Send" button
$Send.doClick()
```

그림3 . RobotManager Input Script

3. Related Work

JPF-AWT: Model checking GUI applications

2011 26th IEEE/ACM International Conference on Automated Software Engineering (ASE 2011)

Java PathFinder의 확장인 JPF - AWT 발표

사용자의 input sequence를 지정하는 스크립트가 주어지면 JPF-AWT는 즉시 해당 어플리케이션을 확인 가능

기존의 GUI 테스트 방법에 비하여 탐색 가능한 상태와 시나리오의 개수가 대폭 증가함
(user scenario 370 -> 28,000, 12 previously unknown errors)

그림2 . RobotManager Input Script

4.Suggested Solution

5. Demonstration