

Detox

A Year in. Building It, Testing With It.

Rotem Mizrachi-Meidan

April 2018



rotemm@wix.com



[@rotemmiz](https://twitter.com/rotemmiz)



medium/@rotemmiz



[github.com/
rotemmiz](https://github.com/rotemmiz)

Hi.

I am Rotem

Software Engineer at Wix.com



01

Flakiness

Flakiness

- Tests sometimes fail for no apparent reason
- Tests are nondeterministic tasks run in different order than the app
- Unclear when the app finished handling user interaction
- Users have to deal with synchronization manually

;

```
...
expectElementToNotExist: async testId => {
  let element;
  for (let i = 0; i < 20; i++) {
    element = await driver.getElementIfExists(testId,
50);
    if (element) {
      // if it's still there check back in a little
bit
      await driver.sleep(50);
    } else {
      element = undefined;
      break;
    }
  }
  expect({element, testId}).not.toBeAValidElement();
},
...
```

Flakiness

q - probability of a failed test
n - number of tests

$$1 - (1 - q)^n$$

$$1 - (1 - 0.005)^{20} = 9.5\%$$

$$1 - (1 - 0.005)^{50} = 22.1\%$$

$$1 - (1 - 0.005)^{100} = 39.4\%$$

- Flaky
no reason
- Slow
- Complex
chance

Black Box



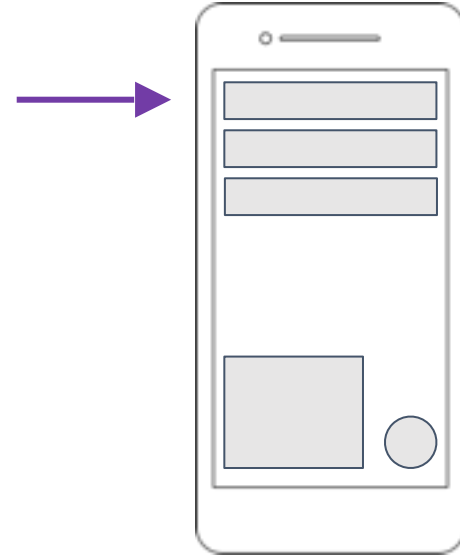
“Black-box testing is a method of [software testing](#) that examines the functionality of an application without peering into its internal structures or workings.”

- Wikipedia

Black Box

Query only resources/APIs provided by the OS.

In this case,
UIAutomator is a viewer that inspects layout
hierarchy, supplied by Android OS.



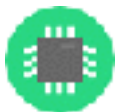
02

Detox

Gray box, not black box



Runs in the same process



Access to memory read the application state



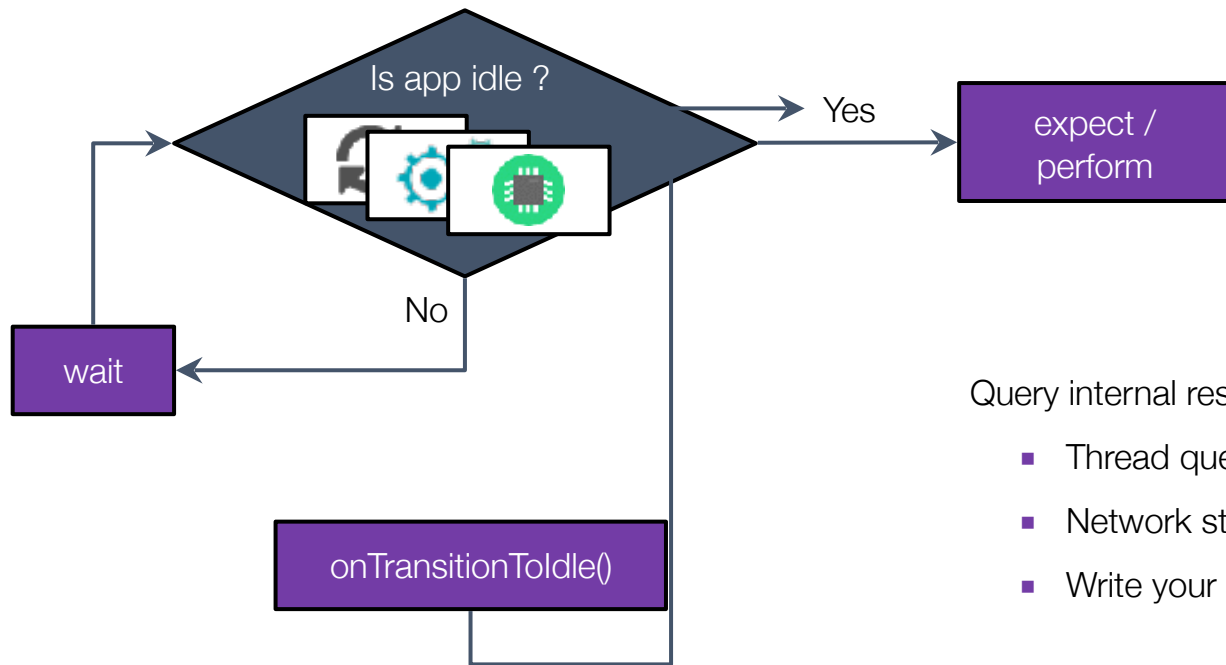
Run on any thread including the main (UI) thread



You don't test what you ship test code is part of the app

Uses EarlGrey and Espresso

Gray Box Sync Mechanisms



Query internal resources:

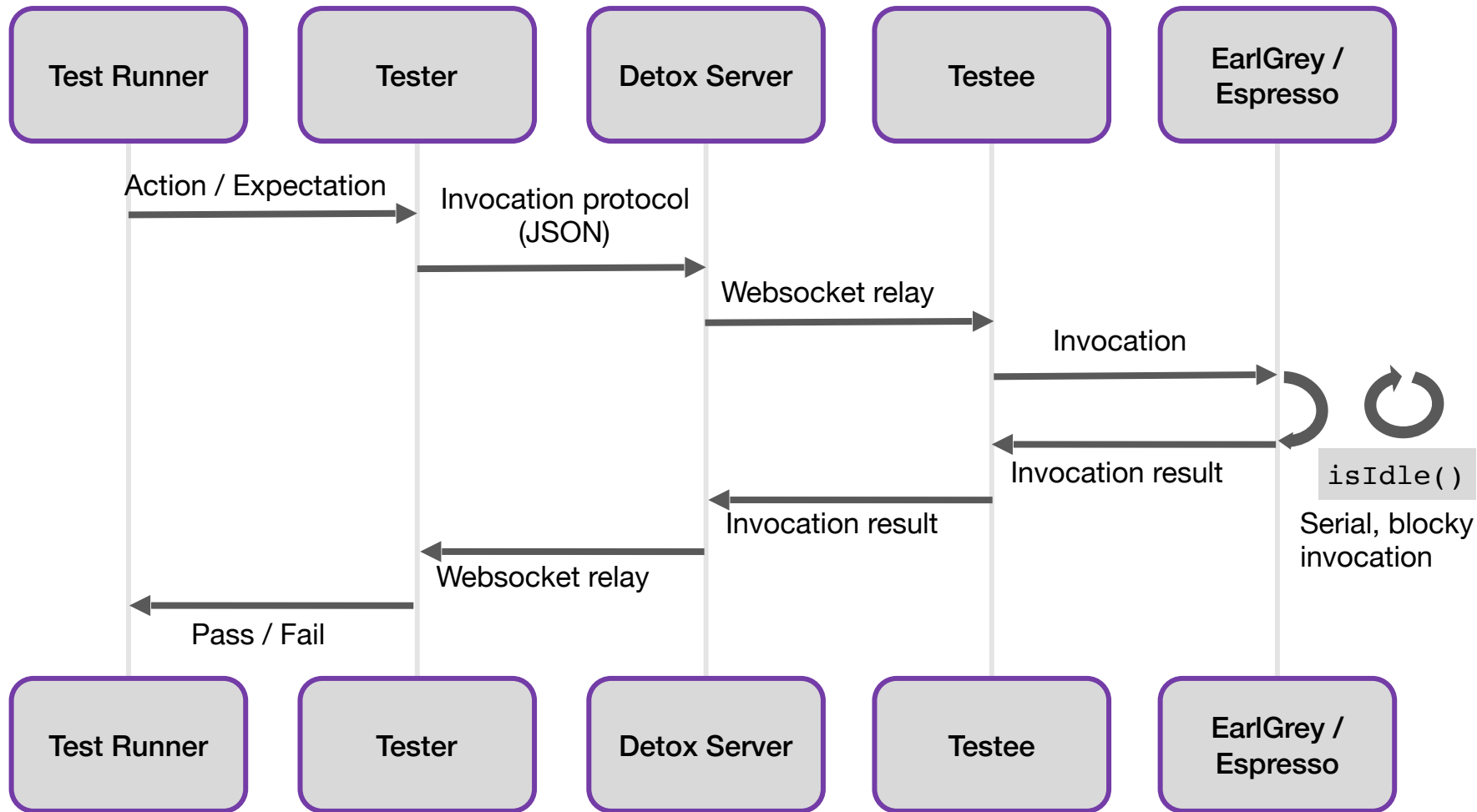
- Thread queues
- Network stack (in flight requests)
- Write your own custom resource monitor

Does not rely on WebDriver

- Async
- Cross platform
- Test Runner Independent
- Debuggable

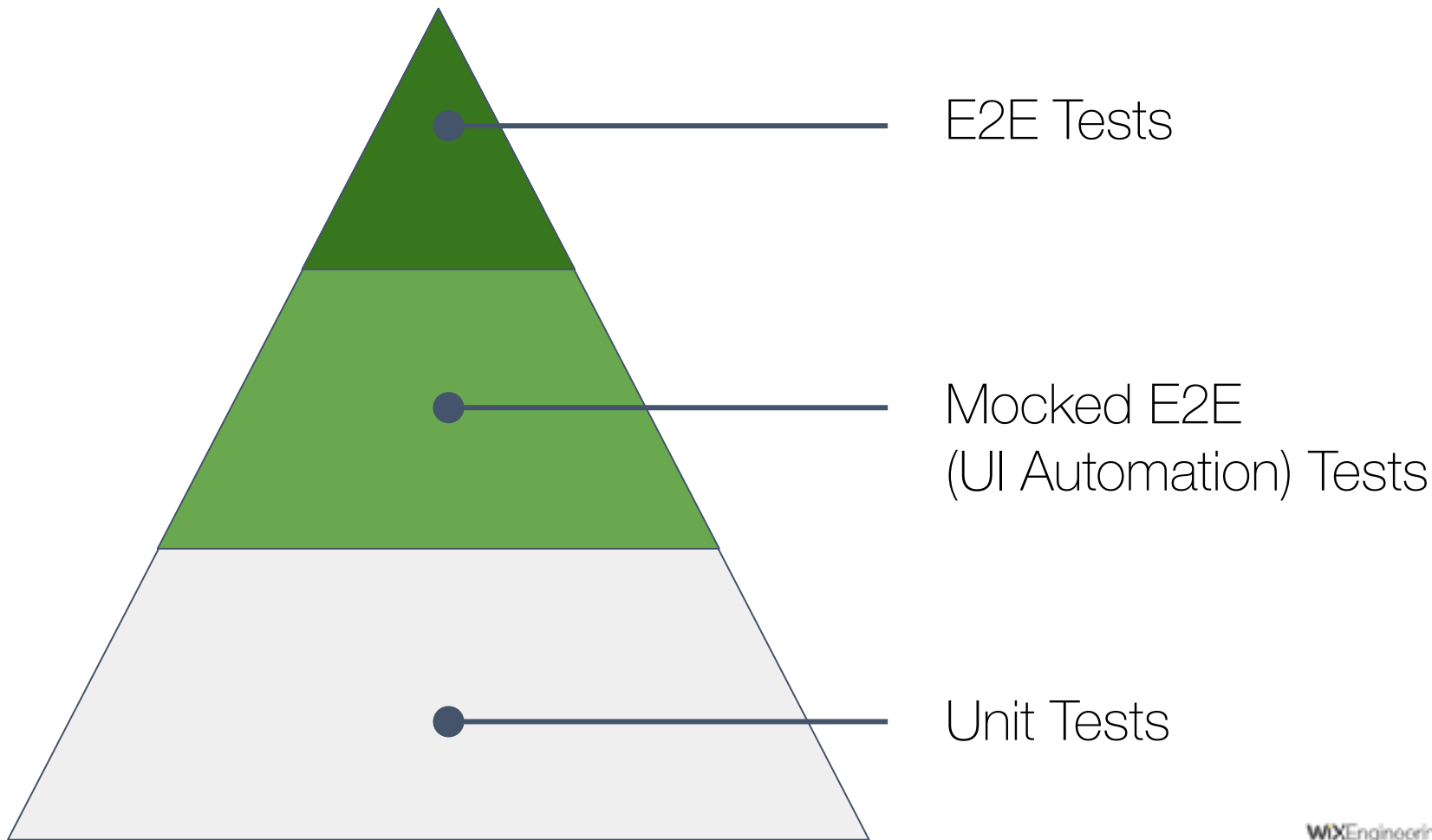
```
describe('Login flow', () => {  
  it('should login successfully', async () => {  
    await device.reloadReactNative();  
    await expect(element(by.id('email'))).toBeVisible();  
  
    await element(by.id('email')).typeText('john@example.com');  
    await element(by.id('password')).typeText('123456');  
    await element(by.id('login')).tap();  
  
    await expect(element(by.text('Welcome'))).toBeVisible();  
    await expect(element(by.id('email'))).toNotExist();  
  });  
});
```

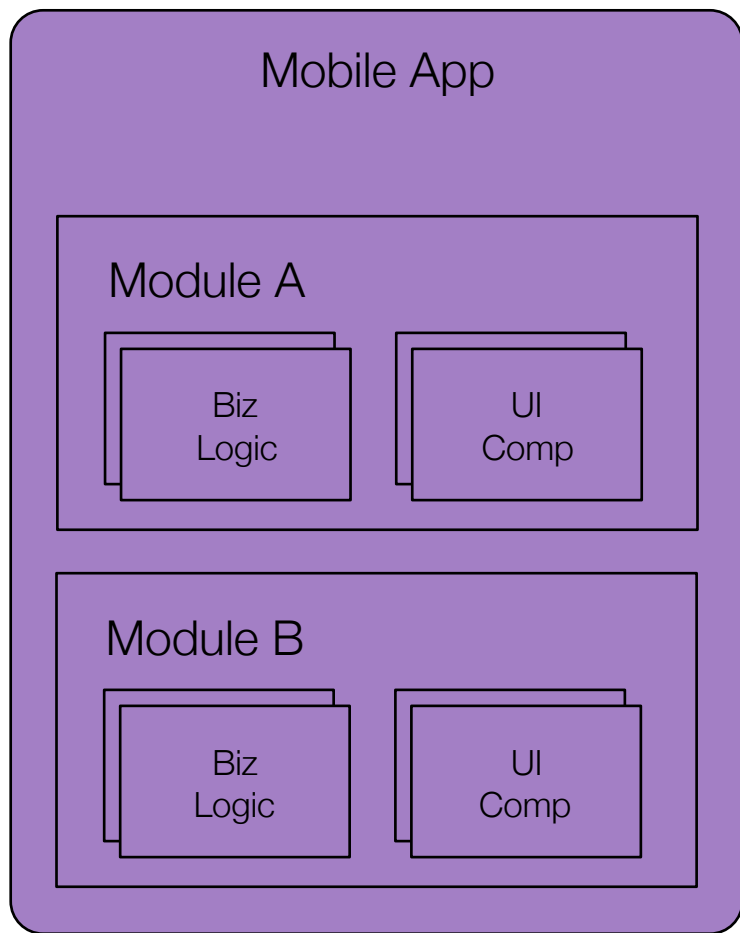

Evaluating expectations on the device



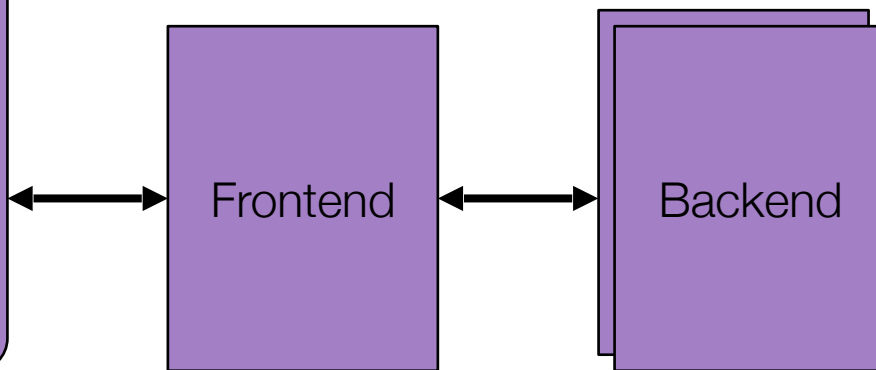
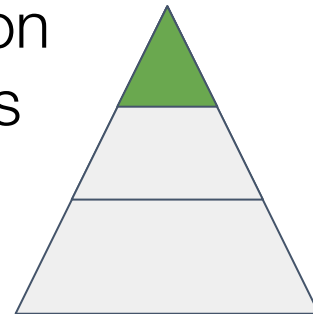
03

Testing with Detox





Production
E2E Tests



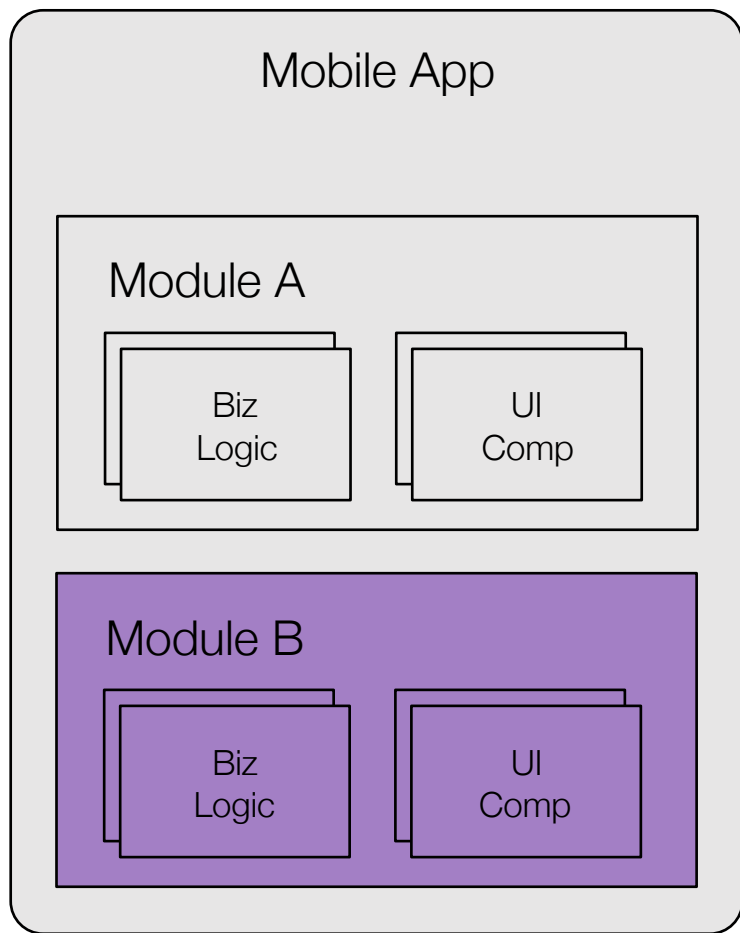
Production E2E

Pros

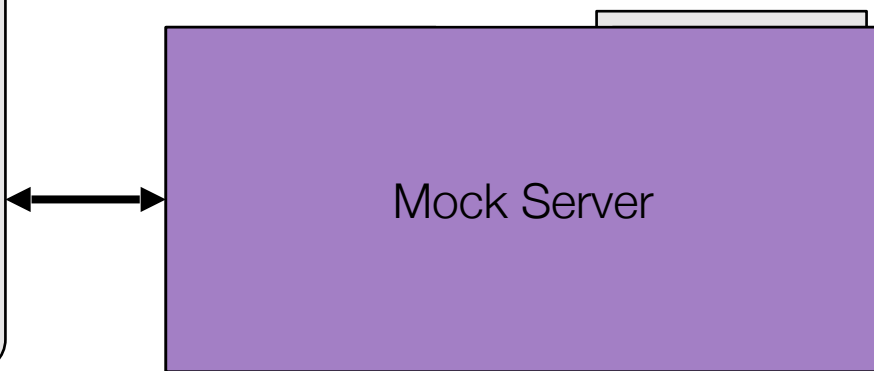
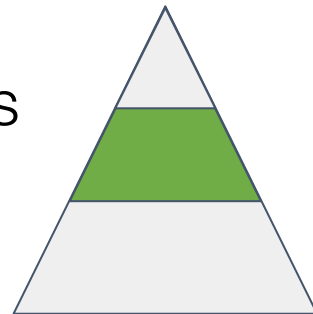
- Real user experience
- Easy to setup
- Easy to write
- High confidence

Cons

- Flaky
- Slow
- Hard to maintain



Mocked
E2E Tests



Mocked E2E

Pros

- Closer to code
- Stable
- Easy to maintain

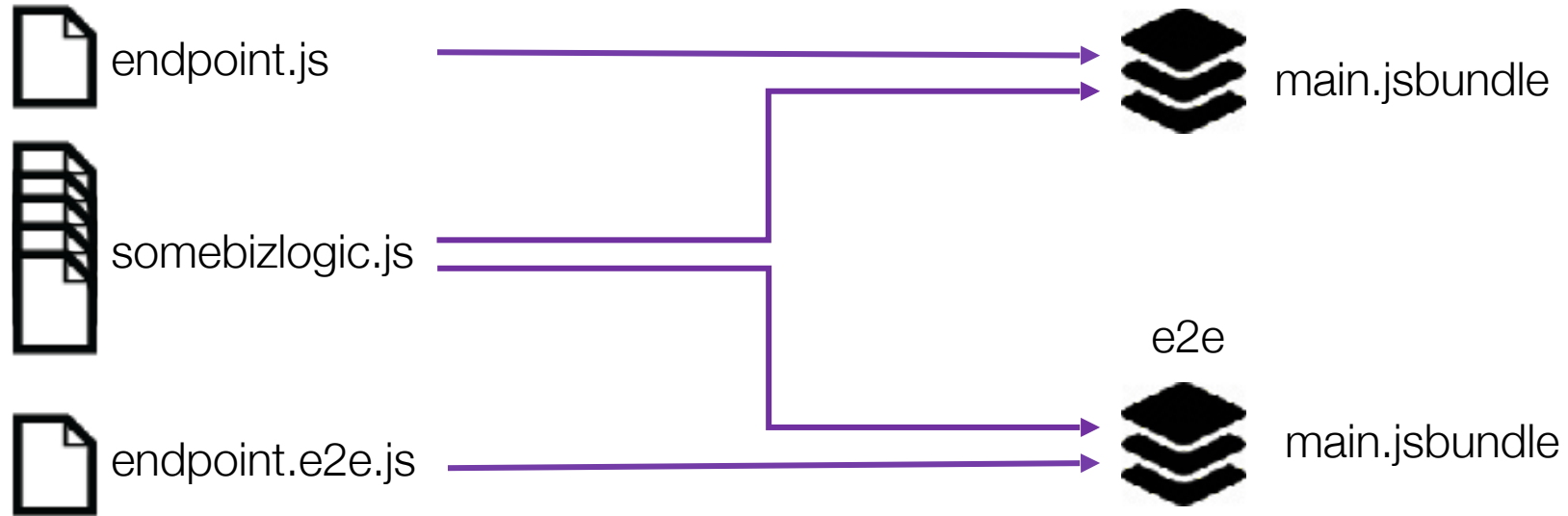
Cons

- Hard to setup
- Hard to write

Mocking for Detox

With React Native

Mocking: Build Flavors



```
npm install react-native-repackager --save-dev
```

```
"scripts": {  
  "postinstall": "react-native-repackager setup"  
}
```



endpoint.js

```
module.exports = {  
  LOGIN_ENDPOINT: 'https://login.wix.com'  
};
```



endpoint.e2e.js

```
module.exports = {  
  LOGIN_ENDPOINT: 'https://localhost:90210'  
};
```



rn-cli.config.js

```
module.exports = {  
  getSourceExts: () => process.env.RN_SRC_EXT ?  
    process.env.RN_SRC_EXT.split(',') : []  
};
```



```
> RN_SRC_EXT=e2e.js react-native start  
  
> RN_SRC_EXT=e2e.js xcodebuild <params>  
  
> RN_SRC_EXT=e2e.js ./gradlew assembleRelease
```



~~endpoint.js~~



endpoint.e2e.js

```
describe('Login', () => {
  const server = new MockServer();

  beforeEach(async () => {
    await server.start();
  });

  afterEach(async () => {
    await server.stop();
  });

  it('should login and persist login after restart', async () => {
    server.forUser(server.testUserId()).withBusinesses(1);

    await device.launchApp({ delete: true, permissions: { notifications: 'YES' } });
    await element(by.text('Log In TestUser1')).tap();
    await expect(element(by.text('feed.FeedScreen'))).toBeVisible();

    await device.launchApp({ newInstance: true });
    await expect(element(by.text('feed.FeedScreen'))).toBeVisible();
  });
});
```

```
class MockNotificationServer {
  start({port}) {
    ...
    this.app.post('/regiscribe', this.onRegiscribe.bind(this));
    this.app.post('/unsubscribe', this.onUnsubscribe.bind(this));
    this.server = this.app.listen(port);
  }
  reset() {
    this.lastRegiscribe = {};
    this.lastUnsubscribe = {};
  }
  onRegiscribe(req, res) {
    this.lastRegiscribe = req;
    res.json({
      installationId: 'MockInstallationId'
    });
  }
  onUnsubscribe(req, res) {
    this.lastUnsubscribe = req;
    res.send('ok');
  }
  expectLastRegiscribe() {
    return expect(this.lastRegiscribe.body);
  }
  expectLastUnsubscribe() {
    return expect(this.lastUnsubscribe.body);
  }
}
```

```
it('should send push token to notification server after user allows push permissions',
  async () => {
    appStateServer.forUser(appStateServer.testUserId()).withBusinesses(1);
    await device.launchApp({ delete: true });
    await element(by.text('Log In TestUser1')).tap();
    notificationServer.expectLastRegiscribe().toNotExist();

    await element(by.text('Allow')).tap();
    await expect(element(by.text('feed.FeedScreen'))).toBeVisible();

    notificationServer.expectLastRegiscribe().toContain({ appleDeviceToken:
    consts.DeviceToken });
  });
```


Mocking

- Launch Args
- User Notifications
- Deep Links
- Runtime Permissions

Don't act according to app state

Control it!

04

What's Next ?

“ ≈ 0 Manual QA”

Developers

- Unit tests
- Mocked E2E
- Contract tests

QA Engineers

- Prod E2E
manual backup
- Scenarios

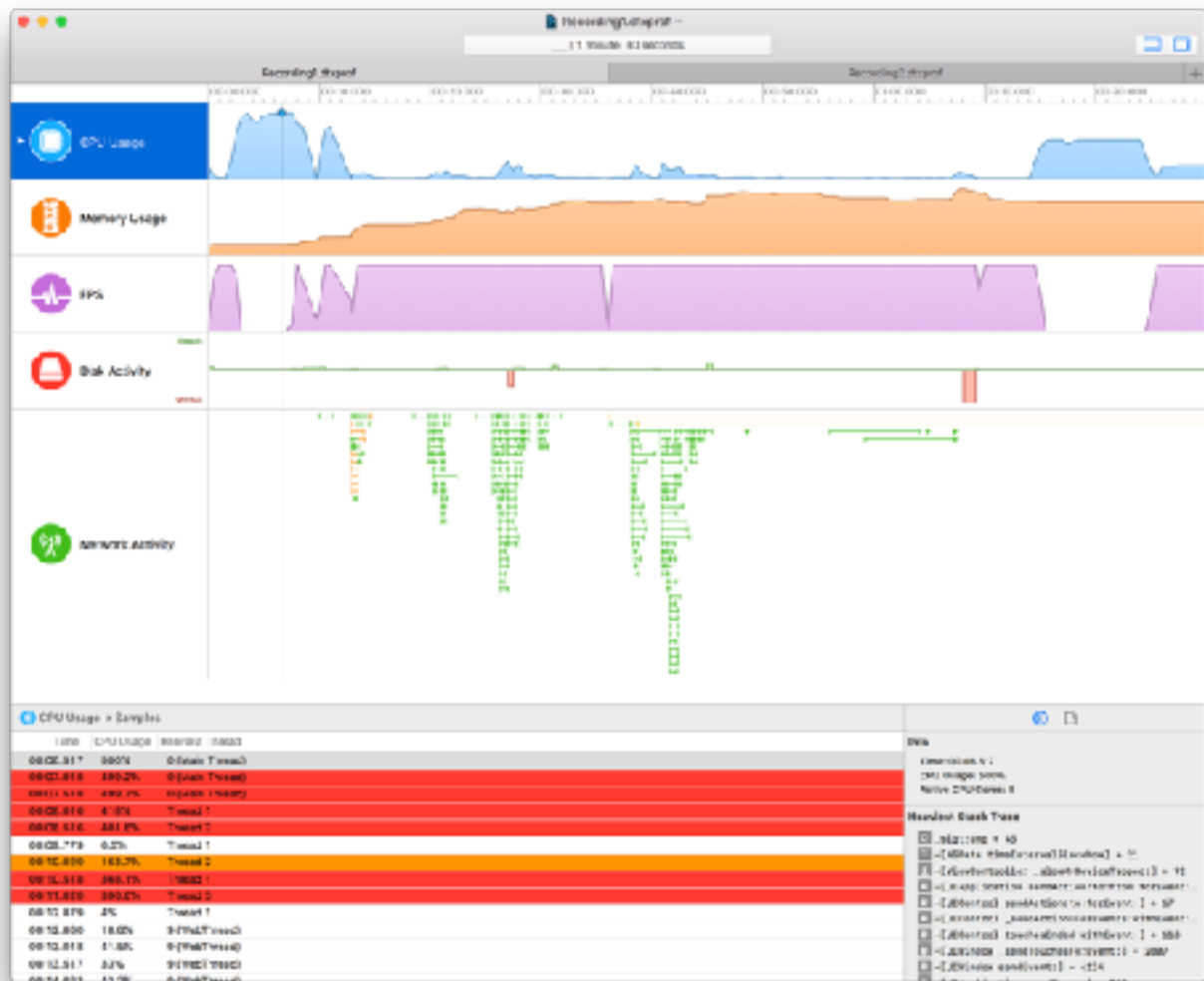
Developers + QA

- Educate
- Sync

Future Plans

- Parallelization through test runners
- Artifacts screenshots, videos, logs
- Better error logging
- Run on Devices
- ...

Detox Instruments



WIXEngineering

Thanks.



medium.com/@rotemmiz



rotemm@wix.com



[@rotemmiz](https://twitter.com/rotemmiz)



medium.com/@rotemmiz



[github.com/
rotemmiz](https://github.com/rotemmiz)