



Maestro:

The awesome mobile testing tool you've never heard of



Maksim Lin
GDE Flutter/Dart
GDG/Flutter Melbourne Organiser

Need to talk fast! 🏃

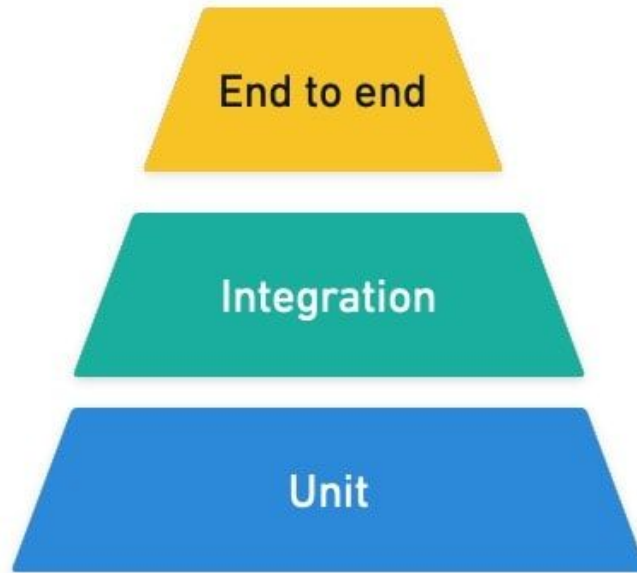
1. Introduction 🗨️

2. Theory 📖

3. Demo 💻

The Testing Pyramid

The Test Pyramid



Engineering == Trade-offs

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

<https://docs.flutter.dev/testing/overview>

End to End (E2E) Testing

- Reserve it for only your **high value** user interactions 💰
- "Smoke testing" 🔥
- Use **release** app builds
- If possible run in **production** environment ✨
- Slow 😞
- Resource intensive (normal CI VMs not a great host!) 😞
- Flaky 😞

What is Maestro?

"Maestro is the simplest and most effective mobile UI testing framework."

- **End to End** testing tool
- ***Open Source*** (<https://github.com/mobile-dev-inc/maestro>)
- For Android & iOS **only** (**inc Flutter** & RN)
- Uses standard platform accessibility IDs/Texts
- ***Leans into*** inherent nature of UI testing **flakiness**

Maestro Features

- Easy to install
- Good DX CLI tooling
- Small YAML-based DSL for interaction scripts
- Javascript available for integration & automation
- GUI web-based Studio "IDE" 🎉🎉
- Built in screen recording (⚠️ uses a web service! ⚠️)

Why not Flutter Integration Tests?

- YAML not Dart allows non Flutter eng to write tests
- Maestro Studio allows non eng to write tests
- No need to install dev tooling Maestro & APK enough
- Using Accessibility IDs no need to know Flutter widget details
- Works with native framework UI not just Flutter widgets
- Works with Webviews

Demo !

Summary

1. Learned what Maestro is
2. How Maestro works
3. How to use Maestro studio
4. How to run Maestro UI tests



@RickyTinez on YT

***Share the love,
share the knowledge,
knowledge is power,
peace!***

Thank You!



`fluttercommunity.social/@maks`



`@maks`



Maksim Lin
GDE Flutter/Dart