Test Automation in Open-Source Android Apps: A Large-Scale Empirical Study

Jun-Wei Lin, Navid Salehnamadi, and Sam Malek
School of Information and Computer Sciences
University of California, Irvine
ASE 2020
Automated Testing of Mobile Apps

Received attention in recent years from both researchers and developers

Researchers
- Automated program repair, automated test transfer, mutation testing, ...
- The extent mobile tests exist
- The type and quality of these tests
- Whether the tests are adopted in a particular way

Developers
- Why and how to adopt automated testing
- The impact on user satisfaction or project popularity in developer community
Automated Testing of Mobile Apps

Received attention in recent years from both researchers and developers

Researchers
- Automated program repair
- The extent mobile tests exist
- The type and quality of these tests
- Whether the tests are adopted in a particular way

Developers
- Why and how to adopt automated tests
- The impact on user satisfaction or popularity in developer community

Holistic view from three complementary perspectives: apps, developers, and impacts
Differences from Prior Work

Scale
- 3.5M+ non-forked repositories on GitHub were scanned
- 12,000+ real-world apps across 16 app markets were identified
  (prior work: ~1000 apps on F-Droid/Google Play)

Curated dataset
- Heuristics to identify non-trivial apps, e.g., apps published in an app market
- Excluding dummy tests generated by Android Studio, e.g., ExampleUnitTest()

Survey involving 148 contributors of the subject apps for the rationale behind our observations
Data Collection

Directory Inspection

GHTorrent 3.5M+ Repos

One Manifest File w/ Gradle Plugin
537K Apps

Manifest File Analysis

2+ Components
287K Apps

Dataset for Study
12,562 Apps

1+ App Markets
19K Apps

Duplicates & Old Apps Removal
Metadata Collection

App Market Matching
Data Collection

- **GHTorrent 3.5M+ Repos**
- **Directory Inspection**
- **Manifest File Analysis**
- **One Manifest File w/ Gradle Plugin 537K Apps**
- **2+ Components 287K Apps**
- **Dataset for Study 12,562 Apps**
- **1+ App Markets 19K Apps**
- **App Market Matching**
- **Duplicates & Old Apps Removal Metadata Collection**
Data Collection

\[
\text{Directory Inspection} \quad \rightarrow \quad \text{One Manifest File w/ Gradle Plugin 537K Apps} \quad \rightarrow \quad \text{2+ Components 287K Apps}
\]

\[
\text{Dataset for Study 12,562 Apps} \quad \rightarrow \quad \text{1+ App Markets 19K Apps}
\]

\[
\text{Duplicates & Old Apps Removal Metadata Collection} \quad \leftarrow \quad \text{App Market Matching}
\]
Data Collection

Directory Inspection

GHTorrent 3.5M+ Repos

One Manifest File w/ Gradle Plugin 537K Apps

2+ Components 287K Apps

Manifest File Analysis

Dataset for Study 12,562 Apps

1+ App Markets 19K Apps

App Market Matching

Duplicates & Old Apps Removal
Metadata Collection
Initial List of GitHub Repos

GHTorrent DB
- A research project that monitors the GitHub public event timeline and populates the metadata of the observed events

Query the GHTorrent DB for projects:
- Written in Java or Kotlin
- Non-forked, non-deleted
- 3.5M+ repos
Filtering Criteria

Goal: to identify non-trivial and real-world apps

Criterion 1
- Exactly one AndroidManifest.xml
- A task to build Android app in build.gradle

Criterion 2
- 2+ components declared in the manifest file

Criterion 3
- The package name must appear in an app market, e.g., Google Play, F-Droid, Anzhi
Data Cleaning and Meta-data Collection

Remove apps
1) With duplicate package names
2) Before 2015

Collect additional meta-data
- Number of forks, stars, contributors, etc., on GitHub
- Category and user ratings on Google Play

Two months for the entire process
### Top 10 App Market Distribution

<table>
<thead>
<tr>
<th>Market*</th>
<th>#Apps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Play</td>
<td>11265</td>
</tr>
<tr>
<td>PlayDrone</td>
<td>539</td>
</tr>
<tr>
<td>fdroid</td>
<td>434</td>
</tr>
<tr>
<td>anzhi</td>
<td>408</td>
</tr>
<tr>
<td>appchina</td>
<td>294</td>
</tr>
<tr>
<td>mi.com</td>
<td>70</td>
</tr>
<tr>
<td>VirusShare</td>
<td>62</td>
</tr>
<tr>
<td>angeeks</td>
<td>41</td>
</tr>
<tr>
<td>1mobile</td>
<td>26</td>
</tr>
<tr>
<td>freewarelovers</td>
<td>12</td>
</tr>
</tbody>
</table>

### Year Distribution

<table>
<thead>
<tr>
<th>Year Created</th>
<th>#Apps</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015</td>
<td>3614</td>
</tr>
<tr>
<td>2016</td>
<td>2330</td>
</tr>
<tr>
<td>2017</td>
<td>1731</td>
</tr>
<tr>
<td>2018</td>
<td>2898</td>
</tr>
<tr>
<td>2019</td>
<td>1989</td>
</tr>
<tr>
<td>Total</td>
<td>12562</td>
</tr>
</tbody>
</table>
Counting Tests

Assumptions for the apps:
- Using JUnit-based testing frameworks, e.g., JUnit, Espresso, Roboelectric, Mockito
- Developed with Android Studio

A method annotated with "@Test" is considered as a test case
- Used by JUnit-based testing frameworks

Exclude the placeholder tests automatically generated by Android Studio, e.g., ExampleUnitTest.java, ExampleInstrumentedTest.java
Developer Survey

Goal: To understand the rationale behind our findings from the dataset

Participant: the main contributor of each subject app
  ◦ e.g., project owner, contributor who has made the most commits, etc.

Questions
  ◦ Demographic, e.g., country, Android development experience
  ◦ Current practices of Android app testing
  ◦ Opinions about our findings in the correlation analysis between the adoption of test automation and the popularity of apps

148 complete responses from 6,837 unique emails sent successfully
Research Questions

App

• How prevalent is test automation in open-source Android apps?
• Is the prevalence of test automation varied across different categories of apps?

Developer

• How prevalent is test automation and what are the reasons for not adopting it?
• The biggest challenges in Android testing?
• The most useful criteria for evaluating Android tests
• Do developers prefer unit or UI testing and why?
• Do the same developers have the same testing habits across apps?

Impact

• Is the practice of Test Pyramid followed by developers?
• How does test automation relate to project popularity?
• How does test automation relate to user satisfaction?
Research Questions

**App**
- How prevalent is test automation in open-source Android apps?
- Is the prevalence of test automation varied across different categories of apps?
- How prevalent is test automation and what are the reasons for not adopting it?
  - The biggest challenges in Android testing?
  - The most useful criteria for evaluating Android tests
  - Do developers prefer unit or UI testing and why?
  - Do the same developers have the same testing habits across apps?
  - Is the practice of Test Pyramid followed by developers?

**Developer**
- How does test automation relate to project popularity?
- How does test automation relate to user satisfaction?
Research Questions

- **RQ1. How prevalent is test automation in open-source Android apps?**
  - Is the prevalence of test automation varied across different categories of apps?
  - How prevalent is test automation and what are the reasons for not adopting it?
  - The biggest challenges in Android testing?
  - The most useful criteria for evaluating Android tests
  - Do developers prefer unit or UI testing and why?
  - Do the same developers have the same testing habits across apps?
  - Is the practice of Test Pyramid followed by developers?

- **Developer**
  - How prevalent is test automation relate to project popularity?
  - How does test automation relate to user satisfaction?
RQ1. Prevalence of Test Automation

Only 7.98% of the subject apps contain tests. Much lower than previous findings (14%-40%)

56% (7017/12562) subject apps contain placeholder tests (and were excluded)
Research Questions

**App**
- How prevalent is test automation in open-source Android apps?
- Is the prevalence of test automation varied across different categories of apps?
- **RQ2. How prevalent is test automation and what are the reasons for not adopting it?**
  - The biggest challenges in Android testing?
  - The most useful criteria for evaluating Android tests
  - Do developers prefer unit or UI testing and why?
  - Do the same developers have the same testing habits across apps?
  - Is the practice of Test Pyramid followed by developers?

**Developer**
- How does test automation relate to project popularity?
- How does test automation relate to user satisfaction?
RQ2. Prevalence of Test Automation (Reported by Developers)

How Developers Test Their Apps

<table>
<thead>
<tr>
<th>Way</th>
<th>#Respondents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manually</td>
<td>130</td>
</tr>
<tr>
<td>With scripted/automated tests</td>
<td>85</td>
</tr>
<tr>
<td>With dedicated QA team or 3rd party testing services</td>
<td>43</td>
</tr>
<tr>
<td>With automatic input generation tools</td>
<td></td>
</tr>
<tr>
<td>Other</td>
<td></td>
</tr>
<tr>
<td>Not at all</td>
<td></td>
</tr>
</tbody>
</table>

57% (85/148) says they are using, but only 8% observed
RQ2. Difficulties in Adopting Test Automation (Reported by Developers)

<table>
<thead>
<tr>
<th>Difficulty</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost to create and maintain automated tests</td>
<td>77</td>
</tr>
<tr>
<td>Time constraints</td>
<td>74</td>
</tr>
<tr>
<td>Size or maturity of the app</td>
<td>66</td>
</tr>
<tr>
<td>Lack of exposure or knowledge of existing frameworks</td>
<td>52</td>
</tr>
<tr>
<td>Cumbersome to use</td>
<td>50</td>
</tr>
<tr>
<td>Lack of support from management or organization</td>
<td>30</td>
</tr>
<tr>
<td>Other</td>
<td>11</td>
</tr>
</tbody>
</table>

The app is not big or complex enough to require automated tests
Research Questions

App

• How prevalent is test automation in open-source Android apps?
• Is the prevalence of test automation varied across different categories of apps?

Developer

• How prevalent is test automation and what are the reasons for not adopting it?
• The biggest challenges in Android testing?
• The most useful criteria for evaluating Android tests
• Do developers prefer unit or UI testing and why?
• Do the same developers have the same testing habits across apps?
• Is the practice of Test Pyramid followed by developers?

Impact

• Popularity metrics that mobile developers care about
  • Attention received from other developers, e.g., stars, forks, contributors on GitHub
  • User ratings on the market

• Whether these popularity metrics are impacted by the adoption of test automation

• RQ3. How does test automation relate to project popularity?
• RQ4. How does test automation relate to user satisfaction?
Statistical Analysis

Two disjoint sets with equal size
- $R_w$: metric values (e.g., number of stars) from the apps with tests
- $R_o$: metric values from the apps without tests

Compute the mean and median of $R_w$ and $R_o$, and the difference between the mean and median

Perform hypothesis testing to determine if the difference observed in $R_w$ and $R_o$ is statistically significant

Repeated for all the popularity metrics
RQ3. How does test automation relate to project popularity?

### Impact of Having Tests on the Popularity of Apps.

- \( R_W \): Apps with Tests.
- \( R_O \): Apps Without Tests.

<table>
<thead>
<tr>
<th></th>
<th>Stars*</th>
<th></th>
<th>Forks*</th>
<th></th>
<th>Contributors*</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Size</td>
<td>Mean</td>
<td>Median</td>
<td>Size</td>
<td>Mean</td>
</tr>
<tr>
<td>( R_W )</td>
<td>629</td>
<td>10.95</td>
<td>0</td>
<td>630</td>
<td>3.74</td>
</tr>
<tr>
<td>( R_O )</td>
<td>629</td>
<td>4.57</td>
<td>0</td>
<td>630</td>
<td>1.31</td>
</tr>
<tr>
<td>( \Delta )</td>
<td>6.38</td>
<td>0</td>
<td>2.43</td>
<td>0</td>
<td>1.13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Commits*</th>
<th></th>
<th>Issues*</th>
<th></th>
<th>Pull Requests*</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Size</td>
<td>Mean</td>
<td>Median</td>
<td>Size</td>
<td>Mean</td>
</tr>
<tr>
<td>( R_W )</td>
<td>628</td>
<td>147.21</td>
<td>84.5</td>
<td>635</td>
<td>10.39</td>
</tr>
<tr>
<td>( R_O )</td>
<td>628</td>
<td>39.93</td>
<td>14</td>
<td>635</td>
<td>1.35</td>
</tr>
<tr>
<td>( \Delta )</td>
<td>107.28</td>
<td>70.5</td>
<td>9.04</td>
<td>0</td>
<td>7.91</td>
</tr>
</tbody>
</table>

*The difference is statistically significant.*
Thoughts from the Survey Participants

- **Causation (57%)**
- **Connection (34%)**
- **Other (e.g., Coincidence) (9%)**

“**I would say they have a direct connection** since the quality and rigidness of the app’s code can definitely influence an app’s popularity.” (direct)

“First you build the app, then it gets popular, then you get resources/motivation to increase it’s quality. That’s when you go to UI tests.” (reverse)

“Projects that become popular end up writing more tests because they need to ensure the stability of the project. As the project becomes more stable (due to more testing) it provides a positive feedback loop.” (bidirectional)
RQ3. How does test automation relate to project popularity?

Popular projects are more likely to adopt test automation practices. 57% of the developers believe it implies causality between them.
RQ4. How does test automation relate to user satisfaction (ratings on Google Play)?

Impact of Having Tests on the User Satisfaction of Apps. 

<table>
<thead>
<tr>
<th></th>
<th>Size</th>
<th>Mean</th>
<th>Median</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rw</td>
<td>211</td>
<td>4.14</td>
<td>4.25</td>
<td>0.0689</td>
</tr>
<tr>
<td>Ro</td>
<td>211</td>
<td>4.2</td>
<td>4.33</td>
<td></td>
</tr>
<tr>
<td>Δ</td>
<td>-0.06</td>
<td>-0.08</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

No association found between them with statistical significance
Thoughts from the Survey Participants

“I think it would depend on the type of application. Games and such are harder to test and the quality of test does not correlate with how fun the game is. For a banking application tests are essential and do effect the quality of the final product.”

“Play Store ratings are a noisy metric of app quality and overall user experience, so the no apparent correlation doesn’t convince me that app quality isn’t impacted at least somewhat by automated testing”
RQ4. How does test automation relate to user satisfaction (ratings on Google Play)?

Users’ satisfaction with apps appears to be unrelated to the adoption of test automation, while half of the developers think differently.
Discussion

Automated testing is not widely adopted

Only 8% of the subject apps have adopted automated testing

- Contradicts earlier studies that have reported higher adoption rate (14%-40%)
- But in line with the general perception that it is challenging to find open-source apps with tests for research purposes

Dataset of real-world and non-trivial apps with automated tests

- For emerging research topics, e.g., automated program repair, automated test transfer, and mutation testing
Discussion

Automated testing can be useful and important

Strong correlation between the adoption of test automation and the popularity of development projects
  ◦ Most of the survey respondents (91%) believe that the correlation is either causation or a connection

Users’ satisfaction appears unrelated to test automation
  ◦ But a considerable amount of survey participants (52%) think that automated testing has impacts on users’ satisfaction
Discussion

Tools and libraries have room for improvement

One of the difficulties reported by developers in adopting test automation is cumbersome tools
  ◦ Steep learning curve, poor documentation, usability, and compatibility issues

UI testing tools could be improved by addressing developers’ concerns about speed, simpleness, and robustness
  ◦ “headless mode” (such as done by Robolectric) to accelerate execution of UI tests
  ◦ Better Interactive tools (such as Espresso Test Recorder) to help developers create UI tests
  ◦ Efforts to prevent or resolve flakiness of UI tests
Conclusion

A holistic view regarding how and why test automation is (not) practically adopted in open-source Android apps

- 12,000+ real-world and non-trivial apps on GitHub
- A survey of 148 developers of these apps

Findings to guide the current practices and future research directions

Future work

- More open-source projects, e.g., apps hosted on Bitbucket
- New research questions, e.g., continuous integration practices

Thank You!