Ph.D itee

INFORMATION TECHNOLOGY ELECTRICAL ENGINEERING

Vincenzo Riccio

Tutor: Anna Rita Fasolino

XXXI Cycle - III year presentation

ENHANCING AUTOMATED GUI EXPLORATION TECHNIQUES FOR ANDROID MOBILE APPLICATIONS

Università degli Studi di Napoli Federico II

Personal Background

- **Candidate:** Vincenzo Riccio
- **Cycle:** XXXI
- **Fellowship**: PhD grant
- **Graduation**: MSc with honors in Computer Engineering at the University of Napoli Federico II
- **Research Activity**: Software Testing Automation
- **Research Field**: Software Engineering
- **Collaborations:**

Universidade do Porto
Faculdade de Engenharia
FEUP

UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

Research Group

- REsEarch gRoup of Software Engineering (REvERSE) at the University of Naples Federico II

- **Mission:** REvERSE@Unina aims at developing novel methods, techniques and tools that advance development and evolution of software systems. We are interested in all the software lifecycle processes, with a special focus on: Software Maintenance, Reverse Engineering, and Testing

Credits Summary

	1st Year	2nd Year	3rd Year	Entire PhD Course	Check
Modules	17	19	0	35,5	**30-70**
Seminars	10,2	5,2	2,7	18,1	**10-30**
Research	34	46	58	138	**80-140**
Total	61,2	70	61	191,6	**>180**

Experience Abroad

UNIVERSITÄT PASSAU

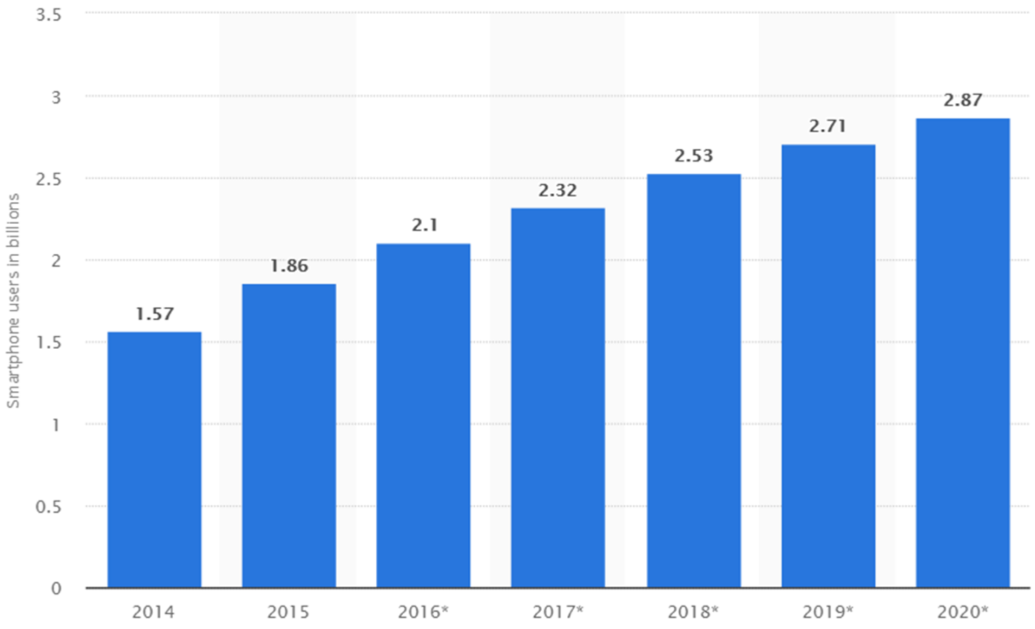
Fakultät für Informatik und Mathematik

- **Topic:** Novel evolutionary search algorithms for testing mobile applications
- **Start:** 17 April 2018
- **End:** 7 August 2018

Prof G. Fraser, Chair of Software Engineering II

Germany
Dresden
Bautzen
Jena
Chemnitz
Zwickau
Ústí nad Labem
Liberec
Fulda
Karlovy Vary
Prague
Bayreuth
Würzburg
Bamberg
Pilsen
Cz
Nuremberg
Ceske Budejovice
Regensburg
Straubing
Český Krumlov
ttgart
Ingolstadt
Passau
Ulm
Augsburg
Linz
Wels
Munich
Steyr
Rosenheim
Salzburg
nstanz
Kempten
Friedrichshafen
Garmisch-Partenkirchen
Berchtesgaden
Austria

Smartphone users worldwide



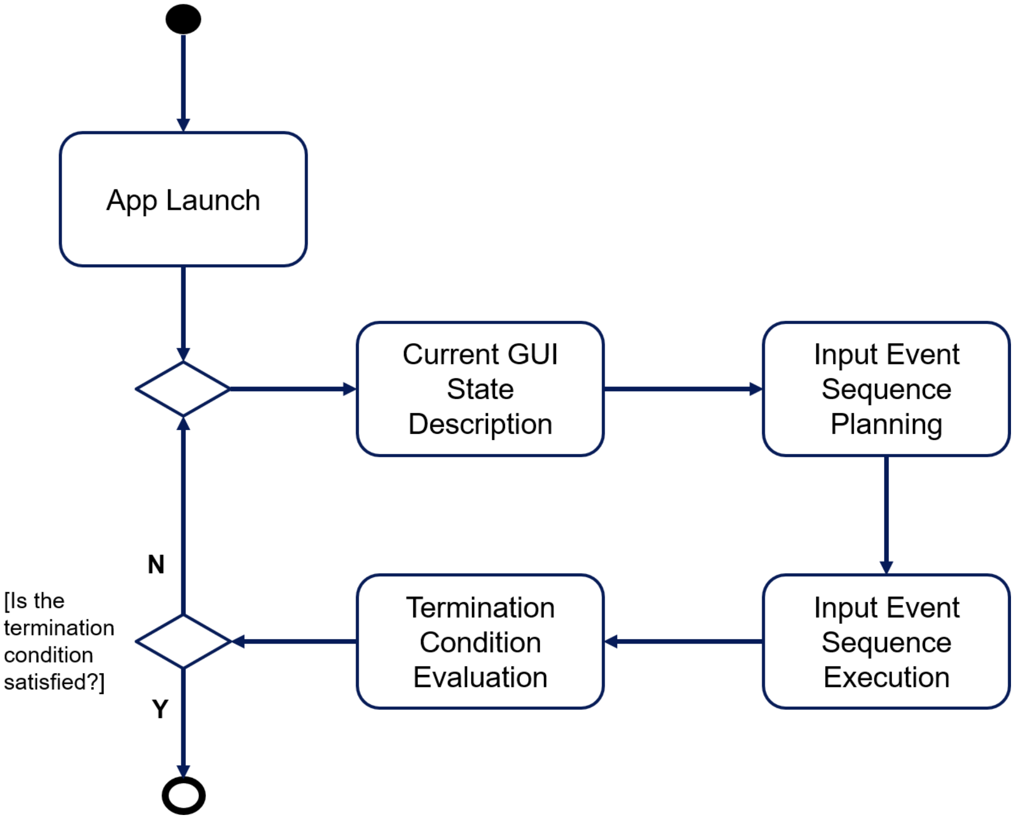
https://www.statista.com/statistics/330695/

- There is a constant demand for new mobile apps
- Android is today the world's most popular mobile operating system

Automation Tools

- The demand for app quality has grown together with their spread
- Automation tools can facilitate software quality engineering activities since they save humans from routine, time-consuming and error-prone manual tasks

Automated GUI Exploration Techniques (AGETs)


App Launch
Current GUI State Description
Input Event Sequence Planning
Input Event Sequence Execution
Termination Condition Evaluation
N
[Is the termination condition satisfied?]
Y

D. Amalfitano, N. Amatucci, AM. Memon, P. Tramontana, AR. Fasolino, “A general framework for comparing automatic testing techniques of Android mobile apps”, Journal of Systems and Software, 2017

Challenges

Enanche AGETs by:

1. targeting mobile-specific features
2. exploiting app-specific knowledge that only human users can provide

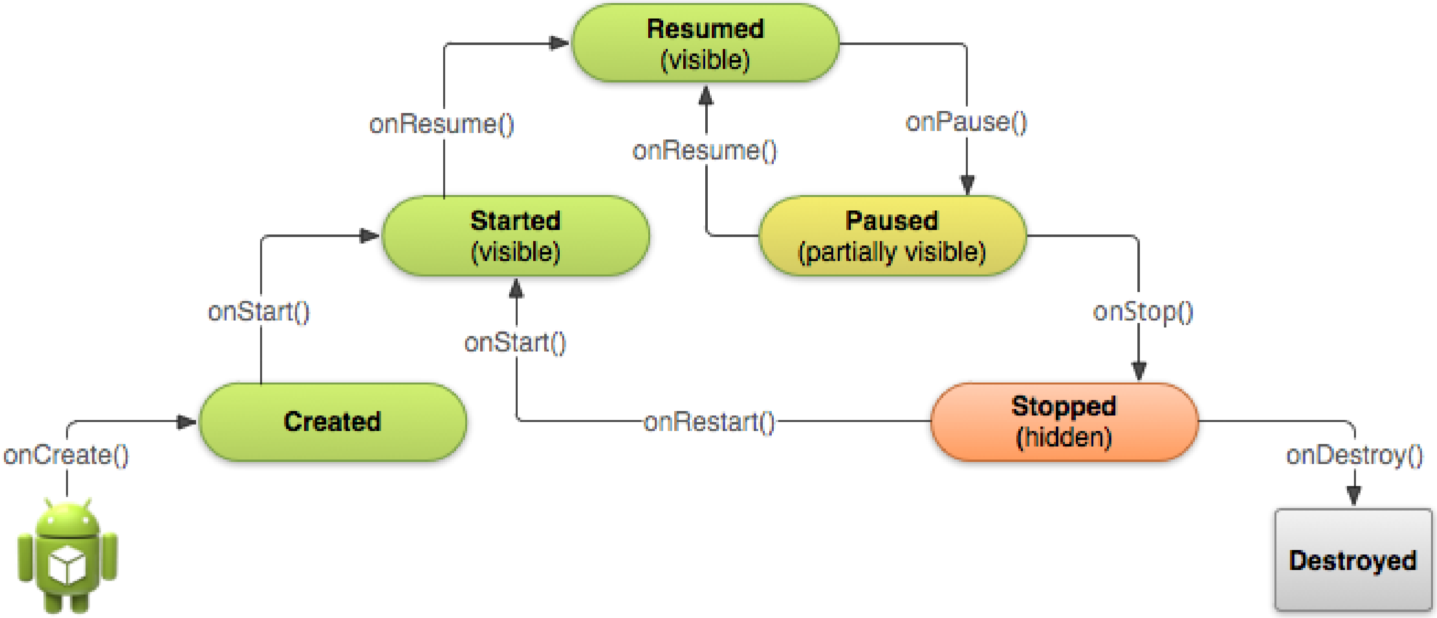
Challenge #1

Targeting mobile-specific features

The Android Activity Lifecycle

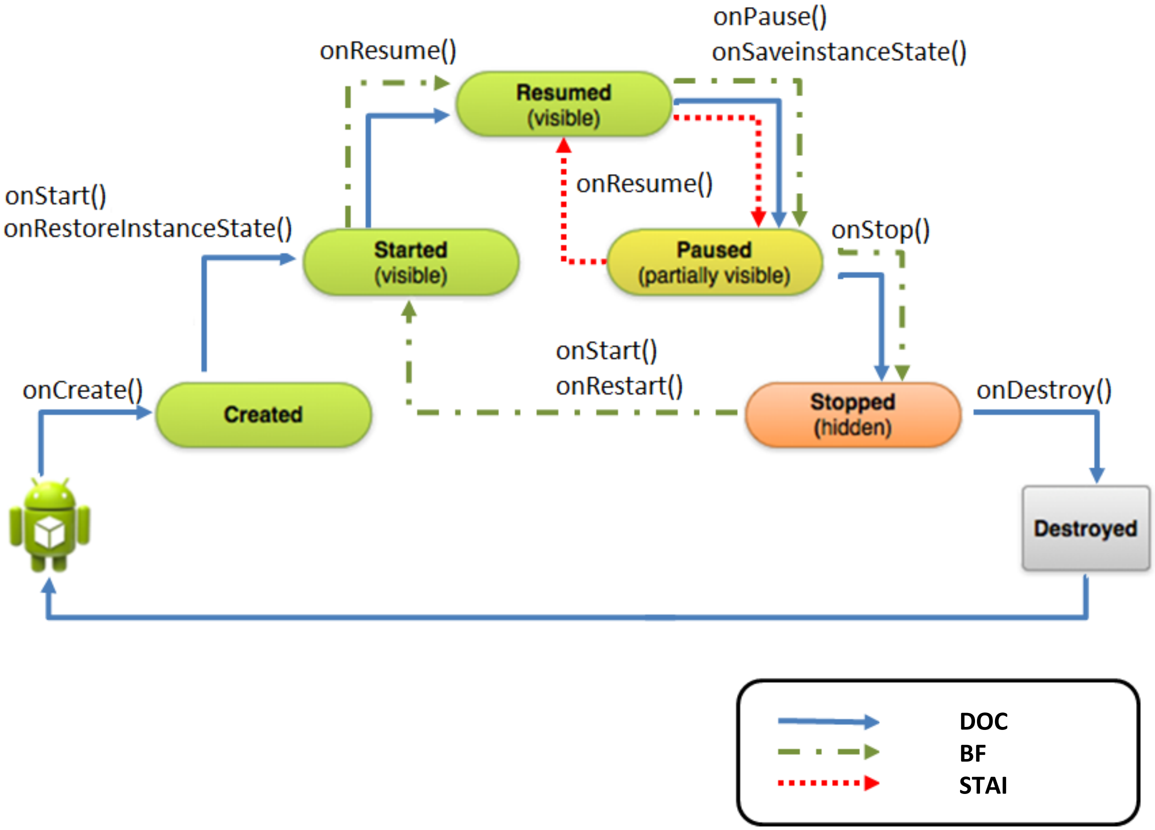
Android Activity Lifecycle

- An Android app is composed by one or more Activities
- Each Activity represents a single screen
- The Android Framework defines a peculiar lifecycle for Activity instances



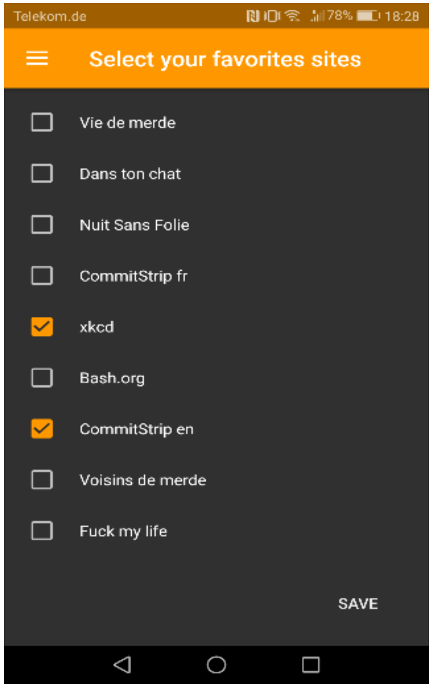
Lifecycle Event Sequences

- Mobile-specific events able to exercise the Activity lifecycle



Motivating Example: GUI Failure

- GUI failures consist in the manifestation of an unexpected GUI state



Telekom.de 78% 18:28

Select your favorites sites

☐ Vie de merde

☐ Dans ton chat

☐ Nuit Sans Folie

☐ CommitStrip fr

☑ xkcd

☐ Bash.org

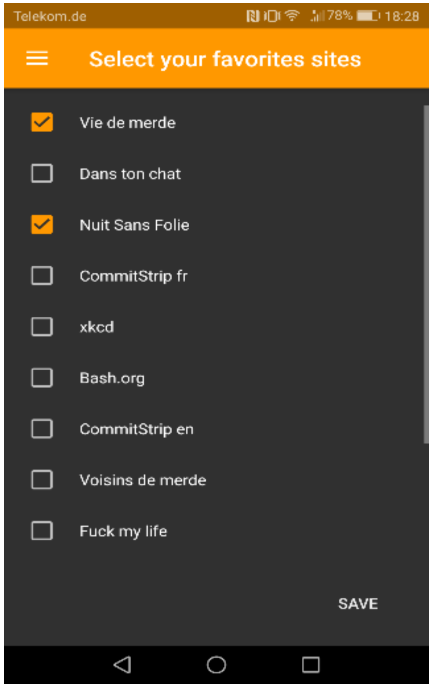
☑ CommitStrip en

☐ Voisins de merde

☐ Fuck my life

SAVE

Background
Foreground



Telekom.de 78% 18:28

Select your favorites sites

☑ Vie de merde

☐ Dans ton chat

☑ Nuit Sans Folie

☐ CommitStrip fr

☐ xkcd

☐ Bash.org

☐ CommitStrip en

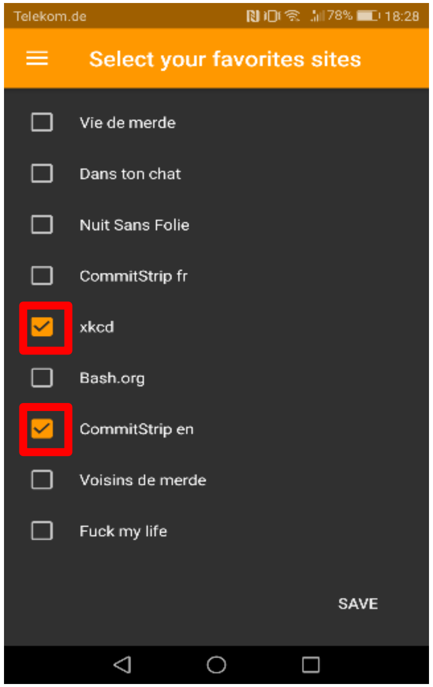
☐ Voisins de merde

☐ Fuck my life

SAVE

Motivating Example: GUI Failure

- GUI failures consist in the manifestation of an unexpected GUI state



Telekom.de 78% 18:28

Select your favorites sites

□ Vie de merde

□ Dans ton chat

□ Nuit Sans Folie

□ CommitStrip fr

☑ xkcd

□ Bash.org

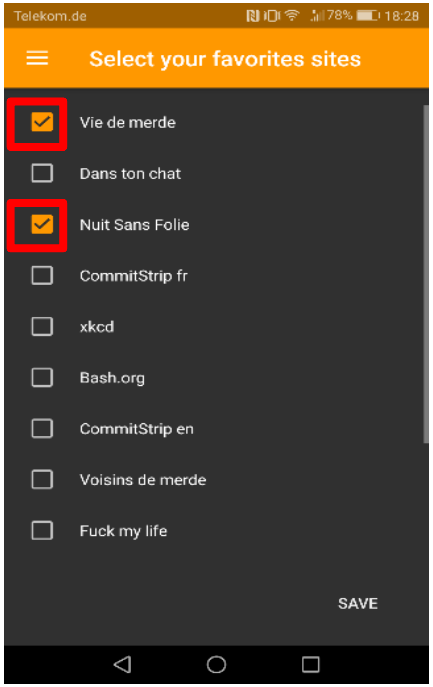
☑ CommitStrip en

□ Voisins de merde

□ Fuck my life

SAVE

Background
Foreground



Telekom.de 78% 18:28

Select your favorites sites

☑ Vie de merde

□ Dans ton chat

☑ Nuit Sans Folie

□ CommitStrip fr

□ xkcd

□ Bash.org

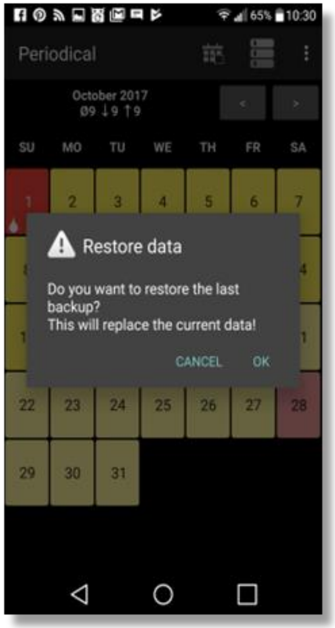
□ CommitStrip en

□ Voisins de merde

□ Fuck my life

SAVE

Exploratory Studies

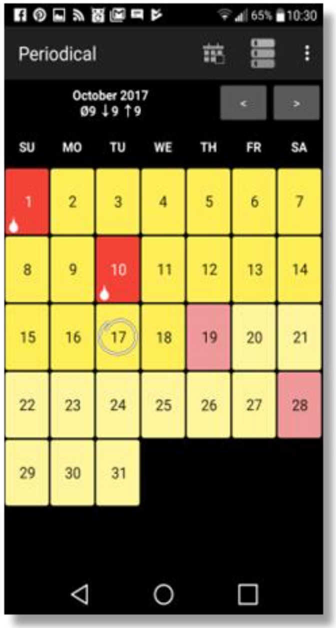


Restore data

Do you want to restore the last backup?
This will replace the current data!

CANCEL OK

Double
Orientation
Change

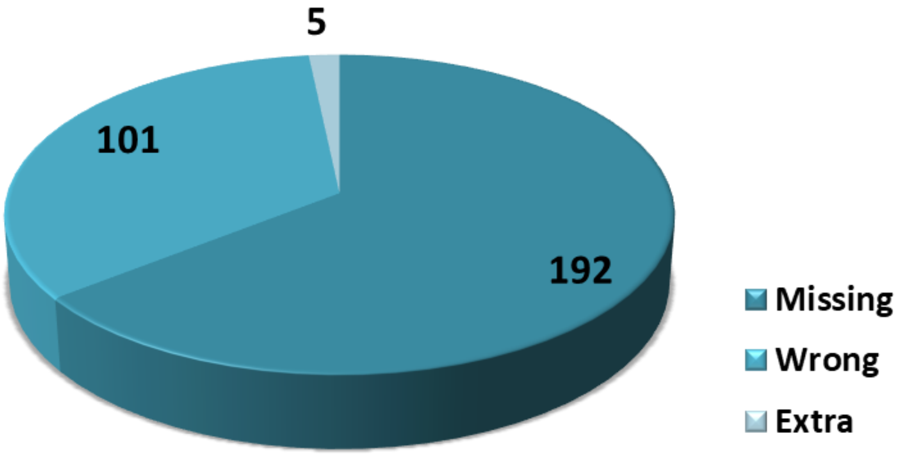


DOC GUI
Failure!!!

Missing
Dialog

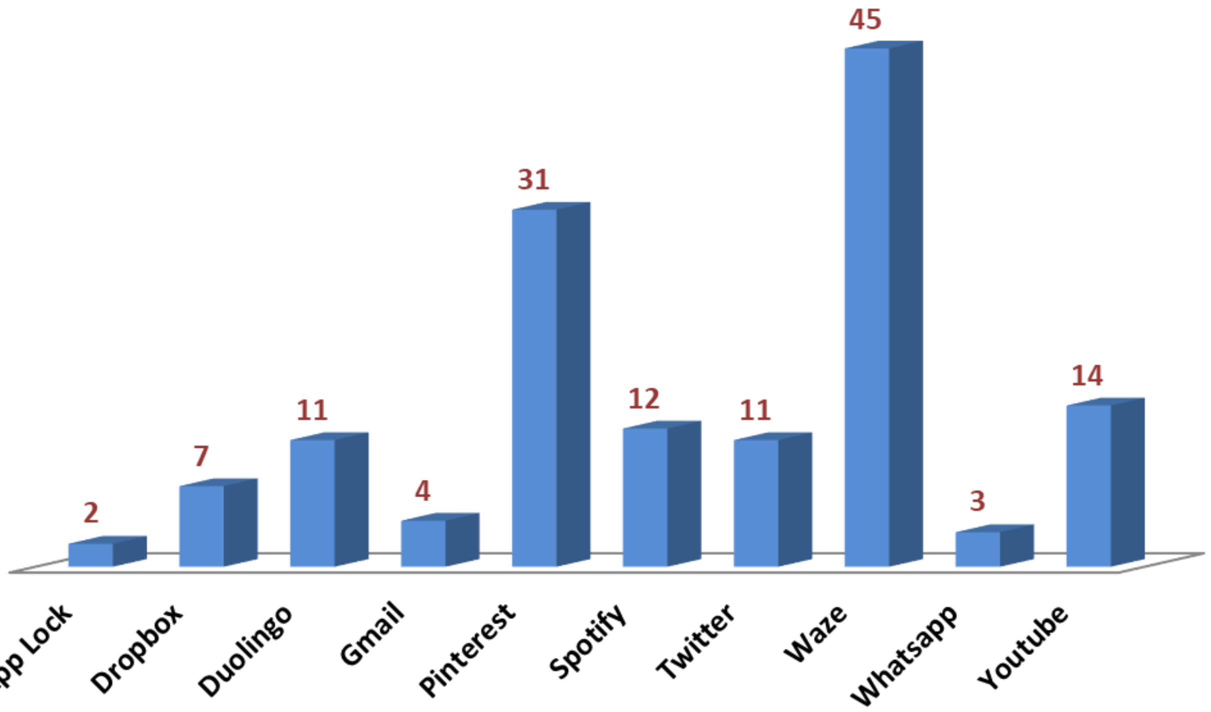
Exploratory Study 1

- 68 open-source apps
- 86% of the considered apps are affected by GUI failures due to orientation changes
- Most of the detected failures involve Dialog objects missing from the GUI after the DOC
- 6 classes of common faults causing GUI failures have been identified

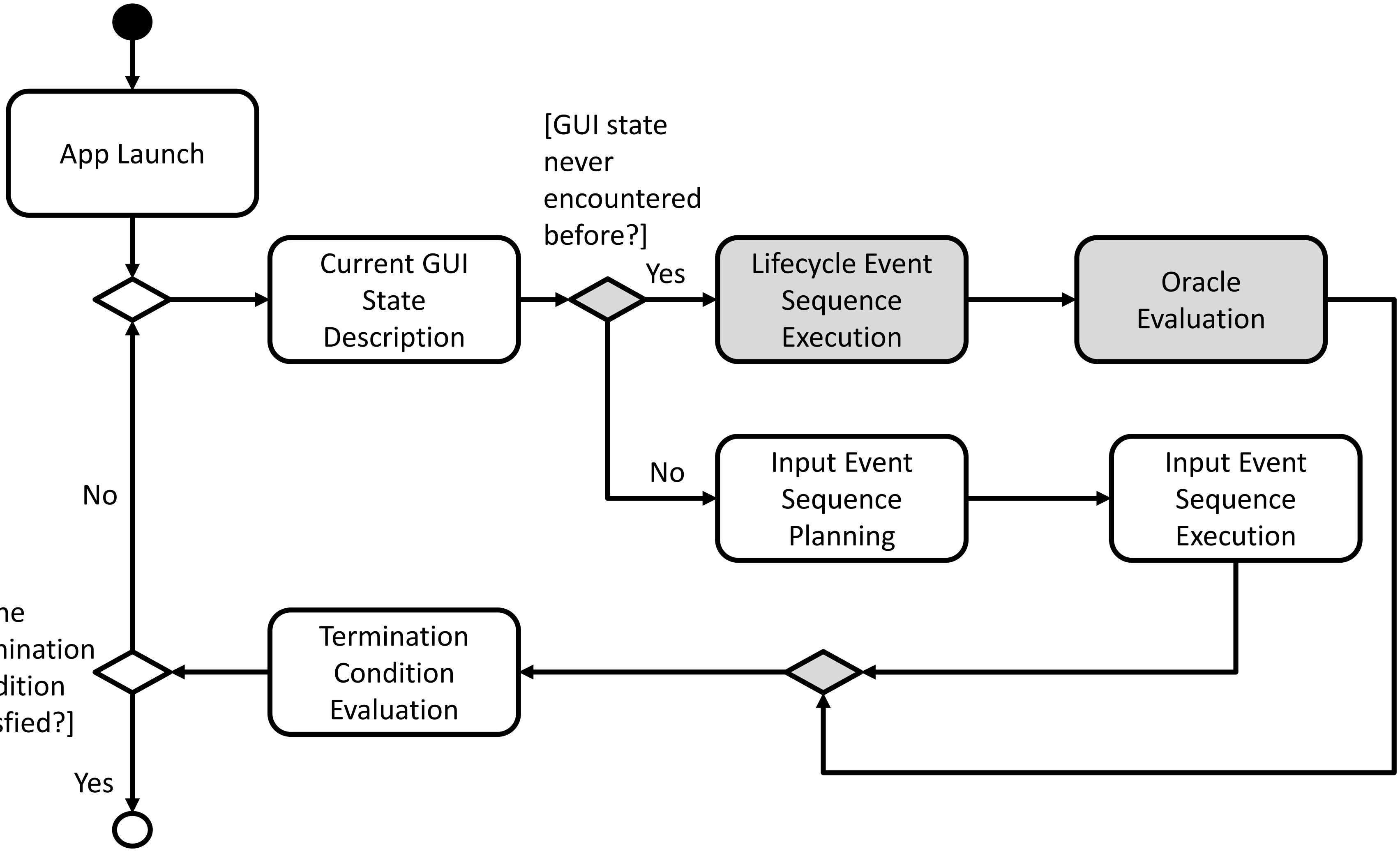


Exploratory Study 2

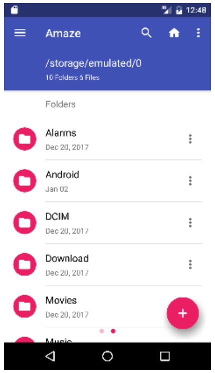
- 15 industrial-strength apps
- All the considered apps are affected by GUI failures due to orientation changes



The ALARic Approach


App Launch
Current GUI State Description
[GUI state never encountered before?]
Yes
Lifecycle Event Sequence Execution
Oracle Evaluation
No
Input Event Sequence Planning
Input Event Sequence Execution
No
[Is the termination condition satisfied?]
Termination Condition Evaluation
Yes

ALARic Workflow Example



12:48

Amaze

/storage/emulated/0

10 Folders 6 Files

Folders

Alarms
Dec 20, 2017

Android
Jan 02

DCIM
Dec 20, 2017

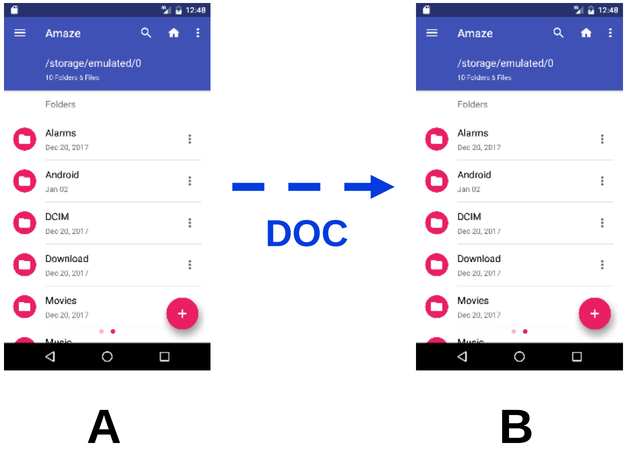
Download
Dec 20, 2017

Movies
Dec 20, 2017

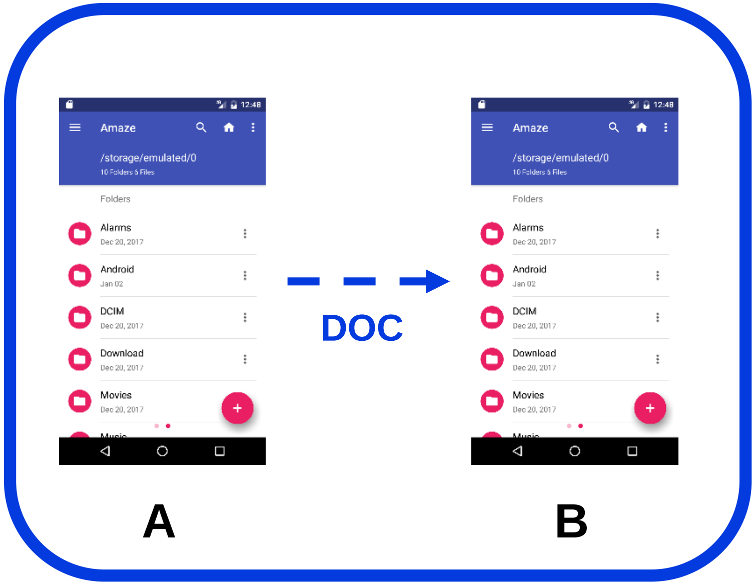
Music

A

ALARic Workflow Example

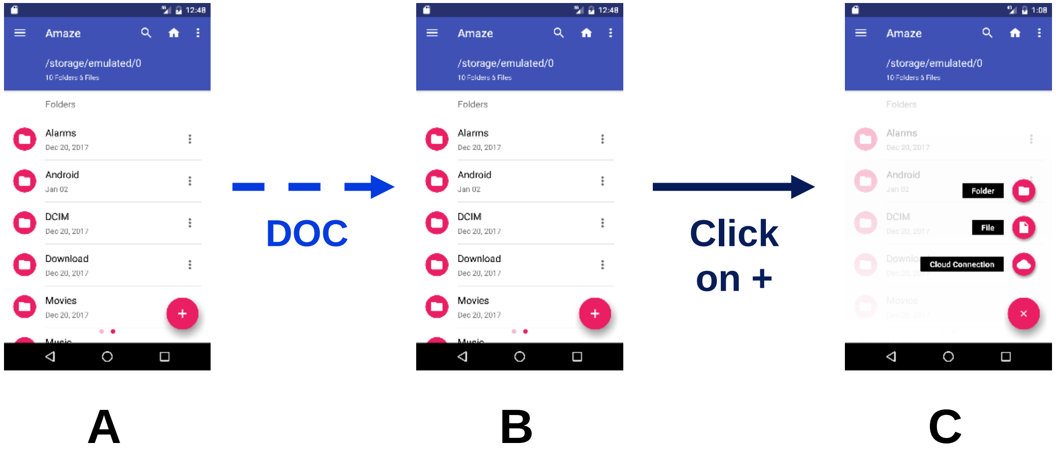


ALARic Workflow Example



B = A

ALARic Workflow Example



DOC

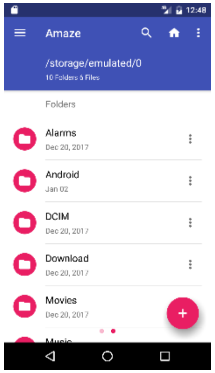
Click on +

A

B

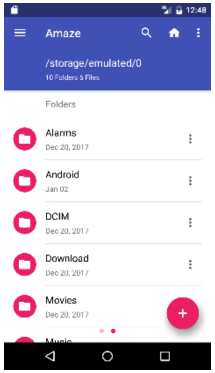
C

ALARic Workflow Example



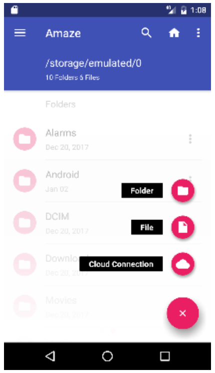
A

DOC



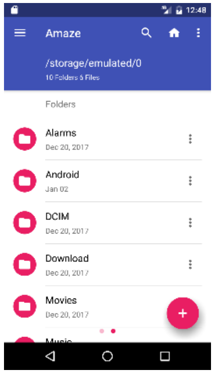
B

Click on +



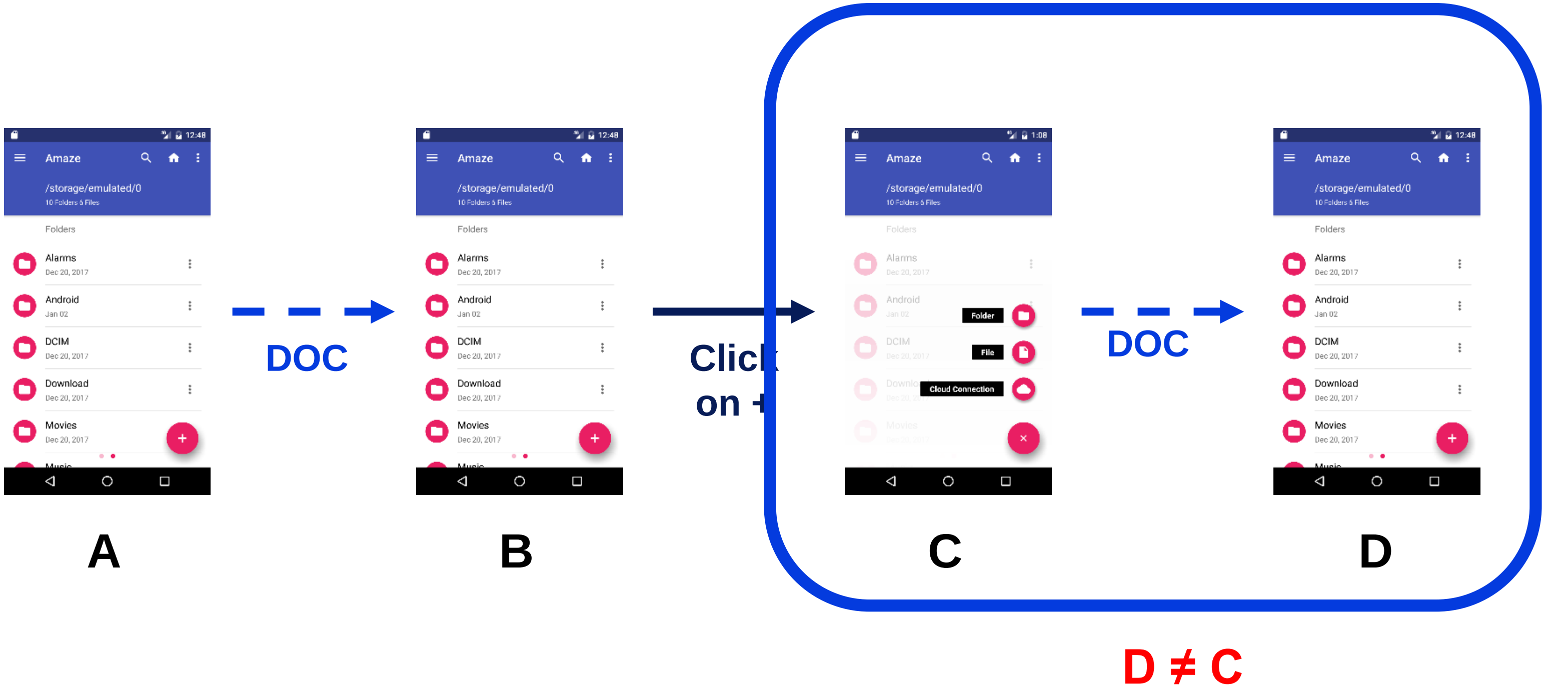
C

DOC

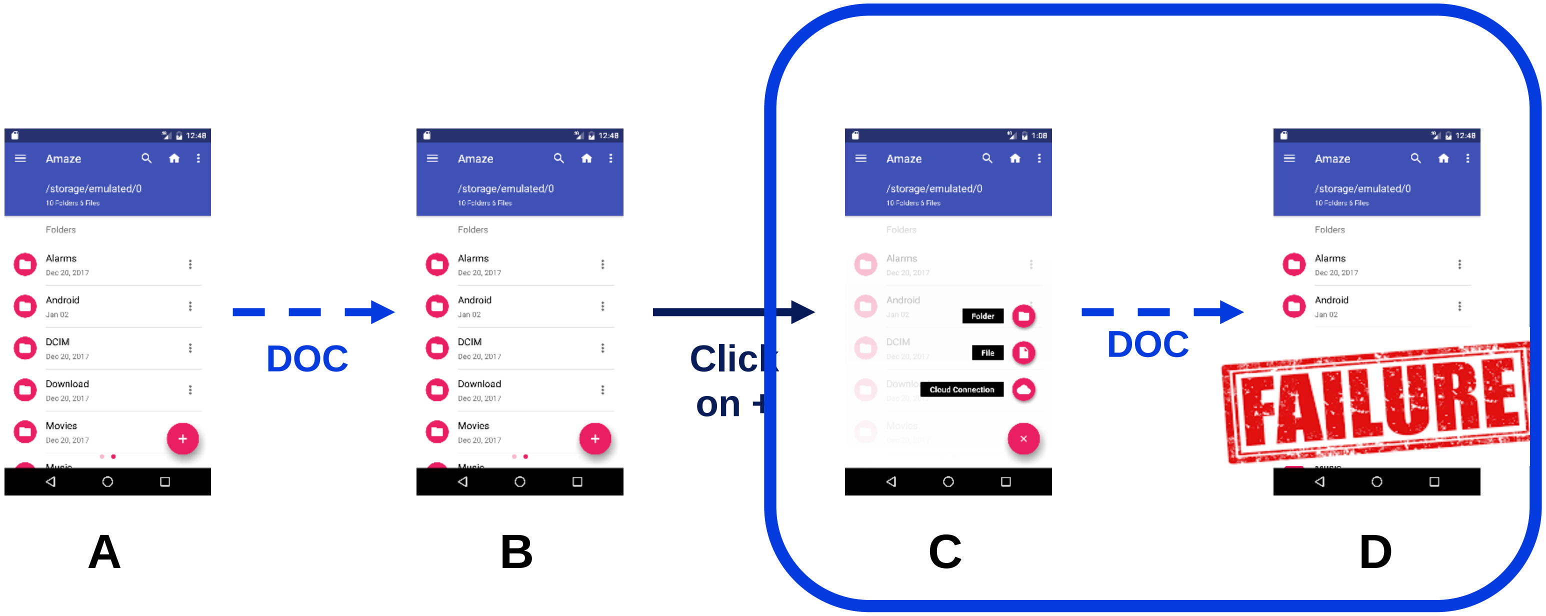


D

ALARic Workflow Example

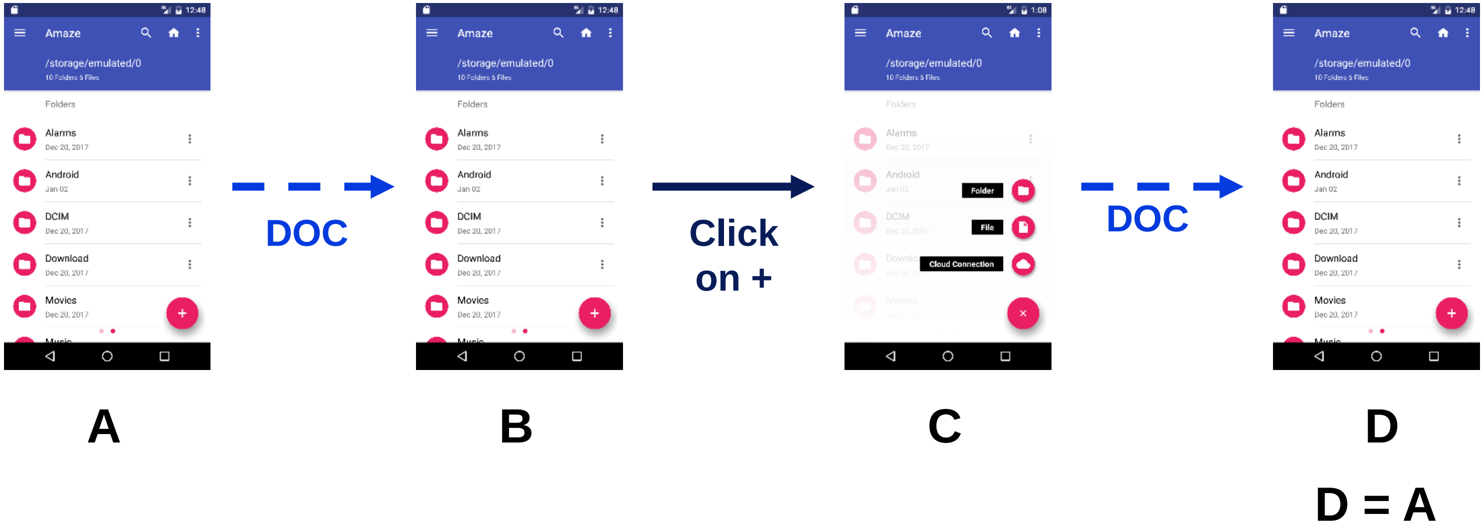


ALARic Workflow Example



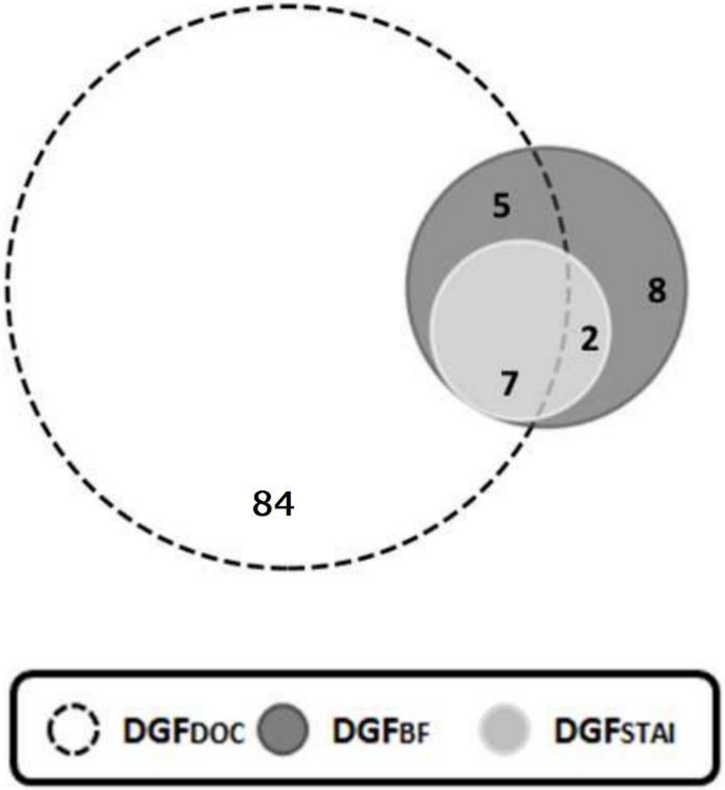
D ≠ C

ALARic Workflow Example



Experimental Results

- ALARic detected 106 distinct GUI failures in 15 analyzed apps

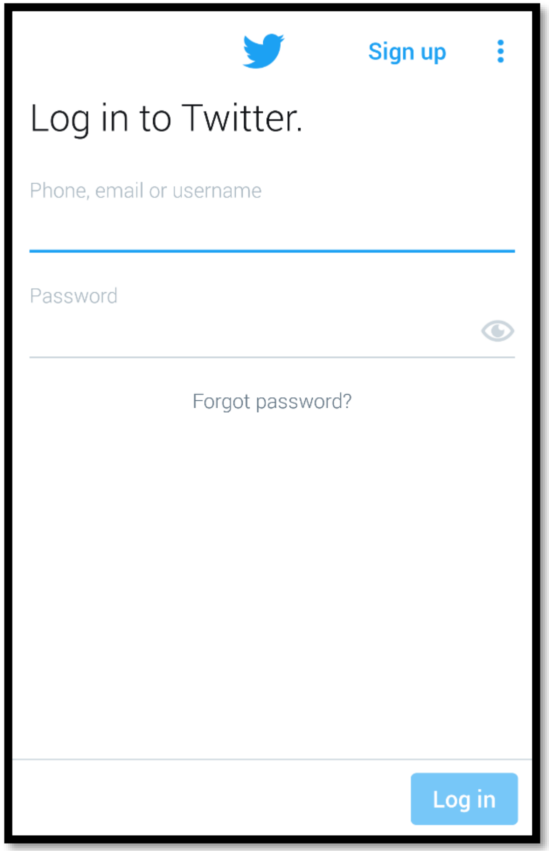


Challenge #2

Exploiting app-specific knowledge that only human users can provide

Gate GUI Unlocking

Gate GUIs



Sign up

Log in to Twitter.

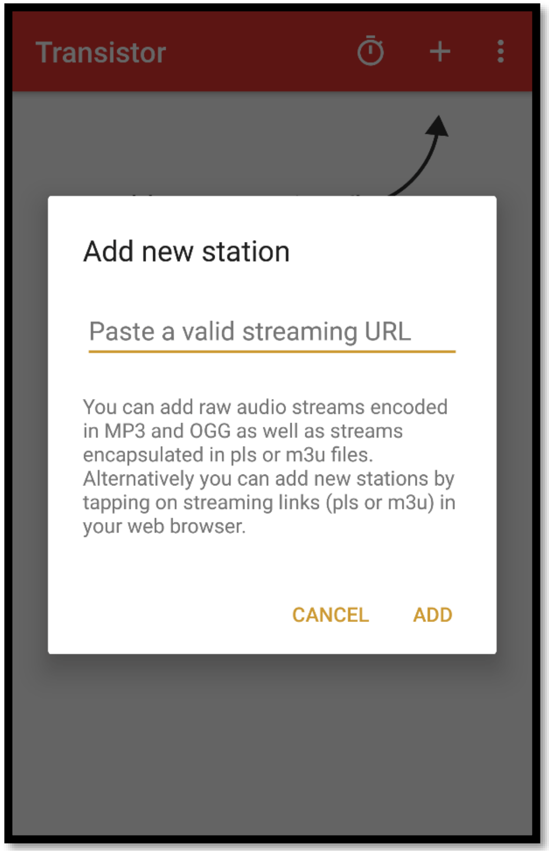
Phone, email or username

Password

Forgot password?

Log in

Login Gate GUI



Add new station

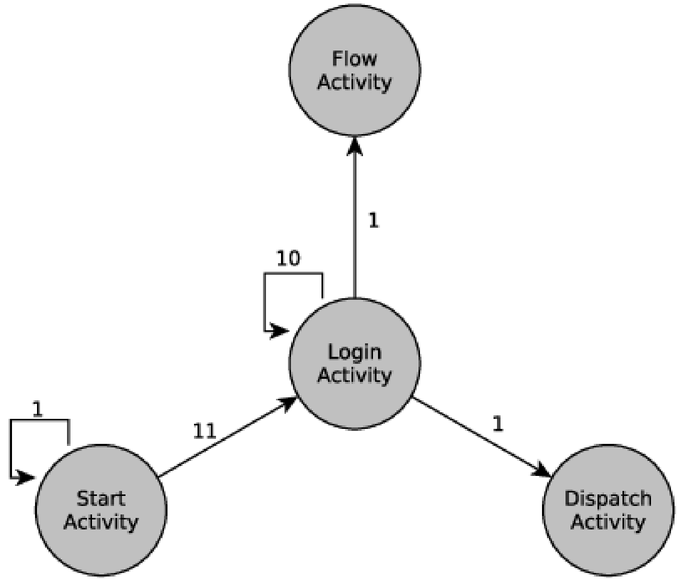
Paste a valid streaming URL

You can add raw audio streams encoded in MP3 and OGG as well as streams encapsulated in pls or m3u files. Alternatively you can add new stations by tapping on streaming links (pls or m3u) in your web browser.

CANCEL ADD

Settings Gate GUI

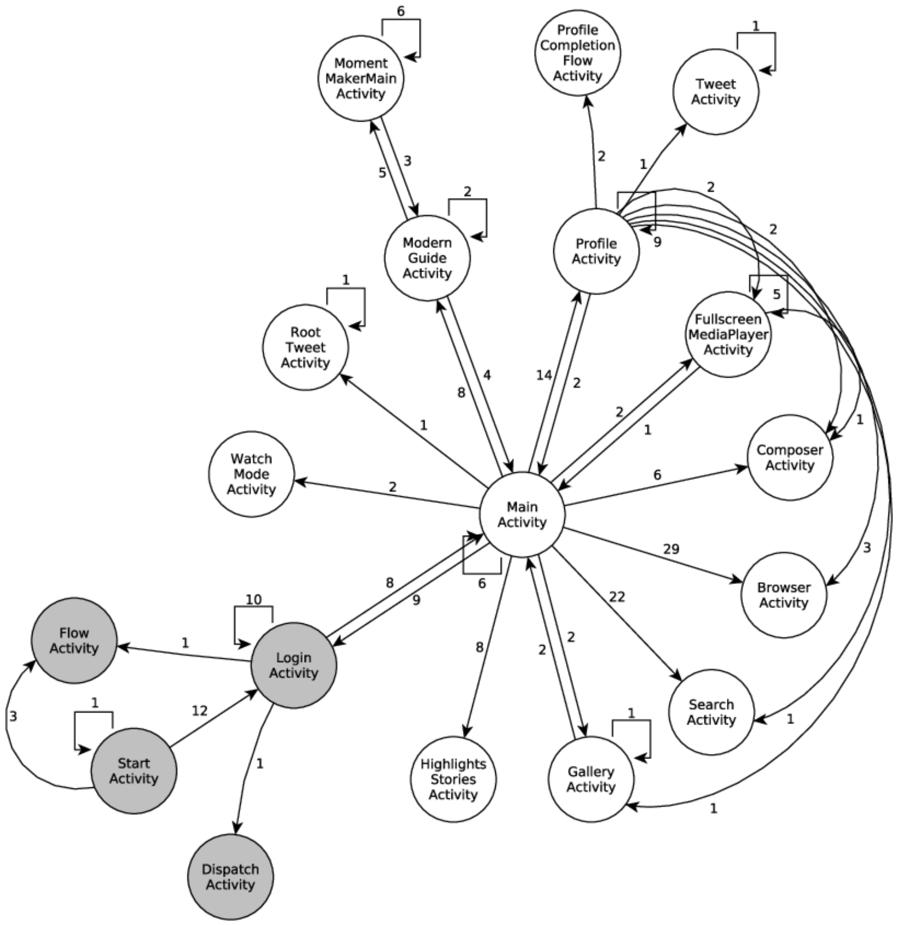
Gate GUI Locked



4 Activities

1 Mb

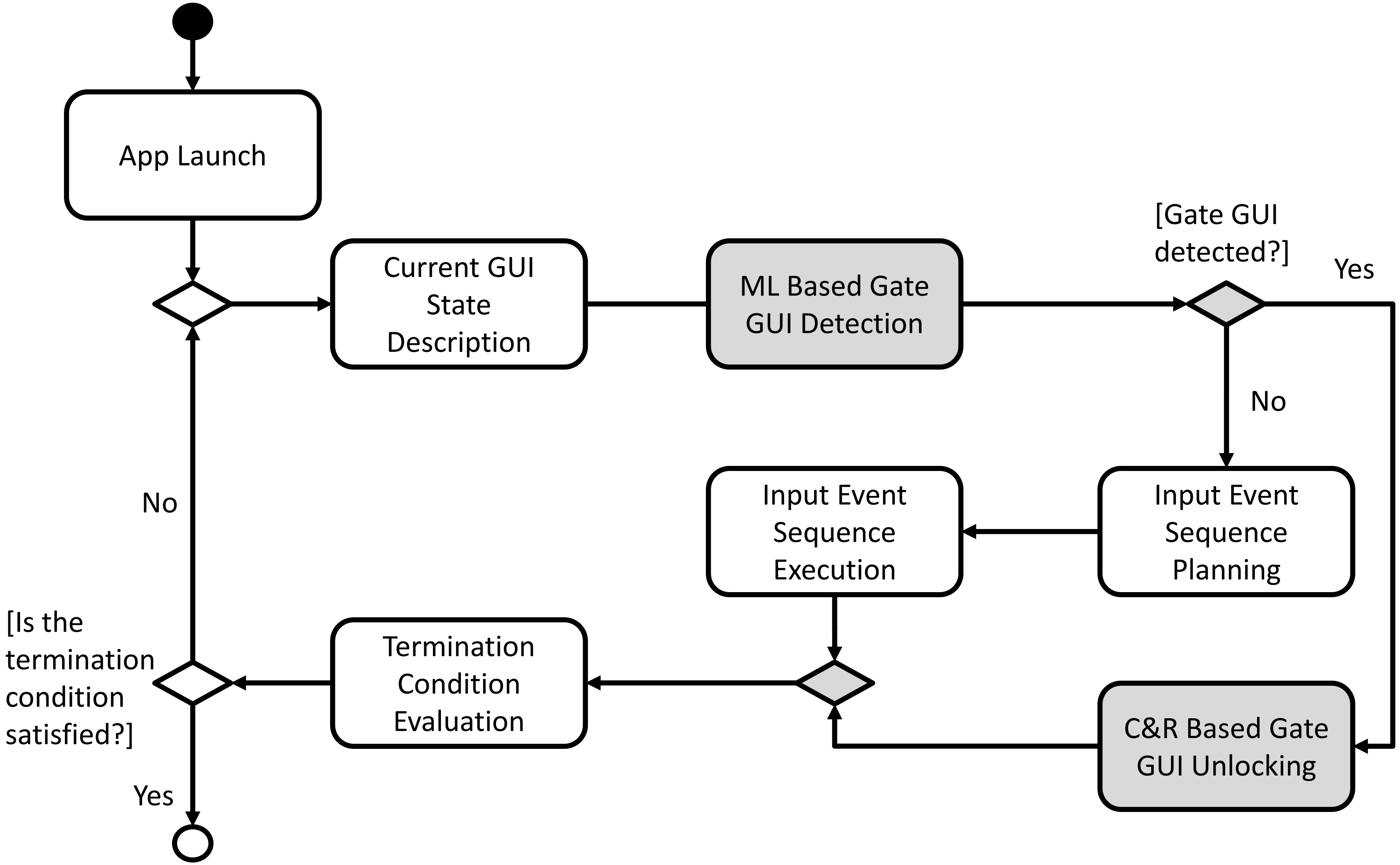
Gate GUI Unlocked



18 Activities

380 Mb

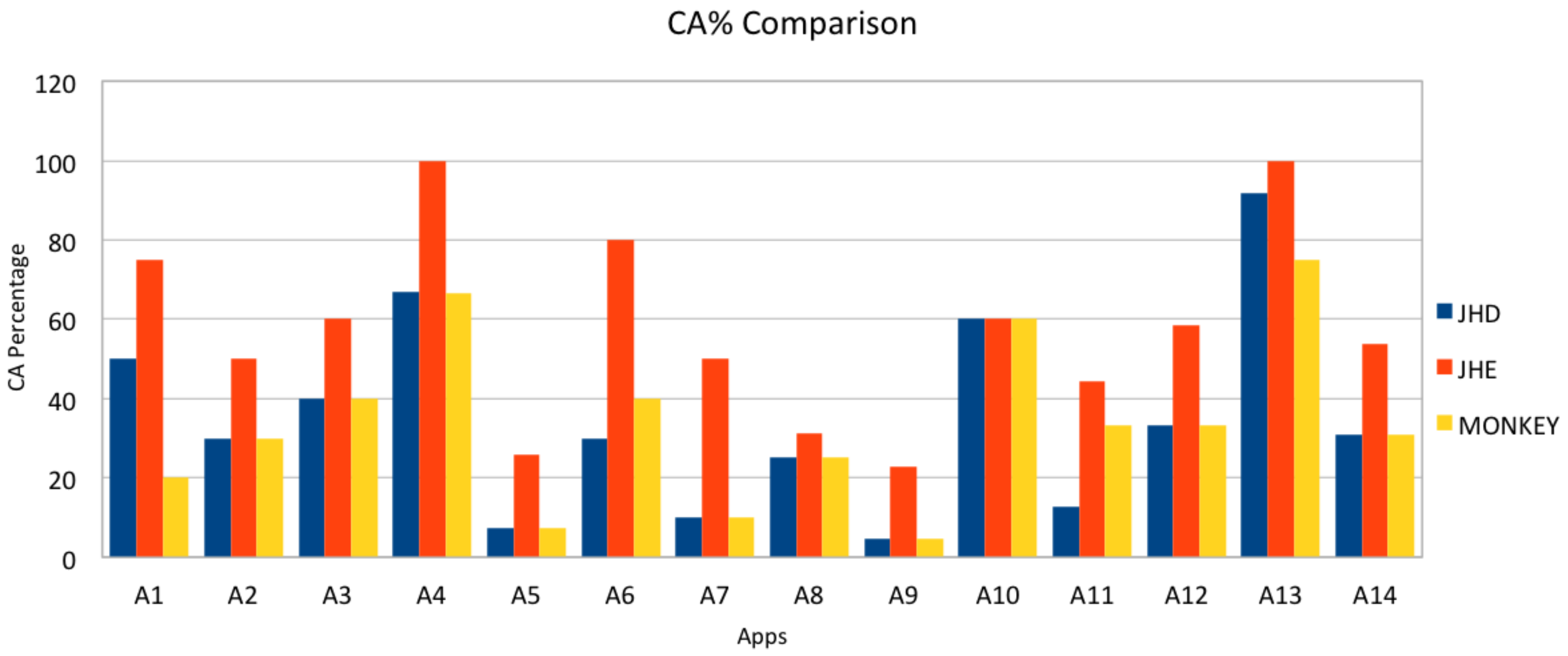
The juGULAR Approach


App Launch
Current GUI State Description
ML Based Gate GUI Detection
[Gate GUI detected?]
Yes
No
Input Event Sequence Planning
Input Event Sequence Execution
C&R Based Gate GUI Unlocking
Termination Condition Evaluation
[Is the termination condition satisfied?]
No
Yes

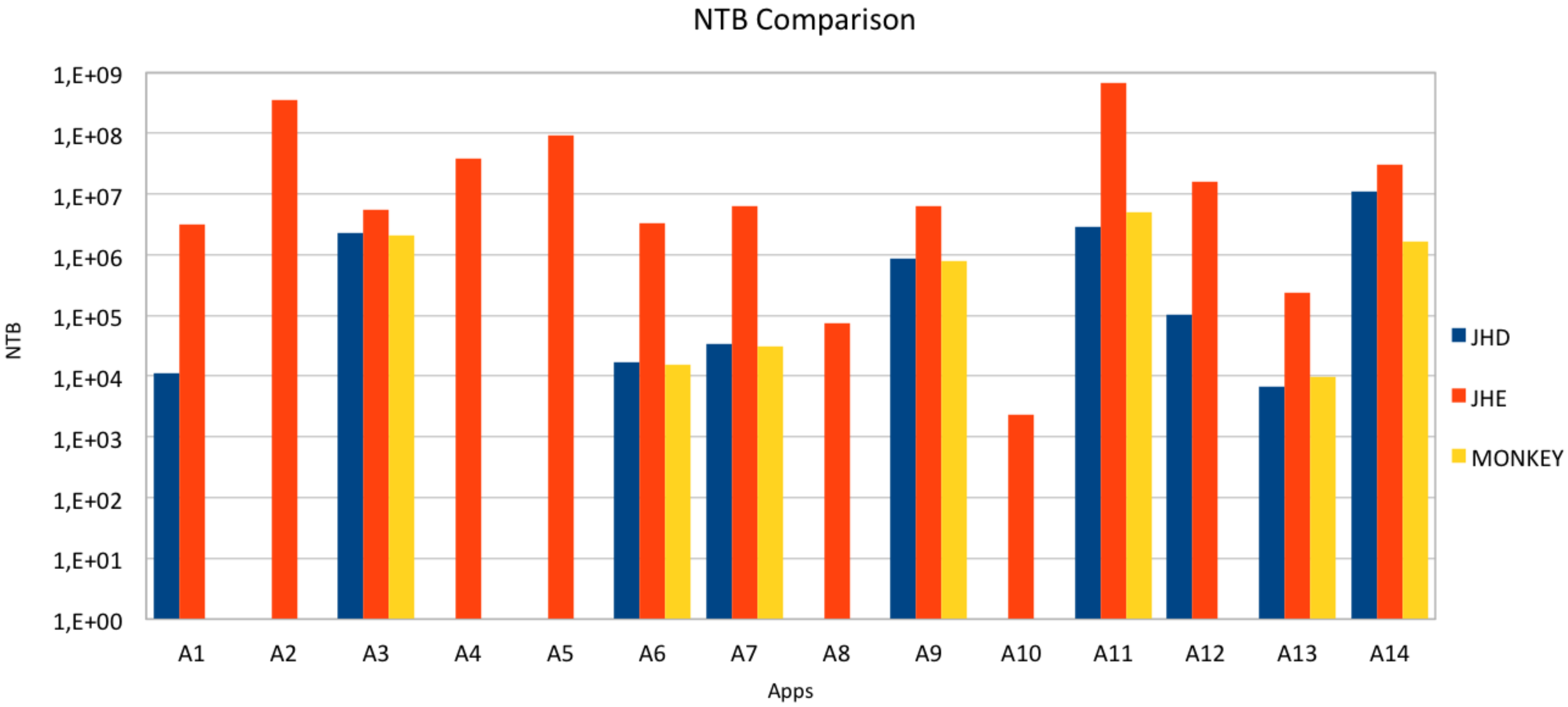
Experimental Evaluation

- Comparison between
  - juGULAR with Hybridization Disabled (**JHD**)
  - juGULAR with Hybridization Enabled (**JHE**)
  - The state-of-the-practice tool, **Monkey**

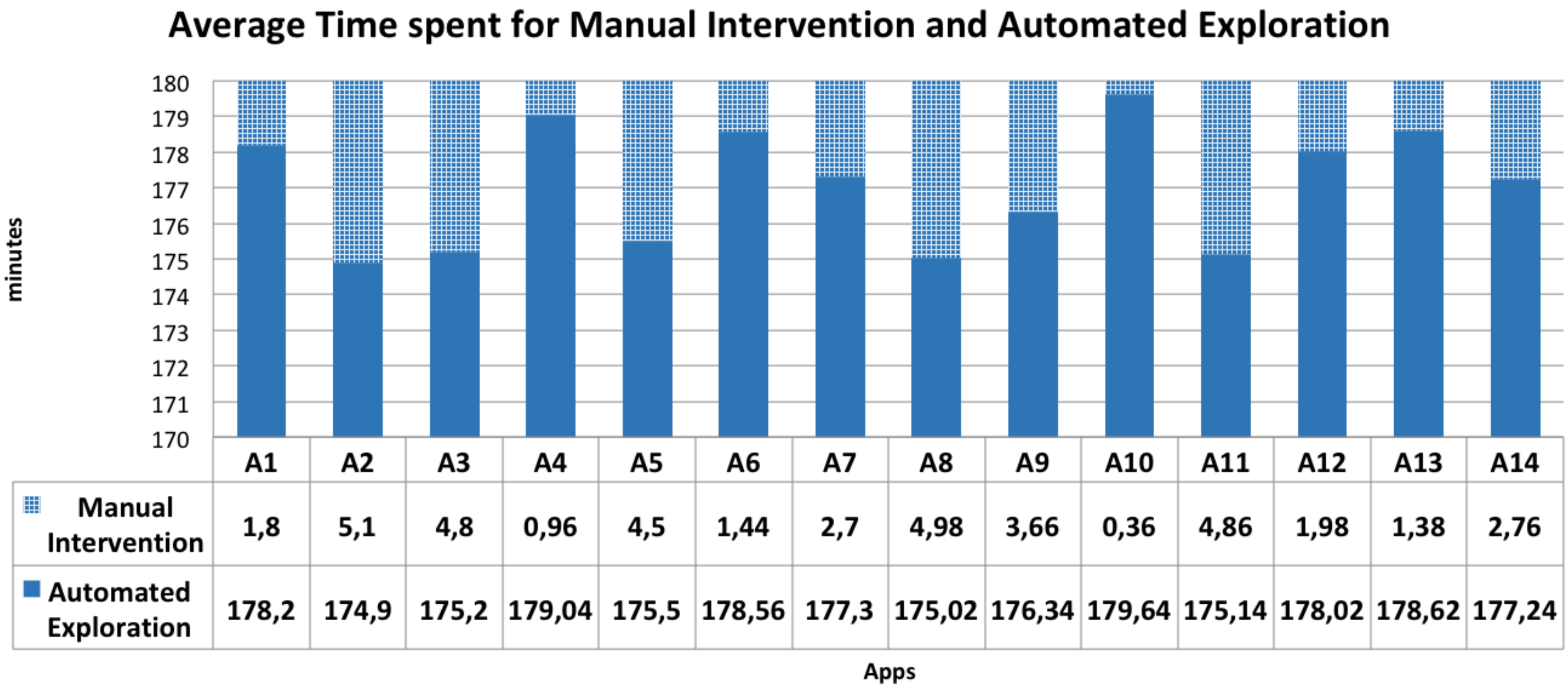
Covered Activities



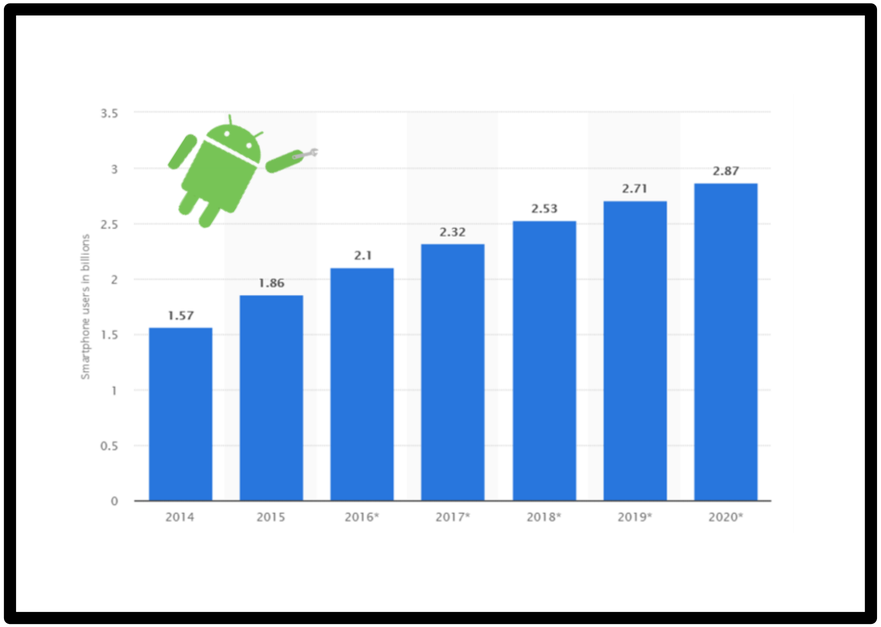
Network Traffic Bytes

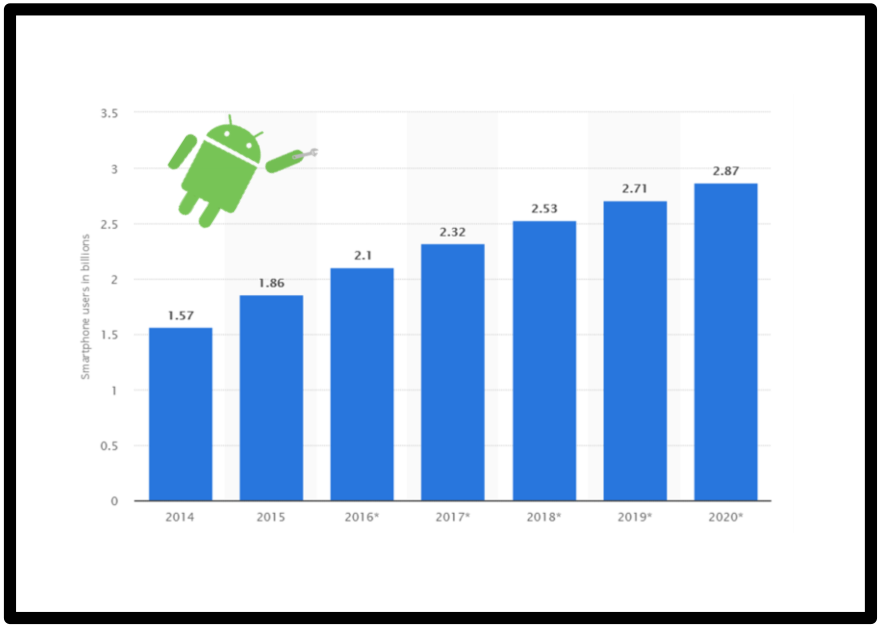


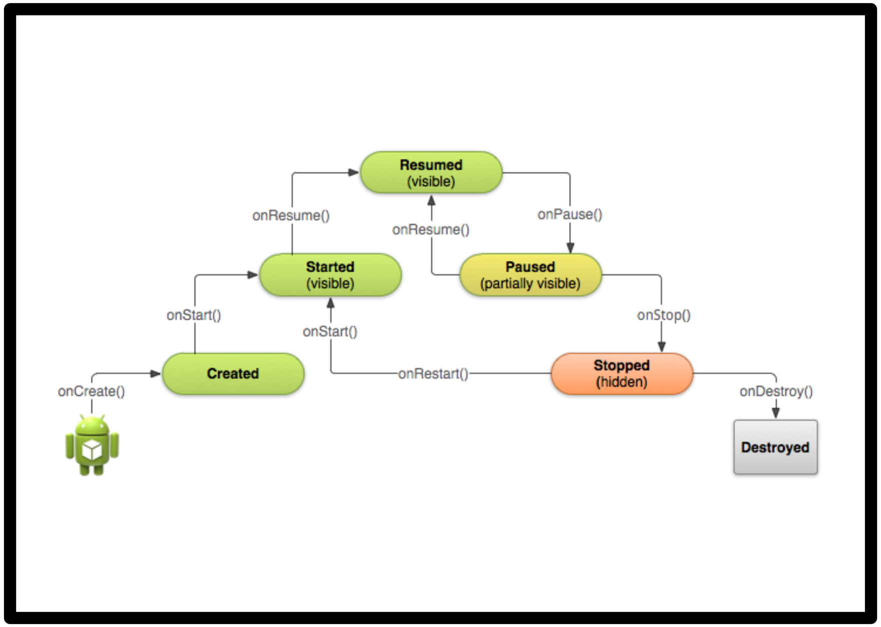
Manual Intervention Percentage

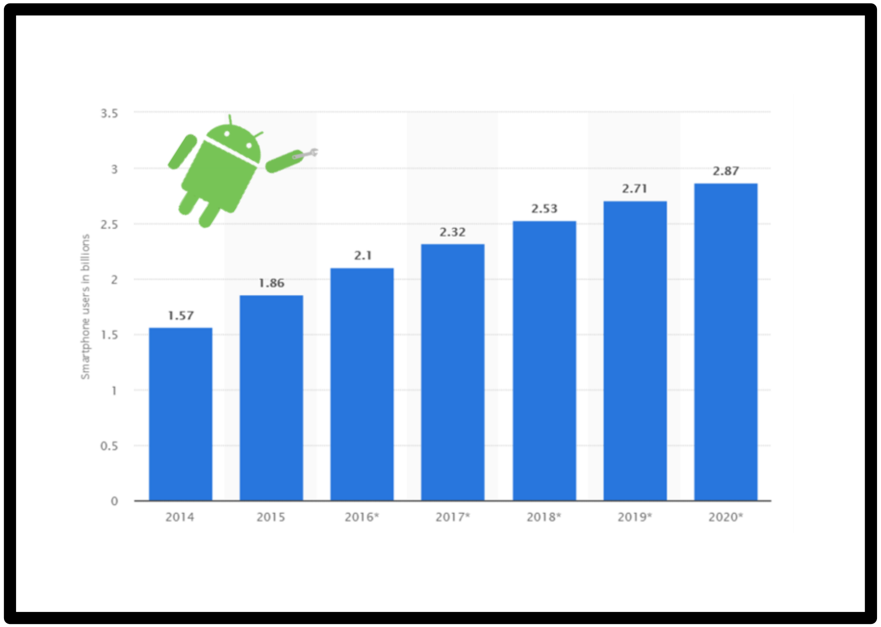


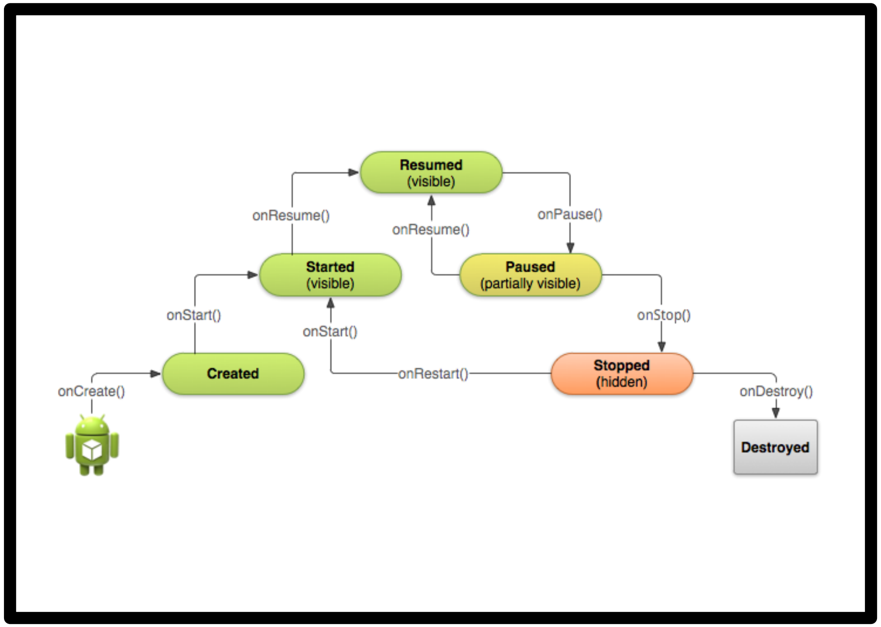
	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14
▦ Manual Intervention	1,8	5,1	4,8	0,96	4,5	1,44	2,7	4,98	3,66	0,36	4,86	1,98	1,38	2,76
■ Automated Exploration	178,2	174,9	175,2	179,04	175,5	178,56	177,3	175,02	176,34	179,64	175,14	178,02	178,62	177,24

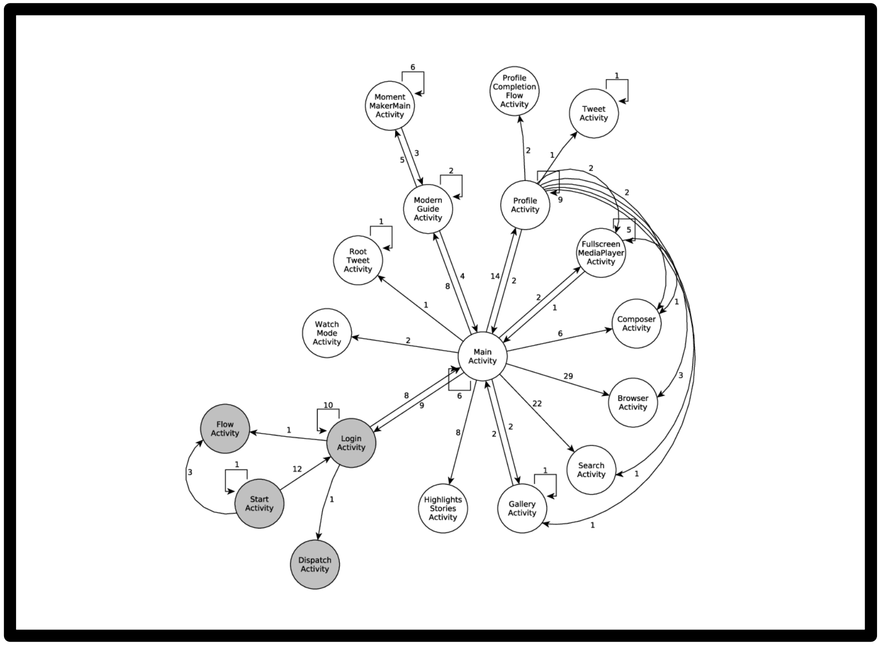

3.5
3
2.5
2
1.5
1
0.5
0
Smartphone users in billions
1.57
1.86
2.1
2.32
2.53
2.71
2.87
2014
2015
2016*
2017*
2018*
2019*
2020*

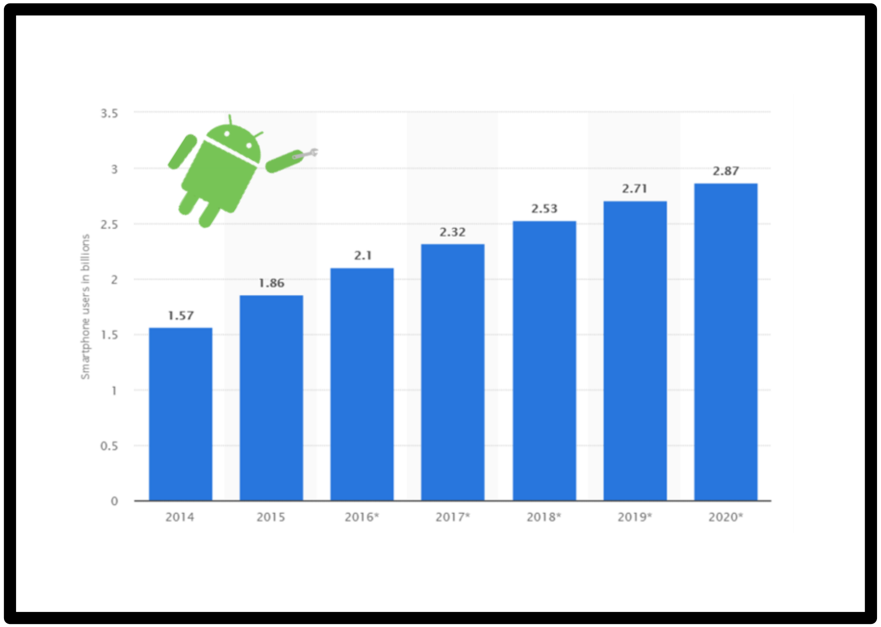

3.5
3
2.5
2
1.5
1
0.5
0
Smartphone users in billions
1.57
1.86
2.1
2.32
2.53
2.71
2.87
2014
2015
2016*
2017*
2018*
2019*
2020*

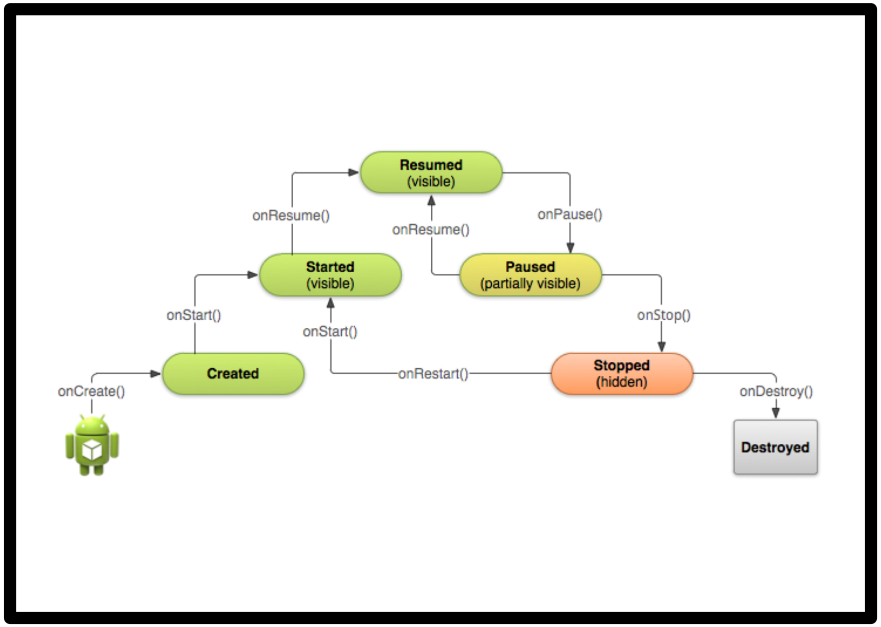

Resumed
(visible)
onResume()
onResume()
onPause()
Started
(visible)
Paused
(partially visible)
onStart()
onStart()
onStop()
Created
onRestart()
Stopped
(hidden)
onCreate()
onDestroy()
Destroyed

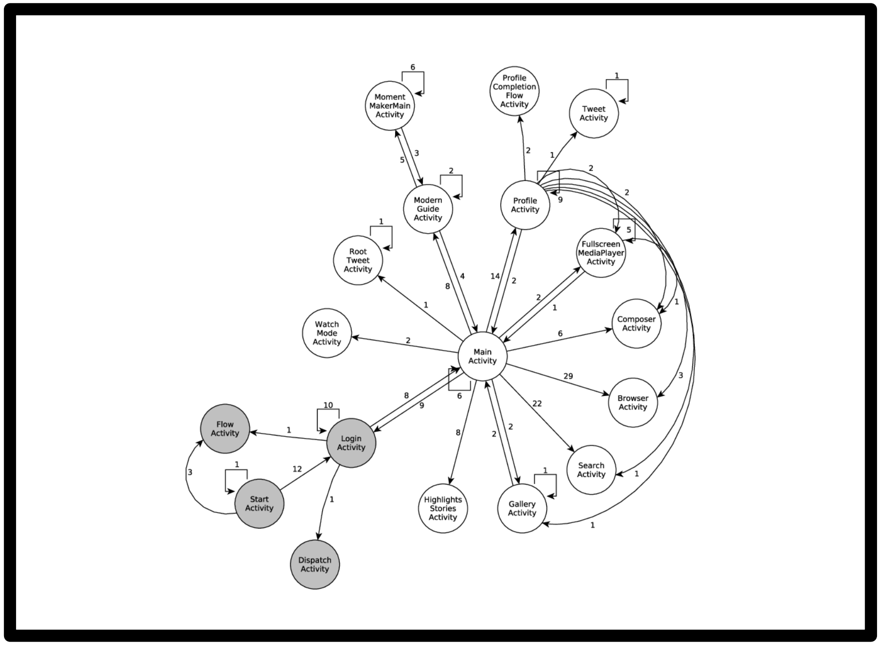

Smartphone users in billions
3.5
3
2.5
2
1.5
1
0.5
0
1.57
1.86
2.1
2.32
2.53
2.71
2.87
2014
2015
2016*
2017*
2018*
2019*
2020*

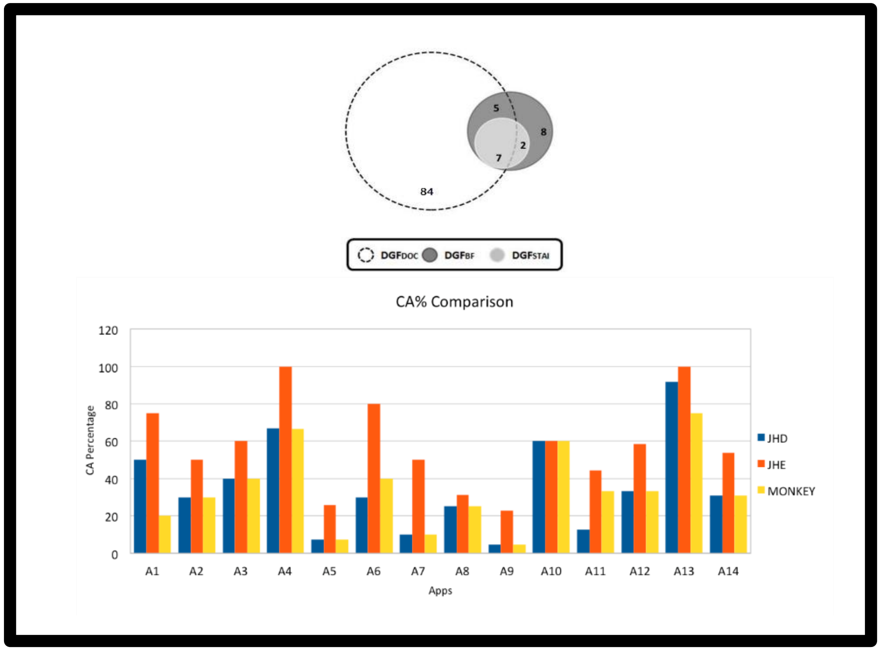

Resumed
(visible)
onResume()
onResume()
onPause()
Started
(visible)
Paused
(partially visible)
onStart()
onStart()
onStop()
onCreate()
Created
onRestart()
Stopped
(hidden)
onDestroy()
Destroyed


Moment MakerMain Activity
Profile Completion Flow Activity
Tweet Activity
Modern Guide Activity
Profile Activity
Root Tweet Activity
Fullscreen MediaPlayer Activity
Watch Mode Activity
Composer Activity
Main Activity
Browser Activity
Flow Activity
Login Activity
Start Activity
Search Activity
Highlights Stories Activity
Gallery Activity
Dispatch Activity
6
3
5
2
2
1
1
2
2
9
5
1
8
4
14
2
1
2
1
6
2
29
22
3
8
9
6
10
1
12
1
3
1
8
2
2
1
1
1


Smartphone users in billions
3.5
3
2.5
2
1.5
1
0.5
0
1.57
1.86
2.1
2.32
2.53
2.71
2.87
2014
2015
2016*
2017*
2018*
2019*
2020*


Resumed
(visible)
onResume()
onResume()
onPause()
Started
(visible)
Paused
(partially visible)
onStart()
onStart()
onStop()
Created
onRestart()
Stopped
(hidden)
onCreate()
onDestroy()
Destroyed


Moment MakerMain Activity
6
Profile Completion Flow Activity
Tweet Activity
1
5
3
2
1
Modern Guide Activity
2
Profile Activity
2
9
2
Root Tweet Activity
1
Fullscreen MediaPlayer Activity
5
8
4
14
2
2
1
1
1
Watch Mode Activity
2
Main Activity
6
Composer Activity
29
3
Browser Activity
8
9
6
22
Flow Activity
10
1
Login Activity
3
1
12
Start Activity
8
2
2
Search Activity
1
1
1
Highlights Stories Activity
Gallery Activity
1
1
Dispatch Activity


5
8
2
7
84
DGF_{DOC}
DGF_{BF}
DGF_{STAI}
CA% Comparison
CA Percentage
120
100
80
60
40
20
0
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
A13
A14
Apps
JHD
JHE
MONKEY

Products (1/2)

- **Journal Papers:**
  - D Amalfitano, V Riccio, ACR Paiva, and AR Fasolino (2018). **Why does the orientation change mess up my Android application? From GUI failures to code faults.** *Software Testing, Verification and Reliability, 28(1)*. Wiley. doi:10.1002/stvr.1654.
    - In collaboration with the University of Porto
    - **Wiley's #Top20Article:** Amongst articles published by Wiley between July 2016 and June 2018, this article received some of the highest downloads in the 12-months post online publication
  - D Amalfitano, V Riccio, N Amatucci, V De Simone, and AR Fasolino (2018) **Combining Automated GUI Exploration of Android apps with Capture and Replay through Machine Learning.** *Information and Software Technology, 105(1)*. Elsevier. doi:10.1016/j.infsof.2018.08.007.

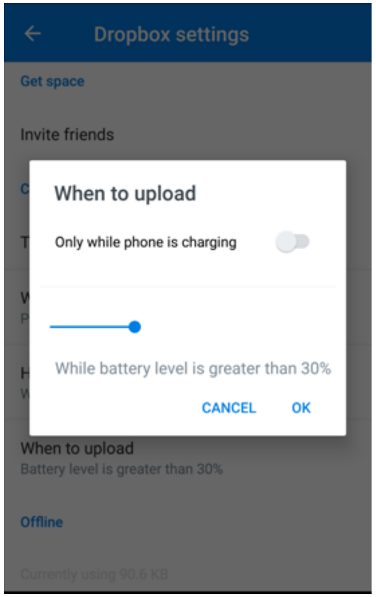
Products (2/2)

- **Conference Papers:**
  - D Amalfitano, V De Simone, A R Fasolino and V Riccio (2015). **Comparing Model Coverage and Code Coverage in Model Driven Testing: An Exploratory Study**, In *Proceedings of the 30th IEEE/ACM International Conference on Automated Software Engineering Workshop (ASEW)*, Lincoln, NE, 2015, pp. 70-73. doi: 10.1109/ASEW.2015.18
  - Domenico Amalfitano, Nicola Amatucci, Vincenzo De Simone, Vincenzo Riccio, and Fasolino Anna Rita (2017). **Towards a Thing-In-the-Loop approach for the Verification and Validation of IoT systems.** In *Proceedings of the 1st ACM Workshop on the Internet of Safe Things (SafeThings'17)*, Rasit Eskicioglu (Ed.). ACM, New York, NY, USA, pp. 57-63. doi: 10.1145/3137003.3137007
  - Vincenzo Riccio, Domenico Amalfitano, and Anna Rita Fasolino (2018). **Is This the Lifecycle We Really Want? An Automated Black-Box Testing Approach for Android Activities.** *The Joint Workshop of 4th Workshop on UI Test Automation and 8th Workshop on TESting Techniques for eventBasED Software (INTUITESTBEDS 2018)*. ACM (In press).

Extra Slides

Motivating Example: Crash

- A crash occurs when an app stops functioning properly and exits unexpectedly



Dropbox settings

Get space

Invite friends

When to upload

Only while phone is charging

While battery level is greater than 30%

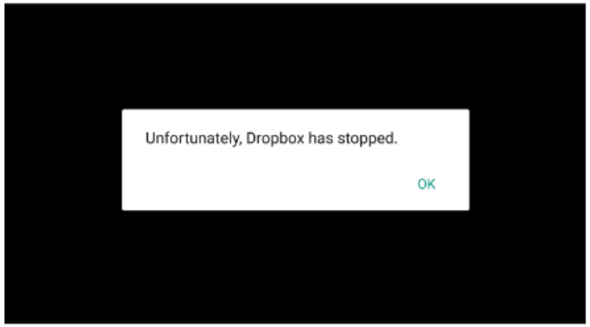
CANCEL OK

When to upload
Battery level is greater than 30%

Offline

Currently using 90.6 KB

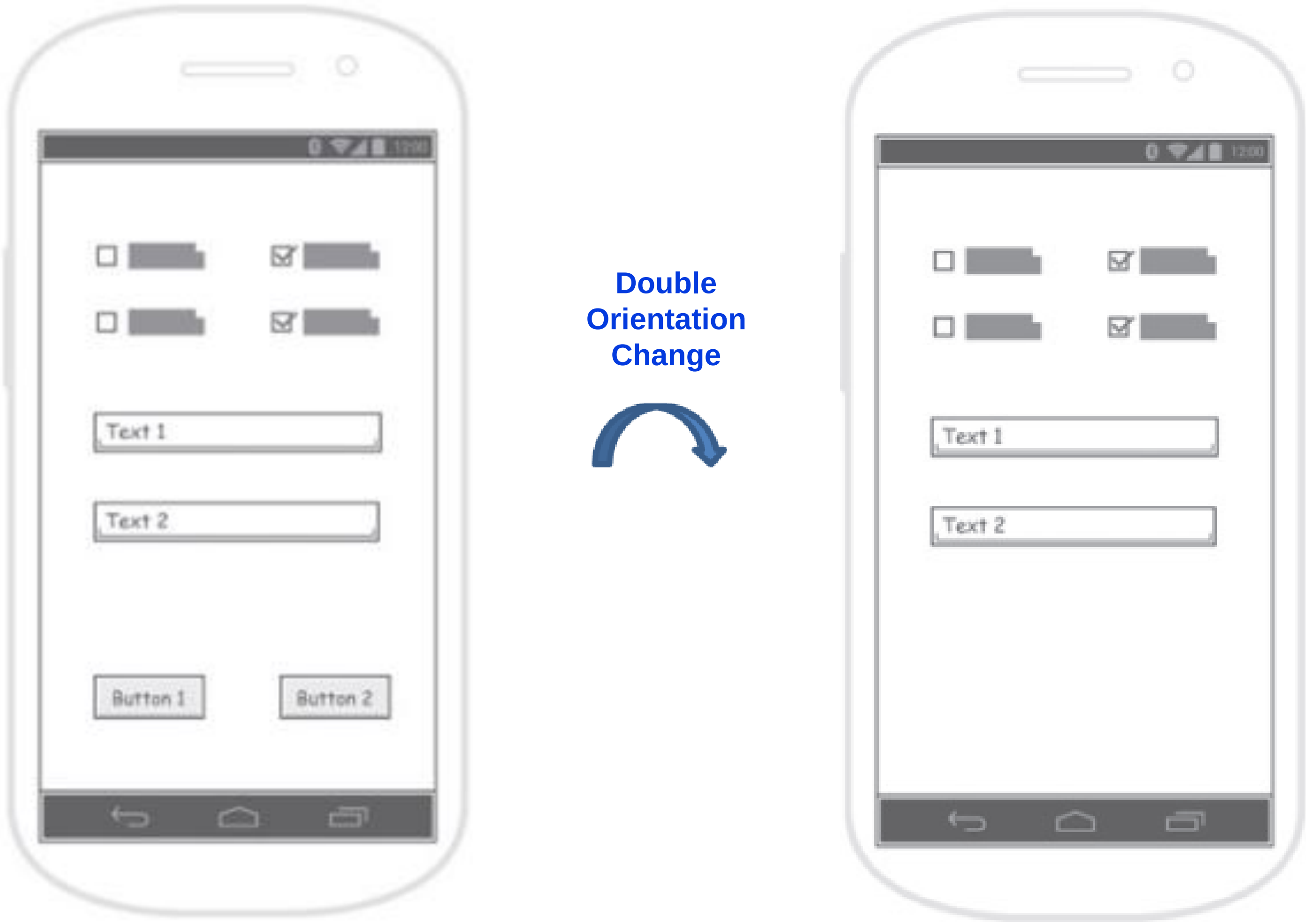
Orientation Change



Unfortunately, Dropbox has stopped.

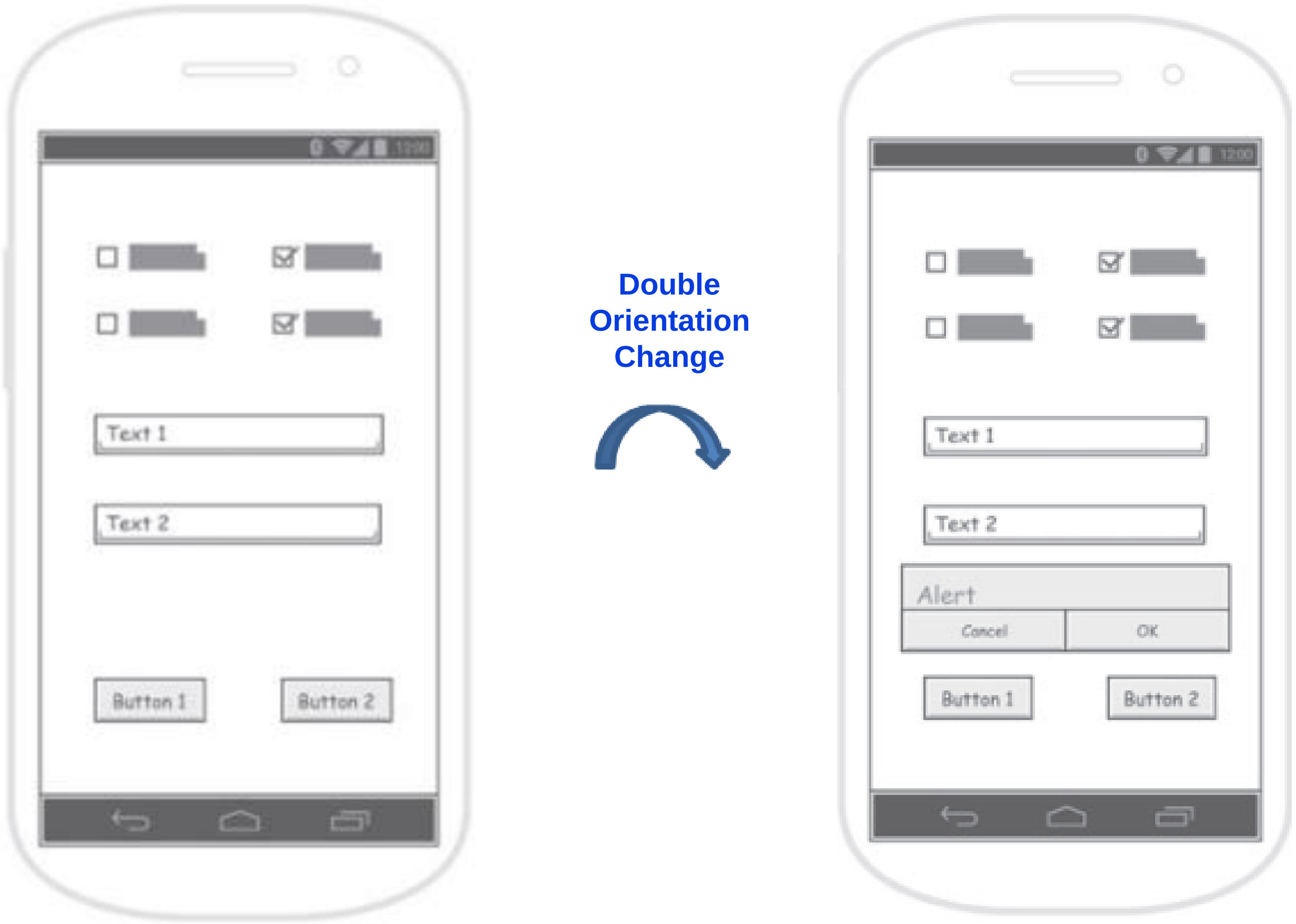
OK

Missing GUI Failure



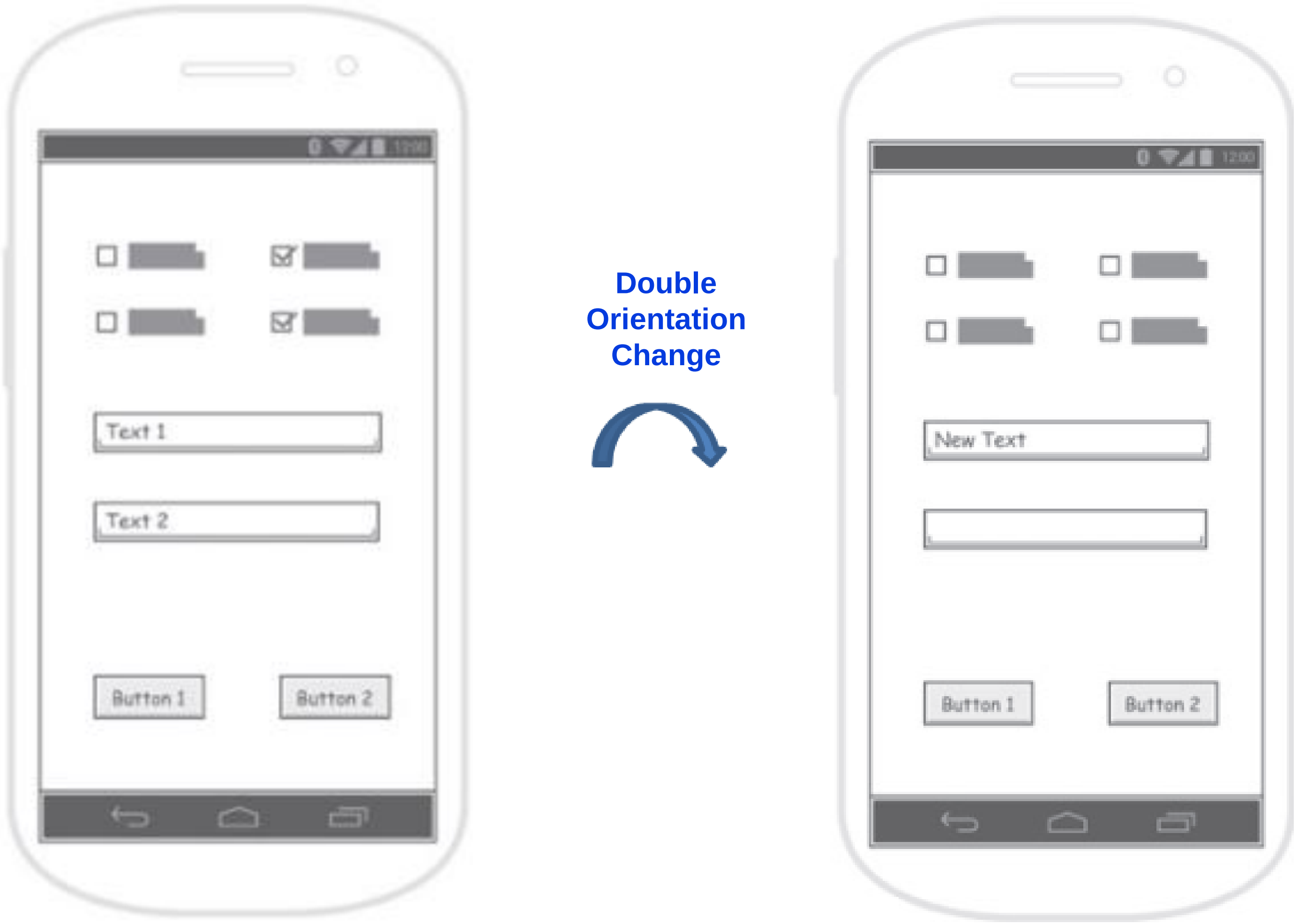
Double Orientation Change

Extra GUI Failure



Double Orientation Change

Wrong GUI Failure



Double Orientation Change

ALARic Description

- **ALARic (Activity Lifecycle Android Ripper)**, a novel fully automated Black-Box Event-based testing technique to detect issues tied to the Activity lifecycle
- It combines:
  - The traditional testing approaches based on dynamic app exploration
  - A strategy that systematically exercises the Activity lifecycle on each GUI state encountered during the exploration
- It relies on:
  - **Lifecycle Event Sequences**, mobile-specific events able to exercise the Activity lifecycle
  - Testing oracles to detect crashes and GUI failures tied to the Activity lifecycle

Experimental Evaluation

- **GOAL:** Evaluate the ability of ALARic to automatically detect crashes and GUI failures tied to the Activity lifecycle
  - **RQ1:** How effective is the ALARic tool in detecting issues tied to the Activity lifecycle in real Android apps?
  - **RQ2:** How does the effectiveness of the ALARic tool in detecting crashes tied to the Activity lifecycle in real Android apps compare to the state-of-the-practice tool, Monkey?

Objects

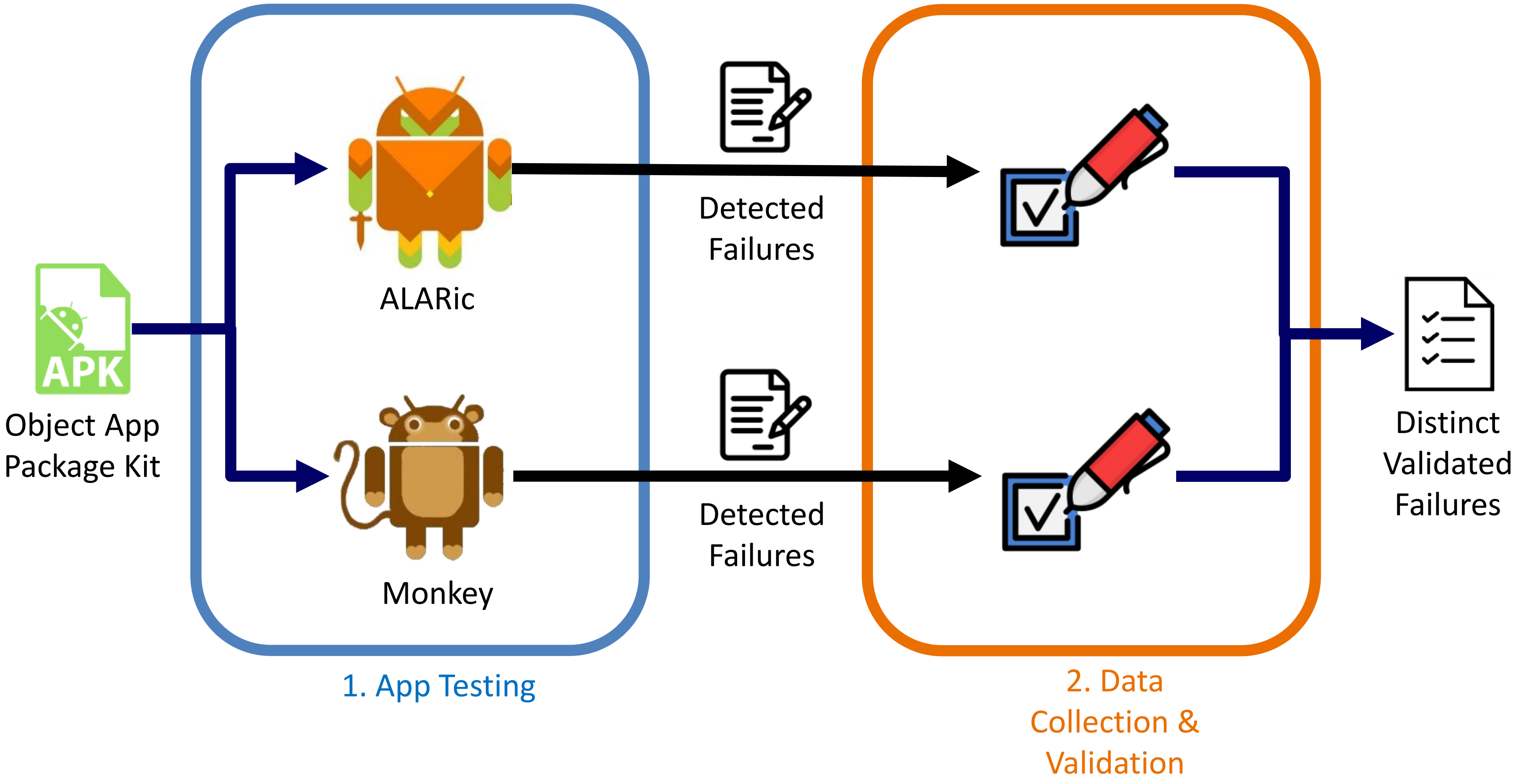
- 15 apps that are distributed by Google Play Store whose source code is available in the F-Droid repository

ID	App	Version	Activities
A1	A Time Tracker	0.21	5
A2	Port Knocker	1.0.9	6
A3	Who Has My Stuff?	1.0.27	4
A4	Agram	1.4.1	5
A5	Alarm Klock	1.9	5
A6	Padland	1.3	10
A7	Syncthing	0.9.1	12
A8	Anecdote	1.1.2	3
A9	Amaze File Manager	3.1.2 RC4	5
A10	Google Authenticator	2.21	5
A11	BeeCount	2.3.9	8
A12	FOSDEM companion	1.4.6	8
A13	Periodical	0.30	6
A14	Taskbar	3.0.2	23
A15	SpaRSS	1.11.8	8

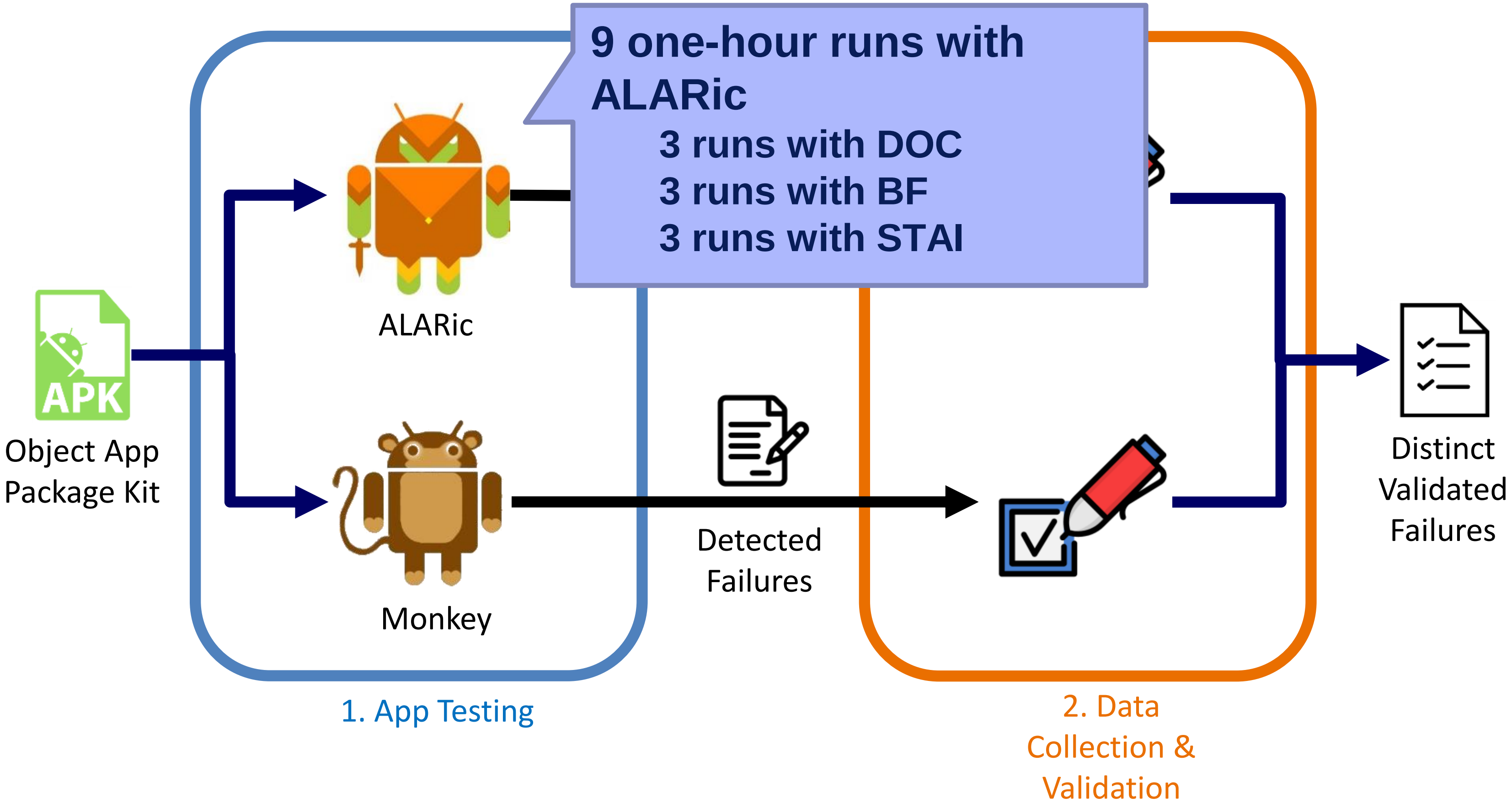
Metrics

- To evaluate the effectiveness of ALARic in detecting GUI failures:
  - **#DGFDOC** number of distinct GUI Failures triggered by DOC
  - **#DGFBF** number of distinct GUI Failures triggered by BF
  - **#DGFSTAI** number of distinct GUI Failures triggered by STAI
  - **#DGFTOTAL** number of distinct GUI Failures triggered by the DOC, BF, STAI
- To evaluate the effectiveness of both the tools in finding Crashes:
  - **#DCDOC** number of distinct crashes triggered by DOC
  - **#DCBF** number of distinct crashes triggered by BF
  - **#DCSTAI** number of distinct crashes triggered by STAI
  - **#DCTOTAL** number of distinct crashes triggered by the DOC, BF, STAI

Experimental Procedure

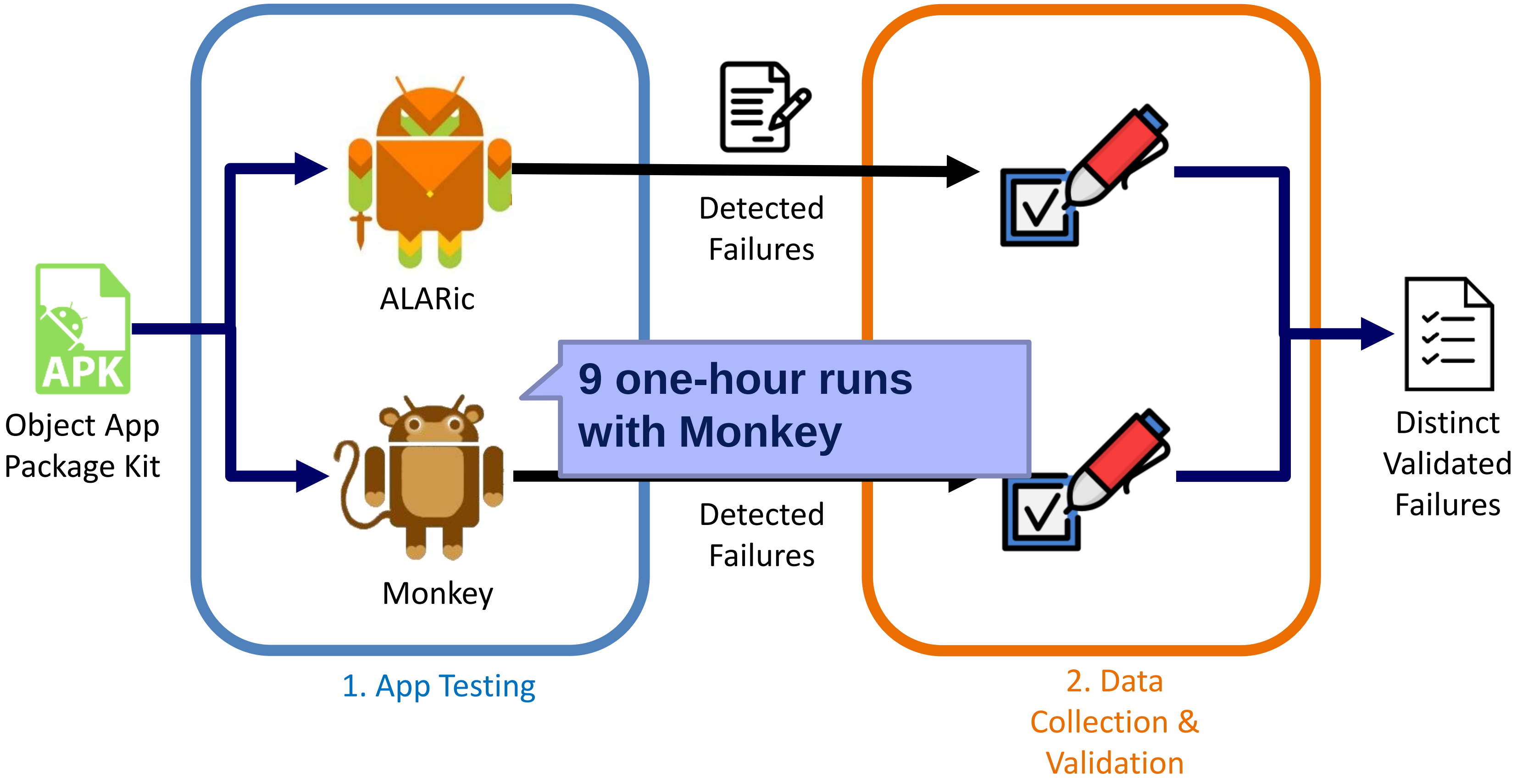


Experimental Procedure



9 one-hour runs with ALARic
- **3 runs with DOC**
- **3 runs with BF**
- **3 runs with STAI**

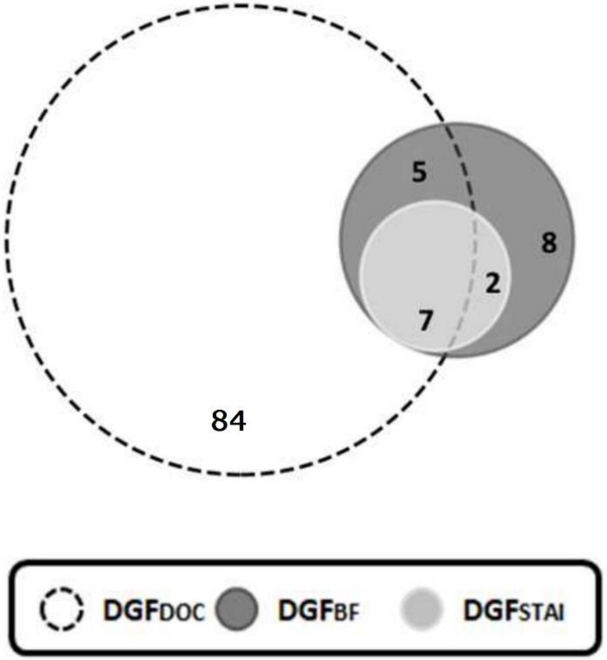
Experimental Procedure

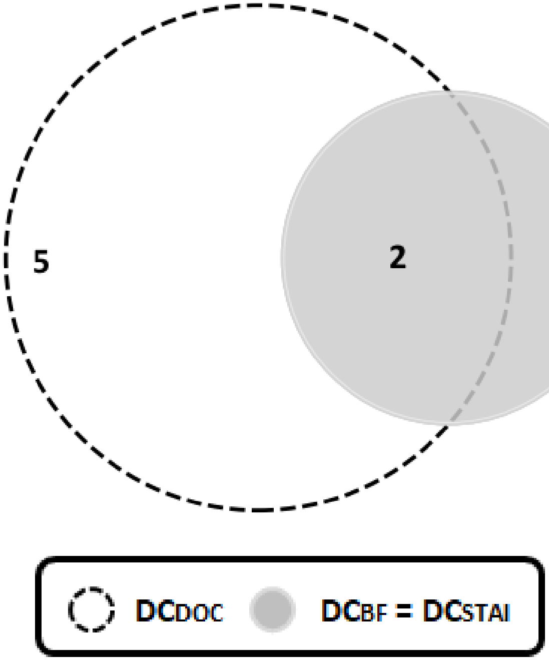


9 one-hour runs with Monkey

Experimental Results: RQ1

- ALARic detected 106 distinct GUI failures and 8 crashes tied to the Activity lifecycle in the 15 analyzed apps





Experimental Results: RQ2

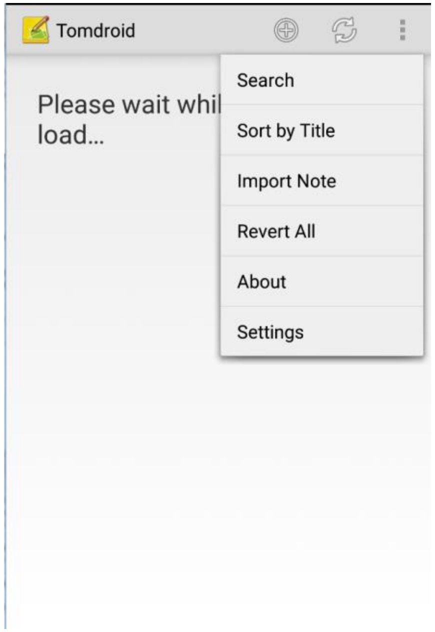
- ALARic outperformed Monkey in the ability to detect issues tied to the Activity lifecycle
  - In total ALARic triggered more crashes than Monkey
  - Monkey seeds events that exercise the Activity lifecycle, e.g. orientation changes, back button press, but it applies them without a proper strategy

App	#DCALARic	#DCMonkey
A4	1	1
A6	1	0
A7	1	0
A9	2	0
A11	1	0
A15	2	1
Total	**8**	**2**

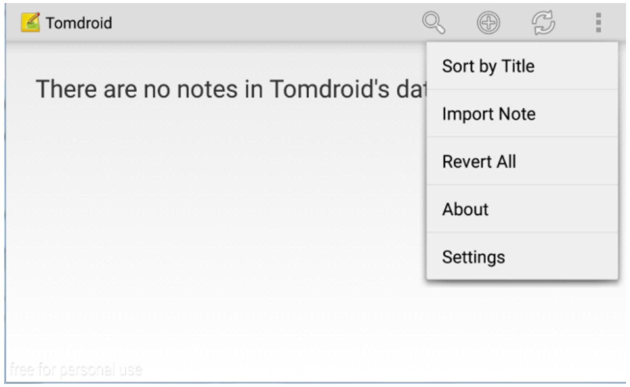
Lesson Learned

- The debugging activity we performed in the failure validation step showed us that the faults causing the failures were mostly located outside the code that overrides the lifecycle callback methods
  - Testers should look for faults that may affect the lifecycle of the Activities also outside the methods that override the lifecycle callbacks
  - Developers should correctly use the Android framework components since they may cause inconsistencies in the app behavior at runtime when Lifecycle Event Sequences occur

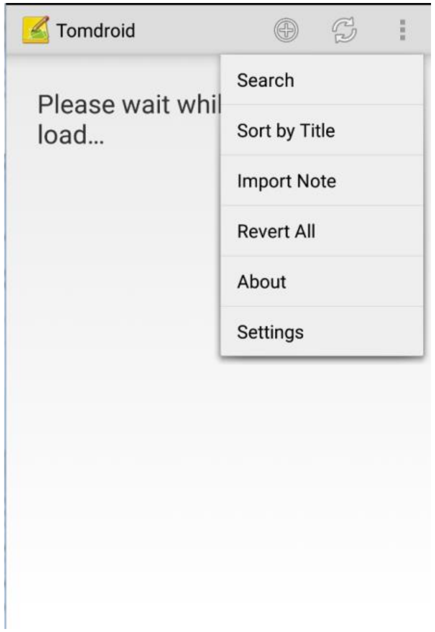
Lifecycle Event Sequences: DOC



Please wait whil
load...

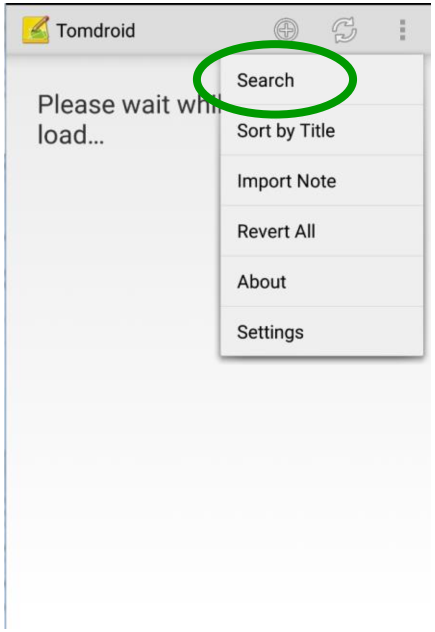


There are no notes in Tomdroid's dat

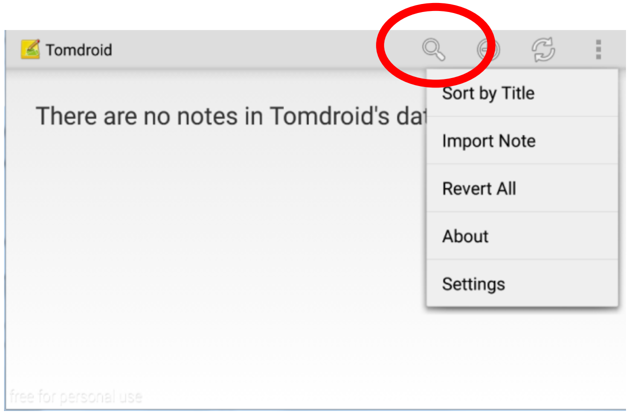


Please wait whil
load...

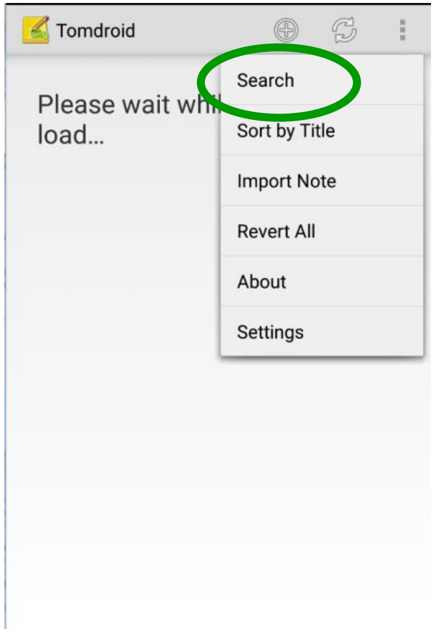
Lifecycle Event Sequences: DOC



Please wait whi
load...

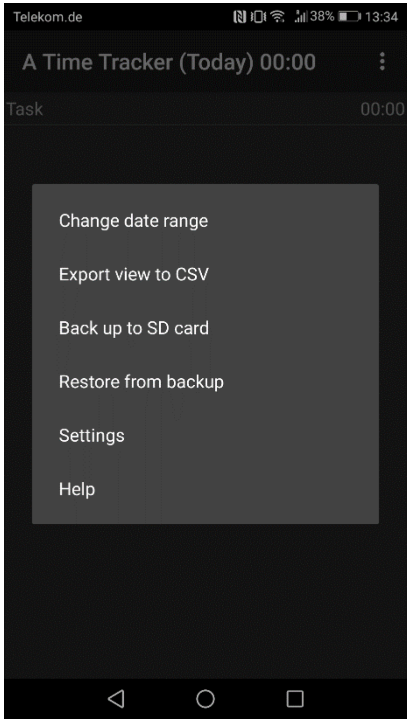


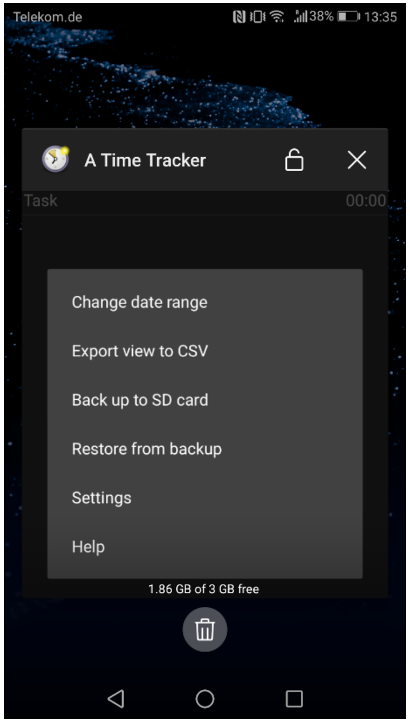
There are no notes in Tomdroid's dat

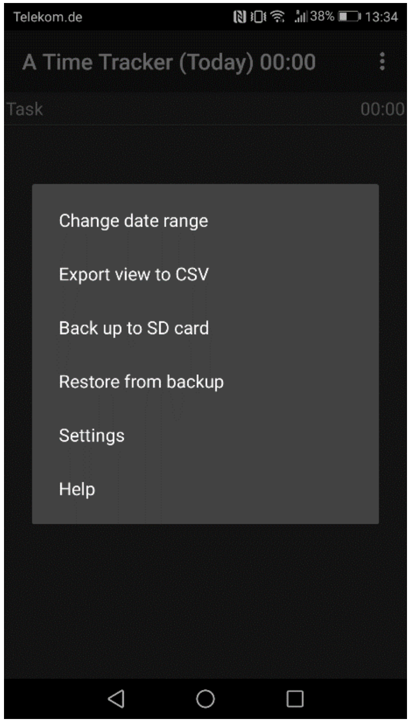


Please wait whi
load...

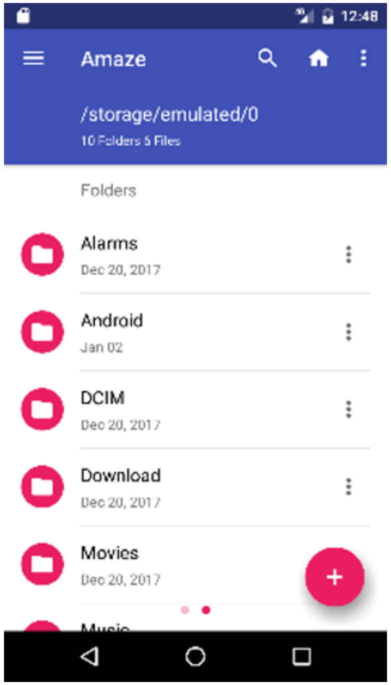
Lifecycle Event Sequences: BF

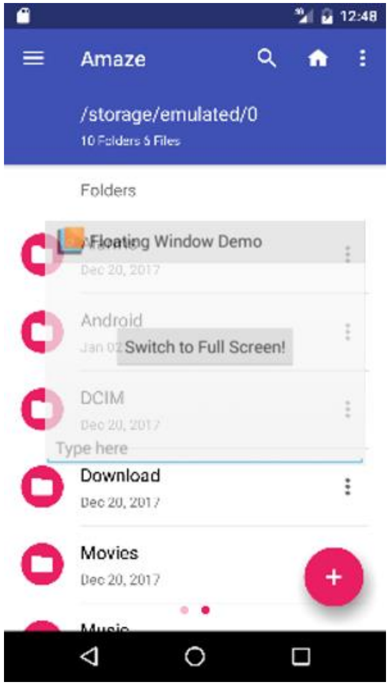


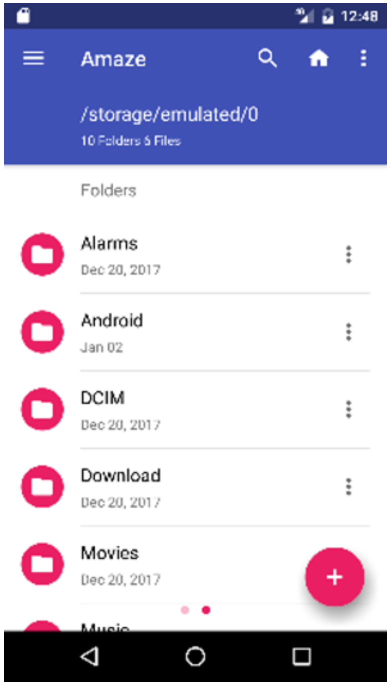




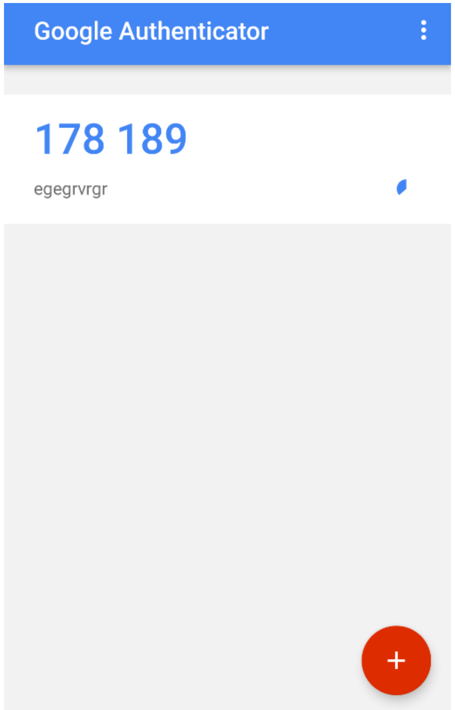
Lifecycle Event Sequences: STAI







False Positive Example #1

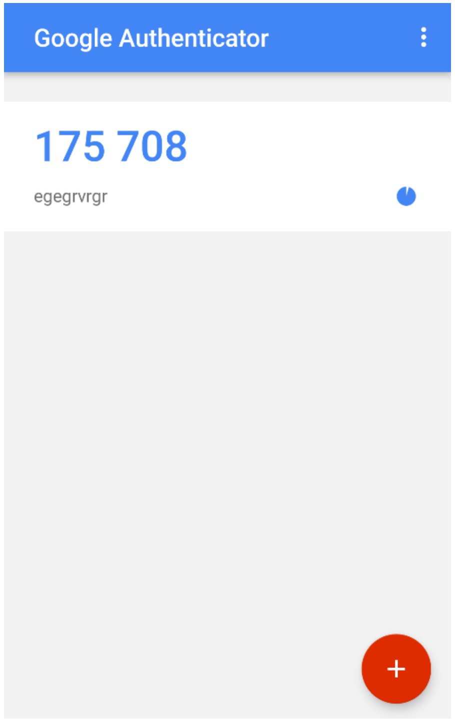


Google Authenticator

178 189

egegrvrgr

DOC
BF
STAI

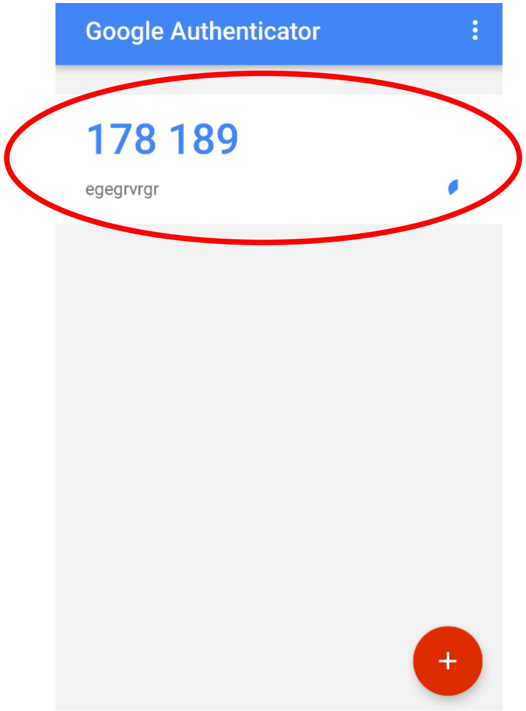


Google Authenticator

175 708

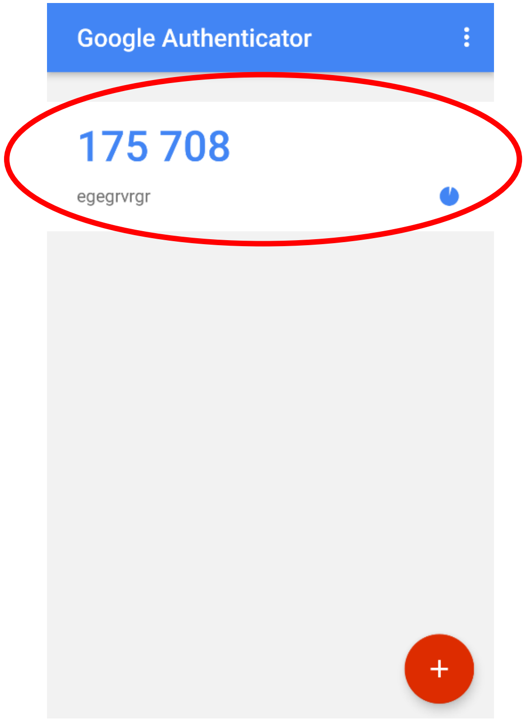
egegrvrgr

False Positive Example #1


Google Authenticator

178 189

egegrvgr

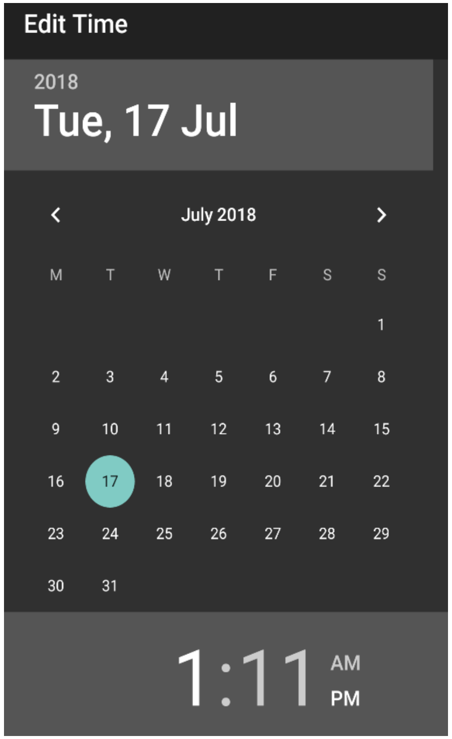

Google Authenticator

175 708

egegrvgr

DOC
BF
STAI

False Positive Example #2



Edit Time

2018

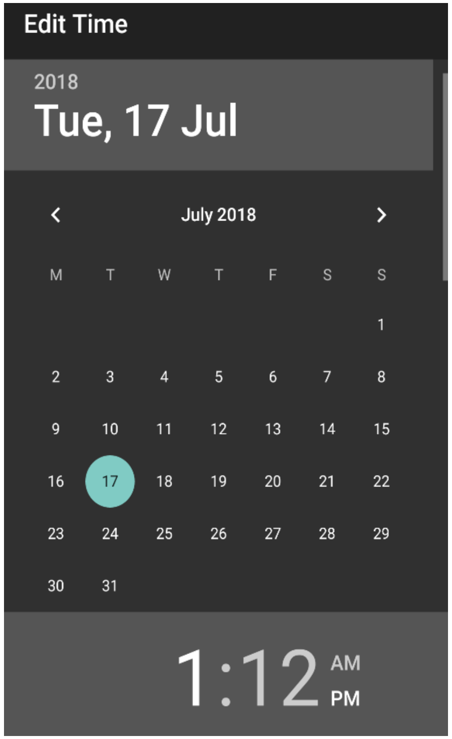
Tue, 17 Jul

July 2018

M	T	W	T	F	S	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

1:11 AM PM

BF



Edit Time

2018

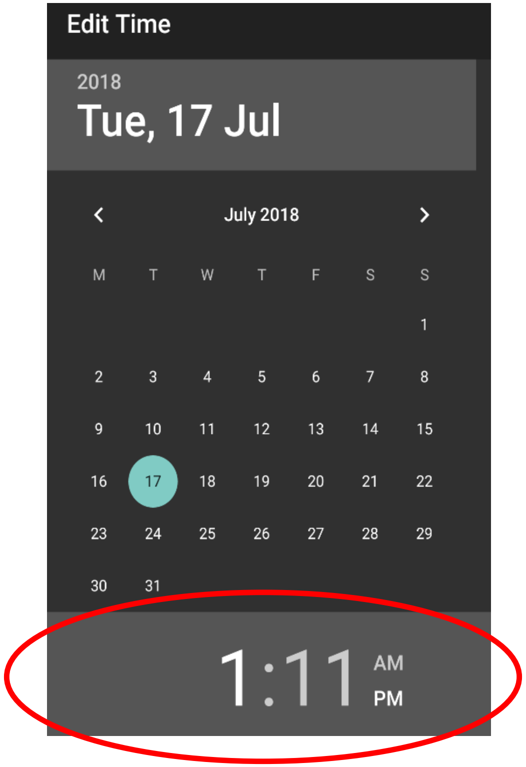
Tue, 17 Jul

July 2018

M	T	W	T	F	S	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

1:12 AM PM

False Positive Example #2



Edit Time

2018

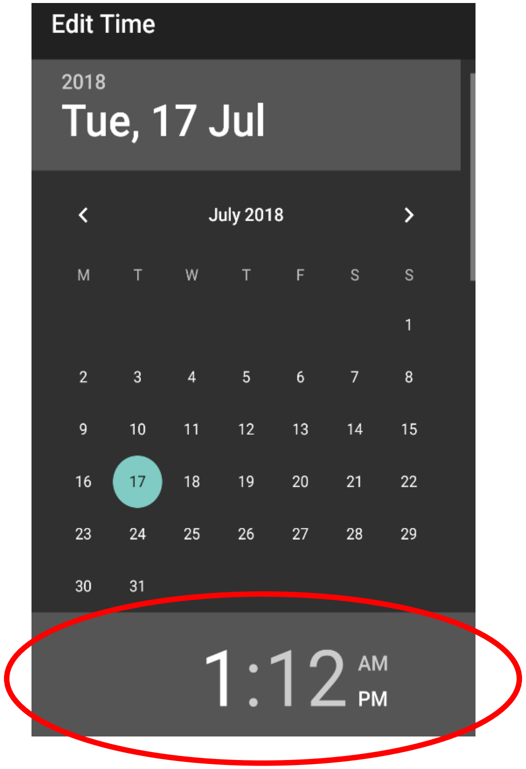
Tue, 17 Jul

July 2018

M	T	W	T	F	S	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

1:11 AM PM

BF



Edit Time

2018

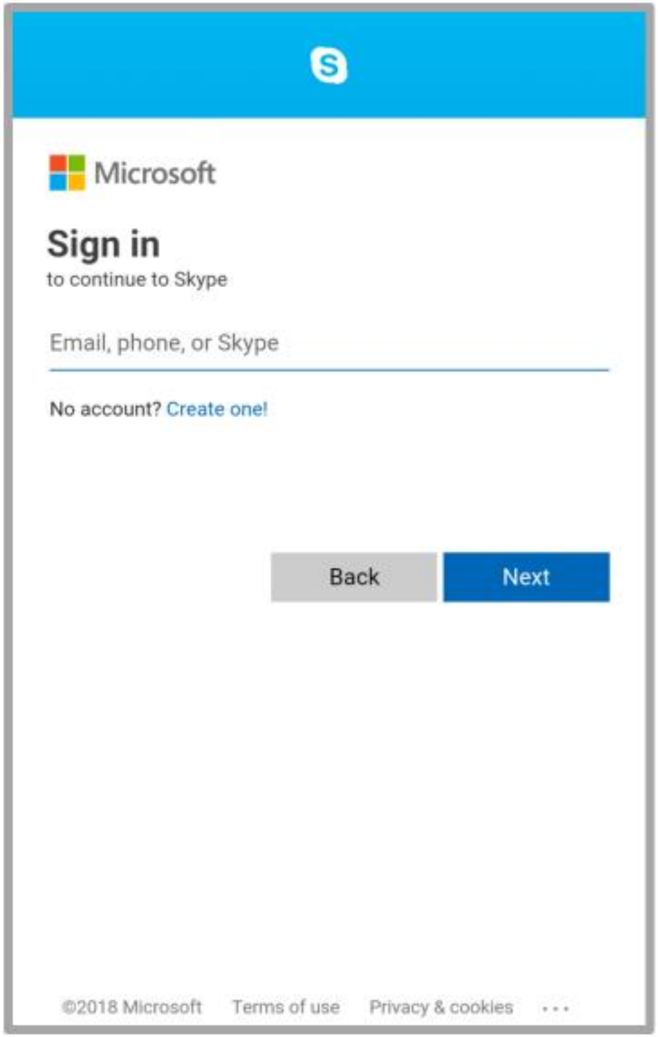
Tue, 17 Jul

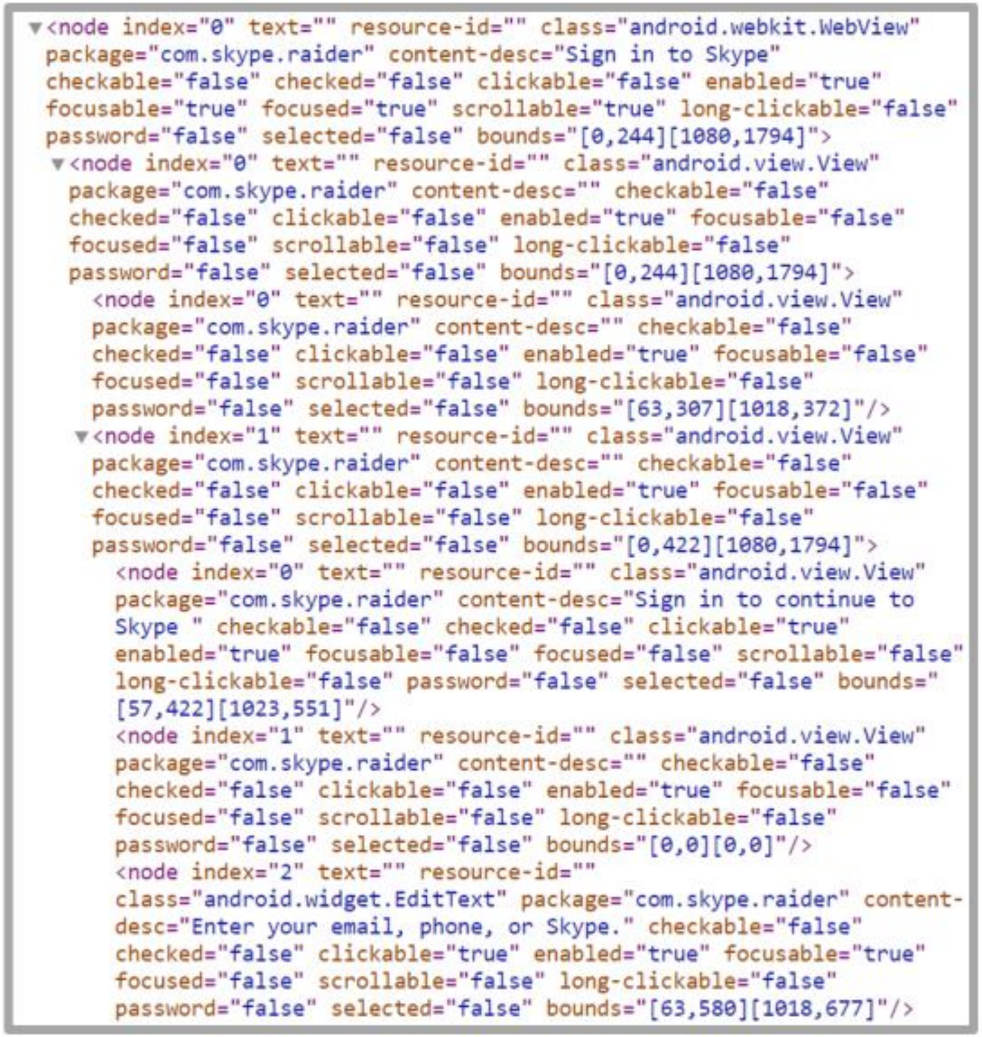
July 2018

M	T	W	T	F	S	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

1:12 AM PM

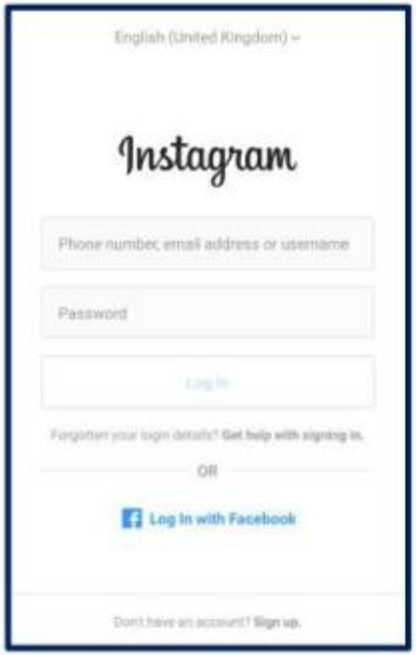
GUI XML Description





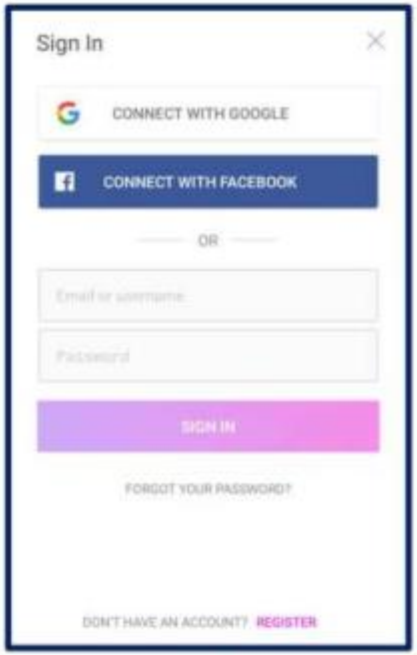
```
▼<node index="0" text="" resource-id="" class="android.webkit.WebView"
 package="com.skype.raider" content-desc="Sign in to Skype"
 checkable="false" checked="false" clickable="false" enabled="true"
 focusable="true" focused="true" scrollable="true" long-clickable="false"
 password="false" selected="false" bounds="[0,244][1080,1794]">
  ▼<node index="0" text="" resource-id="" class="android.view.View"
   package="com.skype.raider" content-desc="" checkable="false"
   checked="false" clickable="false" enabled="true" focusable="false"
   focused="false" scrollable="false" long-clickable="false"
   password="false" selected="false" bounds="[0,244][1080,1794]">
     <node index="0" text="" resource-id="" class="android.view.View"
     package="com.skype.raider" content-desc="" checkable="false"
     checked="false" clickable="false" enabled="true" focusable="false"
     focused="false" scrollable="false" long-clickable="false"
     password="false" selected="false" bounds="[63,307][1018,372]"/>
    ▼<node index="1" text="" resource-id="" class="android.view.View"
     package="com.skype.raider" content-desc="" checkable="false"
     checked="false" clickable="false" enabled="true" focusable="false"
     focused="false" scrollable="false" long-clickable="false"
     password="false" selected="false" bounds="[0,422][1080,1794]">
       <node index="0" text="" resource-id="" class="android.view.View"
       package="com.skype.raider" content-desc="Sign in to continue to
       Skype " checkable="false" checked="false" clickable="true"
       enabled="true" focusable="false" focused="false" scrollable="false"
       long-clickable="false" password="false" selected="false" bounds="
       [57,422][1023,551]"/>
       <node index="1" text="" resource-id="" class="android.view.View"
       package="com.skype.raider" content-desc="" checkable="false"
       checked="false" clickable="false" enabled="true" focusable="false"
       focused="false" scrollable="false" long-clickable="false"
       password="false" selected="false" bounds="[0,0][0,0]"/>
       <node index="2" text="" resource-id=""
       class="android.widget.EditText" package="com.skype.raider" content-
       desc="Enter your email, phone, or Skype." checkable="false"
       checked="false" clickable="true" enabled="true" focusable="true"
       focused="false" scrollable="false" long-clickable="false"
       password="false" selected="false" bounds="[63,580][1018,677]"/>
```

GUI Textual Information Content

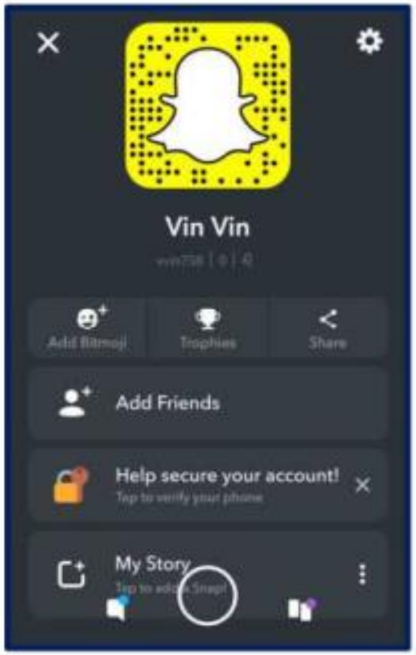


Forgotten your login details? Get help with signing in.

Don't have an account? Sign up.



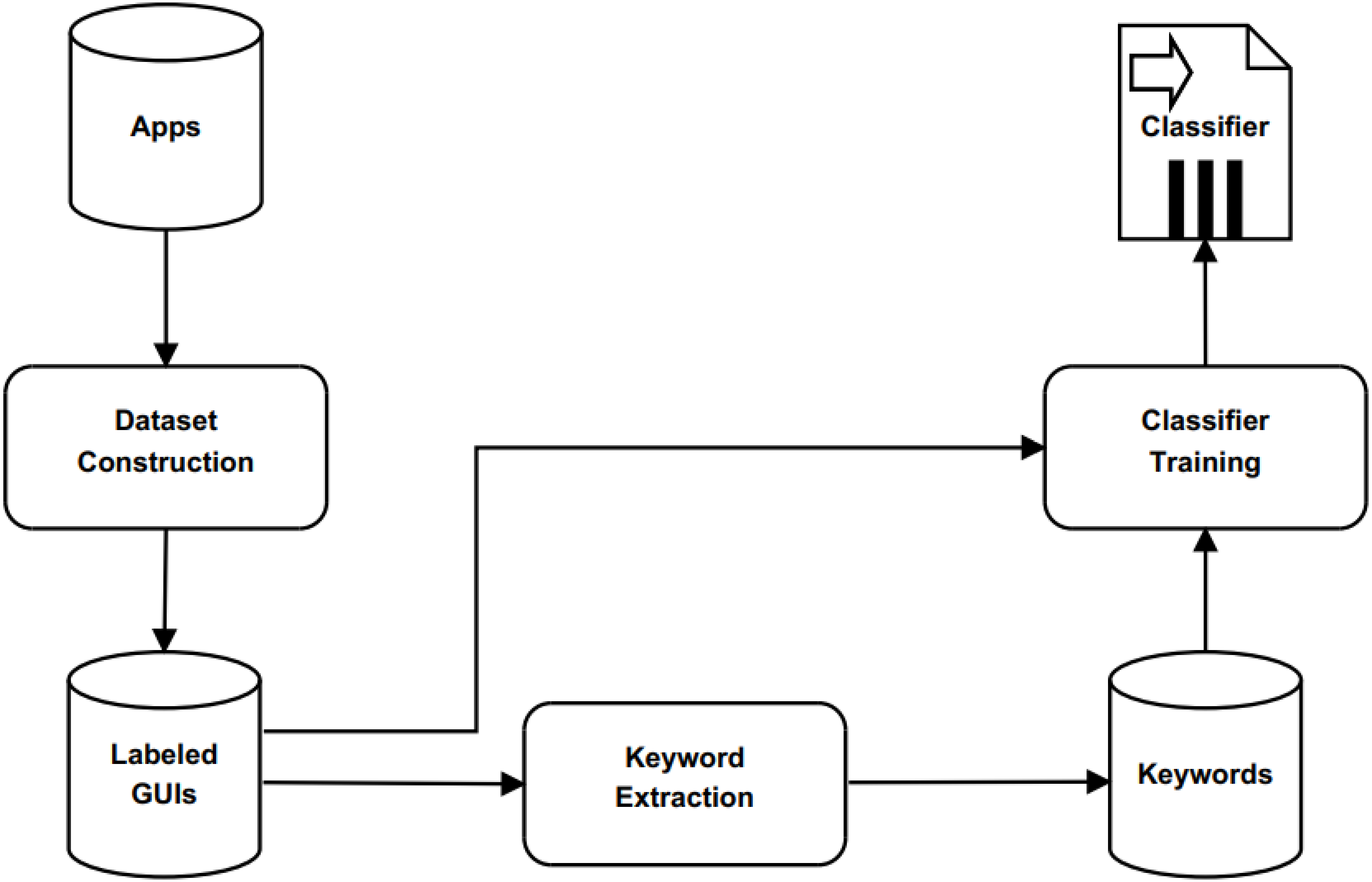
DON'T HAVE AN ACCOUNT? REGISTER



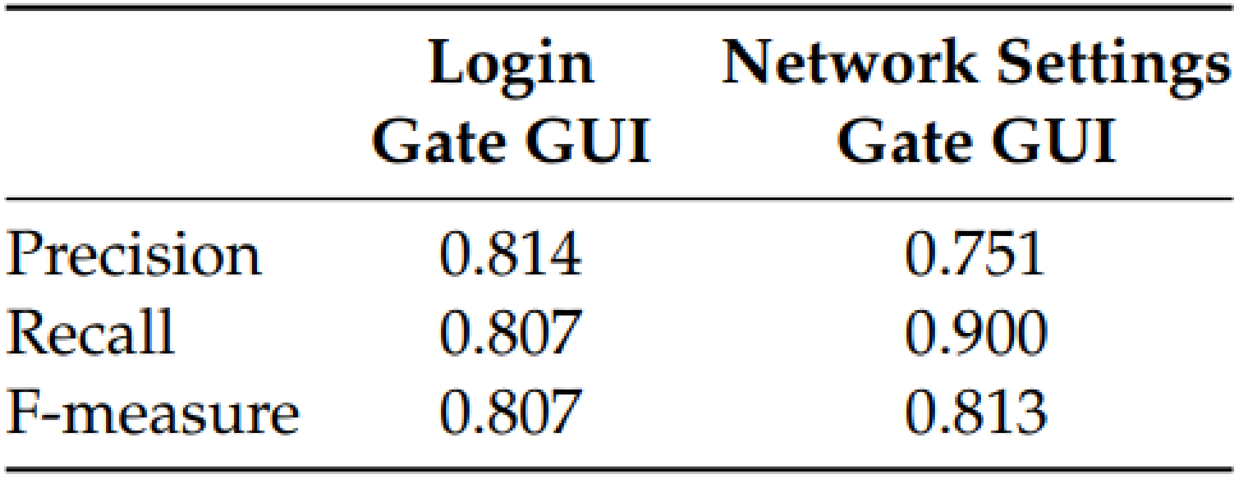
Help secure your account!
Tap to verify your phone

Keyword	(a)	(b)	(c)
forgot	✓	✓	✗
login	✓	✗	✗
password	✓	✓	✗
facebook	✓	✓	✗

ML-based Classifier Training Process

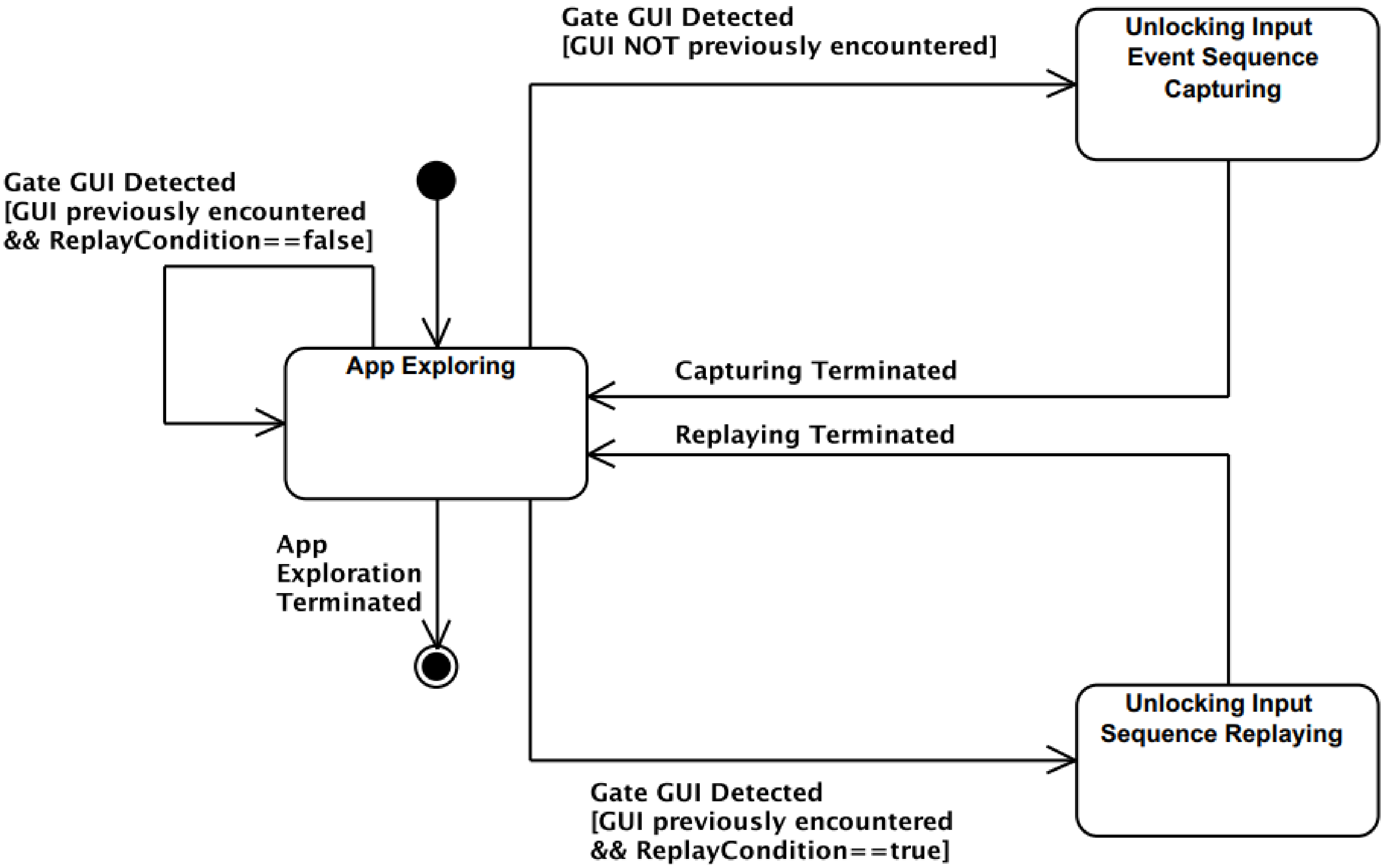


Gate GUI Classifiers’ Performance

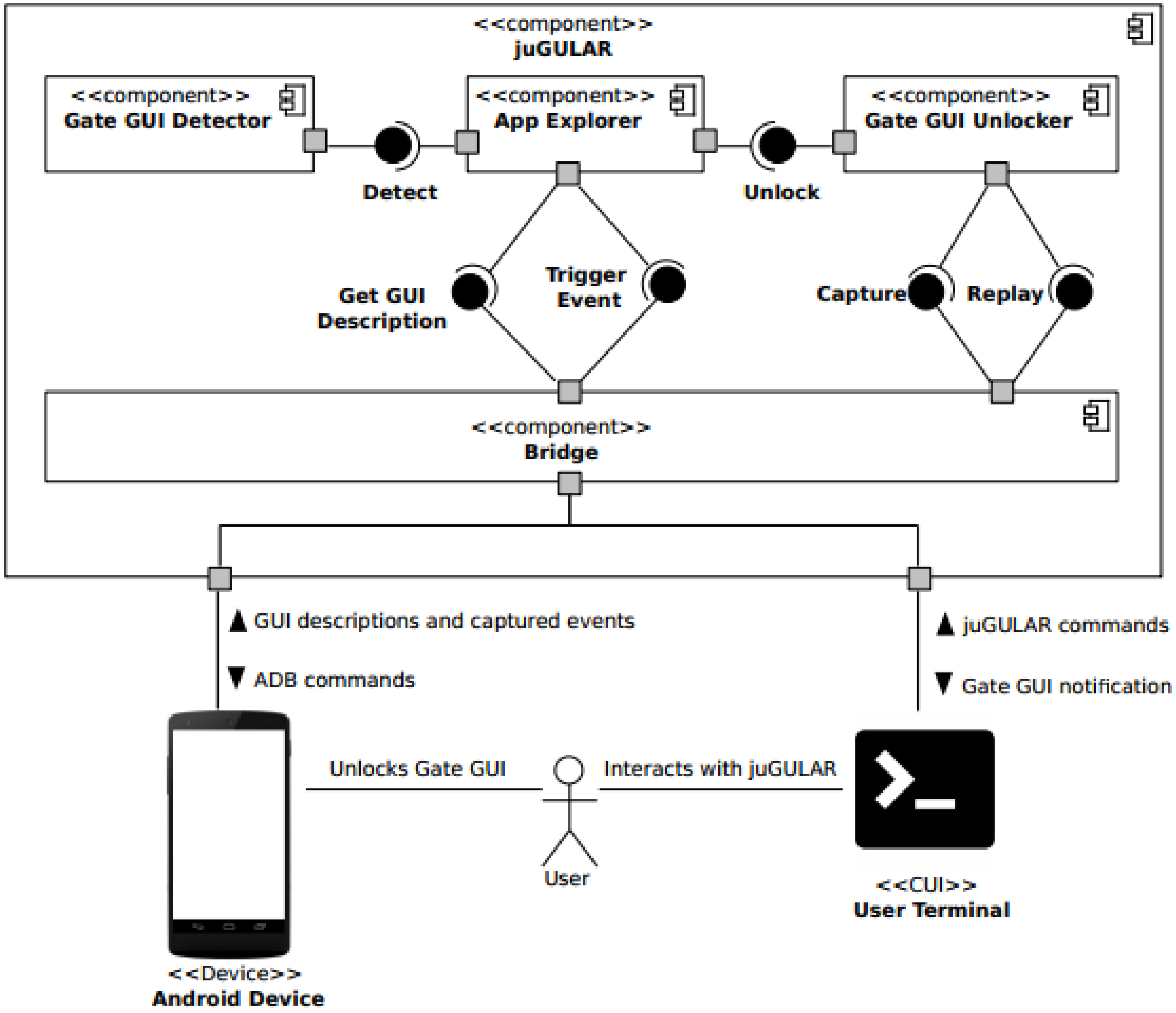


	Login Gate GUI	Network Settings Gate GUI
Precision	0.814	0.751
Recall	0.807	0.900
F-measure	0.807	0.813

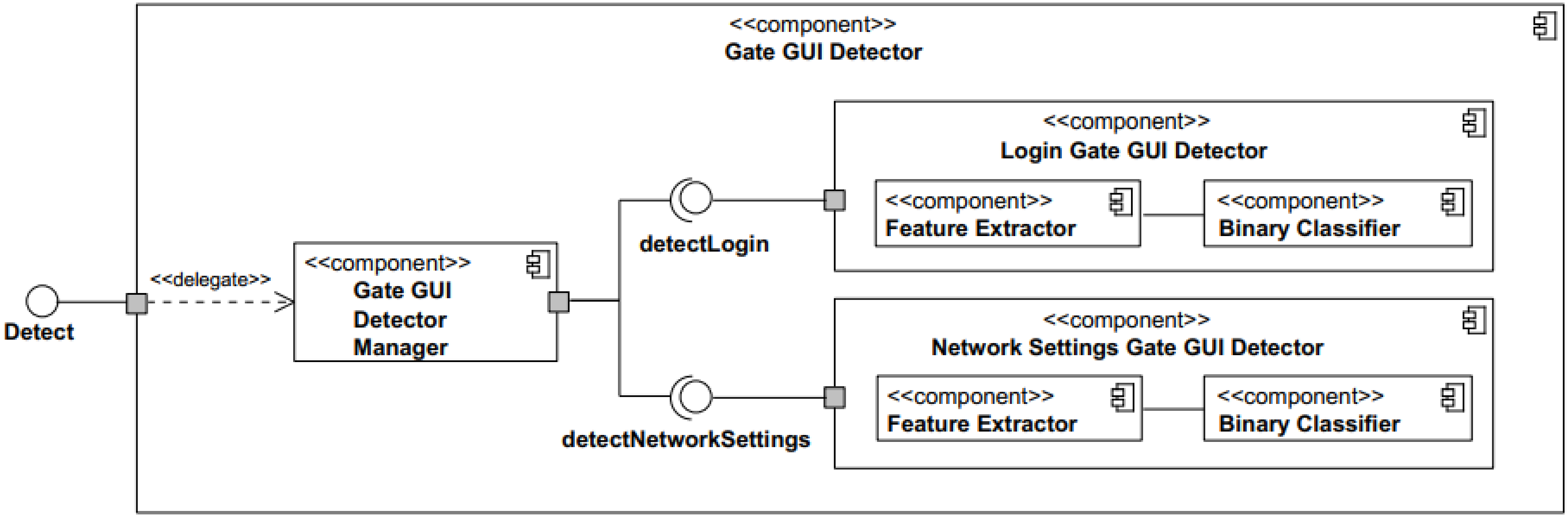
Combining AGET and C&R


Gate GUI Detected
[GUI NOT previously encountered]
Unlocking Input Event Sequence Capturing
Gate GUI Detected
[GUI previously encountered
&& ReplayCondition==false]
App Exploring
Capturing Terminated
Replaying Terminated
App Exploration Terminated
Unlocking Input Sequence Replaying
Gate GUI Detected
[GUI previously encountered
&& ReplayCondition==true]

The juGULAR Platform



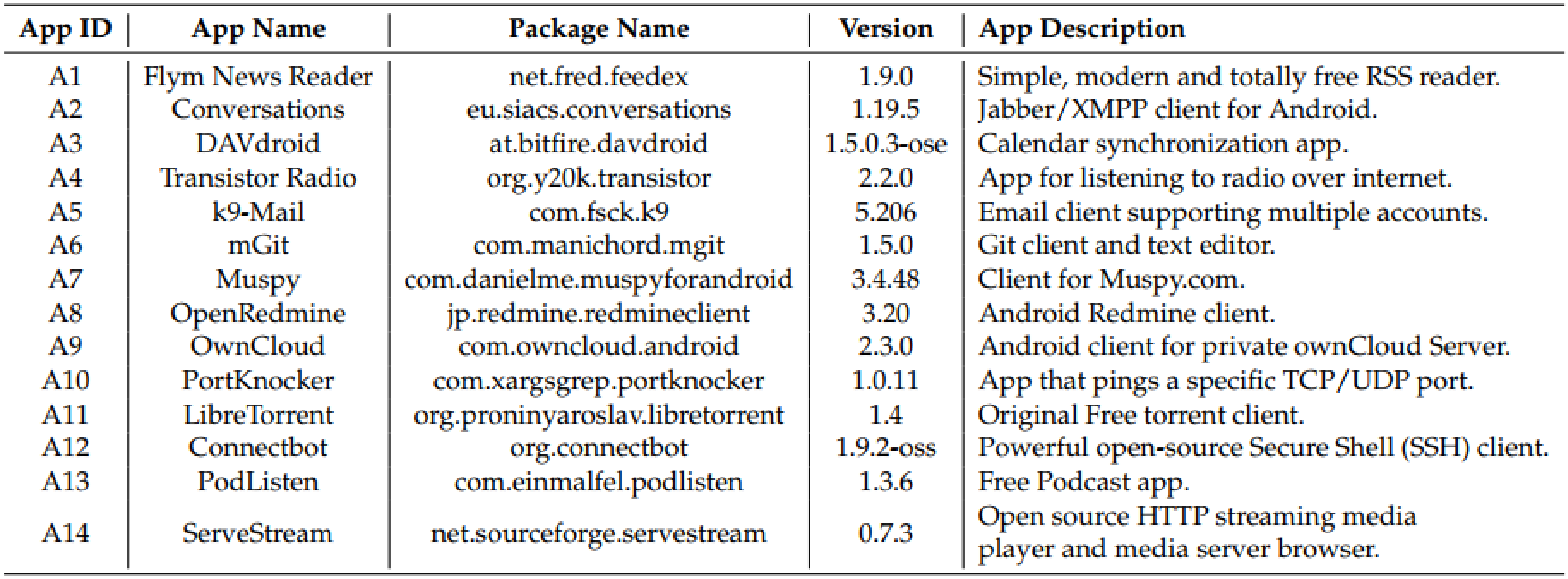
Gate GUI Detector



Experimental Evaluation

- **GOAL:** Understand how the hybridization proposed by juGULAR does impact the ability of fully automated GUI exploration techniques in analyzing apps and at what cost.
  - **RQ1:** How does the hybridization introduced by juGULAR affect the effectiveness of an automated exploration technique?
  - **RQ2:** How does the manual intervention required by juGULAR affect the costs of the hybrid exploration approach?
  - **RQ3:** How does the exploration effectiveness of juGULAR compare to the effectiveness of the AGET implemented by the state-of-the-practice Monkey tool?

Objects



App ID	App Name	Package Name	Version	App Description
A1	Flym News Reader	net.fred.feedex	1.9.0	Simple, modern and totally free RSS reader.
A2	Conversations	eu.siacs.conversations	1.19.5	Jabber/XMPP client for Android.
A3	DAVdroid	at.bitfire.davdroid	1.5.0.3-ose	Calendar synchronization app.
A4	Transistor Radio	org.y20k.transistor	2.2.0	App for listening to radio over internet.
A5	k9-Mail	com.fsck.k9	5.206	Email client supporting multiple accounts.
A6	mGit	com.manichord.mgit	1.5.0	Git client and text editor.
A7	Muspy	com.danielme.muspyforandroid	3.4.48	Client for Muspy.com.
A8	OpenRedmine	jp.redmine.redmineclient	3.20	Android Redmine client.
A9	OwnCloud	com.owncloud.android	2.3.0	Android client for private ownCloud Server.
A10	PortKnocker	com.xargsgrep.portknocker	1.0.11	App that pings a specific TCP/UDP port.
A11	LibreTorrent	org.proninyaroslav.libretorrent	1.4	Original Free torrent client.
A12	Connectbot	org.connectbot	1.9.2-oss	Powerful open-source Secure Shell (SSH) client.
A13	PodListen	com.einmalfel.podlisten	1.3.6	Free Podcast app.
A14	ServeStream	net.sourceforge.servestream	0.7.3	Open source HTTP streaming media player and media server browser.

Metrics

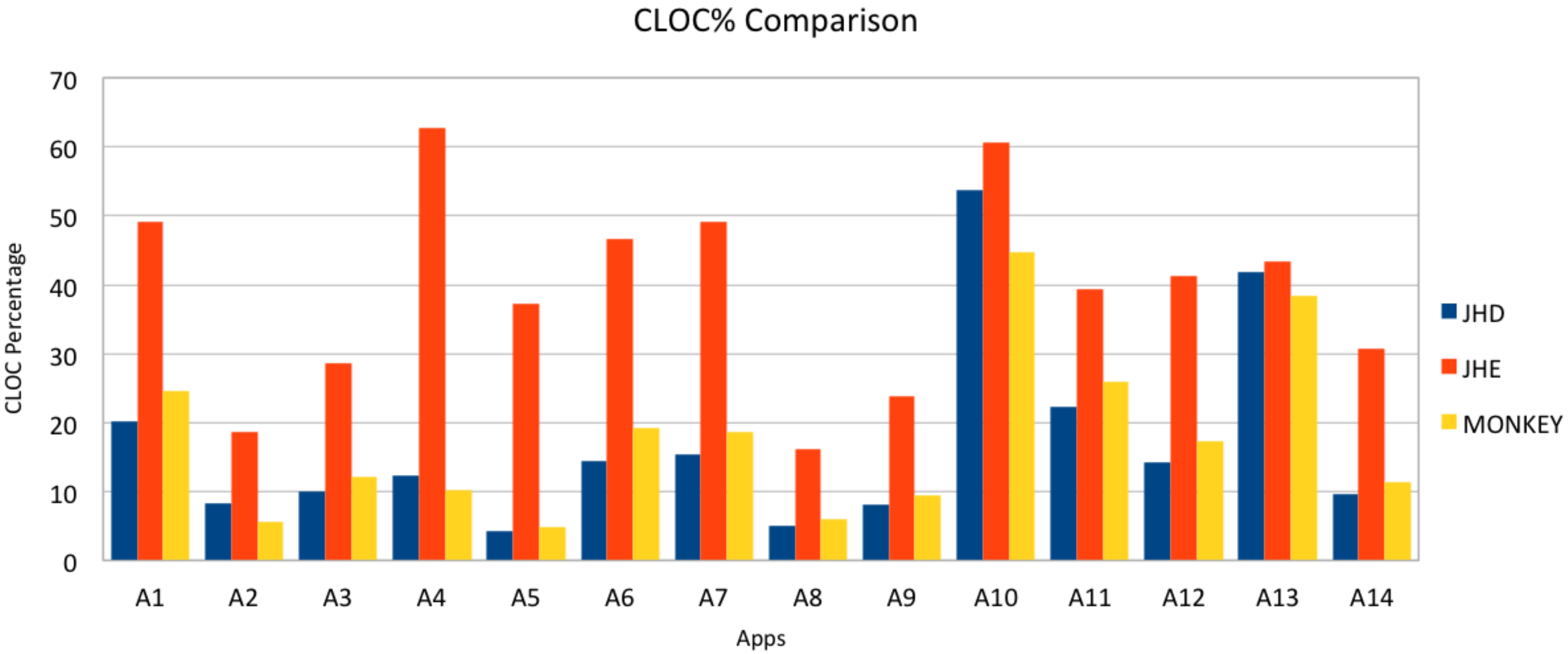
$$CA\% = \frac{\#\,Covered\ Activities}{\#\,App\ Activities} * 100$$

$$CLOC\% = \frac{\#\,Covered\ LOCs}{\#\,App\ LOCs} * 100$$

$$NTB = \#\,App\ Sent\ Bytes\ + \ \#\,App\ Received\ Bytes$$

$$MIT\% = \frac{\sum_i CaptureTime_i}{TotalExplorationTime} * 100$$

Covered Lines Of Code


CLOC% Comparison
CLOC Percentage
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
A13
A14
Apps
JHD
JHE
MONKEY