

# **EXPLORING THE INTEGRATION OF USER FEEDBACK IN AUTOMATED TESTING OF ANDROID APPLICATIONS**

**G. GRANO, A. CIURUMELEA, S. PANICHELLA, F. PALOMBA, H.  
GALL**

**SANER 2018, 20-23 MARCH, CAMPOBASSO (ITALY)**

**149 BILLIONS OF APPS**  
**12 MILLIONS OF DEVS**  
**60 BILLIONS**



**COMPETITION  
SATISFACTION  
QUALITY**

# TESTING TOOLS

## PLETHORA OF ANDROID TESTING TOOLS:

- > **MONKEY**: STATE OF THE PRACTICE
- > **SAPIENZ**: NOW IN FACEBOOK
- > **DYNODROID**
- > ...
- > **AND A LOT OF OTHERS!**



# LIMITATIONS

**THEY ARE NOT SUITED FOR  
GENERATING INPUTS THAT  
REQUIRE HUMAN  
INTELLIGENCE**

**REDUNDANCY OF  
GENERATED INPUT  
SEQUENCES**



# **TOOLS BEHAVIOR**

**1. STACK TRACE**

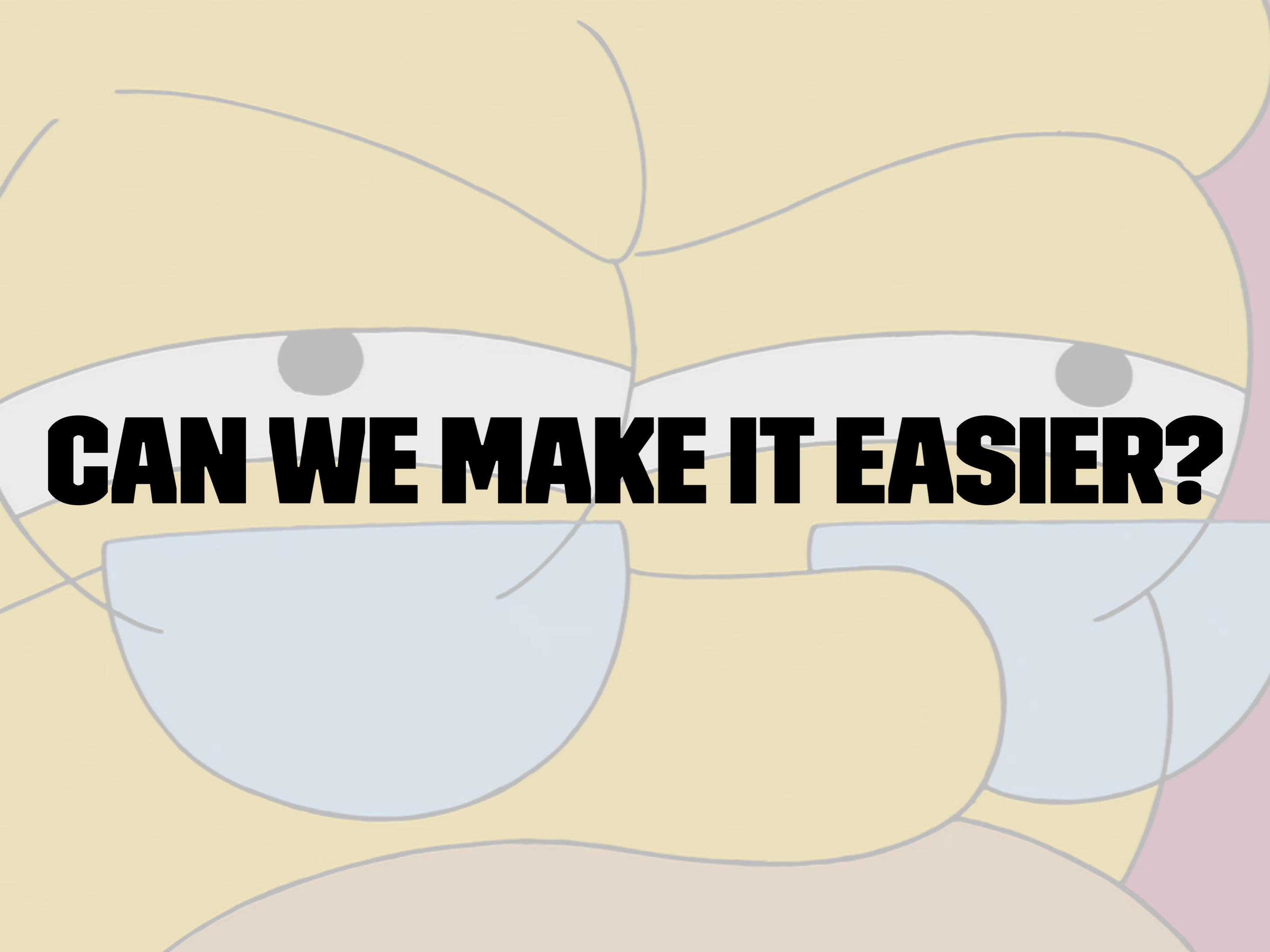
**2. SEQUENCE OF INPUTS**

# STACK TRACE

```
// CRASH: com.danvelazco.fbwrapper (pid 4302)
// Short Msg: java.lang.NullPointerException
// Long Msg: java.lang.NullPointerException
// Build Label: samsung/espressowifixx/espressowifi:4.2.2/JDQ39/P3110XXDMH1:user/release-keys
// Build Changelist: 8291
// Build Time: 1419156873000
// java.lang.NullPointerException
// at com.danvelazco.fbwrapper.activity.BaseFacebookWebViewActivity
//          .onKeyDown(BaseFacebookWebViewActivity.java:649)
// at com.danvelazco.fbwrapper.FbWrapper.onKeyDown(FbWrapper.java:429)
// at android.view.KeyEvent.dispatch(KeyEvent.java:2640)
// at android.app.Activity.dispatchKeyEvent(Activity.java:2433)
// at com.android.internal.policy.impl.PhoneWindow$DecorView.dispatchKeyEvent(PhoneWindow.java:2021)
// at android.view.ViewRootImpl$ViewPostImeInputStage.processKeyEvent(ViewRootImpl.java:3845)
// at android.view.ViewRootImpl$ViewPostImeInputStage.onProcess(ViewRootImpl.java:3819)
// at android.view.ViewRootImpl$InputStage.deliver(ViewRootImpl.java:3392)
// at android.view.ViewRootImpl$InputStage.onDeliverToNext(ViewRootImpl.java:3442)
// at android.view.ViewRootImpl$InputStage.forward(ViewRootImpl.java:3411)
// at android.view.ViewRootImpl$AsyncInputStage.forward(ViewRootImpl.java:3518)
```

# SEQUENCE OF INPUTS

```
type= raw events
count= -1
speed= 1.0
start data >>
LaunchActivity(com.ringdroid,com.ringdroid.RingdroidSelectActivity)
DispatchKey(223989,223989,0,23,0,0,-1,0)
DispatchKey(224204,224204,1,23,0,0,-1,0)
DispatchPointer(224346,224347,0,479.0,774.0,0.0,0.0,0.0,1.0,1.0,0,0)
DispatchPointer(224346,224351,2,479.60635,797.5855,0.0,0.0,0.0,1.0,1.0,0,0)
DispatchPointer(224346,224353,2,482.31937,814.9475,0.0,0.0,0.0,1.0,1.0,0,0)
DispatchPointer(224346,224357,2,483.44247,829.02045,0.0,0.0,0.0,1.0,1.0,0,0)
DispatchPointer(224346,224359,2,486.9434,848.0035,0.0,0.0,0.0,1.0,1.0,0,0)
DispatchPointer(224346,224361,2,490.1806,859.495,0.0,0.0,0.0,1.0,1.0,0,0)
DispatchPointer(224346,224364,2,497.59595,872.6837,0.0,0.0,0.0,1.0,1.0,0,0)
DispatchPointer(224346,224367,2,500.53647,894.2986,0.0,0.0,0.0,1.0,1.0,0,0)
DispatchPointer(224346,224369,1,503.94815,896.686,0.0,0.0,0.0,1.0,1.0,0,0)
DispatchPointer(224374,224374,0,166.0,4.0,0.0,0.0,0.0,1.0,1.0,0,0)
```



**CAN WE MAKE IT EASIER?**

# HISTORY OF SUCCESS

- > RELEASE PLANNING <sup>1 2</sup>
- > CHANGE LOCALIZATION <sup>3 2</sup>
- > USER FEEDBACK CATEGORIZATION <sup>4</sup>

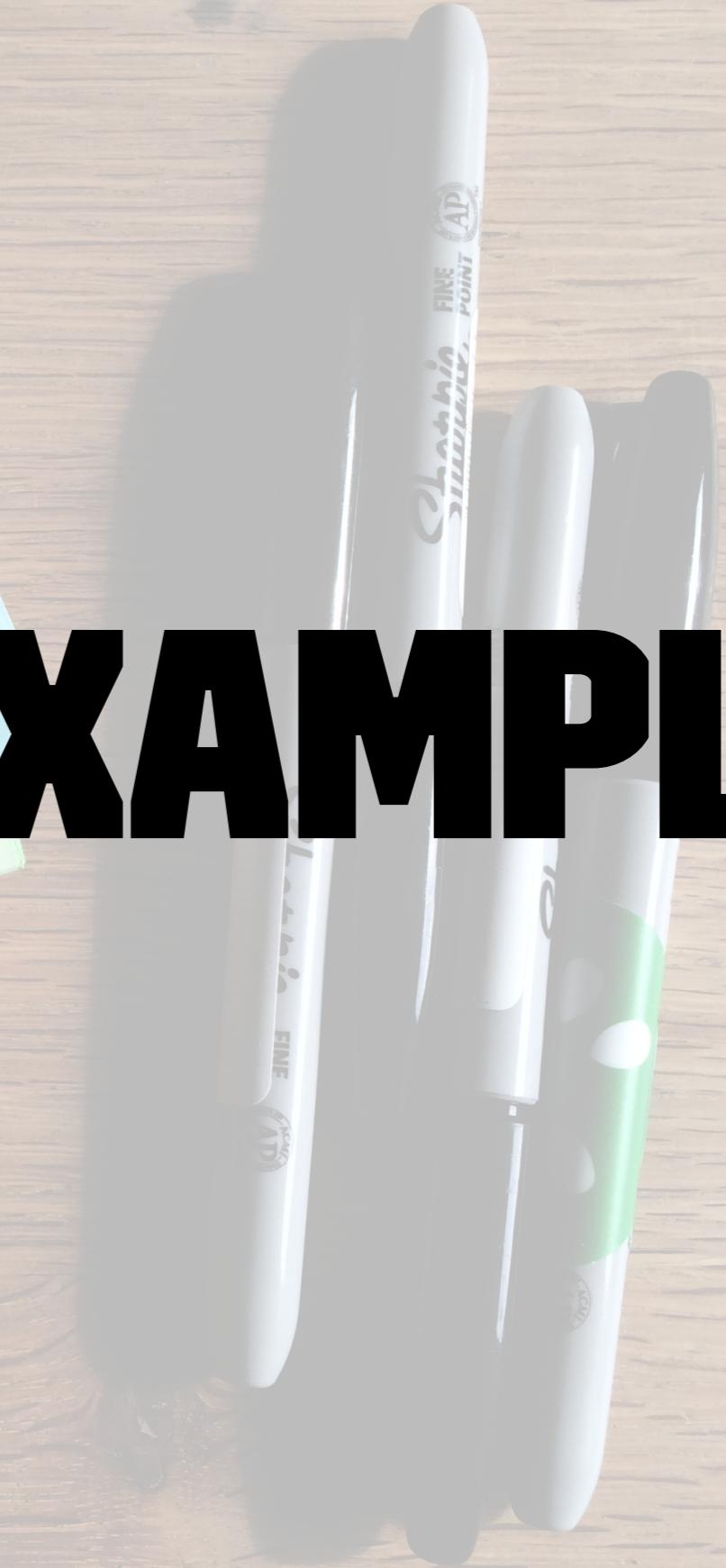
<sup>1</sup> VILLAROEL ET AL - RELEASE PLANNING OF MOBILE APPS BASED ON USER REVIEWS

<sup>3</sup> PALOMBA ET AL - RECOMMENDING AND LOCALIZING CHANGE REQUESTS FOR MOBILE APPS BASED ON USER REVIEWS

<sup>2</sup> CIURUMELEA ET AL - ANALYZING REVIEWS AND CODE OF MOBILE APPS FOR BETTER RELEASE PLANNING

<sup>4</sup> PANICHELLA ET AL - HOW CAN I IMPROVE MY APP? CLASSIFYING USER REVIEWS FOR SOFTWARE MAINTENANCE AND EVOLUTION

# CONCRETE EXAMPLE



# A STACK TRACE

```
Long Msg: java.lang.NumberFormatException: Invalid int: "/"
java.lang.RuntimeException: An error occurred while executing doInBackground()
    at android.os.AsyncTask$3.done(AsyncTask.java:300)
    at java.util.concurrent.FutureTask.finishCompletion(FutureTask.java:355)
    ...
    at com.amaze.filemanager.services.AsyncTask$2.call(AsyncTask.java:288)
    at java.util.concurrent.FutureTask.run(FutureTask.java:237)
    ... 3 more
```

# AN USER REVIEW

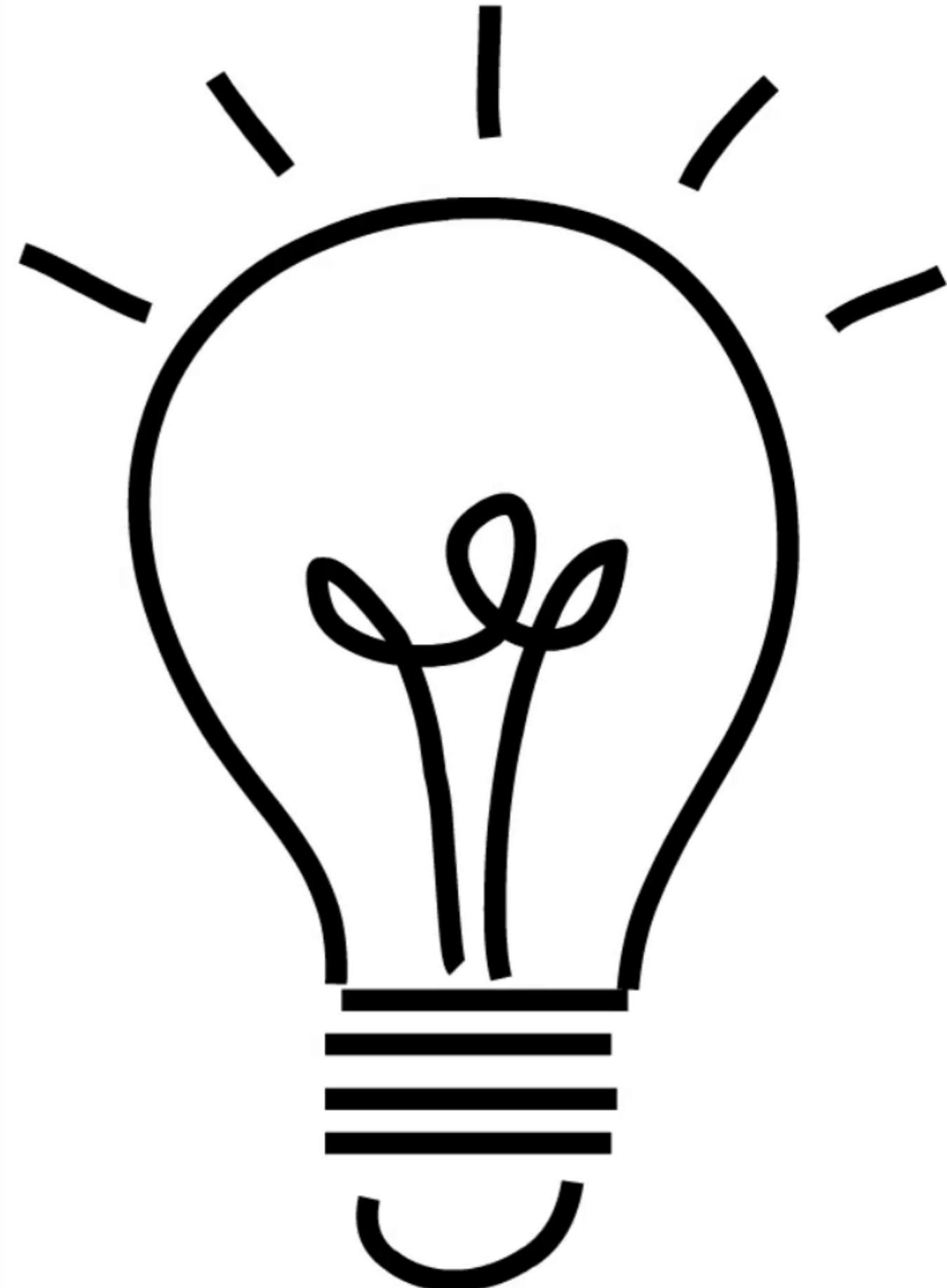
"Love the idea of this app but anytime I leave the page the screen goes completely white and won't come back until force-stopped.

Update: I thought the white screen was because my phone was so outdated but it still does it on my Nexus 6 ...."

# UNDERLINE IDEA

USER REVIEWS MIGHT BE HELPFUL FOR:

- › COMPREHENDING THE CAUSES BEHIND A FAILURE
- › EASING THE DEBUGGING PHASE
- › DISCOVERING ERRORS THAT TOOLS CANNOT REVEAL





# RESEARCH QUESTIONS

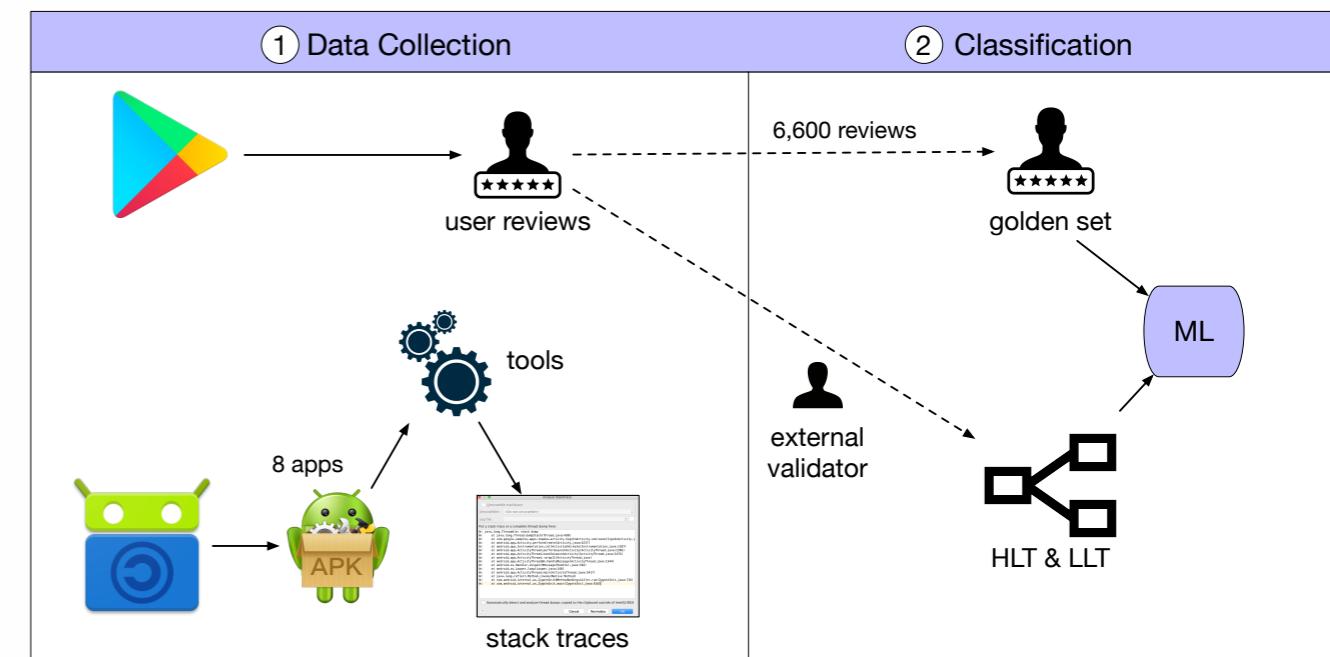


- > **RQ1: WHAT TYPE OF USER FEEDBACK CAN WE LEVERAGE TO DETECT BUGS AND SUPPORT TESTING ACTIVITIES OF MOBILE APPS?**
- > **RQ2: HOW COMPLEMENTARY IS USER FEEDBACK INFORMATION WITH RESPECT TO THE OUTCOMES OF AUTOMATED TESTING TOOLS?**
- > **RQ3: TO WHAT EXTENT CAN WE AUTOMATICALLY LINK THE CRASH-RELATED INFORMATION REPORTED IN BOTH USER FEEDBACK AND TESTING TOOLS?**

# RQ1: WHICH REVIEWS CAN WE USE?

## DATA COLLECTION

- > REVIEWS CRAWLER FOR GOOGLE PLAY STORE
- > MANUALLY VALIDATED FROM AN EXTERNAL VALIDATOR
- > RUN OUR APPS AGAINST MONKEY AND SAPIENZ



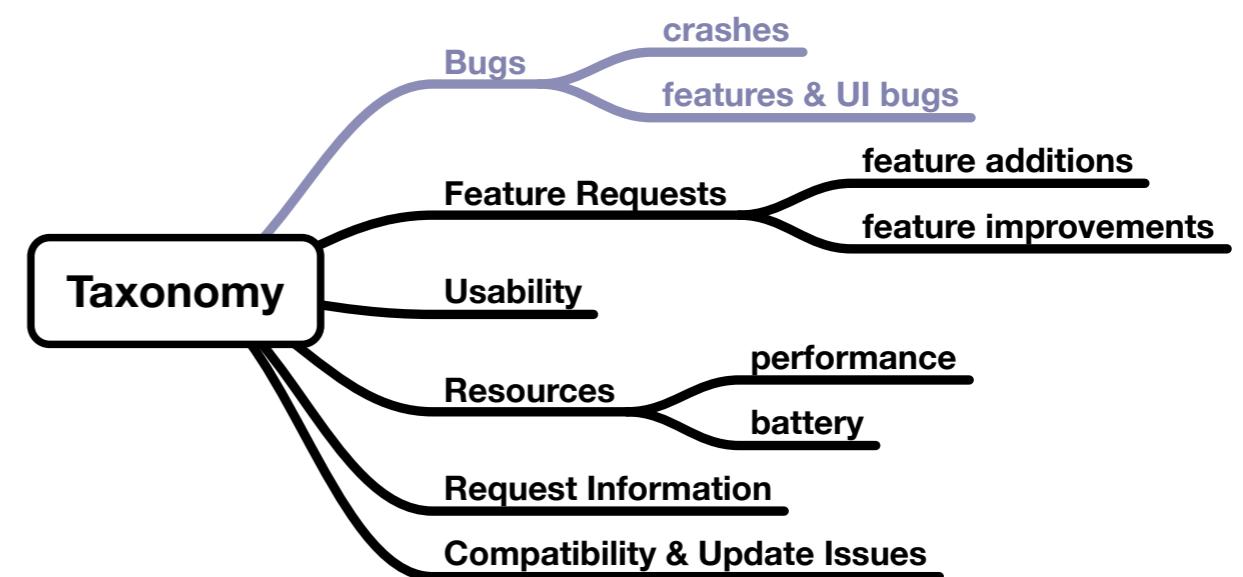
## OUTPUT

- > MACHINE LEARNING CLASSIFIER
- > TWO (HIGH AND LOW) LEVEL TAXONOMY

# RQ1: RESULTS

CATEGORY	PRECISION	RECALL	F1 SCORE
----------	-----------	--------	----------

CATEGORY	PRECISION	RECALL	F1 SCORE
FEATURES & UI BUGS	0.83	0.82	0.83
CRASHES	0.91	0.94	0.92



# RQ1: RESULTS

**WE ARE ABLE TO PREDICT  
WITH GOOD PRECISION  
REVIEWS CLAIMING  
ABOUT BUGS**

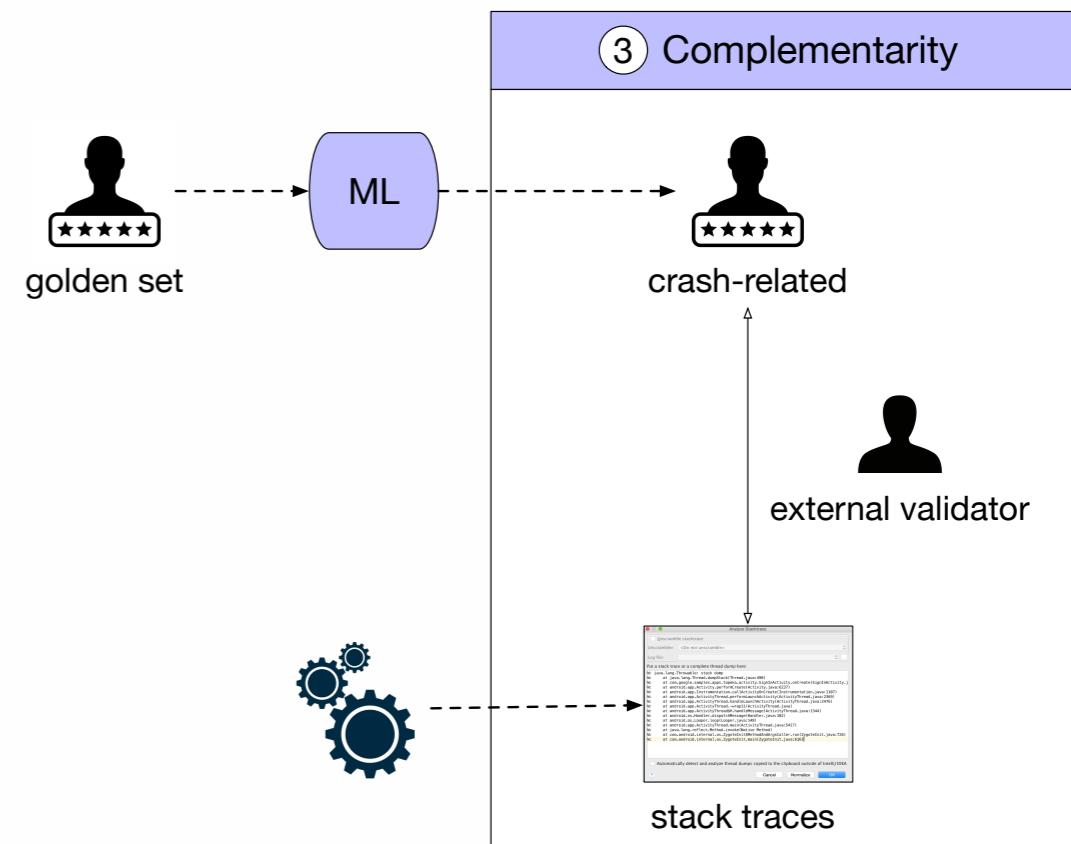


# RQ2: COMPLEMENTARITY

WE GAVE TO AN EXTERNAL INSPECTOR:

- > STACK TRACES
- > EVENT LOGS FOR CRASHES
- > CRASH-RELATED REVIEWS
- > APK AND SOURCE
- > EMULATOR

GOAL: ESTABLISH MANUALLY VALIDATED LINKS  
BETWEEN REVIEWS AND STACK TRACES



# RQ2: RESULTS

APP	COMMON	ONLY REVIEWS	ONLY TOOLS
APP 1	13.6%	68.2%	18.2%
APP 2	23.1%	69.2%	7.7%
AVERAGE	16%	62%	22%

## RQ2: RESULTS

**TESTING TOOLS  
POTENTIALLY MISS  
SEVERAL FAILURES  
EXPERIENCED BY USERS**

...

...

...

...

AVERAGE

16%

62%

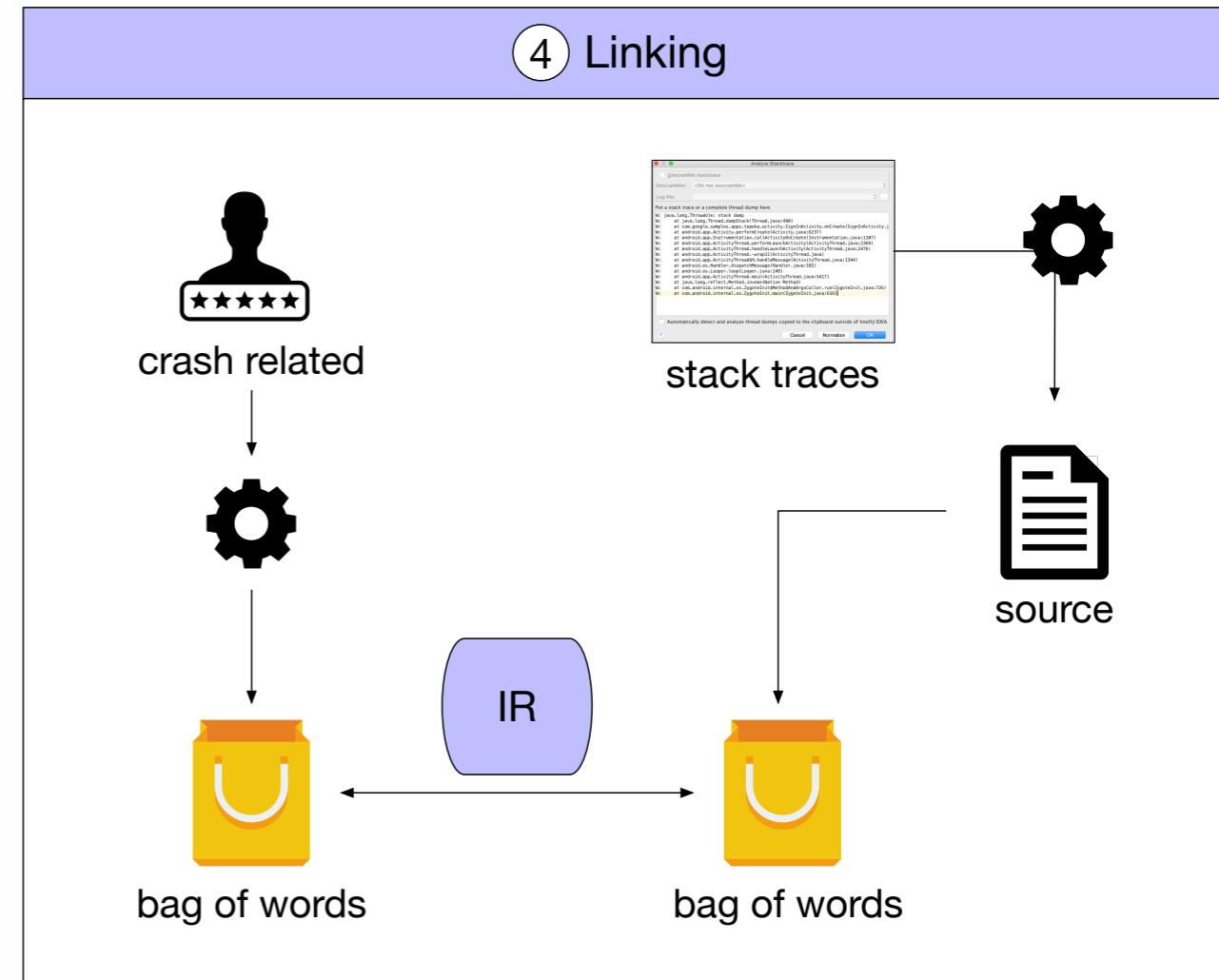
22%

# RQ3: LINKING

**GOAL:** AUTOMATICALLY LINK STACK TRACES WITH USER REVIEWS

## STEPS

- > AUGMENTING STACK TRACE WITH SOURCE CODE INFORMATION
- > PREPROCESSING FOR BOTH SOURCE
- > 2 BAGS OF WORD FOR EACH SOURCE
- > 3 DIFFERENT IR TECHNIQUES: DICE, JACCARD, VSM



# RQ3: RESULTS

APP	PRECISION	RECALL	F1 SCORE
APP 1	67%	57%	62%
APP 2	62%	68%	65%
...	...	...	...
AVERAGE	82%	75%	78%

## RQ3: RESULTS

**GOOD PERFORMANCES IN  
LINKING CRASH-RELATED  
USER REVIEWS AND  
STACK TRACES**

...

...

...

...

AVERAGE

82%

75%

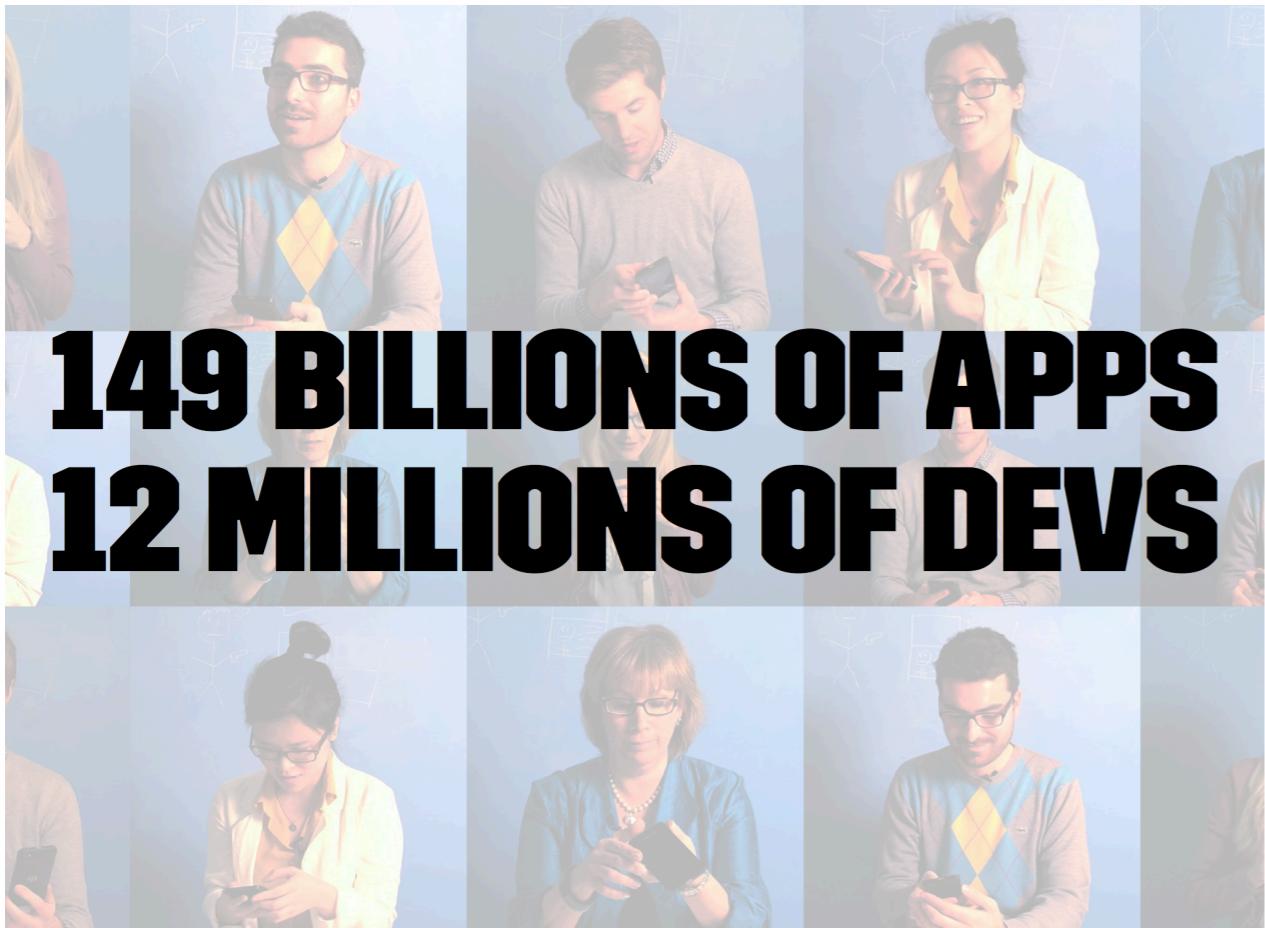
78%

# **FUTURE WORK**

## **USER-ORIENTED TESTING**

- > SUMMARIZATION
- > PRIORITIZATION
- > GENERATION



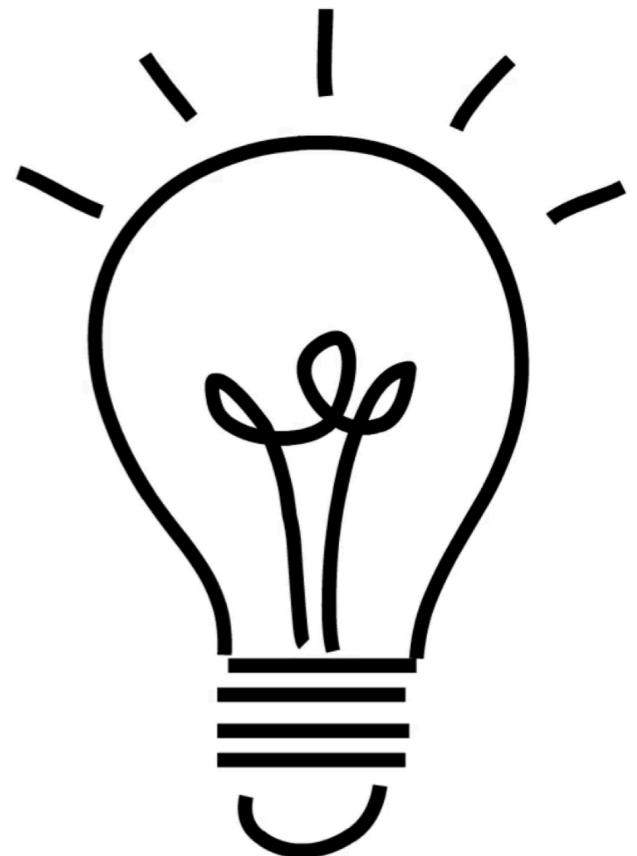


# 149 BILLIONS OF APPS 12 MILLIONS OF DEVS

## UNDERLINE IDEA

USER REVIEWS MIGHT BE HELPFUL FOR:

- COMPREHENDING THE CAUSES BEHIND A FAILURE
- EASING THE DEBUGGING PHASE
- DISCOVERING ERRORS THAT TOOLS CANNOT REVEAL



15 - GIOVANNI GRANO @ S.E.A.L.

## RQ2: RESULTS

APP	COMMON	ONLY REVIEWS	ONLY TOOLS
APP 1	13.6%	68.2%	18.2%
APP 2	23.1%	69.2%	7.7%
...	...	...	...
AVERAGE	16%	62%	22%

## FUTURE WORK

### USER-ORIENTED TESTING

- SUMMARIZATION
- PRIORITIZATION
- GENERATION



25 - GIOVANNI GRANO @ S.E.A.L.