



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

ANÁLISIS DE HERRAMIENTAS PARA MEJORAR LA CALIDAD DE APLICACIONES PARA ANDROID

Memoria Para Optar Al Título De
Ingeniero Civil Informático

CRISTOPHER NICOLÁS OYARZÚN ALTAMIRANO

Departamento de Informática
Valparaíso - Julio 2014

OBJETIVO GENERAL

Estudiar y comparar herramientas que ayuden a mejorar la calidad de las aplicaciones desarrolladas para Android, entregando a los desarrolladores una guía práctica que permita tomar mejores decisiones durante un proyecto.

OBJETIVOS ESPECÍFICOS

- Identificar los distintos problemas existentes durante el desarrollo de aplicaciones Android.
- Estudiar las herramientas que actualmente permiten mejorar la calidad de las aplicaciones, y clasificarlas en base a los distintos problemas que buscan solucionar.

OBJETIVOS ESPECÍFICOS

- En base a la clasificación realizada, llevar a cabo un análisis comparativo entre las herramientas estudiadas.
- En base al análisis realizado, implementar las herramientas que puedan ser más útiles en un entorno real de desarrollo.

¿QUÉ ES ANDROID?

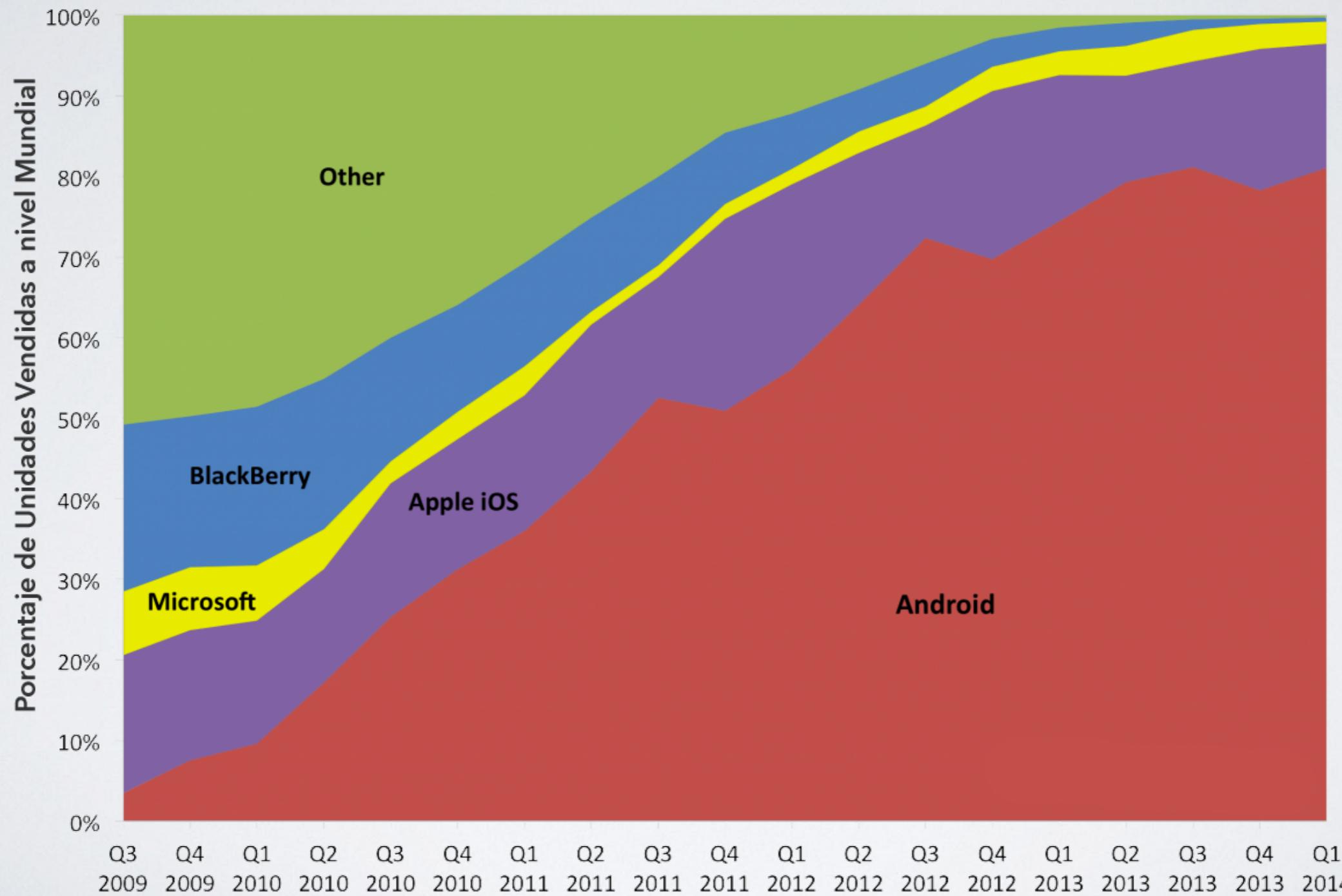




PRIMER DISPOSITIVO

HTC DREAM

PORCENTAJE DEL MERCADO GLOBAL EN DISPOSITIVOS MÓVILES



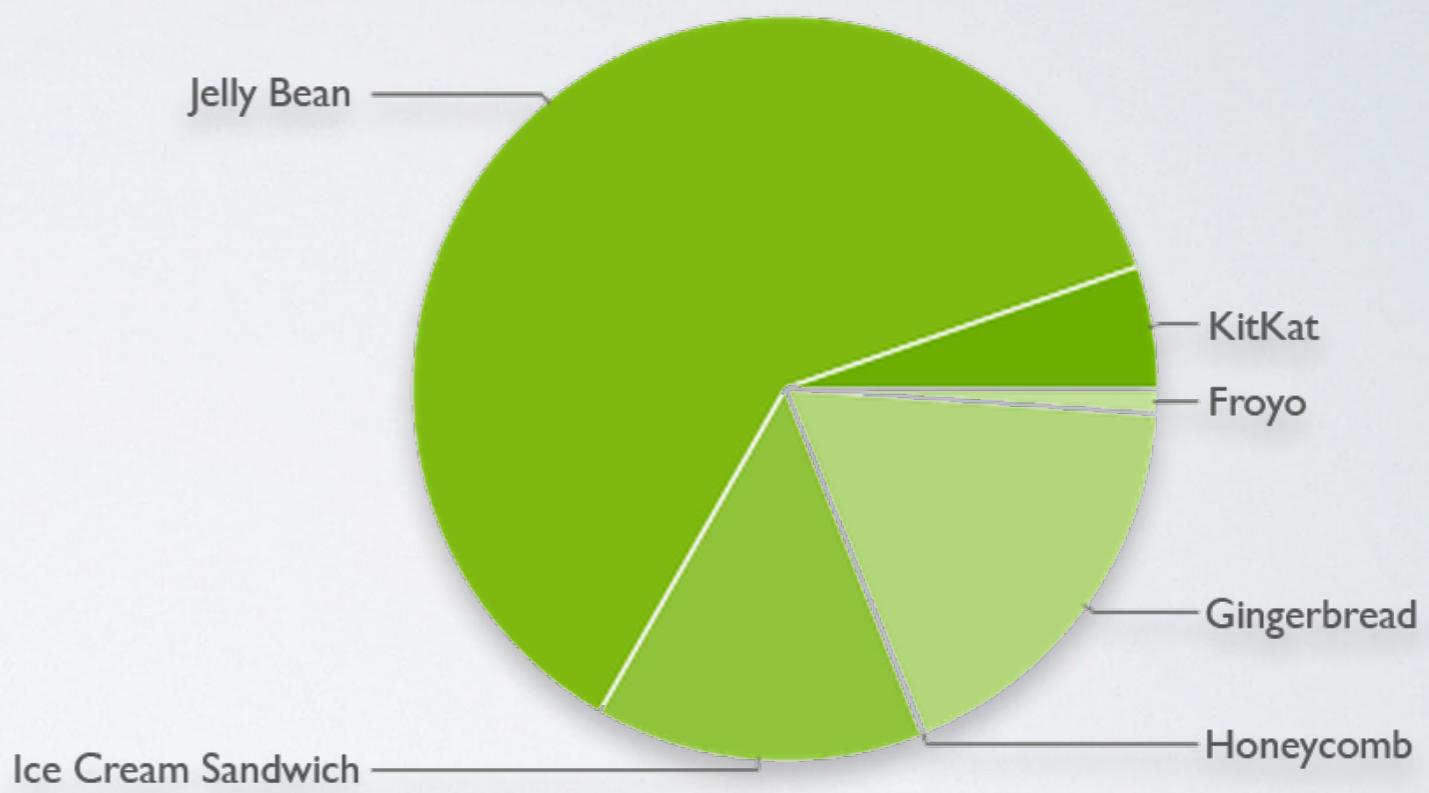
Fuente: IDC, Strategy Analytics

+ 11.000 DISPOSITIVOS



8 VERSIONES DE ANDROID

Version	Codename	API	Distribution
2.2	Froyo	8	1.1%
2.3.3 - 2.3.7	Gingerbread	10	17.8%
3.2	Honeycomb	13	0.1%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	14.3%
4.1.x	Jelly Bean	16	34.4%
4.2.x		17	18.1%
4.3		18	8.9%
4.4	KitKat	19	5.3%



Fuente: <http://developer.android.com/about/dashboards> (Abril 2014)

PROBLEMAS IDENTIFICADOS

- Fragmentación
- Distribución de versiones beta
- Crashes

HERRAMIENTAS DE TESTING UNITARIO

- Framework de Testing de Android
- Robolectric



FRAMEWORK DE TESTING DE ANDROID: JUNIT, ASSERTIONS Y MOCKS

Ventajas

- Acceso a todo el Framework de Android.
- Buena documentación.

Desventajas

- Los tests son más lentos ya que se ejecutan en un dispositivo o emulador.



ROBOLECTRIC

Ventajas

- Los tests se ejecutan en la máquina virtual de Java.
- Comunidad activa, con más de 4 años de desarrollo

Desventajas

- No todo puede hacerse con los mocks de Robolectric.

ANÁLISIS DE HERRAMIENTAS DE TESTING UNITARIO

Herramienta	Gratis	Código Abierto	Madurez
Framework Testing Android	Si	Si	Alta
Robolectric	Si	Si	Alta

ANÁLISIS DE HERRAMIENTAS DE TESTING UNITARIO

Herramienta	Documentación	Usabilidad	Duración
Framework Testing Android	Buena	Fácil	Alta
Robolectric	Media	Fácil	Baja

HERRAMIENTAS DE TESTING DE UI

- uiautomator
- Robotium
- Espresso



UIAUTOMATOR

Ventajas

- Test independientes del proceso en el que la aplicación bajo testing funciona.

Desventajas

- No tiene compatibilidad con versiones más antiguas, ya que funciona desde API 16 en adelante.
- Obtener instancias de objetos por ID's sólo desde API 18.



ROBOTIUM

Ventajas

- Gratuito.
- Fácil Uso.
- Más de 4 años de desarrollo.

Desventajas

- Los tests no se sincronizan con el thread principal.



ESPRESSO

Ventajas

- Creado por un interno de Google basándose en Robotium.
- Se sincroniza con thread principal.
- Tests hasta 3 veces más rápidos que Robotium.

Desventajas

- Falta mayor documentación ya que lleva sólo un año de desarrollo.

ANÁLISIS DE HERRAMIENTAS DE TESTING DE UI

Herramienta	Gratis	Código Abierto	Madurez	Duración
uiautomator	Si	Si	Media	Alta
Robotium	Si	Si	Alta	Media
Espresso	Si	Si	Media	Baja

ANÁLISIS DE HERRAMIENTAS DE TESTING DE UI

Herramienta	Documentación	Usabilidad	Sincronizada con UI thread
uiautomator	Media	Media	No
Robotium	Buena	Fácil	No
Espresso	Media	Media	Si

OTRAS HERRAMIENTAS DE TESTING

- uiautomator viewer
- Monkey
- MonkeyRunner
- Spoon

HERRAMIENTAS DE DISTRIBUCIÓN DE VERSIONES BETA

- Google Play Console
- HockeyApp
- AppBlade



GOOGLE PLAY CONSOLE

Ventajas

- Gratis
- Permite utilizar la tienda oficial de Google para distribuir versiones betas.
- Es posible llegar a mucha más gente.

Desventajas

- Los usuarios testers deben ser miembros de un grupo en Google Groups o una comunidad en Google+.
- El feedback entregado por los usuarios no llega directo a los desarrolladores.



HOCKEYAPP

Ventajas

- Soporte para Android, iOS, WP y Mac OS.
- Se puede enviar feedback directo a desarrolladores.
- Se puede implementar el SDK para recibir reportes de crashes.

Desventajas

- De pago, 30 días de prueba.
- Usuarios deben crearse una cuenta en el sitio de HockeyApp.



APPBLADE

Ventajas

- Soporte para Android, iOS y Blackberry.
- Usuarios pueden enviar feedback sobre la versión.

Desventajas

- De pago, después de incluir 25 usuarios.
- Usuarios deben crearse una cuenta en el sitio de AppBlade.
- Aún le falta desarrollo, poco madura.

ANÁLISIS DE HERRAMIENTAS DE DISTRIBUCIÓN DE VERSIONES

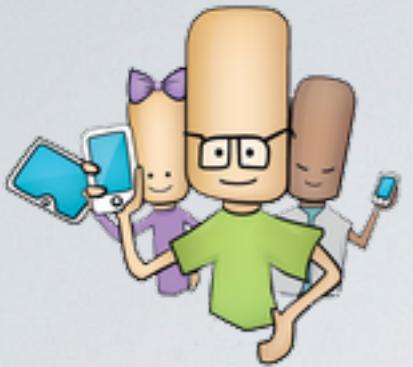
Herramienta	Gratis	Precio	Madurez
Google Play Console	Si	0 USD	Alta
HockeyApp	No	10 USD por usuarios ilimitados	Alta
AppBlade	No	1 USD por cada usuario	Baja

ANÁLISIS DE HERRAMIENTAS DE DISTRIBUCIÓN DE VERSIONES

Herramienta	Feedback	Multiplataforma	Distribución
Google Play Console	No	No	Masiva
HockeyApp	Si	Si	Limitada
AppBlade	Si	Si	Limitada

HERRAMIENTAS DE DISTRIBUCIÓN DE VERSIONES BETA Y TESTING

- The Beta Family
- UserTesting



**The Beta
Family**

THE BETA FAMILY

Ventajas

- Soporte para Android y iOS.
- Algunos usuarios reciben dinero por los tests.

Desventajas

- De pago, se ofrece una opción gratuita, pero con usuarios sin reputación.
- Cuenta con un SDK para grabar la pantalla del usuario y al usuario durante los tests, pero sólo para iOS.



USERTESTING

Ventajas

- Soporte para Android, iOS y sitios web.
- Todos los usuarios graban la pantalla de su dispositivo durante el test.
- Todos los usuarios reciben dinero por los tests.

Desventajas

- De pago, aunque es posible solicitar una prueba gratuita.

ANÁLISIS DE HERRAMIENTAS DE DISTRIBUCIÓN Y TESTING

Herramienta	Gratis	Precio	Madurez
The Beta Family	No	16.15 USD por usuario	Baja
UserTesting	No	50 USD por usuario	Alta

ANÁLISIS DE HERRAMIENTAS DE DISTRIBUCIÓN Y TESTING

Herramienta	Multiplataforma	Testing incluye video	Distribución
The Beta Family	Si	No	Limitada
UserTesting	Si	Si	Limitada

HERRAMIENTAS DE REPORTE DE CRASHES

- Google Play Console
- Crittercism
- Bugsense
- Google Analytics
- ACRA



GOOGLE PLAY CONSOLE

Ventajas

- Gratuito
- Utiliza la tienda oficial de Google para recepción de crashes sin necesidad de implementaciones.
- Reportes vienen con un mensaje por parte del usuario.

Desventajas

- Muy pocos usuarios envían reportes de crashes.



CRITTERCISM

Ventajas

- Recepción de todos los crashes de los usuarios
- Recepción de información muy detallada en cada reporte.
- Soporte para Android, iOS, WP y HTML5.

Desventajas

- De pago.



BUGSENSE

Ventajas

- Recepción de todos los crashes de los usuarios
- Recepción de información útil en cada reporte.
- Soporte para Android, iOS, WP y HTML5.

Desventajas

- De pago.



GOOGLE ANALYTICS

Ventajas

- Gratis.
- Recepción de todos los crashes de los usuarios.
- Recepción de información útil en cada reporte.
- Soporte para Android, iOS y Web.

Desventajas

- Los reportes de crashes se muestran en la plataforma con un día de retraso.



ACRA

Ventajas

- Gratis y de código abierto.
- Se reciben todos los crashes de los usuarios.
- El desarrollador decide que información quiere recibir en cada reporte.

Desventajas

- Es necesario implementar una aplicación web para poder ver los reportes de crashes.

ANÁLISIS DE HERRAMIENTAS DE REPORTE DE CRASHES

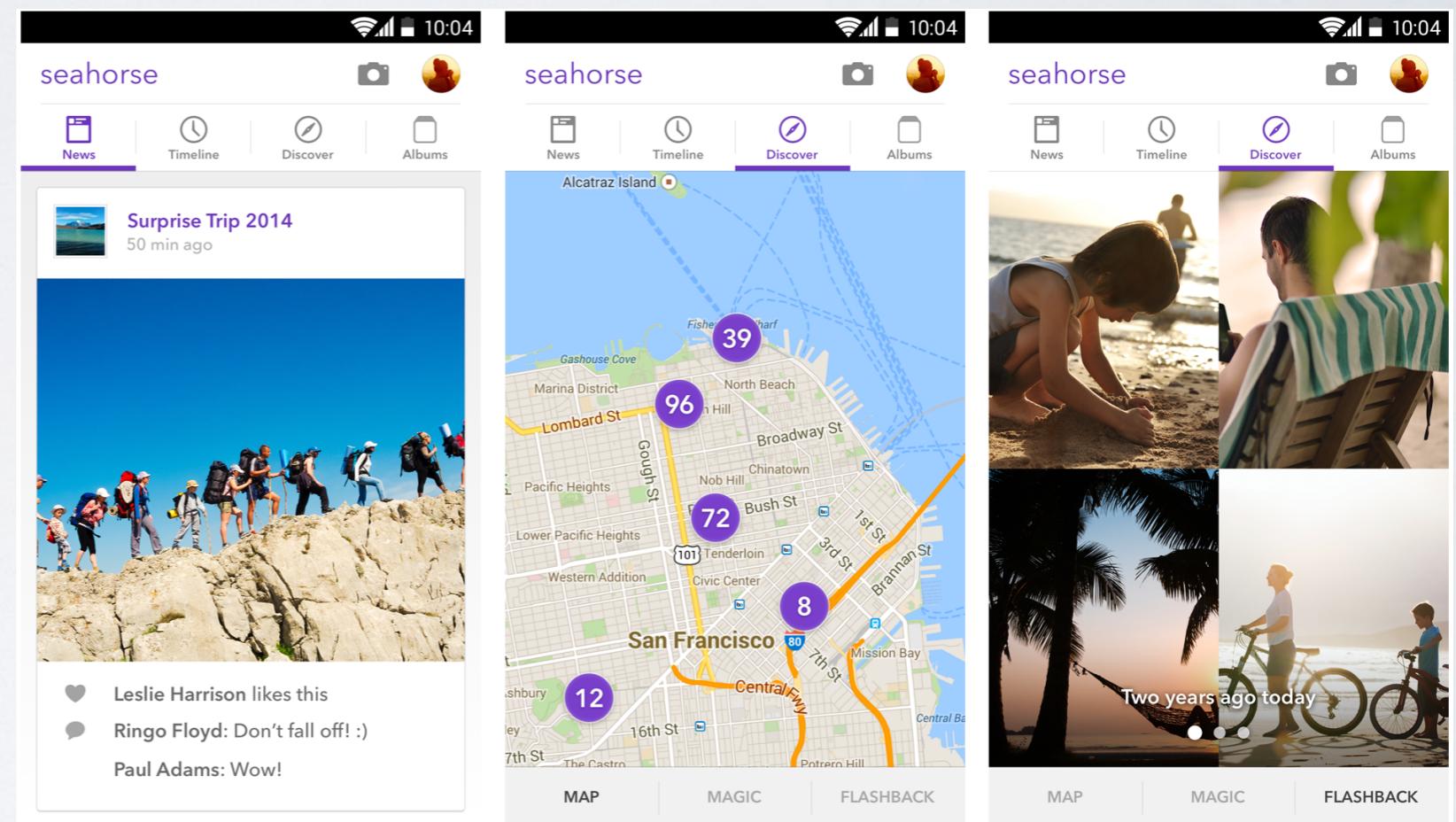
Herramienta	Gratis	Precio	Multiplataforma
Google Play Console	Si	0 USD	No
Crittercism	No	24 USD	Si
BugSense	No	19 USD	Si
Google Analytics	Si	0 USD	Si
ACRA	Si	0 USD	No

ANÁLISIS DE HERRAMIENTAS DE REPORTE DE CRASHES

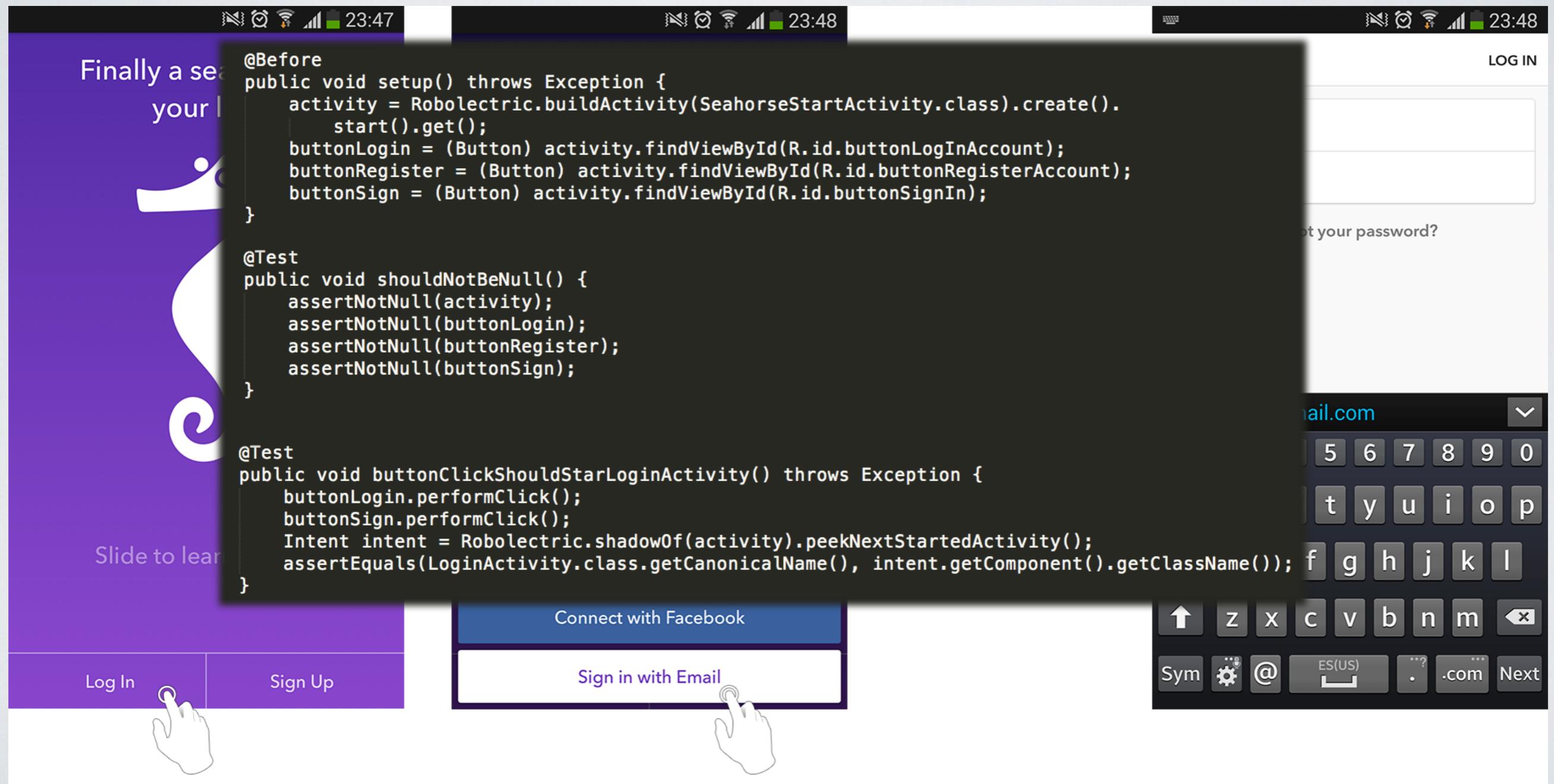
Herramienta	Madurez	Usabilidad	Calidad del Reporte
Google Play Console	Alta	Fácil	Baja
Crittercism	Alta	Fácil	Alta
Bugsense	Alta	Fácil	Media
Google Analytics	Alta	Fácil	Media
ACRA	Media	Difícil	Media

IMPLEMENTACIÓN DE HERRAMIENTAS EN SEAHORSE

- Robolectric
- Robotium + Spoon
- Monkey
- HockeyApp
- UserTesting
- Crittercism



ROBOLECTRIC



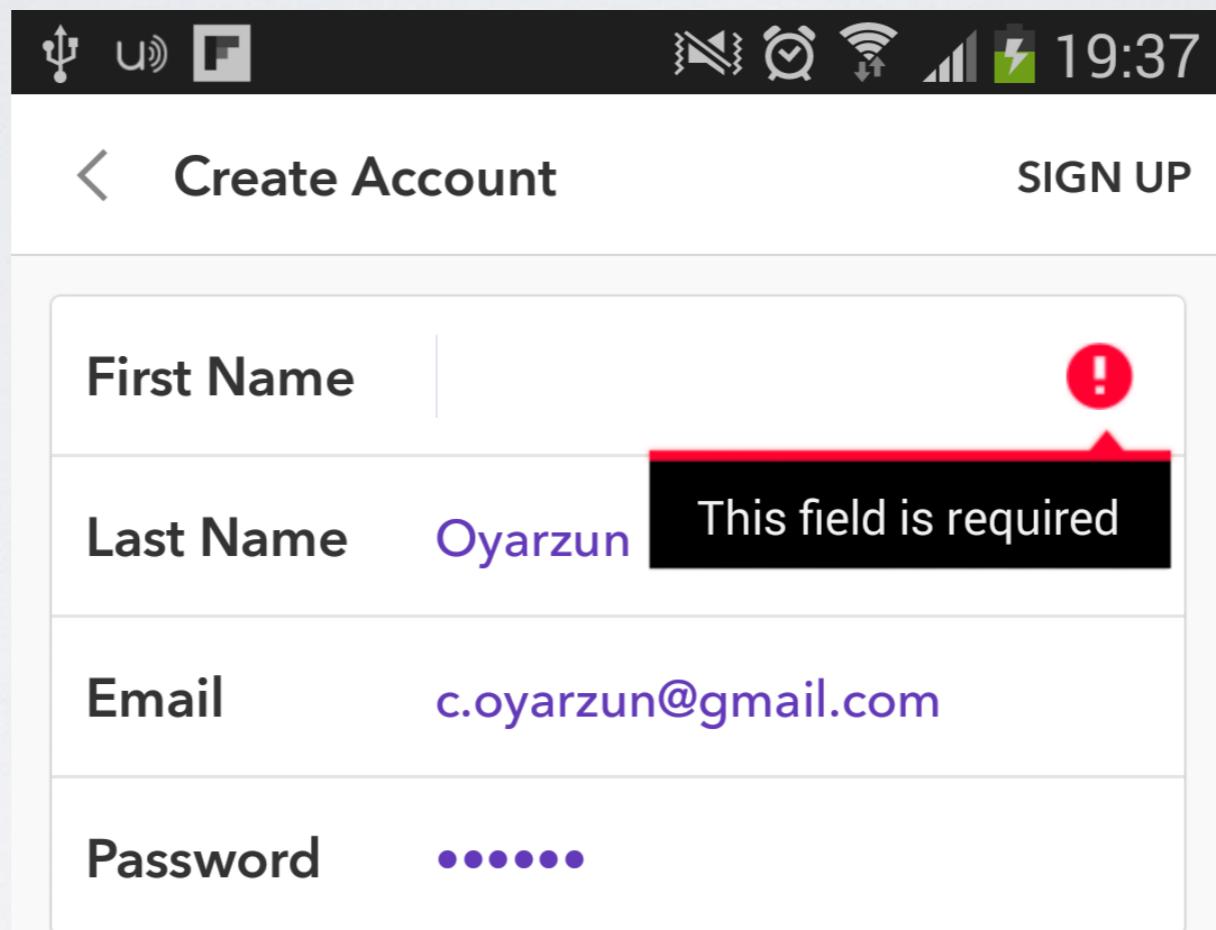
ROBOLECTRIC

```
@Before
public void setup() throws Exception {
    activity = Robolectric.buildActivity(SeahorseStartActivity.class).create().
        start().get();
    buttonLogin = (Button) activity.findViewById(R.id.buttonLogInAccount);
    buttonRegister = (Button) activity.findViewById(R.id.buttonRegisterAccount);
    buttonSign = (Button) activity.findViewById(R.id.buttonSignIn);
}

@Test
public void shouldNotNull() {
    assertNotNull(activity);
    assertNotNull(buttonLogin);
    assertNotNull(buttonRegister);
    assertNotNull(buttonSign);
}

@Test
public void buttonClickShouldStarLoginActivity() throws Exception {
    buttonLogin.performClick();
    buttonSign.performClick();
    Intent intent = Robolectric.shadowOf(activity).peekNextStartedActivity();
    assertEquals(LoginActivity.class.getCanonicalName(), intent.getComponent().getClassName());
}
```

ROBOTIUM + SPOON



By creating your account you're accepting
our [Terms of Service](#) and [Data Use Policy](#).

ROBOTIUM + SPOON

```
Spoon.screenshot(getActivity(), "RegisterActivity");
solo.assertCurrentActivity("SeahorseStartActivity Never Loaded", RegisterActivity.class);
solo.enterText((EditText) solo.getView(R.id.editLastNameRegister), "Oyarzun");
solo.enterText((EditText) solo.getView(R.id.editEmailRegister), "c.oyarzun@gmail.com");
solo.enterText((EditText) solo.getView(R.id.editPasswordRegister), "123456");
Spoon.screenshot(getActivity(), "RegisterActivityWithInfo");
solo.clickOnText(getActivity().getString(R.string.SIGN_UP).toUpperCase());
Spoon.screenshot(getActivity(), "ShowMissingField");
```

ROBOTIUM + SPOON

Genymotion Nexus 4 - 4.2.2 - API 17 - 768x1280

First Name
Last Name
Email
Password

By creating your account you're accepting our [Terms of Service](#) and [Data Use Policy](#).

SIGN UP

samsung GT-I9505

First Name
Last Name
Email
Password

By creating your account you're accepting our [Terms of Service](#) and [Data Use Policy](#).

SIGN UP

Genymotion HTC One - 4.2.2 - API 17 - 1080x1920

First Name
Last Name
Email
Password

By creating your account you're accepting our [Terms of Service](#) and [Data Use Policy](#).

SIGN UP

MONKEY

```
adb shell monkey -p co.seahorse.android -v 2000
```

HOCKEYAPP

Seahorse a Android | Beta

App Overview Versions 23 Crashes 0 Feedback 1 Users 12 Timeline

Add Version Invite User Manage App

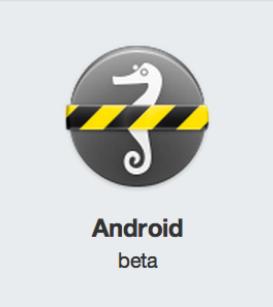
Seahorse a
co.seahorse.android.debug

App ID: 6caf551684093e76263bef69fcab049f ✘
Secret: [Show](#)

Owner
Seahorse, Inc. (Daniel Undurraga)

Download & Feedback
[Public Page](#)

Latest Downloadable Version
[0.9.7-alpha20140619 \(34\)](#)

Android beta

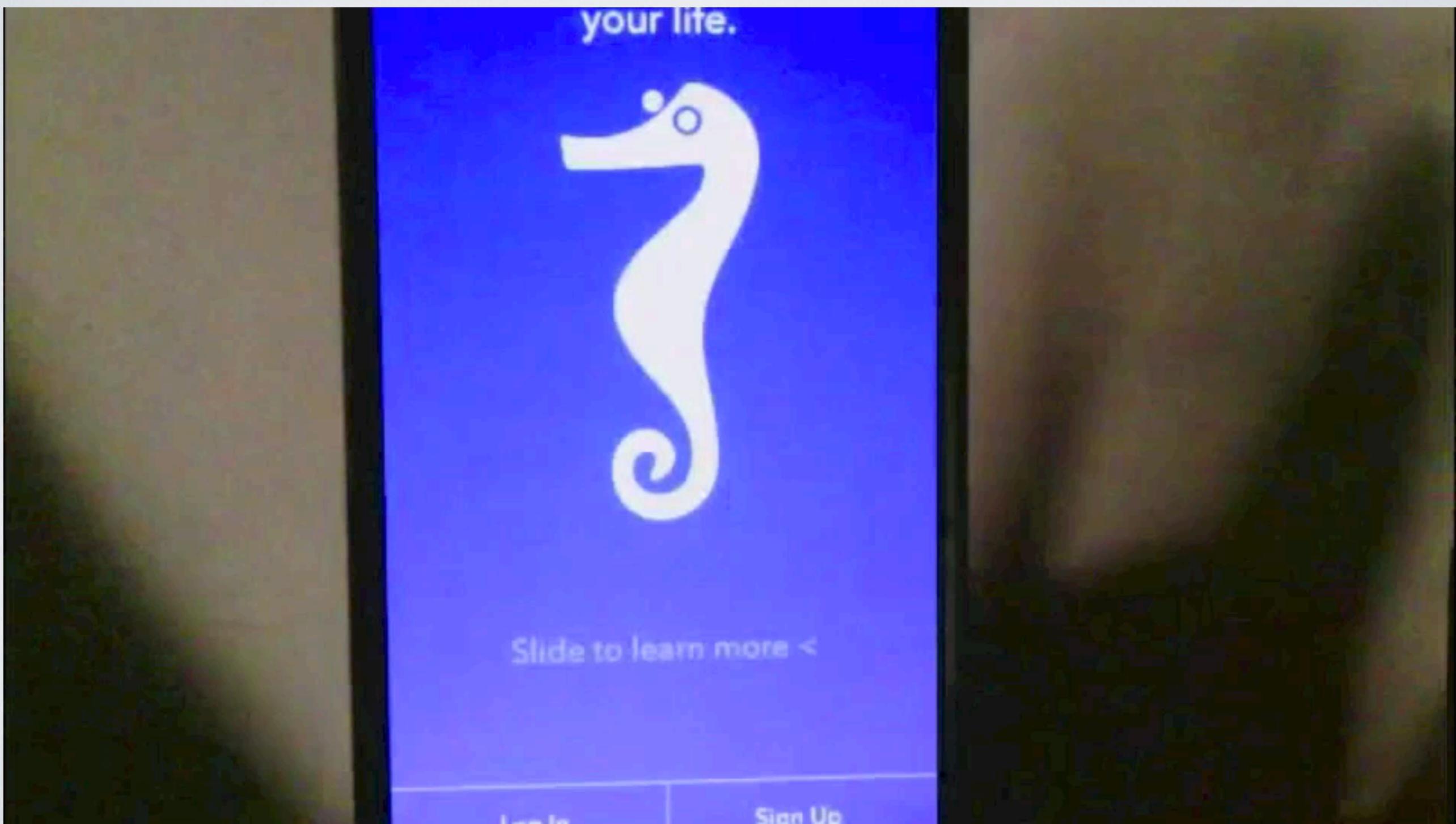
Latest Versions

Name	Code	Devices	Downloads	Crashes	Storage	Last Updated	Actions
0.9.7-alpha20140619	34	0	1	0	21.16MB	19 Jun 2014, 18:20	
0.9.6-alpha20140617	34	0	4	0	20.46MB	17 Jun 2014, 00:57	
0.9.6-alpha20140605	34	0	2	0	20.30MB	05 Jun 2014, 03:14	
0.9.5-alpha20140603	33	0	1	0	20.27MB	03 Jun 2014, 23:36	
0.9.5-alpha20140530	33	0	2	0	20.12MB	31 May 2014, 00:19	

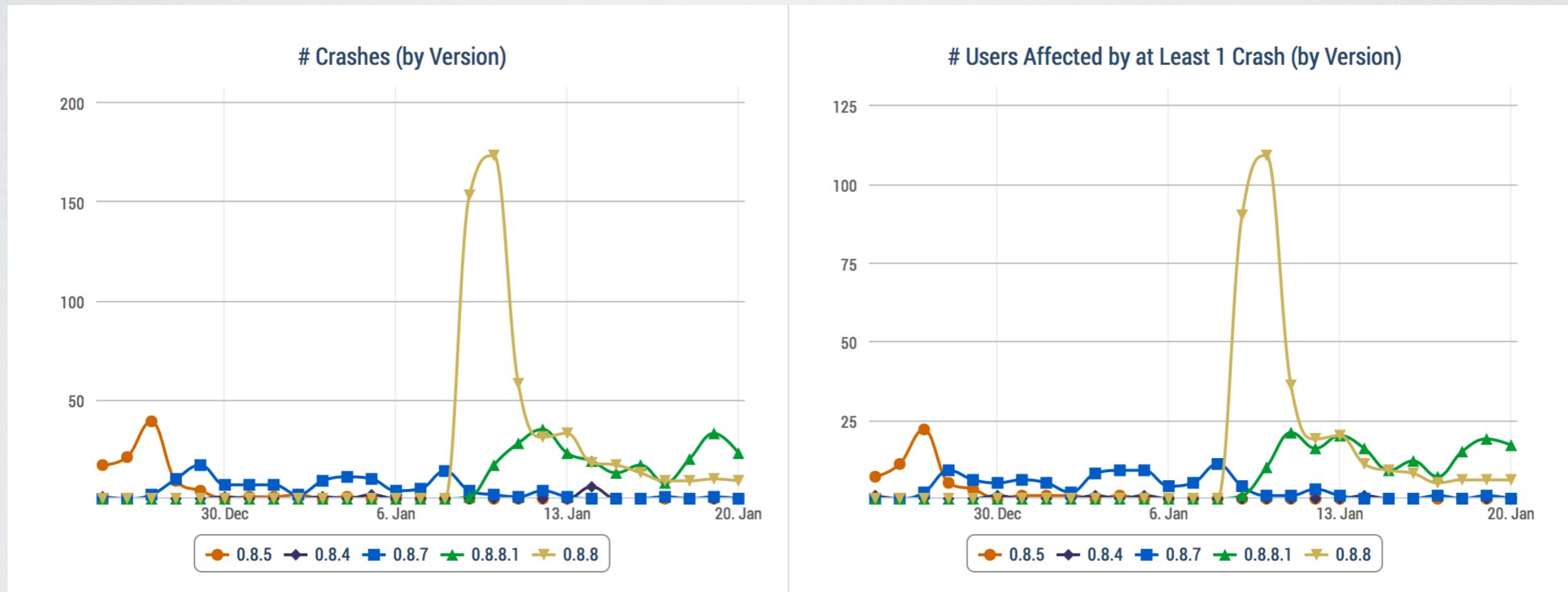
Statistics

Base Metrics Usage Time

USERTESTING



CRITTERCISM



CONCLUSIONES

- La Fragmentación es el principal problema que afecta a los desarrolladores.
- Existe gran cantidad de documentación obsoleta.
- Con respecto al testing unitario se recomienda escribir la mayor cantidad de código independiente de la API de Android.

CONCLUSIONES

- Con respecto al testing de interfaz gráfica, depende en gran medida del contexto en que se quieran usar las herramientas ya que uiautomator, Robotium y Espresso cuentan con ventajas una sobre otra.
- Con respecto a la distribución de versiones, va a depender del grado de distribución que se está buscando.

CONCLUSIONES

- Existen herramientas para combinar la distribución de versiones beta con el testing, aunque tienen un costo asociado.
- En relación a los reportes de crashes, el desarrollador debe fijarse en el nivel de detalle que quiere recibir.

TRABAJO FUTURO

- Existen otras herramientas de testing que se están popularizando relacionadas al cloud testing.
- Se podrían realizar estudios futuros con herramientas como:
 - Herramientas de debugging y análisis de performance.
 - Herramientas de análisis de datos y tracking de eventos.
 - Herramientas de apoyo a bases de datos.
 - Bibliotecas útiles para el desarrollo de aplicaciones.

¿PREGUNTAS?

