

مروری بر آزمایش نرم افزار تلفن همراه در محیط ابری

محمد ملایون^۱، طاهره رستمی^۲

A Study of mobile application testing in the cloud

Mohammad molayoun¹, Tahereh rostami²

Email: mm6889@gmail.com, tehereh.rostami@gmail.com

چکیده

امروزه گسترش و آزمایش نرم افزارهای تلفن همراه با توجه به پیچیدگی و محدود بودن منابع با مشکلات فراوانی روبرو شده است. آزمایش دستی نرم افزارها در انواع دستگاه‌های تلفن همراه زمان بر و پرهزینه است و از طرفی استفاده از شبیه سازها امکان برقراری ارتباط واقعی با خصوصیات یک دستگاه را ایجاد نمی‌کند. نرم افزارهای تلفن همراه بر اساس رضایت کاربران مورد بررسی قرار می‌گیرند که عملکرد و کارایی این نرم افزارها قبل از راه اندازی، در دستگاه‌های تلفن همراه متفاوت آزمایش می‌شوند. این مقاله، به مرور آزمایش نرم افزار در محیط ابری، انواع آزمون‌های قابل اجرا، تفصیل مزایا و معایب آن‌ها پرداخته و در پایان تعدادی از محبوب‌ترین محیط‌های ابری را معرفی می‌کند.

کلمات کلیدی

آزمایش ابری، آزمایش خودکار، تلفن همراه ابری، نرم افزار تلفن همراه.

۱. مقدمه

در سال‌های اخیر، با پیشرفت تلفن‌های همراه هوشمند و سیستم عامل‌های آن‌ها، نیازهای کاربران و برنامه نویسان دستخوش تغییرات فراوانی گردیده که در نهایت منجر به تولید و عرضه نرم افزارها مطابق با میل و رضایت مصرف کنندگان شده است. این نرم افزارها قبل از عرضه در بازار باید از جنبه‌های گوناگونی مورد آزمایش قرار بگیرند تا عملکرد و کارایی آن‌ها تضمین شده باشد. برای اطمینان از میزان سازگاری هر برنامه، می‌بایست آن را در تلفن همراه واقعی نصب و آزمایش کنیم. با توجه به گستردگی دستگاه‌های تلفن همراه، نصب و آزمایش نرم افزار در این مقیاس مستلزم صرف زمان و هزینه بالایی است. تلفن‌های همراه هوشمند برای اجرای نرم افزارها، وابستگی زیادی به منابع، نوع سیستم عامل، وضوح تصویر دستگاه، حسگرها (مکان‌یاب، شتاب سنج و...) دارند. به همین دلیل برنامه نویسان برای آزمایش نرم افزارهای خود از شبیه سازهای محیط تلفن همراه استفاده می‌کنند، اما با توجه به محدود بودن منابع، تنوع در هسته سیستم عامل و وضوح تصویرهای متفاوت، شبیه سازها با مشکلاتی از قبیل دسترسی نداشتن به داده‌های حسگر واقعی و نبود شبیه ساز برای انواع مختلف هسته‌های سیستم عامل مواجه می‌گردند. بنابراین، شبیه سازها در این شرایط نمی‌توانند بستر مناسبی برای آزمایش برنامه‌ها باشند. در جدول ۱ تفاوت‌های محیط شبیه ساز با محیط دستگاه واقعی را مشاهده می‌نمایید.

جدول (۱) تفاوت‌های شبیه ساز و دستگاه واقعی

ویژگی	شبیه	دستگاه واقعی
نمایش پیکسل‌های واقعی	x	✓
استفاده از حسگرها	x	✓
انعطاف در تغییر هسته سیستم عامل	✓	x
محیط واقعی آزمایش	x	✓
تعامل با سخت افزار	x	✓
استفاده شبکه (3G.Wifi)	x	✓

حال این سؤال پیش می‌آید که مناسب‌ترین روش برای آزمایش نرم افزار موبایل چیست؟ پاسخ این سؤال روش آزمایش تلفن همراه از راه دور در محیط ابری است که کاربر از طریق محیط وب به تلفن همراه‌های واقعی دسترسی یافته و از تمامی ویژگی‌های آن برای آزمایش نرم افزار خود استفاده می‌کند. در ادامه این مقاله ابتدا به مرور ساختار کلی این محیط، شرح چگونگی انجام آزمون‌ها، معماری نرم افزاری، مزایا و معایب آن و در پایان تعدادی از محبوب‌ترین محیط‌های ابری را معرفی می‌کنیم.

۲. محیط ابری آزمایش نرم افزار تلفن همراه

ساختار کلی این محیط در شکل ۱ نشان داده شده است. در محیط آنلاین ابری، تلفن‌های همراه در یک فضای آزمایشگاهی از طریق USB به سرورها متصل و با دستورات کنترلی که از کاربر دریافت می‌کنند در تلفن‌های همراه اجرا می‌شوند و توسط Capture کردن، تصویر نمایش داده شده در دستگاه تلفن همراه، نتایج را به کاربر نشان می‌دهند. در این روش کاربر توسط یک رابط کاربری تحت وب با سرور ارتباط برقرار کرده و پس از بارگذاری نرم افزار توسط کاربر، سرور شروع به آزمایش آن در دستگاه‌های انتخابی کاربر می‌کند و نتایج را به او نشان می‌دهد. این محیط از قابلیت کشیدن و رها کردن، لمس کردن، ضربه زدن و هر آنچه در یک تلفن همراه هوشمند توسط کاربر مصرف کننده انجام می‌شود پشتیبانی می‌نماید.

آزمایش نرم افزارها در دستگاه‌های مختلف هم به صورت دستی و هم به صورت خودکار صورت می‌گیرد. در روش دستی کاربر توسط صفحه کلید و موشواره عمل‌های کشیدن (swap) را با رسم یک خط از نقطه X تا Y و با سرعت V انجام می‌دهد و با کلیک کردن می‌تواند عمل لمس را انجام دهد. در روش خودکار ارائه شده دستورات در فایل XML ذخیره، به صورت خودکار بر روی دستگاه‌های مختلف اجرا و نتایج آن به کاربر نمایش داده می‌شود [۱].



شکل (۱) معماری ارتباطی محیط ابری آزمایش نرم افزار تلفن همراه

۳. انواع روش‌های آزمایش

آزمایش‌های صورت گرفته بر روی دستگاه‌های تلفن همراه در حالت اجرای نرم افزار به پنج دسته کلی تقسیم می‌شود: آزمون عملکرد، آزمون کارایی، آزمون سازگاری، آزمون GUI (رابط گرافیکی کاربر) و آزمون امنیت



شکل (۲) آزمون‌ها قابل انجام در محیط ابری آزمایش نرم افزار تلفن همراه

۱.۳. آزمون عملکرد

آزمون عملکرد، عمومی‌ترین آزمون برای بررسی ویژگی‌های یک نرم افزار است. در این آزمایش باید بدانیم که هر نرم افزار چگونه کار می‌کند و در صورت اجرا در دستگاه‌های تلفن همراه متفاوت، عملکرد یکسان و مشابه داشته باشد. به‌عنوان مثال یک برنامه ماشین حساب وظیفه ارائه محاسبات ریاضی را دارد و باید روی هر نوع از دستگاه‌های تلفن همراه که اجرا می‌شود نتیجه یکسانی را نمایش دهد [۲].

روند آزمون عملکرد با استراتژی جعبه سیاه صورت می‌گیرد. در این روش هیچ دسترسی به کد منبع برنامه وجود ندارد و برنامه نویس با خیال آسوده می‌تواند فایل‌های خود را در سرور بارگذاری کند. این آزمایش در فایل APK نرم افزار انجام می‌شود. استراتژی جعبه سیاه از دیدگاه کاربر، برنامه را مورد آزمایش قرار می‌دهد. این به این معناست که هیچ اطلاعاتی از کد و طراحی داخلی برنامه نداشته و با تمرکز بر ورودی‌ها و خروجی‌ها مشخص می‌کند



که یک برنامه، در مقابل یک عمل خاص، چه پاسخی باید بدهد و با رفتار واقعی برنامه در مستندات به بررسی و مقایسه آن می‌پردازد [۳].

۲.۳. آزمون کارایی

کارایی هر برنامه بیان می‌کند، یک نرم افزار تلفن همراه تا چه اندازه می‌تواند بهترین پاسخ را در کوتاه‌ترین مدت زمان ممکن به کاربر بدهد که از طریق محاسبه زمان پاسخگویی و توان عملیاتی آن به دست می‌آید. برای محاسبه زمان پاسخگویی (R)، از فرمول زیر استفاده می‌کنیم:

$$R = (T_o - T_i) - L \text{ seconds/Request}$$

R = زمان پاسخگویی

T_i = زمان درخواست سرویس توسط کاربر

T_o = زمان تکمیل درخواست کاربر

L = زمان تأخیر شبکه (در صورت وجود)

توان عملیاتی (T)، از محاسبه مجموع تعداد درخواست‌های پردازش شده در زمان معین به دست می‌آید و زمان پاسخگویی در این محاسبه تأثیر گذار است. این محاسبه به صورت زیر انجام می‌شود:

$$T = t / \text{Avg} (R) \text{ request/second}$$

$$\text{Avg} (R) = \text{Sum} (R_i) / E \text{ second/request}$$

t = زمان (بر حسب ثانیه)

E = تعداد درخواست‌ها

R_i = زمان پاسخ درخواست i

(Avg (R) = میانگین زمان پاسخ درخواست‌های i

تأخیر در شبکه (L): به مدت زمان تأخیر در شبکه‌ای که تلفن همراه به آن متصل شده و قرار است بر روی آن آزمون صورت گیرد گفته می‌شود. در اینجا شبکه ما محیط ابری است و مقدار این تأخیر به تلفن همراه و شبکه ابری بستگی دارد که برای محاسبه آن از فرمول زیر استفاده می‌شود [۲].

$$L = (RTT/2) + D + E \text{ seconds}$$

RTT = زمان تأخیر گردشی (زمان بین فرستنده و گیرنده)

D = زمان تأخیر ردیابی مسیر حرکت

E = زمان تأخیر نقطه پایان محاسباتی

۳.۳. آزمون سازگاری

سازگاری یک برنامه تلفن همراه در تعداد دستگاه‌های تلفن همراهی که کارایی صحیح نرم افزار را پشتیبانی می‌کنند تعریف می‌شود و به عواملی مانند سخت افزار و نرم افزار دستگاه بستگی دارد. سازگاری یک برنامه با توجه به نتایج آزمون کارایی و با استفاده از معیارهای کارایی (R, T, L) به دست می‌آید. سازگاری (C) یک نرم افزار با فرض محدوده خاصی از مقادیر، برای هر یک از معیارهای کارایی به دست می‌آید. در این محاسبه برای معیارهای کارایی (R, T, L)



(T, L) بازه ۰ تا ۱۰ در نظر می‌گیریم و طبق فرمول زیر درصد سازگاری را محاسبه می‌کنیم [۲].

$$C = (MR + MT + ML) / 3$$

MR = زمان پاسخ اندازه گیر کارایی

MT = توان عملیاتی اندازه گیر کارایی

ML = تأخیر در شبکه اندازه گیر کارایی

در جدول ۲ نمونه‌ای از محاسبات فوق را ملاحظه می‌نمایید. در این مثال، زمان پاسخگویی نرم افزار در ۵ دستگاه مختلف، با فرض زمان ورود درخواست ۱ و کسر تأخیر در شبکه، محاسبه می‌کند.

$$RI = (1.3 - 1) - 0.1 = 0.2$$

جدول (۲) محاسبه زمان پاسخگویی

R	L	To	Ti	
۰.۲	۰.۱	۱.۳	۱	دستگاه ۱
۰.۱	۰.۳	۱.۴	۱	دستگاه ۲
۰.۴	۰.۲	۱.۶	۱	دستگاه ۳
۰.۲	۰.۱	۱.۳	۱	دستگاه ۴
۰.۲	۰.۱	۱.۳	۱	دستگاه ۵

جدول ۳ نشان می‌دهد که هر چقدر مقدار زمان پاسخگویی (R) و تأخیر در شبکه (L) کمتر و توان عملیاتی (T) بیشتر باشد، امتیازی بیشتری خواهیم داشت.

جدول (۳) معیار امتیازدهی به معیارهای (R, T, L)

R	T	L	امتیاز
۰	۵	۰	۱۰
۰.۱	۴	۰.۱	۹
۰.۲	۳	۰.۲	۸
۰.۳	۲	۰.۳	۷
۰.۴	۱	۰.۴	۶
۰.۵	۰	۰.۵	۵
۰.۶	۰.۵	۰.۶	۴
۰.۷	۰.۴	۰.۷	۳
۰.۸	۰.۳	۰.۸	۲
۰.۹	۰.۲	۰.۹	۱
۱	۰.۱	۱	۰



جدول (۴) محاسبه درصد سازگاری

امتیاز	میانگین	
۸	۰.۲۲	میانگین زمان پاسخ (R)
۹	۴.۵	توان عملیاتی (T)*
۹	۰.۱۶	میانگین تأخیر شبکه (L)
	٪۸۶	درصد سازگاری (C)
		* t = ۱ ثانیه

$$C = (8+9+9)/3 = 8.6 \times 10 = 86\%$$

طبق محاسبات صورت گرفته نرم افزار ما در ۸۶٪ از دستگاه‌های انتخابی سازگار است

۴.۳. آزمون رابط گرافیکی کاربر

آزمایش رابط کاربری برنامه برای پیدا کردن Crash در برنامه و بر اساس تمرکز بر فعالیت کاربر انجام می‌شود. با استفاده از ابزار Ripper Android دستورات و فعالیت‌هایی که در برنامه انجام می‌شود بررسی، با دستورات منطقی مقایسه و نتیجه آن در ثبت کننده وقایع ذخیره می‌شود. در این بستر نرم افزاری، ثبت وقایع توسط یک ابزارک (widget) انجام می‌گیرد. برای مثال وقتی کاربر دکمه را فشار می‌دهد و یا متنی را ویرایش می‌کند، رابط گرافیکی دچار تغییر شده و زمان پاسخ رابط کاربری برنامه، به دستورات کاربر با مستندات برنامه مقایسه می‌شود. این بستر برای همه‌ی نسخه‌های مختلف اندروید قابل استفاده نیست. ابزار testdriod برای انجام آزمایش رابط کاربری به صورت موازی است. این بستر با ضبط کد آزمون، اجازه آزمایش کردن خودکار را برای ما فراهم می‌آورد. این عملیات در سه مرحله: تعریف روند آزمون رابط کاربری، انجام آزمایش و به دست آوردن نتایج آزمون صورت می‌گیرد. این بستر برای ضبط وقایع چند لمسی مناسب نیست [۴].

۵.۳. آزمون امنیت

آزمون امنیت در نرم افزارهای تلفن همراه یکی از مهم‌ترین اجزای آزمایش است. این آزمایش با بررسی ورودی و خروجی نرم‌افزار به کشف آسیب پذیری آن می‌پردازد. در محیط ابری آزمایش امنیت نرم‌افزار به صورت خودکار صورت می‌پذیرد. در این محیط با توجه به در دسترس نبودن کد منبع نرم افزار، روش کار به این صورت است که در ابتدا ورودی و خروجی‌های برنامه مشخص و توسط یک ماژول تولید ورودی Input Generator موارد آزمون تولید می‌شود. پس از تولید موارد آزمون نرم افزار بر روی یک شبیه ساز اجرا می‌شود. در این شبیه ساز امکان آزمایش چندین نرم افزار مشابه با مقادیر ورودی تولید شده وجود دارد. نتایج آزمون توسط ماژول Monitoring Facilities بررسی و در مخزن خطاها ذخیره می‌شود. این نتایج توسط موتور تجزیه و تحلیل با نتایج واقعی آزمون بررسی شده و برای کاربر مورد نظر نمایش داده می‌شود [۵].

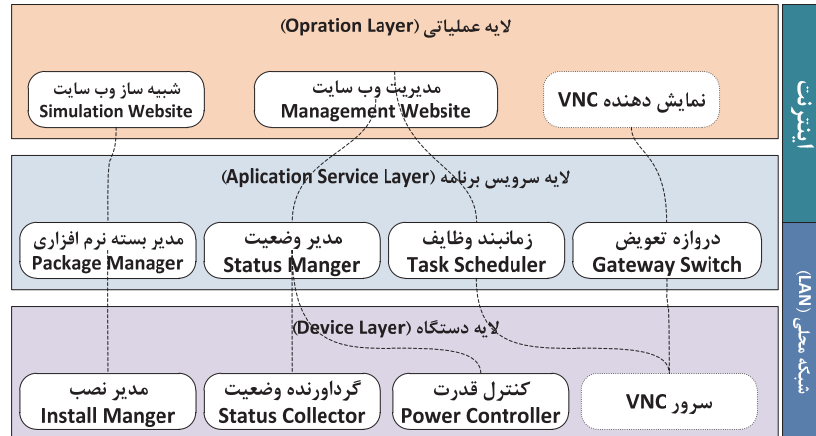


جدول (۵) آزمون‌ها در یک نگاه

هدف	آزمون
عملکرد یکسان در دستگاه‌های مختلف	آزمون عملکرد
بهترین پاسخ در کوتاه‌ترین زمان	آزمون کارایی
بهترین کارایی در بیشترین دستگاه	آزمون سازگاری
اجرای صحیح با تمرکز بر فعالیت کاربر	آزمون رابط گرافیکی
کشف آسیب پذیری	آزمون امنیت

۴. معماری محیط ابری آزمایش نرم افزار تلفن همراه

ساختار کلی محیط ابری آزمایش نرم افزار تلفن همراه، در شکل ۳ نمایش داده شده است. همان‌طور که مشاهده می‌کنید این محیط در سه لایه طراحی شده است. لایه اول، لایه دستگاه است که شامل مجموعه‌ای از نرم افزارهای تعبیه شده در حال اجرا در تلفن همراه هوشمند است. لایه خدمات نرم افزار، یک پلی برای ارتباط بین دو لایه دیگر است و برنامه‌های سرور در آن اجرا می‌شود. لایه عملیاتی، برای کاربران از راه دور طراحی شده است که آن‌ها می‌توانند از طریق کلاینت VNC Viewer به تلفن همراه هوشمند متصل شوند. تمامی لایه‌ها در محیط شبکه پیاده سازی شده‌اند به این صورت که لایه دستگاه در شبکه LAN و لایه عملیاتی در اینترنت اجرا می‌شود. لایه خدمات نرم افزار به هر دو این شبکه‌ها دسترسی دارد [۶].



شکل (۳) معماری سه لایه‌ای محیط ابری آزمایش نرم افزار تلفن همراه

در این قسمت به توضیح ماژول‌های مختلف این لایه‌ها می‌پردازیم.

۴.۱. لایه دستگاه

- ۱- مدیریت نصب: پس از بارگذاری پکیج نرم افزار توسط کاربر این ماژول با ارسال دستور نصب به دستگاه آن را نصب می‌کند.
- ۲- سرور VNC: کاربر برای اینکه از نتایج دستورات اجرا شده در گوشی تلفن همراه مطلع شود نیاز به مشاهده صفحه نمایش گوشی تلفن همراه دارد. سرور VNC با capture کردن تصویر با فرکانس بالا تصاویر را به کاربر ارسال می‌کند.
- ۳- گردآورنده وضعیت: یک مکانیسم مانیتورینگ است. وقتی دستگاه تلفن همراه متوقف شود، ماژول مدیریت



وضعیت که در لایه بالاتر کار می‌کند با راه اندازی مجدد دستگاه را آماده کار می‌کند. ماژول گردآورنده وضعیت اطلاعات CPU و RAM را به منظور آگاهی از وضعیت دستگاه‌های تلفن همراه به مدیر وضعیت ارسال می‌کند.

۴- کنترل قدرت: وقتی دستگاه متوقف شده است، مدیر وضعیت دستور راه اندازی مجدد می‌دهد و کنترل کننده قدرت این عملیات را با اجرای دستور PowerManager.reboot انجام می‌دهد [۶].

۲.۴. لایه سرویس برنامه

- مدیر بسته نرم افزاری: این ماژول فایل بارگذاری شده توسط کاربر را در پوشه‌ای در سرور ذخیره می‌کند. ویژگی دیگر این ماژول ارتباط آن با ماژول مدیر نصب است به این صورت که وقتی کاربر می‌خواهد لیست نرم افزارهای قابل نصب را مشاهده کند، ماژول مدیر نصب درخواست خود را به مدیر بسته نرم افزاری ارسال می‌کند و این ماژول لیست نرم افزارهای کاربر را در قالب یک فایل JSON ارسال می‌کند.

۲- مدیر وضعیت: این ماژول بر وضعیت تلفن همراه نظارت دارد. گردآورنده وضعیت در لایه دستگاه، اطلاعات CPU و میزان استفاده از حافظه را جمع آوری کرده و اطلاعات خود را به این ماژول ارسال می‌کند.

۳- زمان بند وظایف: وقتی که کاربر از راه دور به تلفن هوشمند متصل شد یک جلسه در ماژول زمان بند وظیفه ایجاد می‌گردد. زمانی که ارتباط کاربر قطع یا زمان جلسه از حد مجاز بگذرد، دستور قطع ارتباط را ارسال می‌کند.

۴- دروازه تعویض: برای اتصال از راه دور به دستگاه هوشمند نیاز به آدرس و شماره پورت داریم. برای هر دستگاه هوشمند یک پورت VNC از شماره ۵۹۰۱ به صورت یکتا در نظر گرفته شده است و آدرس عمومی بر روی سرور برنامه قرار دارد. وقتی VNC Viewer درخواست خود را برای سرور ارسال می‌کند این ماژول ابتدا پورت اتصال را بازیابی و از ماژول زمان بند وظایف، در مورد وضعیت دستگاه مورد نظر سؤال و چنانچه دستگاه آزاد بود ارتباط برقرار می‌شود [۶].

۳.۴. لایه عملیاتی

در این لایه، فعالیت کاربران مدیریت می‌شود. این محیط دو گروه از کاربران را پشتیبانی می‌کند: کاربران و مدیران سیستم. کاربران با ورود به محیط کاربری خود می‌توانند لیست دستگاه‌های خود را مشاهده کنند و به گسترش نرم افزارهای خود بپردازند. مدیران برای مدیریت محیط ابری و تعمیر و نگه داری وارد ناحیه کاربری می‌شوند، مدیر سیستم می‌تواند تمامی تلفن همراهها را راه اندازی مجدد کنند و در صورت نیاز آنها را فعال و غیرفعال کند [۶].

۵. ویژگی‌های کلیدی این سرویس

در این محیط از گوشی‌های تلفن واقعی استفاده می‌شود و یک امتیاز ویژه برای برنامه نویسان است و تجربه آزمایش در گوشی واقعی را به برنامه نویس می‌دهد. در این سرویس وجود انواع تلفن همراه و به روز بودن دستگاه‌ها یک ضرورت است. از دیگر ویژگی‌های کلیدی این سرویس توانایی استفاده از تلفن همراه‌های توزیع شده است و تلفن همراهها می‌توانند در مراکز مختلفی مستقر باشند و از طریق سرور اصلی در اختیار کاربران قرار گیرند [۱].

با توجه به اینکه محیط ابری در بین کاربران یک محیط ناامن تلقی می‌گردد، این سرویس از فایل APK جهت اجرای آزمون‌های خود استفاده می‌کند و برنامه نویس از بابت از دست رفتن کد برنامه خود هیچ نگرانی ندارد. این آزمایش به عنوان آزمون جعبه سیاه شناخته می‌شود و یک ویژگی مثبت این سرویس است [۲].

مرکز تجربه اپل (ACE) سخت افزار و محصولات خود را جهت تمرین در این مرکز به نمایش می‌گذارد. محیط



ابری آزمایش تلفن همراه، می‌تواند امکان نمایش از راه دور دستگاه‌های تلفن همراه را فراهم کند. کاربران در هر لحظه می‌توانند ویژگی‌های جدید محصولات تلفن همراه را در مرورگر وب تجربه کنند. این ویژگی می‌تواند میلیون‌ها دلار صرفه جویی در زمینه برگزاری نمایشگاه‌ها به عمل آورد [۱].

استفاده از رابط کاربری تحت وب و همچنین استفاده از زبان XML، جهت طراحی آزمون خودکار توسط کاربر، قابلیت مثبتی در جهت استفاده‌ی راحت‌تر کاربران از این سرویس است.

۶. بررسی مشکلات و راهکارهای موجود

همان‌طور که در بخش‌های قبل توضیح دادیم در این محیط از دستگاه‌های واقعی استفاده می‌شود و این روند مشکلاتی را به همراه دارد. وقتی تلفن همراه توسط کاربر در حال استفاده است سرور برای جلوگیری از تداخل، استفاده آن از lock می‌کند و پس از اتمام آزمون آن را آزاد می‌کند. برای حل این مشکل می‌توان از یک زمان‌بند استفاده کرد و کاربران آزمون‌های خود را در فایل XML ذخیره کنند و بر روی سرور قرار دهند و سرور طبق زمان‌بندی به ترتیب آزمون‌ها را اجرا و نتایج را در پروفایل کاربر قرار دهد. این روش انتظار پاسخ آزمایش، برای بعضی از کاربران مطلوب نخواهد بود. همچنین در زمانی که کاربران این سرویس افزایش پیدا کنند، زمان‌بند سرور قادر به پاسخگویی به نیاز کاربران نیست.

مشکل بعدی، نمایش نتایج برای کاربر است. برای نمایش نتایج به صورت گرافیکی کاربر نیاز دارد صفحه نمایش تلفن همراه را ببیند که دو روش استفاده از فناوری ajax به همراه پلاگین‌های jquery و Capture کردن تصاویر و ارسال آن در وب است و تصویر به صورت متناوب در بازه‌های زمانی به روزرسانی می‌شود. این عمل با مشکلاتی همچون مصرف ترافیک شبکه و تأخیر در ارسال مواجه است.

روش دوم استفاده از سرور VNC، یک سیستم تعریف شده برای انتقال محتویات کامپیوتر با استفاده از پروتکل RFB (RemoteFrameBuffer) است و تصاویر توسط VNC Viewer مشاهده می‌شود. این روش نیازمند اضافه کردن یک سرور و ارتباط با گوشی تلفن همراه است و برای مشاهده حتماً نیاز به نصب نسخه کلاینت VNC Viewer است.

۷. محبوب‌ترین محیط‌های ابری آزمایش نرم افزار موبایل

قصد داریم تعدادی از پروژه‌های اجرا شده از این مدل را معرفی و بررسی کوتاهی در مورد آن‌ها داشته باشیم. Xamarin یکی از محبوب‌ترین ابزارهای آزمایش آنلاین نرم افزار موبایل است که انواع مختلف عملیات لمسی همانند کشیدن، ضربه، دو بار ضربه زدن، فشار قلم را پشتیبانی می‌کند. این ابزار با امکان سفارشی سازی عملیات لمسی به دلخواه کاربر می‌تواند هرگونه حرکات لمسی را ایجاد و در دستگاه‌ها اجرا نماید. قابلیت تغییر موقعیت مکانی GPS و پشتیبانی کامل کلید فیزیکی (Home, Power, Volume) و دوربین دستگاه تمامی امکانات لازم را در اختیار کاربر قرار می‌دهد. نرم افزارهای معروف Foursquare, EBay, Flip board برای آزمایش نرم افزارهای خود از این سرویس استفاده کرده‌اند. این ابزار پلن رایگان ندارد و برای استفاده از این سرویس باید مبلغی بین ۱۰۰۰ الی ۱۲۰۰۰ هزار دلار با توجه به نوع خدماتی را نیاز دارید پرداخت کنید [۷].

Keynote از تنوع دستگاه‌های تلفن همراه بسیار زیادی بهره می‌برد و قابلیت افزودن دستگاه‌های سفارشی مانند سامانه‌های هوشمند خودرویی و یا هر دستگاه سفارشی دیگر را دارد. برای ارتباط شبکه مخابراتی خود از معروف‌ترین شبکه‌ها استفاده می‌کند و در تمامی مرورگرهای وب قابل اجرا است. قابلیت ضبط جلسه آزمون برای اجرای خودکار آن



در دستگاه‌های دیگر از ویژگی‌های مثبت این سرویس است. تمرکز این سرویس برای اجرای خودکار آزمون‌ها است. شرکت‌های Humana, BBVA, Aricent از معروف‌ترین مشتریان این سرویس می‌باشند. این سرویس دارای پلن رایگان با امکانات محدود است و برای استفاده از پلن‌های حرفه‌ای باید ۱۸۰ الی ۷۵۰ دلار ماهیانه پرداخت کنید [۸].

شرکت Perfecto Mobile یک سرویس آزمایش ابری نرم‌افزار موبایل برای کارهای تیمی برنامه‌نویسان ارائه داده است که از ویژگی‌های آن می‌توان به اختصاص یک دستگاه تلفن همراه به یک گروه کاری، اشتراک نتایج، آزمون‌ها و قابلیت کنترل دسترسی کاربران با استفاده از Active Directory یا LDAP اشاره کرد [۹].

بازار آزمایش آنلاین شرکت SOASTA علاوه بر امکانات عمومی، سرویس آزمایش چند لمسی را به صورت ویژه ارائه داده است و همچنین قابلیت استفاده از حسگرهای نوری و دور-نزدیک از ویژگی‌های منحصر به فرد این سرویس است [۱۰].

پروژه Sauce LABS علاوه بر پشتیبانی از آزمایش نرم‌افزارها به صورت فایل APK، سرویس ویژه‌ای برای آزمایش کد برنامه ارائه داده است که توسط آن می‌تواند از یک Debugger آنلاین برای برطرف نمودن مشکلات برنامه در حین اجرا بهره‌مند شد [۱۱].

۸. جمع بندی و نتیجه گیری

اهمیت وجود یک محیط واقعی برای آزمایش نرم افزار تلفن همراه یک ضرورت است. این مقاله با معرفی محیط ابری و انواع آزمایش‌های قابل اجرا، نشان داد که این محیط قادر است یک نرم افزار را در یک شرایط واقعی از لحاظ میزان کارایی و عملکرد، درصد سازگاری، رابط کاربری و امنیتی مورد آزمایش قرار دهد.

در بخش بعدی به شرح ساختار سه لایه‌ای آن، و چگونگی مدیریت ارتباط کاربر با دستگاه را پرداخته است. از ویژگی‌های مثبت این بستر می‌توان به واقعی بودن محیط آزمایش، قابلیت در دسترس بودن از هر نقطه، استفاده از فایل apk برای تضمین امنیت کد برنامه و رابط کاربری تحت وب آن اشاره کرد. این محیط دارای نقاط ضعف همچون هزینه پیاده سازی بالا، عدم امکان استفاده همزمان کاربران از یک دستگاه است.

در ادامه، تعدادی از محبوب‌ترین پروژه‌های موجود به همراه امکانات آن را معرفی کردیم و مشاهده نمودیم، معروف‌ترین نرم افزارهای تلفن همراه برای بهبود عملکرد و آزمایش نرم افزارهایشان، به استفاده از این سرویس پرداخته‌اند.

همان‌طور که ملاحظه کردید این محیط‌ها برای پیاده سازی و سرویس دهی از دستگاه‌های واقعی تلفن همراه استفاده می‌کنند، پیاده سازی این سرویس پر هزینه بوده و کاربران برای استفاده از این سرویس‌ها باید به صورت ماهانه هزینه‌هایی پرداخت کنند.

در حال حاضر محققان در حال تلاش برای بهبود این محیط، با ترکیبی از شبیه سازها و فناوری مجازی سازی هستند. عدم دسترسی به داده‌های واقعی حسگر در شبیه سازها توسط روش مجازی سازی حسگرها [۱۲] بر طرف شده است و امید آن است که با پیشرفت این روش یک محیط شبیه سازی شده با تمامی امکانات یک دستگاه واقعی را شاهد باشیم تا برنامه نویسان و کاربران با خیال آسوده به تولید نرم افزار و استفاده از آن‌ها بپردازند. امیدوارم این مقاله برای شما مفید قرار گرفته باشد و زمینه ساز تحقیقات بعدی شما باشد.



۹. منابع و مراجع

- [1] J.-F.Huang, Y.-Z.Gong "Remote mobile test system: a mobile phone cloud for application testing" IEEE 4th International Conference. Cloud Computing Technology and Science (CloudCom), 1-4, 2012.
- [2] C.Mano Prathibhan, A.Malini, N.Venkatesh, K.Sundarakantham, "An Automated Testing Framework for Testing Android Mobile Applications in the Cloud" IEEE International Conference. Advanced Communication Control and Computing Technologies (ICACCCT), 1216-1219, 2014.
- [3] http://en.wikipedia.org/wiki/Functional_testing
- [4] A.Malini, N.Venkatesh, Dr.K.Sundarakantham, Dr.S.Mercyshalinie "Mobile Application Testing on Smart Devices Using MTAAS Framework in Cloud" IEEE International Conference. Computer and Communications Technologies (ICCCT), 1-5, 2014.
- [5] S.Malek, N.Esfahani, T.Kacem, R.Mahmood, N.Mirzaei, A.Stavrou "A Framework for Automated Security Testing of Android Applications on the Cloud" IEEE Sixth International Conference. Software Security and Reliability Companion (SERE-C), 35-39, 2012.
- [6] Y.liu, Y.Li, J.Niu, Q.Cao "An online physical simulation platform for Android programming" IEEE Sixth International Conference. Technology for Education (T4E), 26-27, 2014.
- [7] www.xamarin.com/test-cloud.
- [8] www.keynote.com/solutions/testing/mobile-testing.
- [9] www.perfectomobile.com.
- [10] www.soasta.com/products/touchtest.
- [11] www.saucelabs.com/mobile.
- [12] J.-H.Jeong, S.-J.Hur "Sensor Device Virtualization for Mobile Clouds" IEEE International Conference. Consumer Electronics (ICCE), 288-289, 2014.