

Tesi di laurea

Testing di applicazioni Android con tecniche di Capture & Replay

Anno Accademico 2020/2021

Relatore

Ch.mo prof. Porfirio Tramontana

Candidato

Claudia Orlando

Matr. M54/694

Introduzione

- Le piattaforme mobili stanno diventando sempre più diffuse e, conseguentemente, le applicazioni mobili (o semplicemente le app) che funzionano su tali piattaforme.
- Le app devono essere testate per assicurarsi che si comportino correttamente ad input e condizioni diverse. Ciò è il motivo fondamentale per il quale numerose aziende spendono considerevoli quantità di denaro e di risorse per la cosiddetta attività di Quality Assurance (QA), ed, in particolar modo, per l'attività di testing.
- Nel caso delle applicazioni Android, il panorama è ulteriormente complicato a causa della frammentazione dell'ecosistema Android.
- Ottenere, quindi, la sicurezza di un'app che funzioni correttamente su tutta la gamma di dispositivi Android e la versione del sistema operativo corrispondente è particolarmente impegnativo ed oneroso.



Outline

- Introduzione agli strumenti di Capture & Replay
- Appium: punti di forza e punti di debolezza
- Casi di studio: introduzione e domande poste
- Risultati numerici
- Risposta alle domande e discussioni/osservazioni
- Conclusioni



Introduzione agli strumenti di Capture & Replay

- I tool di Capture & Replay consentono a tester senza competenze di programmazione e di testing avanzate, di realizzare degli script di test da sequenze di interazione dell'utente con l'AUT (Application Under Test), con inserimenti da tastiera o movimenti del mouse, acquisendo, in questo modo, scenari di utilizzo reali. A partire da questo script, il tool riproduce sul programma, o successive versioni del medesimo, le operazioni registrate senza richiedere l'intervento attivo di un utente. Ciò, ovviamente, riduce l'effort che si avrebbe con l'esecuzione manuale dei test.
- La nostra analisi è stata quindi orientata allo studio dei tool di Capture & Replay, attualmente in commercio, esaminando, per ognuno di essi, le caratteristiche ed il loro scenario di utilizzo in modo da stabilire quale possa essere il tool di nostro utilizzo. La discriminante sulla quale si è basata la nostra scelta è stata l'assenza di codice sorgente e, quindi, la sola presenza dell'APK.

	Tool	Tecnologia Target	Disponibilità	Costo	APK/source code	Script Rilegibili
1	Sikuli	Visual Tool	✓	Open Source	APK	✗
2	EyeAutomate	Visual Tool	✓	Open Source	APK	✗
3	RANDR	C&R Tool	✗	A pagamento	Source code	✓
4	QDBR	Tool di debugging	✗	Open Source	APK	✗
5	RERAN	C&R Tool	✗	Open Source	APK	✗
6	APPIUM	C&R Tool	✓	Open Source	APK	✓
7	ROBOTIUM	C&R Tool	Disponibile per vecchie versioni di Android	primi 10 test sono gratuiti	APK	✓
8	SELENDROID	C&R Tool	In disuso	Open Source	APK	✓
9	Espresso Test Recorder	C&R Tool	✓	Open Source	Source code	✓

- A valle della nostra analisi, come si può ben osservare dalla tabella riportata che rappresenta la classificazione dei tool esaminati in base ai criteri target di nostro interesse, la scelta del tool è ricaduta su Appium.



Appium: Punti di forza

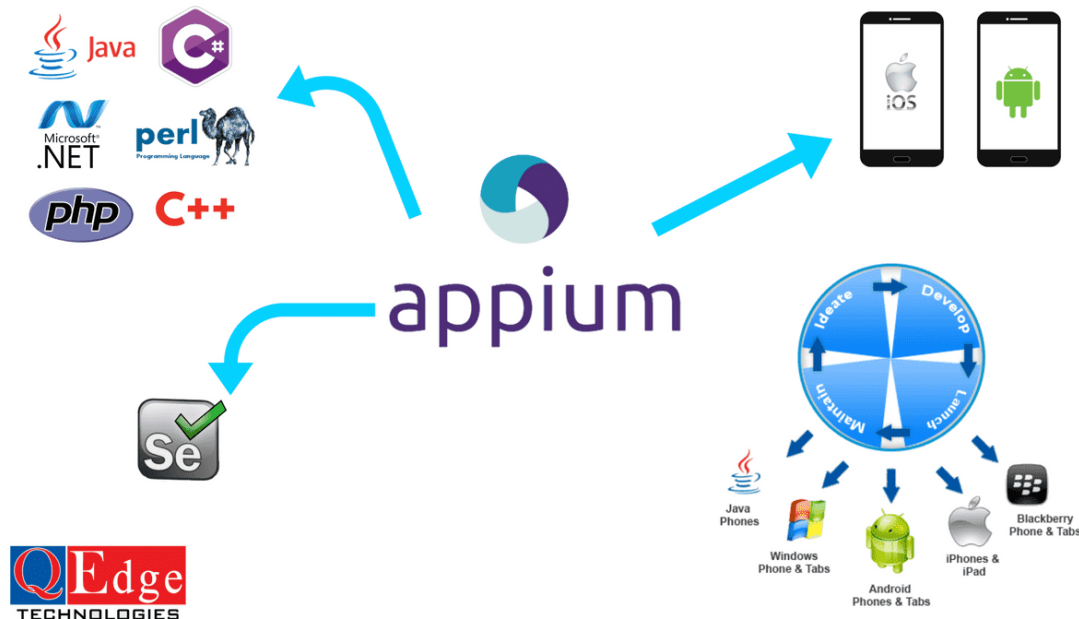
- Open Source
- Semplice ed intuitivo, grazie alla presenza di numerosi tutorial di supporto
- Non richiede una conoscenza di programmazione approfondita
- Consente di automatizzare test per applicazioni mobile sia su piattaforma Android che iOS riducendo l'effort che si avrebbe con l'esecuzione manuale
- Replay in un IDE di propria scelta utilizzando qualsiasi linguaggio compatibile con il WebDriver come Java, Python, C, ecc.
- Non è necessario il source code: è sufficiente l'APK
- Non è necessario ricompilare o modificare l'app grazie all'utilizzo di API di automazione standard su tutte le piattaforme.



Appium: Punti di debolezza



- Alcune versioni di Appium possono non risultare compatibili con le versioni OS di Android ed iOS
- Alcune feature non sono presenti (come lo scroll). Bisogna, quindi, rinunciare parzialmente alla fase di Replay, aggiungendo stringhe di codice che dovranno essere istanziate ogni volta



Casi di studio: Introduzione

Le applicazioni oggetto della nostra analisi sono:

- Munchlife



App contatore che consente di tenere traccia del livello del personaggio mentre si gioca a carte.

- Trolly



Applicazione open source per la lista della spesa

- DIAB



App open source che aiuta a gestire il diabete tenendo traccia dei valori di glucosio e delle iniezioni di insulina

- App per veicoli commerciali



App che consente al driver di controllare dal device le funzionalità dell'abitacolo e del veicolo

Casi di studio: domande poste

Le domande, oggetto della nostra analisi, sono le seguenti:

- **Quanto spesso i test di Appium sono rieseguibili?**
- **Quanto spesso i test di Appium sono replicabili e quali sono le cause di mancata replicabilità?**
- **Quali sono le limitazioni che incontra il tester nell'utilizzo di Appium?**
- **Quali sono i vantaggi di Appium rispetto ad altri strumenti simili?**
- **Quanto Appium è adatto a risolvere i problemi di testing di applicazioni Android in ambito industriale?**



Risultati numerici

		WVGA (Nexus S) API 26	WVGA (Nexus S) API 27	WVGA (Nexus S) API 28	Nexus 10 API 27	Nexus 10 API 24	WVGA (Nexus S) API 23	Galaxy Nexus API 23	Galaxy Nexus API 23	Totale Test Eseguiti
Munchlife	Verifica della rieseguibilità dei test registrati	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	48/48
	Portrait/Landscape	0/6	N.A.	N.A.	6/6	N.A.	0/6	N.A.	N.A.	6/18
	Language	6/6	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	6/6
Trolly	Verifica della rieseguibilità dei test registrati	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	48/48
	Portrait/Landscape	6/6	N.A.	N.A.	N.A.	6/6	N.A.	N.A.	N.A.	12/12
	Language	0/6	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	0/6
DIAB	Verifica della rieseguibilità dei test registrati	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	48/48
	Portrait/Landscape	0/6	N.A.	N.A.	N.A.	N.A.	0/6	N.A.	N.A.	0/12
	Language	0/6	N.A.	N.A.	N.A.	N.A.	0/6	N.A.	N.A.	0/12
Applicazione per veicoli commerciali	Verifica della rieseguibilità dei test registrati	6/6	6/6	6/6	6/6	6/6	6/6	6/6	6/6	48/48
	Portrait/Landscape	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
	Language	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.

La tabella riporta sulle righe le applicazioni, oggetto del nostro studio, e sulle colonne le caratteristiche degli emulatori sui quali sono stati registrati i casi di test.

Risposte alle domande poste (1/3)

1

- **Quanto spesso i test di Appium sono rieseguibili?**

La rieseguibilità dei test, nelle stesse condizioni della registrazione, dà sempre esito positivo (24/24).

2

- **Quanto spesso i test di Appium sono replicabili e quali sono le cause di mancata replicabilità?**

Le cause di mancata replicabilità, in condizioni diverse dalle quali è stato registrato il test, sono essenzialmente riconducibili a due tipologie di test: Portrait/Landscape (18/42) e Language (6/24). Il primo è legato al fatto che il layout portrait ha delle dimensioni ben diverse da quello landscape, mentre il secondo è legato al nome differente degli id, caratterizzanti gli oggetti dell'app, che cambiano a seconda della lingua settata nelle impostazioni del device.

Risposte alle domande poste (2/3)

3

- **Quali sono le limitazioni che incontra il tester nell'utilizzo di Appium?**
In Appium sono assenti alcune feature, come quella dello scroll. In tal caso, si ritiene che il test non sia eseguito correttamente in quanto si rinuncia parzialmente alla fase di Capture e si inseriscono manualmente stringhe di codice che simulino la feature desiderata (6/6).

4

- **Quali sono i vantaggi di Appium rispetto ad altri strumenti simili?**
Esso è un tool open source che si presta bene all'automatizzazione di procedure di test per applicazioni mobile sia su piattaforma Android che iOS. Non è necessario ricompilare o modificare l'AUT, grazie all'utilizzo di API di automazione standard su tutte le piattaforme. Infine, non richiede la presenza del codice sorgente

Risposte alle domande poste (3/3)

5

- **Quanto Appium è adatto a risolvere problemi i problemi di testing di applicazioni Android in ambito industriale?**

Le app in ambito industriale hanno feature più complesse, come quella dello scroll ma anche la procedura di login. Il fallimento di quest'ultima rappresenta un limite intrinseco di questa tipologia di tecniche e non è una problematica legata alla scelta del tool. Invece, il problema legato alle feature mancanti può essere ovviato con l'aggiunta di stringhe di codice che simulino il comportamento desiderato. A meno di questi casi, Appium si presta bene al testing di applicazioni Android in ambito industriale.

Conclusioni

- Appium è un tool open source che consente di automatizzare l'attività di testing su app sia su piattaforma Android che su piattaforma iOS, riducendo l'effort che si avrebbe con l'esecuzione manuale.
- Non è necessaria la presenza del codice sorgente.
- Non richiede una conoscenza di programmazione e di testing avanzate.
- Non essendo stato sviluppato da Google, potrebbe essere indietro rispetto ad alcuni aggiornamenti.
- Alcune feature non sono implementate. In tal caso, è necessario rinunciare parzialmente alla fase di Capture ed inserire manualmente stringhe di codice che simulino la feature mancante.
- Si presta bene ad attività di testing su applicazioni Android in ambito industriale.