Università Ca'Foscari Venezia

Dipartimento di Scienze Ambientali, Informatica e Statistica

Piano di testing

Ingegneria del Software 2017/18

Docente del corso: A. Cortesi
Tutore del corso: A. Spanò
Data di consegna: 31/01/2018
V. 2.0

TEAM : Electrici Team
Corazza Sara 857682
Mutterle Ilaria 860720
Rigon Daniele 857319
Ruffato Denny 859171

Indice

Capitolo 1

Introduzione

Lo scopo di questo documento è quello di fornire informazioni in merito alle attività di testing della nostra applicazione: descrivere il processo che verrà messo in pratica per trovare eventuali errori, come questi verranno valutati e registrati, quali vincoli vanno rispettati e quali sono i requisiti richiesti.

L'obiettivo di questa fase è quello di trovare gli errori con il minimo numero di casi di prova e di stabilire la gerarchia dei test da eseguire, i metodi di esecuzione, i criteri di accettazione.
La scelta condivisa è quella di affrontare questa fase con un approccio bottom-up: verificare man mano i singoli progressi dell'applicativo per raggiungere un test complessivo del progetto finale.

In una seconda fase, procederemo nel testare i diversi percorsi di utilizzo degli strumenti forniti dall'applicazione.

Capitolo 2

Strategia di testing

La nostra strategia di testing si divide in tre fasi:

- **Unit testing**: Questo approccio serve per scoprire subito se ci sono errori nelle singole unità. Risulta più semplice trovare errori e ipotizzarne la natura: se sono circoscritti a una piccola porzione e non fanno già parte di tutto il progetto. Inoltre, se svolto correttamente, limita gli errori a problemi di interazione fra le singole unità, rendendo più semplice la seconda fase, ovvero l'*incremental testing*.

- **Incremental testing**: In questa fase, a poco a poco, faremo collaborare i vari moduli, testandoli prima di integrare quelli che seguono. Questa tecnica permette una tempestiva individuazione degli errori perchè, aggiungendo i moduli singolarmente, è facile individuare quale sia quello che causa malfunzionamenti. Se si decidesse di testare il progetto completo, sarebbe molto più complesso e costoso individuare e correggere malfunzionamenti, in caso ce ne fossero.

- **Stress testing**: In questa fase, controlleremo il comportamento dell'applicazione in casi estremi come la mancanza di connessione Internet o di segnale GPS. L'idea è che se si verifica uno di questi problemi, l'applicazione dovrà continuare a funzionare, visualizzando a schermo un errore che informa l'utente. Vi saranno inoltre due casi limite:

  - il campo di ricerca dei distributori è troppo ampio;
  - il campo di ricerca dei distributori non dà alcun risultato.

  Nel primo caso, per evitare rallentamenti nell'esecuzione dell'applicazione, durante il test si provvederà ad aumentare il raggio di ricerca dei distributori fino a quando l'applicazione non subirà dei rallentamenti notevoli

per quanto riguarda il tempo di risposta. Raggiunto il raggio massimo di ricerca, verranno modificate le impostazioni di base dell'applicazione per fare in modo che l'utente non superi un certo raggio di ricerca, o comunque vengano visualizzati dei messaggi di avvertimento o di errore. Nel secondo caso, invece, all'utente verrà mostrato un messaggio per avvertirlo che non è stato trovato alcun distributore in zona e sarà consigliato di aumentare il raggio di ricerca, per cercare di ricevere una quantità maggiore di risultati.

Capitolo 3

Tracciabilità dei requisiti

I requisiti che verranno testati e il giudizio che verrà dato a test eseguito sono riportati nella tabella seguente:

Nome del requisito	**Criterio di valutazione del test**
Visualizzazione mappa	Valutazione positiva: la mappa viene visualizzata e non causa problemi durante l'interazione dell'utente con essa (tocco e trascinamento).
Scelta del carburante	Valutazione positiva: sulla mappa vengono visualizzati tutti e soli i carburanti spuntati dall'utente.
Scelta di un itinerario	Valutazione positiva: visibile l'itinerario corretto sulla base dei dati immessi dall'utente.
Ricerca posizione	Valutazione positiva: visualizzata con successo la posizione richiesta dall'utente.
Navigazione veloce	Valutazione positiva: visibile l'itinerario corretto che va dalla posizione dell'utente a quella indicata (scelta) dall'utente.

Capitolo 4

Elementi testati

Gli elementi testati nella fase di *unit testing* sono le singole classi e, ancora più nello specifico, i singoli metodi che svilupperemo.

Nella fase di *incremental testing* le varie unità che sono analizzate nello *unit testing* vengono fatte collaborare e si testeranno piccoli gruppi di esse. Continueremo fino al raggiungimento di una forma dell'applicazione comparabile a quella che verrà fornita all' utente; inizialmente si tratterà di un collaudo automatizzato e successivamente manuale.

Nella fase di *stress testing* verrà testato il comportamento dell'applicazione in mancanza di alcuni requisiti come ad esempio la connessione dati o il GPS.

Capitolo 5

Tempo e risorse allocate

Il modus operandi scelto per procedere col progetto richiederà una fase continua di prove durante le fasi iniziali, al fine di trovare quanti più bug possibili allo scopo di migliorare l'applicazione.

Successivamente si procederà ad una fase di ottimizzazione per rendere l'applicazione più stabile e performante possibile, compatibilmente con le capacità di ognuno dei membri del team.

Capitolo 6

Procedura di registrazione dei test

I test che eseguiremo verranno poi riportati in una tabella di cui forniamo un esempio di seguito:

Data	**Descrizione**	**Elementi coinvolti**	**Risultato**

Nello specifico la **descrizione del test** indica cosa ci si aspetta che succeda in mancanza di malfunzionamenti, gli **elementi interessati** sono i singoli moduli se saremo nella fase di *unit testing* o un insieme di essi nella fase di *incremental testing* e infine il **risultato** darà informazioni sul successo o fallimento e una rapida descrizione del problema riscontrato.

Capitolo 7

Requisiti hardware e software

I requisiti fondamentali sono:

- Sistema operativo Android versione 5.1 o successivi
- Connessione a Internet
- Geolocalizzazione

I dispositivi che useremo per i test sono:

- **ASUS Zenfone max** Specifiche: Android Lollipop 5.0 RAM 2 GB
- **LG G5 SE** Specifiche: Android Nougat 7.0 RAM 3 GB
- **WIKO Lenny 3** Specifiche: Android Marshmallow 6.0 RAM 1 GB

Ulteriori dispositivi utilizzati dai tester verranno eventualmente inseriti nei documenti relativi alla documentazione dei risultati di *testing*.

Capitolo 8

Vincoli per il testing

Un testing esaustivo richiederebbe di effettuare prove su tutti i dispositivi Android in modo da verificare il corretto funzionamento su hardware differenti. Siamo però limitati ad utilizzare i nostri dispositivi ed eventualmente usare emulatori per simularne alcuni altri, ma ovviamente non tutti.

I tempi di consegna del progetto implicano un tempo limitato e quindi il *testing* andrà effettuato nel migliore dei modi ma entro i tempi prestabiliti.

I test e tutti i risultati verranno registrati sulla tabella descritta nel paragrafo ***Procedura di registrazione dei test***.

Per quanto concerne, invece, la data di inizio del *testing*, considerando la strategia da noi scelta, sarà poco dopo la data di inizio dello sviluppo della applicazione, non appena i primi moduli/unità saranno pronti per lo *unit testing*.