15/11/2020

Ingegneria del Software

Piano di Testing

RAMMA

Michele Lotto:	875922
Andrea Chinellato:	875422
Alessandro Furlan:	854723
Michele Gatto:	875446
Riccardo Zuliani:	875532

Indice

1. Introduzione

Il presente documento ha lo scopo di esemplificare e raccogliere tutte le attività e le strategie necessarie all'attività di testing del Progetto Rent-sciò.

Avendo adottato un modello di tipo evolutivo che prevede una prototipazione pianificata e una fase di testing alla fine di ogni singola creazione di un prototipo, saranno svolte le seguenti attività:

- **Processo di testing adottato**: In questo processo verranno mostrate le tecniche e le strategie utili per una fase di testing coordinate ed organizzata.

- **Modalità di verifica dei singoli requisiti**: Questa sezione risulta essere fondamentale, in quanto denota le differenze fra l'effettivo sviluppo dell'applicativo e le idee pensate e trascritte nel Documento di Specifica dei Requisiti. Sarà dunque importante apportare e valutare le modifiche necessarie per poter rimanere coerenti con i requisiti da noi stabiliti per questo primo prototipo.

- **Elementi testati**: Ogni caso d'uso progettato e sviluppato va testato nei minimi particolari. Per questo motivo occorrerà creare diverse sezioni in cui si andranno a testare i singoli elementi e i singoli casi di utilizzo dell'utente per poter rendere l'applicazione il più stabile e libera da bug possibile.

- **Schedulazione del testing**: Il documento rappresenterà visualmente tramite tabella l'attività di testing dalla prima schermata fino all'ultima attività disponibile contenuta nell'applicativo.

- **Procedure di registrazione dei test**: In questa sezione verranno raccolti tutti i dati relativi agli esiti dei test effettuati. Il primo prototipo raccoglierà gli esiti dei test effettuati in base alle specifiche relative al Documento di Specifica dei Requisiti.

- **Requisiti di hardware e software utilizzati**: Saranno indicati i requisiti minimi richiesti per poter eseguire l'applicazione sia dal punto di vista hardware che dal punto di vista software.

- **Vincoli per il testing**: Saranno fissati diversi obiettivi da raggiungere affinché l'applicazione possa ritenersi stabile e valida e la fase di testing conclusa con successo.

2. Processo di testing

Il gruppo ha deciso di adottare una tecnica di testing basato sull'approccio bottom-up. Questo approccio consisterà nel collaudare tutti i moduli di basso livello, procedendo ad analizzare ogni singolo modulo testandone la sua correttezza. Poi si procederà ai moduli più ad alto livello andando a testare per ultima l'interfaccia grafica dopo aver testato la correttezza del codice scritto.

Abbiamo scelto questo tipo di strategia perché ci è risultata più semplice e intuitiva da realizzare, al contrario dell'approccio Top-Down.

La correttezza del codice verrà messa in primo piano dagli sviluppatori e responsabili dello stesso. Ma ciò non risulta essere sufficiente appieno per poter testare il completo funzionamento dell'applicativo, in quanto possono generarsi errori derivanti da altri aspetti esterni al codice (es. Bug interfaccia grafica, guasto del dispositivo ecc.).

Il gruppo ha deciso di utilizzare le seguenti strategie:

- **THREAD TESTING**, per definizione questa strategia è adatta per sistemi real-time (nel nostro caso l'utilizzo delle API di Google Maps) verrà usata per le operazioni più complesse e lunghe come la connessione continuativa fra dispositivi.

Nel nostro caso il commerciante ha la necessità di comunicare in modalità peer-to-peer in maniera continuativa con il cliente, ricevendo informazioni come posizione e notifiche relative al noleggio.

Per il primo prototipo vi sarà solamente un commerciante ed un cliente, ma la questione diverrà più rilevante nel secondo prototipo in cui si passerà a più commercianti che comunicano con clienti diversi.

- **INCREMENTAL TESTING**, strategia che verrà adottata dall'inizio alla fine del progetto per tenere sotto controllo le funzioni implementate e l'integrazione tra di loro.

Questa strategia risulta essere la migliore per tener traccia in maniera corretta ed organizzata delle funzionalità che il prototipo assume durante il suo sviluppo.

Quindi verrà usato per le attività principali di sincronizzazione dei dati, comunicazione e interazione tra cliente e commerciante.

È una strategia che ci permetterà di controllare e caricare ogni volta il sistema con nuove funzionalità per controllare il loro corretto funzionamento.

Un esempio del vantaggio di questa strategia è per esempio l'inutilità di poter avviare una nuova corsa, se poi non è possibile effettuare una nuova registrazione all'interno del database.

3. Tracciabilità dei requisiti

I requisiti funzionali del sistema verranno verificati seguendo le specifiche dei requisiti indicate nel **DOCUMENTO DI ANALISI E SPECIFICA**.

Qui sotto riportiamo un esempio dei requisiti funzionali e possibili test da effettuare.

Nelle fasi successive saranno creati ulteriori casi e scenari di test utili per migliorare e trovare eventuali falle nel sistema.

ID Requisito	Nome Requisito	Test	Esito
CM1	Visualizzare mappa e dettagli	Avviare l'app e eseguire il login con un account commerciante.	Positivo, se il commerciante riesce a vedere la mappa, completa di corse attive e posizione dinamica dei clienti.
CM2	Aggiungere e rimuovere veicoli.	Dalla schermata di visualizzazione parco mezzi; aggiungere diversi veicoli con numeri diversi di posti (numeri positivi e negativi) e successivamente rimuoverli.	Positivo, se il commerciante riesce a visualizzare nel parco mezzi i veicoli aggiunti correttamente con un numero di posti realistico.
CM3	Creare corsa	Dalla schermata principale selezionare Nuova corsa. Selezionare un veicolo e controllare che sia disponibile. Una volta completata la procedura del cliente, controllare che la nuova corsa appaia sulla schermata principale del commerciante e che il collegamento sia attivo.	Positivo se, al tap del pulsante, il commerciante avrà generato un nuovo QRcode per connettersi con il cliente e far partire dunque la corsa.
CM4	Rimuovere corsa	Dalla schermata principale selezionare "Rimuovi corsa" nella visualizzazione delle corse attive.	Positivo, se dopo il tap sul pulsante rimuovi corsa, il sistema effettua correttamente l'aggiornamento della corsa terminandola.

CM5	Impostare area limitata	Selezionare nella mappa diversi punti per poi costruire l'area limitata. Provare a costruirne diverse e/o di aggiornare l'esistente area.	Positivo, se nella mappa compare l'area selezionata.
CM6	Visualizzare parco mezzi	Dalla schermata principale selezionare "Visualizza parco mezzi".	Positivo se, al tap del pulsante verrà mostrato il parco mezzi con la possibilità di aggiungere o eliminare un veicolo e di poter controllare la disponibilità dei suoi mezzi.
CM7	Notifica eventi (Commerciante)	Durante la corsa, chiedere al tester del cliente di eseguire delle azioni non consentite, quali uscire dall'area limitata e superare il limite di velocità. Questi eventi verranno notificati al Commerciante.	Positivo, se dopo l'evento il sistema recepisce in un tempo accettabile l'evento e lo comunica al commerciante.
CL1	Attivare corsa	Dalla schermata principale, selezionare "Attiva corsa" e scansionare il QR creato dal commerciante.	Positivo, se dopo la scansione del QR viene visualizzata la mappa con la possibilità di attivare le funzioni specifiche della corsa per il cliente.
CL2	Visualizzare mappa e posizione	Entrare nella pagina principale in modalità cliente.	Positivo, se il cliente riesce a vedere la propria schermata.
CL3	Segnalare problema	Tramite l'apposito pulsante presente nella schermata principale. Il cliente può contattare telefonicamente il commerciante.	Positivo, se il cliente riesce ad eseguire una chiamata verso il commerciante, ed esso riceve la chiamata.
CL4	Riconnettere la corsa attiva	Avviare app e eseguire il login. Il collegamento con il commerciante sarà attivo se appariranno le opzioni di corsa per il cliente.	Positivo se, il cliente, una volta interrotta la sua corsa per qualsiasi motivo, riesce a riconnettersi al database e all'applicazione, proseguendo la sua corsa.

CL5	Notifica eventi (Cliente)	Durante la corsa, provare a eseguire delle azioni non consentite, quali il superamento del limite di velocità e l'uscita dall'area limitata. L'app notificherà l'utilizzatore per ognuna di queste.	Positivo, se il sistema reagisce in un tempo adeguato all' evento comunicandolo al cliente.
CL6	Ritorno al negozio	Dalla schermata del commerciante, selezionare la voce ritorno al negozio, aprendo così Google Maps in modalità navigazione.	Positivo se la navigazione verrà avviata in modo corretto.

4. Elementi testati

Per la parte di coding sarà molto importante prendere in considerazione sia la parte applicativa presente su smartphone, sia la parte di Database al quale si collegheranno gli utenti (clienti o commercianti), in quanto devono essere effettuati test mirati e precisi in modo tale da scovare bug e inconsistenze nell'intero sistema.

Analizzando ogni singola attività assieme ad ogni funzione performata, saremmo capaci di comprendere quelle parti del progetto che richiederanno più impegno e manodopera per essere risolte o modificate.

In questo caso il testing maggiore sarà fatto sullo smartphone (considerando l'account del cliente e del commerciante) e sul Database (insieme di dati utili per essere salvati e utilizzati in maniera sicura dal commerciante).

5. Schedulazione del testing: tempo e risorse allocate

Il tempo e le risorse saranno gestite in base alla priorità e alle strategie di testing adottate:

- **INCREMENTAL TESTING**, verrà usato ogni volta che aggiungeremo una nuova funzione o un'attività al sistema.

Sarà la strategia più importante perché ci permetterà di controllare in modo continuativo l'intero sistema, per questo motivo in termini di tempo e risorse risulterà essere oneroso, perché senza aver gestito problemi di avvio di determinate attività o l'avvio dell'applicazione stessa, il progetto non potrà progredire nelle fasi successive.

- **THREAD TESTING**, verrà utilizzato in tutti prototipi in quanto nel primo prototipo un commerciante e un cliente dovranno scambiarsi informazioni riguardo la posizione e avvisi vari, mentre nei prototipi successivi il contesto si evolverà a più commercianti con più clienti.

In particolare, il commerciante avrà la possibilità di tener traccia a livello visivo sulla mappa (oltre che a livello descrittivo con tabelle riassuntive) tutte le posizioni aggiornate in tempo reale dei clienti che hanno attivato una corsa con lui.

6. Procedure di registrazione dei test

Il gruppo ha deciso di annotare ogni test che verrà effettuato tramite questo tipo di tabella. Annoteremo il responsabile del Test, assegnando un assistente, specificando la descrizione del test, l'esito e un'eventuale annotazione che servirà al gruppo per poter aggiustare e correggere in corso d'opera problematiche e incongruenze emerse.

Specifica test	Descrizione	Responsabile tester e Assistente	Risultato test	Annotazione

I risultati saranno poi oggetto di studio e analisi da parte del gruppo per poter tracciare una panoramica dei vari esiti in modo tale da poter risolvere e revisionare il codice nella maniera più consona al progetto.

7. Requisiti hardware e software utilizzati

I requisiti per il testing riguardano i nostri dispositivi e la possibilità (vista la situazione d'emergenza) di potersi muovere in una zona contrassegnata per il testing per poter verificare il funzionamento corretto dell'applicativo. In particolare, i dispositivi dovranno avere:

- Dispositivo smartphone Android (API ≥ 24) con l'applicazione finale installata
- Disponibilità di WI-FI o Connessione Dati
- PC collegato ad internet per provare gli account Commerciante/Cliente.
- PC dotati di IDE per coding
- Conoscenza del linguaggio Android, Java, XML.
- Conoscenza del funzionamento della console Firebase e il suo collegamento con le applicazioni web e Android

8. Requisiti hardware e software utilizzati

I test devono essere conclusi prima della scadenza finale prevista per il giorno 15/01/2021.

Entro tale data il sistema dovrà essere stato testato nella sua interezza, essere stabile ed eseguire tutte le azioni e i casi d'uso elencati nel **Documento di Analisi e Specifica dei Requisiti** relativi al terzo prototipo da noi proposto secondo il modello evolutivo.