



Test Incident Report OutiftMaker

Riferimento	
Versione	1.0
Data	26/01/2024
Destinatario	Studenti di Ingegneria del Software 2023/24
Presentato da	Avallone Francesca Buongiorno Rocco Pio Letterese Michele
Approvato da	



Sommario

Revision History.....	3
1. Introduzione	4
1.1 Scopo del sistema.....	4
1.2 Scopo del documento	4
1.3 Riferimenti	4
2. Ambiente di esecuzione	5
3. Test Incident Report	6



Revision History

DATA	VERSIONE	DESCRIZIONE	AUTORI
26/01/2024	0.1	Prima stesura	Francesca Avallone Rocco Pio Buongiorno Michele Letterese
27/01/2024	0.2	Compilazione Test Incident Report per ciascun Test Case	Francesca Avallone Rocco Pio Buongiorno Michele Letterese
27/01/2024	1.0	Revisione finale documento	Francesca Avallone Rocco Pio Buongiorno Michele Letterese



1. Introduzione

1.1 Scopo del sistema

Lo scopo principale del sistema è quello di fornire agli utenti dei consigli di abbigliamento personalizzati in base alle loro preferenze e al contesto. Adattare le raccomandazioni di outfit in base al gusto personale dell'utente considerando tipologie preferiti, colori preferiti, e altre preferenze individuali. Questo, tenendo conto dei capi di abbigliamento già presenti nell'armadio virtuale personale dell'utente. Creando o generando degli outfit con essi, ad hoc per le preferenze richieste dall'utente avendo anche la possibilità di visionare anche dopo gli outfit creati o generati nel proprio archivio outfit. Tutto questo comodamente nella tasca dell'utente.

1.2 Scopo del documento

Il seguente documento riporta e descrive le attività di Testing effettuate per garantire il corretto funzionamento dell'applicazione OutfitMaker.

All'interno del documento saranno riportate le funzionalità testate, con i relativi strumenti e strategie utilizzati.

Le funzionalità testate saranno le seguenti:

- Controllo capi minimi
- Inserimento Capo
- Ricerca Capo in Armadio Virtuale

1.3 Riferimenti

- Test Plan (TP)
- Test Case Specification (TCS)
- Test Incident Report (TIR)



2. Ambiente di esecuzione

Per l'esecuzione dei test è stato utilizzato l'ambiente Android Studio, sviluppando un'applicazione Android, è fondamentale garantire la corretta funzionalità delle singole unità di codice (unit test) al fine di identificare e risolvere eventuali problemi.

Per la gestione delle dipendenze durante i test di unità, si fa uso di Mockito, una libreria ampiamente utilizzata per la creazione di oggetti simulati (mock objects). I mock objects consentono di emulare il comportamento di database o altre classi, restituendo risultati specifici in base al contesto del test.



3. Test Incident Report

Test Case ID	Test Incident ID	Tester	Data	Output
TC1_UC1_RF7_GG.1_1	TI1_UC1_RF7_GG.1_1	Rocco Pio Buongiorno	26/01/2024	Passed
TC1_UC1_RF7_GG.1_2	TI1_UC1_RF7_GG.1_2	Rocco Pio Buongiorno	26/01/2024	Passed
TC1_UC1_RF7_GG.1_3	TI1_UC1_RF7_GG.1_3	Rocco Pio Buongiorno	26/01/2024	Passed
TC1_UC1_RF7_GG.1_4	TI1_UC1_RF7_GG.1_4	Rocco Pio Buongiorno	26/01/2024	Passed
TC1_UC1_RF7_GG.1.1_1	TI1_UC1_RF7_GG.1.1_1	Rocco Pio Buongiorno	26/01/2024	Passed
TC1_UC1_RF7_GG.1.1_2	TI1_UC1_RF7_GG.1.1_2	Rocco Pio Buongiorno	26/01/2024	Passed
TC2_UC2_RF4_GAV.1_1	TI2_UC2_RF4_GAV.1_1	Michele Letterese	27/01/2024	Passed
TC2_UC2_RF4_GAV.1_2	TI2_UC2_RF4_GAV.1_2	Michele Letterese	27/01/2024	Passed
TC2_UC2_RF4_GAV.1_3	TI2_UC2_RF4_GAV.1_3	Michele Letterese	27/01/2024	Passed
TC2_UC2_RF4_GAV.1_4	TI2_UC2_RF4_GAV.1_4	Michele Letterese	27/01/2024	Passed
TC2_UC2_RF4_GAV.1_5	TI2_UC2_RF4_GAV.1_5	Michele Letterese	27/01/2024	Passed
TC2_UC2_RF4_GAV.1_6	TI2_UC2_RF4_GAV.1_6	Michele Letterese	27/01/2024	Passed
TC3_UC3_RF3_GAV.2_1	TI3_UC3_RF3_GAV.2_1	Francesca Avallone	27/01/2024	Passed
TC3_UC3_RF3_GAV.2_2	TI3_UC3_RF3_GAV.2_2	Francesca Avallone	27/01/2024	Passed
TC3_UC3_RF3_GAV.2_3	TI3_UC3_RF3_GAV.2_3	Francesca Avallone	27/01/2024	Passed
TC3_UC3_RF3_GAV.2_4	TI3_UC3_RF3_GAV.2_4	Francesca Avallone	27/01/2024	Passed