



UNIVERSITÀ DEGLI STUDI DI SALERNO

Corso di Laurea in Informatica, prof. Gravino Carmine, a.a.
2022/2023

Progetto di Ingegneria del Software



Test Plan (TP)

Versione	0.5
Data	17/02/2023
Presentato da	Guerrera Marco, Lamberti Salvatore, Napolitano Margherita Maria, Zullo Nicola Mario

Revision History

<i>Data</i>	<i>Versione</i>	<i>Descrizione</i>
09/12/2022	0.1	Aggiunta approccio e specifica test case (MMN)
12/12/2022	0.2	Modifica approccio
12/12/2022	0.2	Aggiunta specifica test case (SL)
13/12/2022	0.3	Aggiunta Strumenti per il testing (NMZ)
14/12/2022	0.4	Aggiunta specifica Test Case (MG)
14/12/2022	0.4	Aggiunta requisiti da testare e non; Pass/Fail Criteria (SL)
14/12/2022	0.4	Aggiunta Introduzione, relazione con gli altri documenti, Panoramica del sistema (MG)
14/12/2022	0.4	Aggiunta specifica Test Case (NMZ)
17/02/2023	0.5	Revisione

INDICE

1. Introduzione
2. Relazione con gli altri documenti
3. Panoramica del sistema
4. Funzionalità da testare e non
5. Pass/Fail criteria
6. Approccio
7. Strumenti per il testing (hardware/software)
8. Specifica dei test cases

1. Introduzione

Tommit si propone di aiutare gli studenti ad organizzare il proprio studio, suddividendo le ore in base alle loro necessità e nella ricerca di gruppi di studio inerenti alle materie di loro interesse.

Il Test Plan ha l'obiettivo di descrivere ed analizzare le attività di testing del sistema per garantire un corretto funzionamento e fruizione per gli utenti della piattaforma.

In questo documento sono riportati gli strumenti e le strategie di testing scelti dal team per presentare al cliente un prodotto privo di malfunzionamenti.

Sono stati presi in considerazione i seguenti casi d'uso per l'esecuzione dei test:

- **Richiesta Creazione di un gruppo studio**
- **Login**
- **Registrazione**
- **Richiesta Aggiunta Appunti**

2. Relazione con altri documenti

RAD: I test case vengono presi dai casi d'uso presenti nel RAD.

SDD: I test case devono rispettare la suddivisione in sottosistemi presentata nell'SDD.

3. Panoramica del sistema

Il sistema proposto presenta una architettura di tipologia "three-tier".

Verranno usati HTML5, CSS, Bootstrap, JS per la generazione del front-end.

La logica applicativa e quindi il back-end sarà implementata tramite Tomcat.

Per la gestione del database sarà usato MySQL.

4. Funzionalità da testare e non

Di seguito l'elenco delle varie features su cui verrà effettuato il testing per le varie gestioni

- Gestione utente
 - registrazione utente
- Gestione Gruppo Studio
 - richiesta creazione gruppo studio
 - iscrizione gruppo studio
- Gestione Timer
 - personalizzazione timer

Le funzionalità su cui non verranno effettuate le attività di testing riguardano requisiti funzionali di bassa o media priorità. Sono inoltre escluse le funzionalità che non prevedono input manuale da parte dell'utente.

5. Pass/Fail Criteria

Le attività di testing sono utili per identificare all'interno del sistema eventuali errori, che successivamente verranno eliminati mediante interventi mirati

L'esito di un test case è valutato mediante un oracolo, inteso come il risultato atteso della sua esecuzione, basandosi sui requisiti a sua disposizione.

Un test ha successo (pass) se, dato un input al sistema, l'output ottenuto è diverso dall'output atteso dall'oracolo.

Un test fallisce (fail) se, dato un input al sistema, l'output ottenuto è uguale all'output atteso dall'oracolo.

Tutto il testing sarà valido se tutti i seguenti vincoli saranno rispettati:

- Testare tutti i requisiti funzionali ad alta priorità;
- Effettuare test di regressione ogni volta che si introducono nuove caratteristiche al sistema o vengono modificate quelle presenti.

6. Approccio

Il Testing è il processo di individuazione delle differenze tra il comportamento atteso specificato da modelli di sistema e il comportamento osservato del sistema implementato.

L'intera fase di testing ha come scopo principale la produzione di un prodotto di qualità che soddisfi i requisiti specificati. Le attività di testing saranno suddivise in:

-Unit Testing: trova differenze tra una specifica di un oggetto(object design model) e la sua realizzazione come componente. Il testing di unità è eseguito dagli sviluppatori e si concentra sugli oggetti e i sottosistemi del sistema software per controllare che ognuno di essi sia codificato correttamente ed esegua la funzionalità prevista. I casi di test saranno definiti e documentati direttamente nel codice attraverso l'uso di framework per il testing di classi Java JUnit.

-Testing funzionale: ha l'obiettivo di validare i requisiti funzionali. Consiste nell'individuare i possibili faults generati dagli input degli utenti.

7. Strumenti per il testing.

Gli strumenti di cui necessitiamo per effettuare il testing sono: un computer con un browser installato ed una connessione ad Internet poiché il nostro progetto è web-based.

Per il testing di sistema sarà usato Selenium.

8. Specifica dei test cases

8.1 Richiesta Aggiunta Appunti (UC_RAA_1)

VINCOLI

Parametro: Titolo	
Nome Categoria	Scelte per la categoria
Lunghezza_Titolo[LN]	1.Lunghezza < 1 [errore] 2.Lunghezza >= 1 [PROPERTY_LN_OK]
Parametro: Testo	
Nome Categoria	Scelte per la categoria
Lunghezza_Testo[LT]	1.Lunghezza < 1 [errore] 2.Lunghezza >= 1 [PROPERTY_LN_OK]
Parametro: Materia	
Nome Categoria	Scelte per la categoria
Correttezza_Materia[CM]	1. Correttezza!=Umanistica OR Correttezza!=Scientifico OR Correttezza!=Artistica OR Correttezza!=Informatica OR Correttezza!=Lingue OR Correttezza!=Sanitario [errore] 2. Correttezza=Umanistica OR Correttezza=Scientifico OR Correttezza=Artistica OR Correttezza=Informatica OR Correttezza=Lingue OR Correttezza=Sanitario [PROPERTY_LN_OK]

TEST FRAMES

Test Case ID	Test Frame	Esito
TC_1.1_1	LN1	Errato: Lunghezza del motivo non valida
TC_1.1_2	LN2, LT1	Errato: Lunghezza del testo non valida
TC_1.1_3	LN1, LT1	Errato: Lunghezza del titolo non valida, lunghezza del testo non valida
TC_1.1_4	LN2, LT2, CM1	Errato: Correttezza materia non valida
TC_1.1_5	LN2, LT2, CM2	Corretto

8.2 Richiesta creazione Gruppo Studio (UC_RCGS_1)

VINCOLI

Parametro: obiettivi del gruppo	
Nome Categoria	Scelte per la categoria
Lunghezza_Obiettivi [LNO]	1.Lunghezza < 1 [errore] 2.Lunghezza >= 1 [PROPERTY_LNO_OK]
Parametro: materia di studio	
Nome Categoria	Scelte per la categoria
Lunghezza_Materia [LNM]	1.Lunghezza < 1 [errore] 2.Lunghezza >= 1 [PROPERTY_LNM_OK]
Parametro: luogo di incontro	
Nome Categoria	Scelte per la categoria
Lunghezza_LuogoDiIncontro [LNL]	1.Lunghezza < 1 [errore]

	2.Lunghezza >= 1 [PROPERTY_LNL_OK]
Parametro: nome del gruppo	
Nome Categoria	Scelte per la categoria
Lunghezza_Nome [LNN]	1.Lunghezza < 1 [errore] 2.Lunghezza >= 1 [PROPERTY_LNN_OK]

TEST FRAMES

Test Case ID	Test Frame	Esito
TC_1.2_1	LNO1	Errato: Lunghezza degli obiettivi non valida
TC_1.2_2	LNO2, LNM1	Errato: Lunghezza della materia non valida
TC_1.2_3	LNO2, LNM2, LNL1	Errato: Lunghezza del luogo non valida
TC_1.2_4	LNO2, LNM2, LNL2, LNN1	Errato: Lunghezza nome non valida
TC_1.2_5	LNO2, LNM2, LNL2, LNN2	Corretto

8.3 Registrazione Utente (UC_R_1)

VINCOLI

Parametro: nome	
Nome Categoria	Scelte per la categoria
Lunghezza_Nome [LN]	1.Lunghezza < 1 OR Lunghezza >50 [errore] 2. 50 > Lunghezza >= 1 [PROPERTY_LN_OK]
Parametro: cognome	
Nome Categoria	Scelte per la categoria
Lunghezza_Cognome [LC]	1.Lunghezza < 1 OR Lunghezza >50 [errore] 2. 50 > Lunghezza >= 1 [PROPERTY_LC_OK]
Parametro: CF (codice fiscale)	
Nome Categoria	Scelte per la categoria
Lunghezza_CF [LCF]	1.Lunghezza < 16 OR Lunghezza >16 [errore] 2. Lunghezza = 16 [PROPERTY_LCF_OK]
Parametro: username	
Nome Categoria	Scelte per la categoria
Lunghezza_Username [LU]	1.Lunghezza < 1 OR Lunghezza >50 [errore] 2. 50 > Lunghezza >= 1 [PROPERTY_LU_OK]
Parametro: email	

Nome Categoria	Scelte per la categoria
Lunghezza_Email [LE]	1.Lunghezza < 1 [errore] 2. Lunghezza > 1[PROPERTY_LE_OK]
Formato_Email[FE]	1. “@” not in email [errore] 2. “@” in email [PROPERTY_FE_OK]
Parametro: password	
Nome Categoria	Scelte per la categoria
Lunghezza_Password [LP]	1.Lunghezza < 1 OR Lunghezza >16 [errore] 2.16 > Lunghezza >= 1 [PROPERTY_LP_OK]
Parametro: confermaPassword	
Nome Categoria	Scelte per la categoria
Formato_ConfermaPassword [FCP]	1. confermaPassword != password [errore] 2. confermaPassword == password [PROPERTY_FCP_OK]

TEST FRAMES

Test Case ID	Test Frame	Esito
TC_1.3.1	LN1	Errato: Lunghezza del nome non valida
TC_1.3.2	LC1, LP1	Errato: Lunghezza del cognome non valido, Lunghezza della password non valida
TC_1.3.3	LP2, FCP1	Errato: confermaPassword diversa da password
TC_1.3.4	LN2, LC2, LCF2, FE1	Errato: Formato email non valido, carattere “@” non presente

TC_1.3.5	LN2, LC2, LCF2, LE2, FE2, LP3, FCP2	Corretto
----------	-------------------------------------	----------

8.4 Login UC_L_1

VINCOLI

Parametro: Username	
Nome Categoria	Scelte per la categoria
Match Username [MUN]	1. Match con username= false[errore] 2. Match con username= true [PROPERTY_UN_OK]
Campo Username Vuoto [CUV]	1. Parametro username vuoto = true [errore] 2. Parametro username vuoto = false [PROPERTY_CUV_OK]
Parametro: Password	
Nome Categoria	Scelte per la categoria
Match Password [MP]	1. Match con password= false[errore] 2. Match con password= true [PROPERTY_MP_OK]
Campo Password Vuoto [CPV]	1. Parametro password vuoto = true [errore] 2. Parametro password vuoto = false [PROPERTY_CPV_OK]

TEST FRAMES

Test Case ID	Test Frame	Esito
TC_1.4_1	MUN1	Errato: Match Username fallito
TC_1.4_2	MP1	Errato: Match Password fallito
TC_1.4_3	CUV1	Errato: Campo Username vuoto

TC_1.4_4	CPV1	Errato: Campo Password vuoto
TC_1.4_5	MUN2, MP2, CUV2, CPV2	Corretto