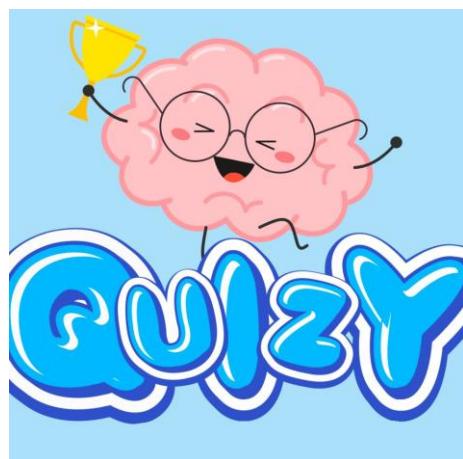


Università degli Studi di Salerno
Corso di Ingegneria del Software

**Quiz
Test Plan
Versione 0.2**



Data: 22/12/2025

Progetto: Quizy	Versione: 0.2
Documento: Test Plan	Data: 21/12/2025

Coordinatore del progetto:

Nome	Matricola
Davide Nino Longobardi	0512119971

Partecipanti:

Nome	Matricola
Roberto Caiazza	0512116335
Fabiana Paciello	0512119578

Scritto da:	Roberto Caiazza
-------------	-----------------

Revision History

Data	Versione	Descrizione	Autore
20/12/2025	0.0	Stesura dei primi 4 punti del documento	Davide Nino / Roberto Caiazza / Fabiana Paciello
21/12/2025	0.1	Stesura completo del documento	Davide Nino / Roberto Caiazza / Fabiana Paciello
27/12/2025	0.2	Aggiunta completamento del punto 9	Roberto Caiazza

Indice

1.	INTRODUZIONE	3
2.	Relationship to other document.....	4
3.	System overview	4
4.	Features to be tested/not to be tested.....	4
5.	Pass/Fail criteria	4
6.	Approach.....	4
7.	Suspension and resumption	4
8.	Testing materials.....	4
9.	Test cases.....	4
10.	Testing schedule	4

1. Introduzione

Il documento viene redatto per fornire una panoramica sui test che vengono effettuati sull'applicazione Quizy. L'obiettivo di questo documento è definire l'ambito, l'approccio, le risorse e la pianificazione delle attività di testing necessarie per validare il sistema. Questo piano è stato redatto per fornire un framework operativo condiviso tra sviluppatori e tester, assicurando che le verifiche siano eseguite in modo sistematico, efficiente ed economico.

2. Relationship to other document

- Requirements Analysis Document (RAD)
- System Design Document (SDD)
- Object Design Document (ODD)

3. System overview

Questa sezione identifica i componenti software oggetto di verifica durante le fasi di unit testing e integration testing. L'unità minima di test corrispondono alle classi contenute nei seguenti sottosistemi logici, organizzati per livello e la persistenza dei dati.

Interface Layer

Rappresenta il front-end dell'applicazione installato sui dispositivi mobili. La componente *general_interface(UI)* nel package *ViewsQuizy*, gestisce le Activity Android e la cattura degli eventi utente. Il test e la corretta visualizzazione delle schermate.

Application Layer

Costituisce il nucleo della logica di business, è composto da quattro componenti principali che saranno quattro componenti principali che saranno testati isolatamente.

- **UserManage:** gestisce l'autenticazione, il login, il logout e il recupero password.
- **QuizManage:** Componente critico che gestisce il ciclo di vita dei quiz
- **GestureAccoun:** Gestire le operazioni amministrative riservare al *Gestore Account*

Persistence Layer

Livello responsabilità della memorizzazione permanente dai dati. Comprende *persistenceManage*, gestisce il mapping relazione verso il DBMS MySQL e garantisce l'integrità delle transazioni.

4. Features to be tested

A livello di funzionalità, gli aspetti del sistema che verranno testati sono.

- Registrazione Utente
- Creazione Quiz
- Creazione Task

- Cambio Ruolo

5. Pass/Fail criteria

Un test viene considerato *Passed* quando non si sono riscontrate failure durante l'esecuzione, mentre viene considerato *Failed* quando sono state riscontrate una o più failure.

6. Approach

L'approccio del testing si divide in:

Testing di unità: verrà utilizzata la tecnica di "Black-Box" testing. Si focalizza sul comportamento di I/O, senza preoccuparsi della struttura interna della componente. Se per ogni dato input siamo in grado di prevedere l'output, allora l'unità super il test. Riduciamo il numero di casi di test effettuando una partizione dividendo le condizioni di input in classi di equivalenza e si scelgono i test case per ogni classe di equivalenza.

Testing di integrazione: si utilizza la strategia *bottom-up*. I sottosistemi al livello più basso della gerarchia sono testati individualmente. I successivi sottosistemi ad essere testati sono quelli che chiamano i sottosistemi testati in precedenza.

Testing di sistema: è l'ultimo testing prima della messa in uso del sistema. Prevede il controllo delle funzionalità del sistema secondo i requisiti specificati.

7. Suspension and resumption

La fase di testing sarà sospesa quando si otterranno i risultati attesi, rispettando però i tempi di consegna del progetto.

La fase di testing sarà ripresa se si effettueranno modifiche al sistema e verranno rieseguiti di nuovo i casi di test.

8. Testing materials

Materiale richiesto per il testing sarà:

- Uno (o più) dispositivo di elaborazione
- Android Studio

9. Test cases

UC 1 Registrazione

UC 2 Login

UC 3 Partecipazione Quiz

UC 4 Creazione quiz

UC 5 Controllo dei problemi

UC 6 Cambio ruolo

10. Testing schedule

- Responsabilità: Il Team Lead supervisiona; gli sviluppatori eseguono Unit, Integration e System Test.
- Rischi: Ritardi nell'implementazione API Restful potrebbero far slittare i test di integrazione frontend.
- Pianificazione: Settimana 1-2 (Unit Testing), Settimana 3 (Integration Testing), Settimana 4 (System e Regression Testing).