

# TEST NON SINCRONIZZATO SU PROD POST 1

## Sommario

---

TEST OUT OF SYNC ON PROD POST 1 TEST OUT OF SYNC ON PROD POST 1 TEST OUT OF SYNC ON PROD POST 1 TEST OUT OF SYNC ON PROD POST 1 TEST OUT OF SYNC ON PROD POST 1

## Tipi in base allo scopo del test

### Testing funzionale

[Il testing funzionale](#) verifica il funzionamento del sistema assicurandosi che le caratteristiche e le funzionalità funzionino in base ai requisiti specificati.

- [Prova del fumo](#): Eseguo controlli di base sulle funzionalità critiche per garantire la stabilità dell'applicazione per ulteriori test.
- [Test di integrità](#): Verifica alcune funzionalità specifiche dopo alcune modifiche minori o correzioni di bug per assicurarsi che funzionino come previsto.
- [Test di integrazione](#): Verifica l'interazione tra i diversi moduli per verificarne il corretto funzionamento.
- [Test del sistema](#): Verifica l'intera applicazione per convalidare la funzionalità end-to-end.
- [Prova di regressione](#): Ripete il test delle funzionalità esistenti dopo le modifiche o le correzioni dei bug per assicurarsi che non vengano introdotti nuovi difetti.
- [User Acceptance Testing \(UAT\)](#): Condotta da utenti finali o clienti per verificare che il software soddisfi i requisiti aziendali.

### Test non funzionali

[Il test non funzionale](#) valuta le prestazioni del sistema in termini di usabilità, prestazioni, sicurezza e compatibilità.

- [Test di usabilità](#): Valuta la facilità d'uso dell'applicazione, l'interfaccia utente e l'esperienza complessiva dell'utente.
- [Test di compatibilità](#): Verifica che l'applicazione funzioni correttamente su browser, dispositivi e sistemi operativi diversi.

- [Test delle prestazioni](#): Misura la velocità, la reattività, la stabilità e il comportamento dell'applicazione in base a carichi di lavoro diversi.
- [Test di sicurezza](#): Identifica le vulnerabilità e garantisce la protezione dei dati sensibili e delle risorse di sistema.

## Processo di testing manuale

Il diagramma seguente fornisce una rappresentazione dettagliata delle fasi del processo di test manuale.

### Processo di testing manuale

- **Analisi dei requisiti**: comprendere e analizzare i requisiti del progetto, le aspettative degli utenti e le esigenze aziendali per identificare gli elementi da testare.
- **Creazione del piano di test**: consente di creare un piano di test che definisce l'ambito, gli obiettivi, le risorse, la pianificazione e l'approccio al test.
- **Creazione di test case**: progettazione di test case dettagliati con passaggi di test, risultati previsti e dati di test per convalidare la funzionalità dell'applicazione.
- **Configurazione ambiente di test**: preparare l'hardware, il software, gli strumenti e i dati di test necessari per simulare l'ambiente di produzione.
- **Esecuzione test case**: eseguire i test case e confrontare i risultati effettivi con i risultati previsti per identificare eventuali problemi.
- **Registrazione dei difetti**: consente di registrare i difetti con dettagli completi quali la descrizione, la gravità, le fasi di riproduzione e le prove di supporto.
- **Correzione e ripristino difetti**: verificare che i difetti segnalati siano stati corretti e che il problema non esista più.
- **Test di regressione**: nuovo test interessato e funzionalità correlate per garantire che le modifiche recenti non abbiano introdotto nuovi difetti.
- **Report chiusura test**: preparare un report finale che riepiloghi le attività di test, i risultati, lo stato dei difetti e le principali lezioni apprese.

## Strumenti utilizzati per il test manuale

- **TestRail**: strumento di gestione dei test utilizzato per organizzare, eseguire e tenere traccia di test case manuali con funzionalità di reporting dettagliate.
- **Xray (per Jira)**: uno strumento di test integrato in Jira che supporta la gestione manuale dei test, la tracciabilità e il rilevamento dei difetti.
- **Qase**: una piattaforma di gestione dei test basata su cloud che semplifica la creazione, l'esecuzione e la collaborazione tra team dei test case.
- **Zephyr**: una soluzione di gestione dei test che consente di gestire le attività di test manuali

con una solida integrazione e creazione di report Jira.

- Tuskr: un leggero strumento di gestione dei test basato su cloud per la creazione, la gestione e il monitoraggio efficiente dei test case manuali.

## Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l'accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).