

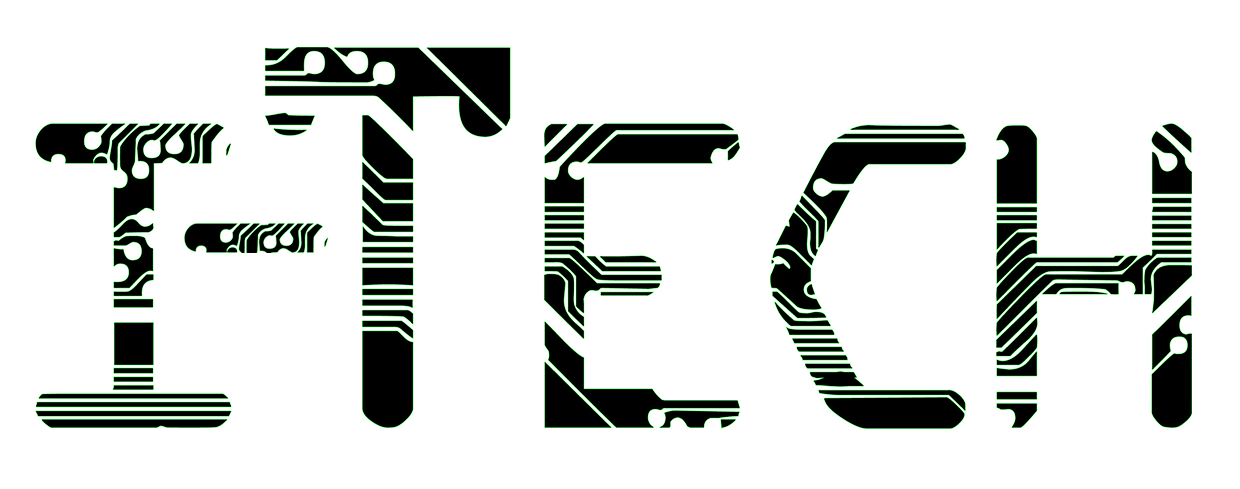
**Università degli Studi di Salerno**

**Anno Accademico 2016/2017**

**Corso di Ingegneria del Software**

**System Test Plan**

**V 1.3**

****

Introduzione 6

Scopo 6

Obiettivi 6

Scenari di testing 6

Documentazione allegata al System Test Plan 7

Relazione con il documento di analisi dei requisiti (RAD) 7

Relazione con il documento di system design (SDD) 7

Relazione con il documento di object design (ODD) 7

Panoramica generale del sistema oggetto di testing 8

Funzionalità da testare 9

Criteri Pass/Failed 9

Strategia globale e approccio 10

Testing di unità 10

Testing di Integrazione 10

Testing di sistema 10

Sospensione e ripresa 11

Criteri di sospensione 11

Criteri di ripresa 11

Materiale per il testing 12

Test cases 13

Gestione registrazione 13

Registrazione 13

Category partition 13

Specification 15

Gestione autenticazione 18

Login 18

Category partition 18

Specification 18

Gestione riparazioni 20

Richiesta preventivo 20

Category partition 20

Invio preventivo 20

Category partition 20

Accetazione/rifiuto preventivo 21

Category partition 21

Gestione acquisti 22

Carrello 22

Category partition 22

Specification 22

Gestione magazzino 24

Inserimento prodotti 24

Category partition 24

Specification 27

Carico/scarico prodotti 29

Category partition 29

Modifica prezzo 29

Category partition 29

Gestione utenti 30

Modifica dati [admin] 30

Category partition 30

Mail 30

Password 30

Modifica dati [clienti] 31

Category partition 31

Indirizzo 31

Telefono 31

Mail 31

Password 32

Pianificazione del testing 33

Determinazione dei ruoli 33

Determinazione dei rischi 33

Organizzazione delle attività di testing 33

Stress and Time Testing 34

Test click - 250 utenti 34

10 click al secondo 34

Run test until 10 click per user 34

Grafico tridimensionale 35

Grafico bidimensionale 36

Index 36

Info riparazioni 37

Registrazione 37

Prodotti – Computer 38

Prodotti – Smartphone 38

Test click - 1000 utenti 39

100 click al secondo 39

Run test until 100 click per user 39

Grafico tridimensionale 40

Grafico bidimensionale 41

Index 41

Info riparazioni 42

Registrazione 43

Prodotti – Computer 43

Prodotti – Smartphone 44

Time test – 1500 utenti 45

Time: 25 minuti 45

Click delay 5 secondi 45

Grafico tridimensionale 46

Grafico bidimensionale 47

Index 47

Info riparazioni 48

Registrazione 48

Prodotti – Computer 49

Prodotti – Smartphone 49

Time test – 750 utenti 50

Time: 50 minuti 50

Click delay 1 secondi 50

Grafico tridimensionale 51

Grafico bidimensionale 52

Index 52

Info riparazioni 53

Registrazione 53

Prodotti – Computer 54

Prodotti – Smartphone 54

RAMP test – 1000 utenti 55

Time: 30 minuti 55

Click delay 3 secondi 55

Grafico tridimensionale 56

Grafico bidimensionale 57

Index 57

Info riparazioni 58

Registrazione 58

Prodotti – Computer 59

Prodotti – Smartphone 59

RAMP test – 1000 utenti 60

Time: 60 minuti 60

Click delay 1 secondi 60

Grafico tridimensionale 61

Grafico bidimensionale 62

Index 62

Info riparazioni 63

Registrazione 63

Prodotti – Computer 64

Prodotti – Smartphone 64

Test incident 65

Bug Report 65

Gestione registrazione 65

BR\_1.1 65

BR\_1.2 65

BR\_1.3 66

Gestione autenticazione 66

BR\_2.1 66

Gestione utente 66

BR\_3.1 66

BR\_3.2 66

BR\_3.3 67

Gestione prodotti 67

BR\_4.1 67

BR\_4.2 67

Gestione riparazioni 68

Gestione ordine 68

BR\_6.1 68

BR\_6.2 68

Gestione dati 68

Gestione interfaccia 68

BR\_8.1 68

BR\_8.2 69

Security and Penetration Test 70

SQL Injection 70

JavaScript-HTML test 70

Glossario 71

**Top Manager:**

*Prof. De Lucia Andrea*

**Team di sviluppo:**

|  |  |
| --- | --- |
| Nome e Cognome | Matricola |
| *Stefano Foresta* | 0512100504 |
| *Gennaro Franzese* | 0512100270 |

**Revision History:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Autore** | **Data** | **Descrizione** | **Versione** |
| *Stefano Foresta* | *13/12/16* | *Struttura documento* | *v 0.1* |
| *Gennaro Franzese* | *13/12/16* | *Stesura capitolo uno* | *v 0.3* |
| *Gennaro Franzese* | *14/12/16* | *Stesura capitolo due* | *v 0.4* |
| *Stefano Foresta* | *14/12/16* | *Revisione capitoli uno e due* | *v 0.4.1* |
| *Stefano Foresta* | *14/12/16* | *Stesura capitolo tre* | *v 0.6* |
| *Gennaro Franzese* | *07/01/17* | *Stesura capitolo quattro, cinque e sei* | *v 0.7* |
| *Gennaro Franzese* | *10/01/17* | *Stesura capitolo sette* | *v 0.7.4* |
| *Stefano Foresta* | *20/01/17* | *Stesura capitolo otto, inizio fase test* | *v 0.8* |
| *Stefano Foresta* | *30/01/17* | *Revisione capitoli da uno a otto* | *v 0.9* |
| *Gennaro Franzese* | *05/02/17* | *Stesura capitolo su Stress and Timing Testing* | *v 1.0* |
| *Stefano Foresta* | *08/02/17* | *Stesura capitolo Test Incident* | *v 1.1* |
| *Gennaro Franzese* | *10/02/17* | *Stesura capitolo sicurezza* | *v 1.2* |
| *Stefano Foresta e Gennaro Franzese* | *18/02/17* | *Revisione totale documento* | *v 1.3* |

# Introduzione

## Scopo

Questo documento specifica un piano di test della web application “**I-Tech**”. Esso descrive la strategia di sperimentazione e l’approccio al test della stessa che il team utilizzerà per verificare che la domanda soddisfa i requisiti stabiliti dal business plan prima del rilascio. In questa attività andremo a rilevare gli eventuali errori prodotti all’interno del codice, per evitare che essi si presentino nel momento in cui il sistema verrà utilizzato dall’utente finale.

Verranno testate esclusivamente le funzionalità implementate e specificate nell’ODD. La fase di testing è quindi strettamente legata alle fasi ad essa precedenti; ogni documento, risultato delle differenti fasi di sviluppo, sarà un punto di partenza indispensabile e centrale per poter effettuare un testing corretto e adeguato al sistema software sviluppato.

## Obiettivi

* Soddisfare le specifiche, le esigenze e le regole di progetto;
* Supportare gli standard aziendali e software specificati;
* Soddisfare i criteri di input dell’utente che testerà il programma.

## Scenari di testing

Le attività di test che si andranno ad eseguire sono state pianificate per le seguenti operazioni:

* Gestione registrazione
* Gestione autenticazione
* Gestione riparazioni
* Gestione acquisti
* Gestione magazzino
* Gestione utenti

# Documentazione allegata al System Test Plan

Il test plan ha, come sottoscritto nell’introduzione, una relazione stretta con il resto dei documenti che sono stati prodotti durante l’elaborazione del prodotto software in questione, poiché in essi è stata pianificata la modalità di sviluppo e di futuro utilizzo dello stesso. Questo quindi permette di rilevare le eventuali differenze tra il comportamento atteso e quello osservato del sistema (cosa fondamentale). Di seguito verranno riportate le relazioni tra il test plan e la documentazione precedente.

## Relazione con il documento di analisi dei requisiti (RAD)

Riguarda particolarmente i requisiti funzionali e non funzionali che sono stati specificati in precedenza e descritti nel RAD in maniera accurata.

## Relazione con il documento di system design (SDD)

In tale documento vengono specificati i sottosistemi e la divisione in livelli è quindi, obbiettivo fondamentale di tale test, è di rimanere quanto più fedele a tale suddivisione.

## Relazione con il documento di object design (ODD)

Troviamo, in questo documento, la specifica sulle interfacce che saranno poi oggetto di test.

# Panoramica generale del sistema oggetto di testing

Come stabilito nel System Design Document la struttura del nostro sistema è divisa secondo una architettura “Two Layers” cioè a due livelli: Interface and Application Layer e Storage Layer. In questo caso il livello più alto è legato indissolubilmente con il livello applicativo che a sua volta eseguirà le operazioni nel database, cercando di garantire il più possibile basso accoppiamento e alta coesione tra le varie classi. Il sistema inoltre è stato suddiviso in sottosistemi più piccoli che cooperano per il funzionamento ottimale dell’applicazione sviluppata.

Quasi ognuna di queste gestioni prevede principalmente operazioni di inserimento, modifica, cancellazione e visualizzazione: saranno proprio queste funzionalità ad essere testate nel corso della fase di testing del sistema.

# Funzionalità da testare

Di seguito è riportato l’elenco delle funzionalità che verranno testate:

1. Gestione registrazione
   1. Registrazione
2. Gestione autenticazione
   1. Login
3. Gestione riparazioni
   1. Richiesta preventivo
   2. Invio preventivo
   3. Accettazione/rifiuto preventivo
   4. Inizio riparazione
   5. Comunicazione riparazione effettuata
4. Gestione acquisti
   1. Carrello
   2. Finalizzazione acquisto
5. Gestione magazzino
   1. Inserimento prodotti
   2. Carico/Scarico prodotti
   3. Modifica prezzo prodotti
6. Gestione utenti
   1. Modifica dati

# Criteri Pass/Failed

I dati di input del test saranno suddivisi in classi di equivalenza, ovvero verranno raggruppati in insiemi dalle caratteristiche comuni, per i quali sarà sufficiente testare un solo elemento rappresentativo. Un input, quindi, avrà superato un test se l’output risultante sarà quello atteso, cioè quello che è stato specificato dal team rispetto al test case.

# Strategia globale e approccio

Le tecniche di testing adottate riguarderanno inizialmente il testing di unità dei singoli componenti, come detto in precedenza, in modo da testare nello specifico la correttezza di ciascuna unità. Seguirà il testing di integrazione, che focalizzerà l’attenzione principalmente sul test delle interfacce delle suddette unità. Infine verrà eseguito il testing di sistema, che vedrà come oggetto di testing l’intero sistema assemblato nei suoi componenti. Quest’ultimo servirà soprattutto a verificare che il sistema soddisfi le richieste del cliente.

## Testing di unità

Durante questa fase, verranno ricercate le condizioni di fallimento, isolando i componenti ed usando test driver e stub, cioè implementazioni parziali di componenti che dipendono o da cui dipendono le componenti da testare. La strategia utilizzata per il testing si baserà esclusivamente sulla tecnica Black-Box, che si focalizza sul comportamento Input/Output, ignorando la struttura interna della componente. Al fine di minimizzare il numero di test cases, i possibili input verranno partizionati in classi di equivalenza e per ogni classe verrà selezionato un test case. Gli stati erronei scovati in questa, come in qualsiasi altra fase di testing, che comporteranno un fallimento del sistema dovranno essere tempestivamente corretti per poter ripristinare il testing al più presto.

## Testing di Integrazione

In questa fase si procederà all’integrazione delle componenti di una funzionalità che verranno testate nel complesso attraverso una strategia Bottom-Up. Si passerà, poi, alla funzionalità successiva fino ad esaurire le funzionalità implementate. Quest’approccio mira principalmente a ridurre le dipendenze tra funzionalità differenti e a facilitare la ricerca di errori nelle interfacce di comunicazione tra i due sottosistemi.

## Testing di sistema

Lo scopo di questa fase di testing è quello di dimostrare che il sistema soddisfi effettivamente i requisiti richiesti e sia, quindi, pronto all’uso. Come per il testing di unità, si cercherà di testare le funzionalità più importanti per l’utente e quelle che hanno una maggiore probabilità di fallimento. Si noti che, come per il testing di unità, si procederà attraverso tecnica Black-Box.

# Sospensione e ripresa

## Criteri di sospensione

La fase di testing del sistema verrà sospesa quando si raggiungerà un compromesso tra qualità del prodotto e costi dell’attività di testing. Il testing verrà quindi portato avanti quanto più possibile nel tempo senza però rischiare di ritardare la consegna finale del progetto e senza uscire fuori budget.

## Criteri di ripresa

In seguito alle modifiche o correzioni delle componenti che introdurranno errori o fallimenti, i test case verranno sottoposti nuovamente al sistema assicurandosi così di aver risolto effettivamente il problema.

# Materiale per il testing

Per l’attività di testing è necessario un pc con connessione ad internet dato che il sistema è accessibile da remoto e il database si trova su un server dedicato.

Caratteristiche pc utilizzati:

|  |  |
| --- | --- |
| Marca | IBM |
| Modello | T60 |
| Processore | Intel Core Duo processor T2300 - 1.60 GHz |
| Ram | 2 GB |
| Schermo | 14” |
| Risoluzione | 1024x768 |

|  |  |
| --- | --- |
| Marca | Asus |
| Modello | X556UA-XO607T |
| Processore | Intel - i5-7200U |
| Ram | 8 GB |
| Schermo | 15,6” |
| Risoluzione | 1280x720 |

|  |  |
| --- | --- |
| Marca | Apple |
| Modello | MacBook Pro |
| Processore | Intel Core i5 dual-core a 2,5GHz |
| Ram | 16 GB |
| Schermo | 13” |
| Risoluzione | 2560 x 1600 |

# Test cases

## Gestione registrazione

### Registrazione

#### Category partition

|  |  |
| --- | --- |
| **Parametro:** *nome*  **Formato:** *[A-Za-z]* | |
| *Lunghezza[ln]* | 1. <3 or >30 [error\_ln] 2. 3<nome<30 [ln\_pass] |
| *Formato[fn]* | 1. formato errato [error\_fn] 2. formato giusto [fn\_pass] |
| *Assenza parametri [apn]* | 1. campo vuoto [error\_apn] 2. campo compilato [apn\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *cognome*  **Formato:** *[A-Za-z]* | |
| *Lunghezza[lc]* | 1. <3 or >30 [error\_lc] 2. 3<cognome<30 [lc\_pass] |
| *Formato[fc]* | 1. formato errato [error\_fc] 2. formato giusto [fc\_pass] |
| *Assenza parametri [apc]* | 1. campo vuoto [error\_apc] 2. campo compilato [apc\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *indirizzo*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[li]* | 1. <3 or >50 [error\_li] 2. 3<indirizzo<50 [li\_pass] |
| *Formato[fi]* | 1. formato errato [error\_fi] 2. formato giusto [fi\_pass] |
| *Assenza parametri[api]* | 1. campo vuoto [error\_api] 2. campo compilato [api\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *codice fiscale*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[lcf]* | 1. 16< or >16 [error\_lcf] 2. 16 [lcf\_pass] |
| *Formato[fcf]* | 1. formato errato [error\_fcf] 2. formato giusto [fcf\_pass] |
| *Assenza parametri[apcf]* | 1. campo vuoto [error\_apcf] 2. campo compilato [apcf\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *telefono*  **Formato:** *[0-9]* | |
| *Lunghezza[lt]* | 1. 10< or >11 [error\_lt] 2. 10<telefono<11 [lt\_pass] |
| *Formato[ft]* | 1. formato errato [error\_ft] 2. formato giusto [ft\_pass] |
| *Assenza parametri[apt]* | 1. campo vuoto [error\_apt] 2. campo compilato [apt\_pass] |

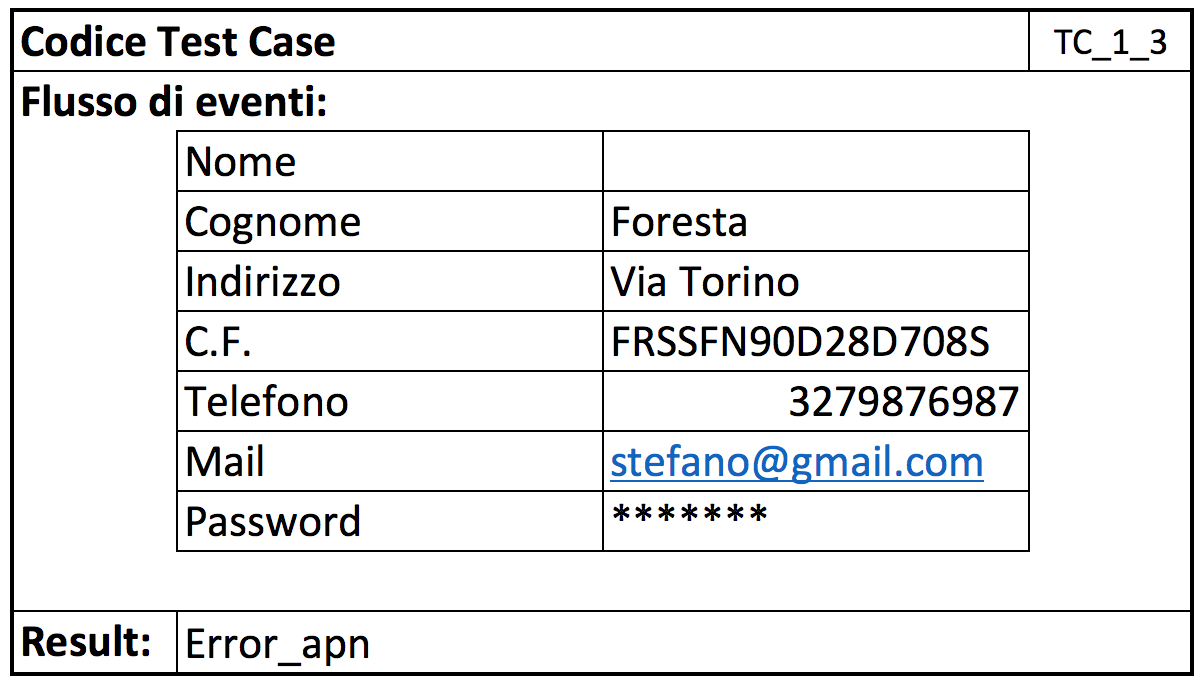
|  |  |
| --- | --- |
| **Parametro:** *mail*  **Formato:** *[A-Za-z0-9]@[a-z]* | |
| *Lunghezza[lm]* | 1. 5< or >30 [error\_lm] 2. 5<mail<30 [lm\_pass] |
| *Formato[fm]* | 1. formato errato [error\_fm] 2. formato giusto [fm\_pass] |
| *Assenza parametri[apm]* | 1. campo vuoto [error\_apm] 2. campo compilato [apm\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *password*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[lp]* | 1. 7< or >20 [error\_lp] 2. 7<password<20 [lp\_pass] |
| *Formato[fp]* | 1. formato errato [error\_fp] 2. formato giusto [fp\_pass] |
| *Assenza parametri[app]* | 1. campo vuoto [error\_app] 2. campo compilato [app\_pass] |

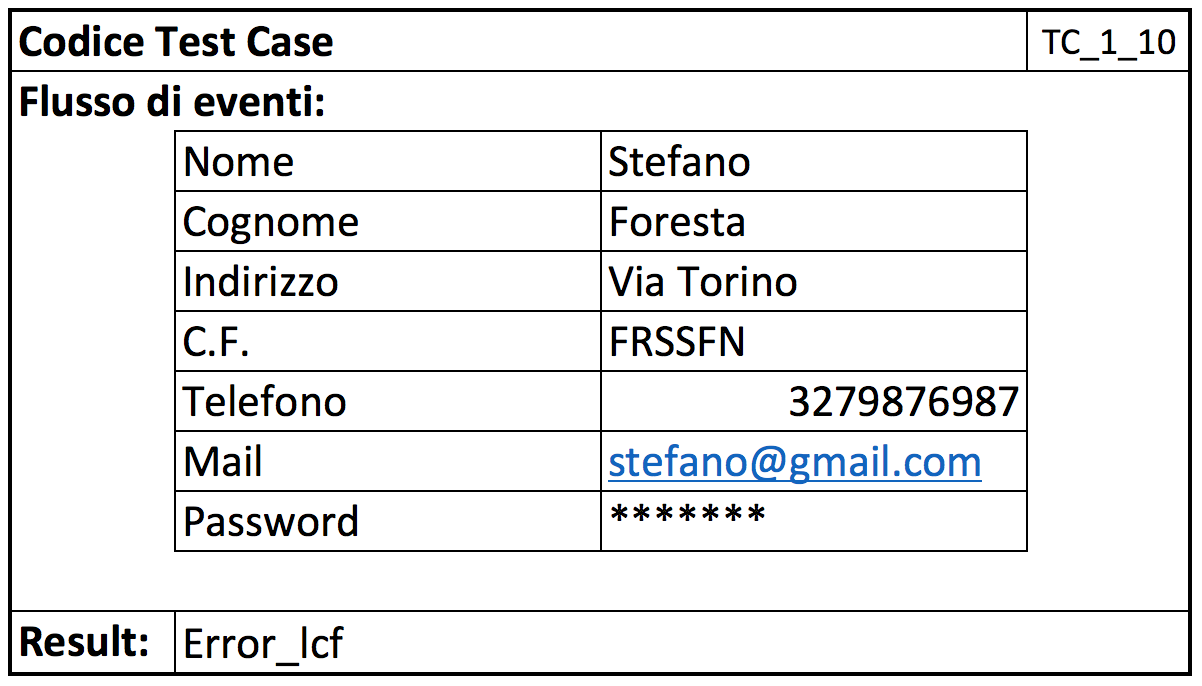
|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_1\_1 | ln\_a | Errore |
| Test\_case\_1\_2 | ln\_b, fn\_a | Errore |
| Test\_case\_1\_3 | ln\_b, fn\_b, apn\_a | Errore |
| Test\_case\_1\_4 | ln\_b, fn\_b, apn\_b, lc\_a | Errore |
| Test\_case\_1\_5 | ln\_b, fn\_b, apn\_b, lc\_b, fc\_a | Errore |
| Test\_case\_1\_6 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_a | Errore |
| Test\_case\_1\_7 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_a | Errore |
| Test\_case\_1\_8 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_a | Errore |
| Test\_case\_1\_9 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_a | Errore |
| Test\_case\_1\_10 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_a | Errore |
| Test\_case\_1\_11 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_a | Errore |
| Test\_case\_1\_12 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_a | Errore |
| Test\_case\_1\_13 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_a | Errore |
| Test\_case\_1\_14 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_b, ft\_a | Errore |
| Test\_case\_1\_15 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_b, ft\_b, apt\_a | Errore |
| Test\_case\_1\_16 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_b, ft\_b, apt\_b, lm\_a | Errore |
| Test\_case\_1\_17 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_b, ft\_b, apt\_b, lm\_b, fm\_a | Errore |
| Test\_case\_1\_18 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_b, ft\_b, apt\_b, lm\_b, fm\_b, apm\_a | Errore |
| Test\_case\_1\_19 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_b, ft\_b, apt\_b, lm\_b, fm\_b, apm\_b, lp\_a | Errore |
| Test\_case\_1\_20 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_b, ft\_b, apt\_b, lm\_b, fm\_b, apm\_b, lp\_b, fp\_a | Errore |
| Test\_case\_1\_21 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_b, ft\_b, apt\_b, lm\_b, fm\_b, apm\_b, lp\_b, fp\_b, app\_a | Errore |
| Test\_case\_1\_22 | ln\_b, fn\_b, apn\_b, lc\_a, fc\_b, apc\_b, li\_b, fi\_b, api\_b, lcf\_b, fcf\_b, apcf\_b, lt\_b, ft\_b, apt\_b, lm\_b, fm\_b, apm\_b, lp\_b, fp\_b, app\_b | Corretto |

#### Specification

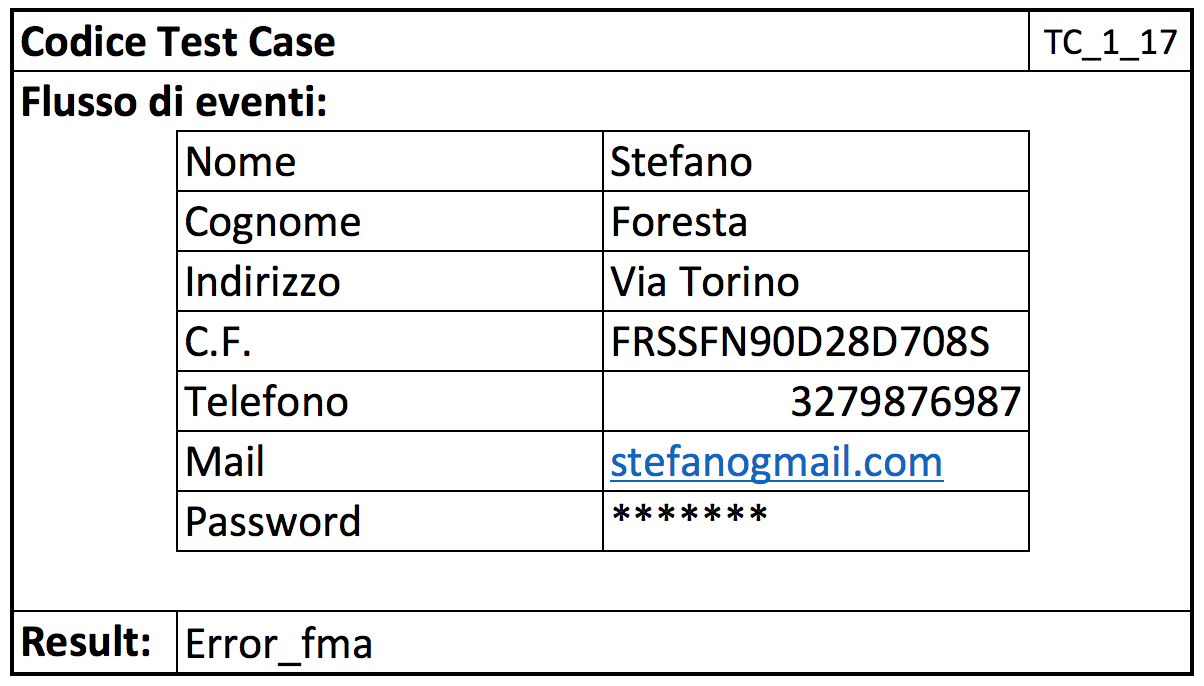
…



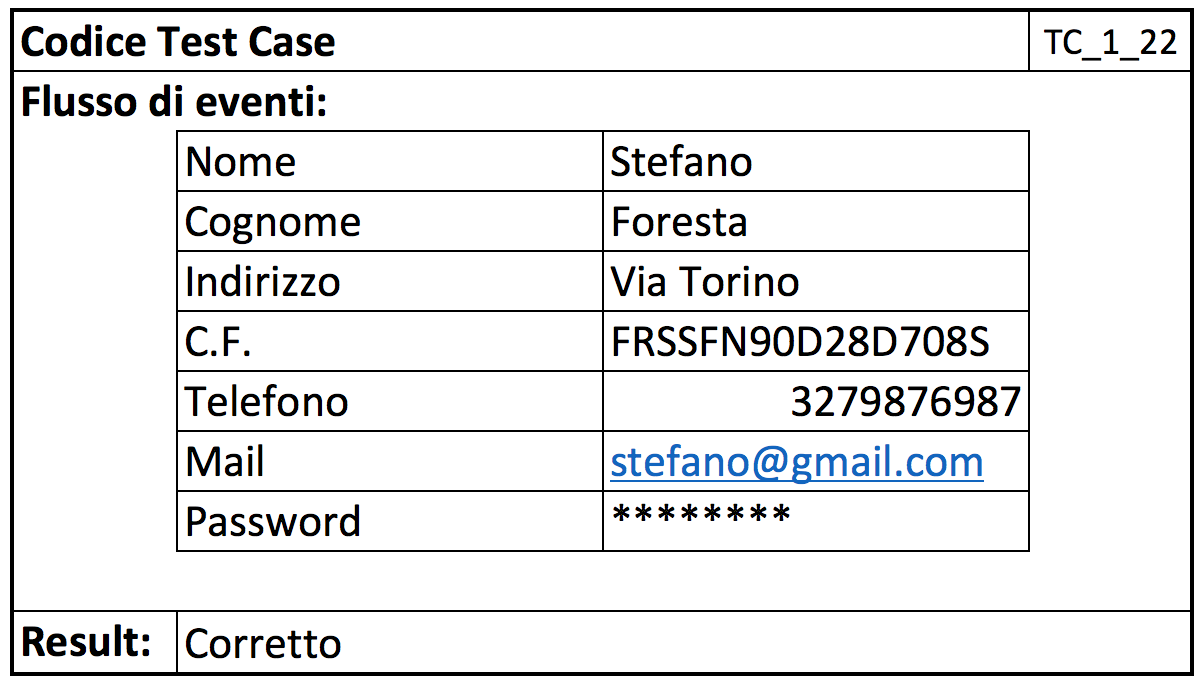
…



…



…



## Gestione autenticazione

### Login

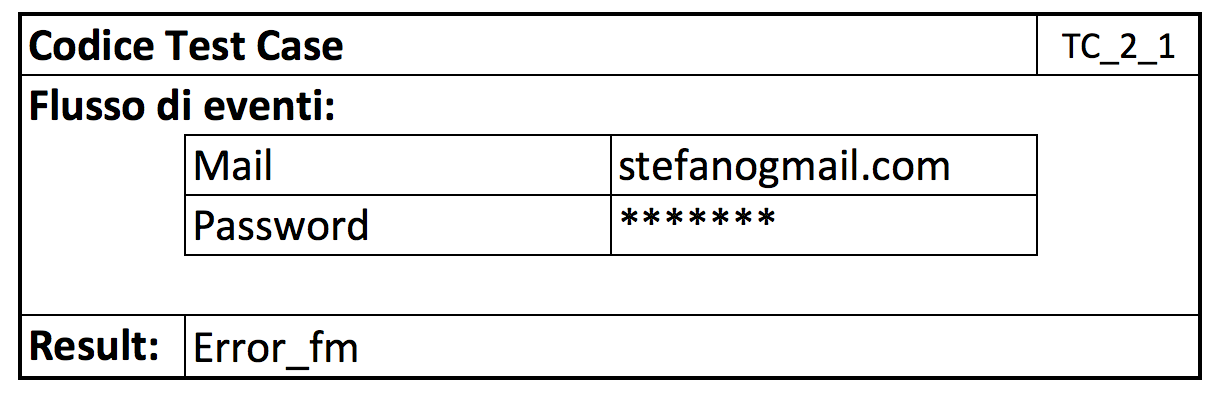
#### Category partition

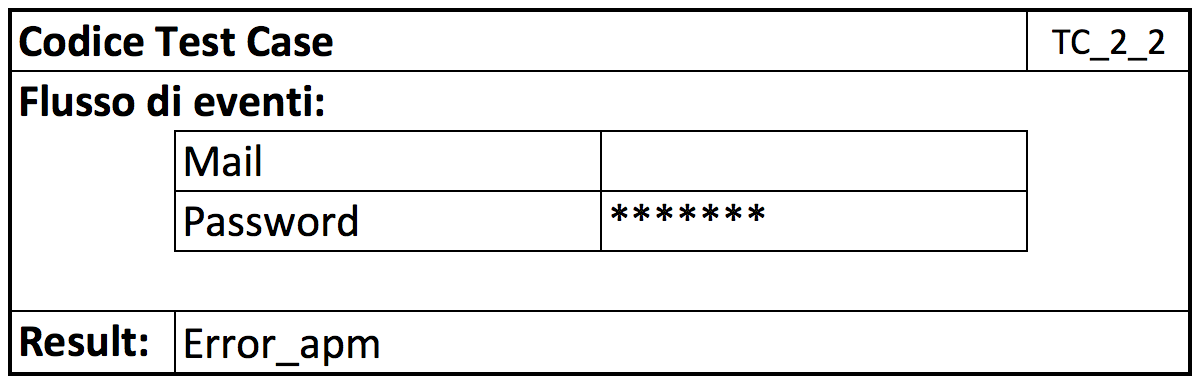
|  |  |
| --- | --- |
| **Parametro:** *mail*  **Formato:** *[A-Za-z0-9]@[a-z]* | |
| *Formato[fm]* | 1. formato errato [error\_fm] 2. formato giusto [fm\_pass] |
| *Assenza parametri[apm]* | 1. campo vuoto [error\_apm] 2. campo compilato [apm\_pass] |

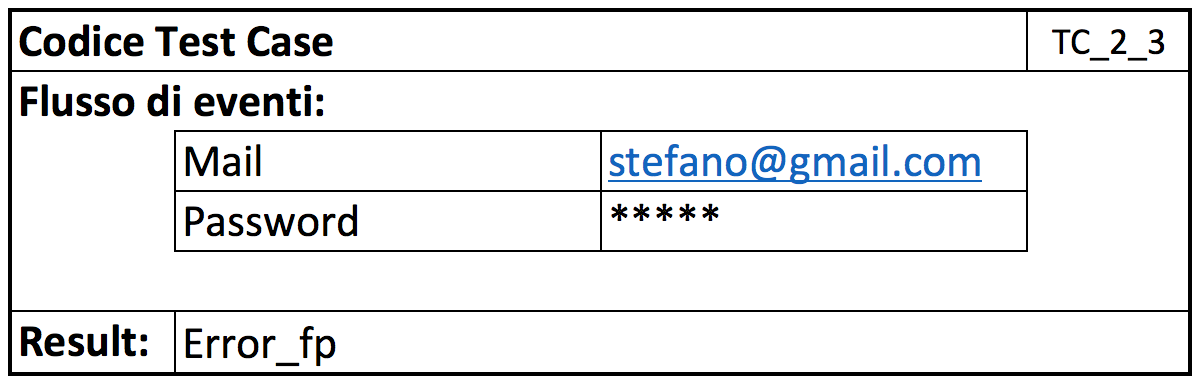
|  |  |
| --- | --- |
| **Parametro:** *password*  **Formato:** *[A-Za-z0-9]* | |
| *Formato[fp]* | 1. formato errato [error\_fp] 2. formato giusto [fp\_pass] |
| *Assenza parametri[app]* | 1. campo vuoto [error\_app] 2. campo compilato [app\_pass] |

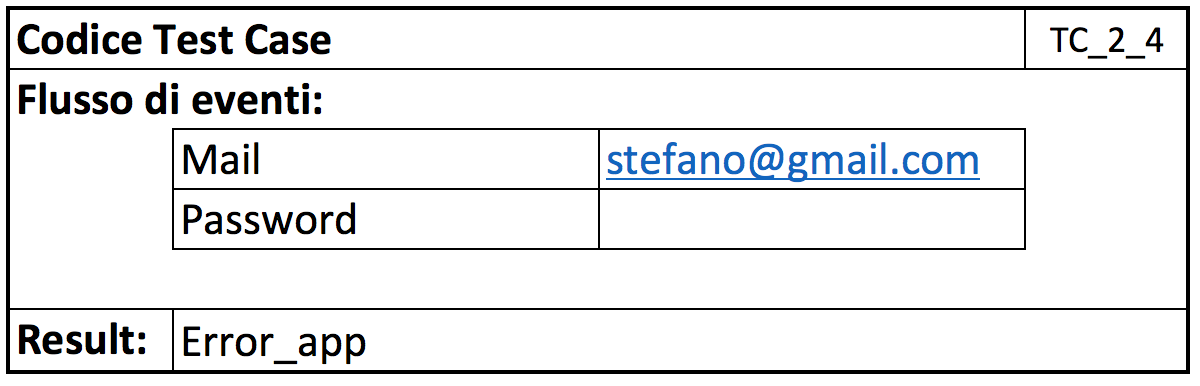
|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_2\_1 | fm\_a | Errore |
| Test\_case\_2\_2 | fm\_b, apm\_a | Errore |
| Test\_case\_2\_3 | fm\_b, apm\_a, fp\_a | Errore |
| Test\_case\_2\_4 | fm\_b, apm\_a, fp\_b, app\_a | Errore |
| Test\_case\_2\_5 | fm\_b, apm\_a, fp\_b, app\_b | Corretto |

#### Specification











## Gestione riparazioni

### Richiesta preventivo

#### Category partition

|  |  |
| --- | --- |
| **Parametro:** *descrizione*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[ld]* | 1. 1< or >200 [error\_ld] 2. 1<descrizione<200 [ld\_pass] |
| *Formato[fd]* | 1. formato errato [error\_fd] 2. formato giusto [fd\_pass] |
| *Assenza parametri[apd]* | 1. campo vuoto [error\_apd] 2. campo compilato [apd\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_3\_1 | ld\_a | Errore |
| Test\_case\_3\_2 | ld\_b, fd\_a | Errore |
| Test\_case\_3\_3 | ld\_b, fd\_b | Errore |
| Test\_case\_3\_4 | ld\_b, fd\_b, apd\_a | Errore |
| Test\_case\_3\_5 | ld\_b, fd\_b, apd\_b | Corretto |

### Invio preventivo

#### Category partition

|  |  |
| --- | --- |
| **Parametro:** *prezzo*  **Formato:** *[0-9]* | |
| *Lunghezza[lpr]* | 1. 1< or >5 [error\_l*pr*] 2. 1<descrizione<5 [l*pr*\_pass] |
| *Formato[fpr]* | 1. formato errato [error\_f*pr*] 2. formato giusto [f*pr*\_pass] |
| *Assenza parametri[appr]* | 1. campo vuoto [error\_ap*pr*] 2. campo compilato [ap*pr*\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_4\_1 | l*pr*\_a | Errore |
| Test\_case\_4\_2 | l*pr*\_b, f*pr*\_a | Errore |
| Test\_case\_4\_3 | l*pr*\_b, f*pr*\_b | Errore |
| Test\_case\_4\_4 | l*pr*\_b, fd*pr*, ap*pr*\_a | Errore |
| Test\_case\_4\_5 | l*pr*\_b, f*pr*\_b, ap*pr*\_b | Corretto |

### Accetazione/rifiuto preventivo

#### Category partition

|  |  |
| --- | --- |
| **Parametro:** *accetta*  **Formato:** *[checked - unchecked]* | |
| *Selezionato[sa]* | 1. non selezionato 2. selezionato [se sr non è selezionato] |

|  |  |
| --- | --- |
| **Parametro:** *rifiuta*  **Formato:** *[checked - unchecked]* | |
| *Selezionato[sr]* | 1. non selezionato 2. selezionato [se sa non è selezionato] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_5\_1 | sa\_a, sr\_a | Errore |
| Test\_case\_5\_2 | sa\_b, sr\_a | Corretto |
| Test\_case\_5\_3 | sa\_a, sr\_b | Corretto |

## Gestione acquisti

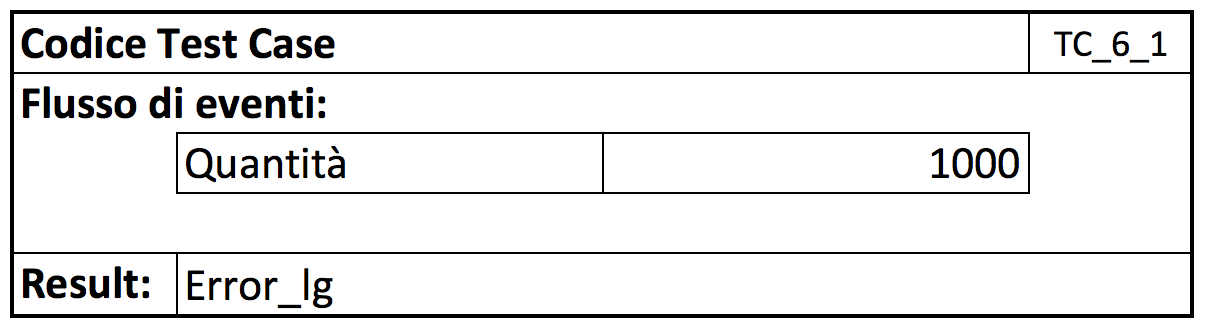
### Carrello

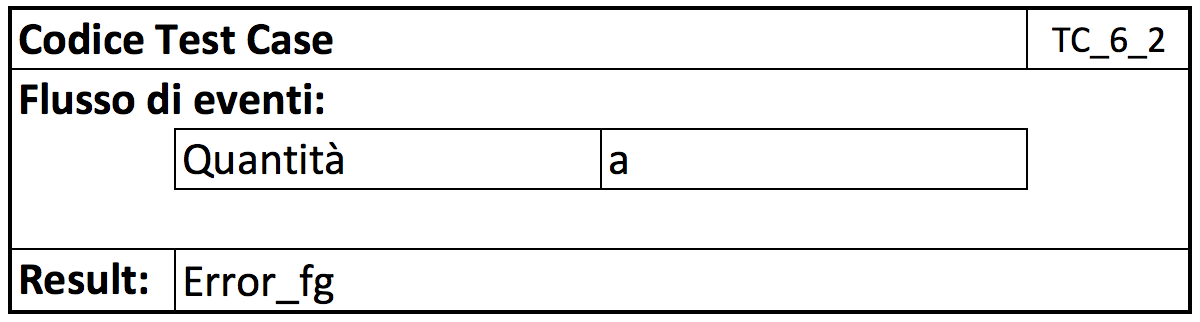
#### Category partition

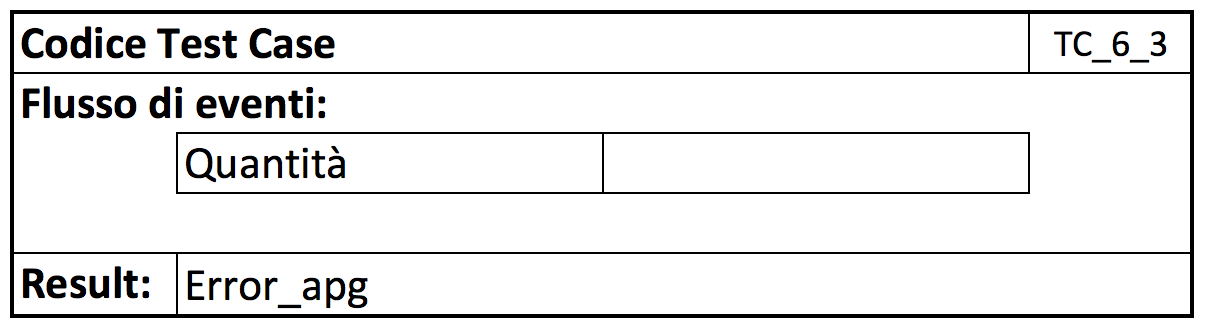
|  |  |
| --- | --- |
| **Parametro:** *quantità*  **Formato:** *[0-9]* | |
| *Lunghezza[lq]* | 1. 1< or >3 [error\_l*q*] 2. 1<quantità<3 [l*q*\_pass] |
| *Formato[fq]* | 1. formato errato [error\_f*q*] 2. formato giusto [f*q*\_pass] |
| *Assenza parametri[apq]* | 1. campo vuoto [error\_ap*q*] 2. campo compilato [ap*q*\_pass] |
| *Quantità maggiore della disponibilità [qmdq]* | 1. quantità maggiore [error\_qmdq] 2. quantità regolare [qmdq\_pass] |

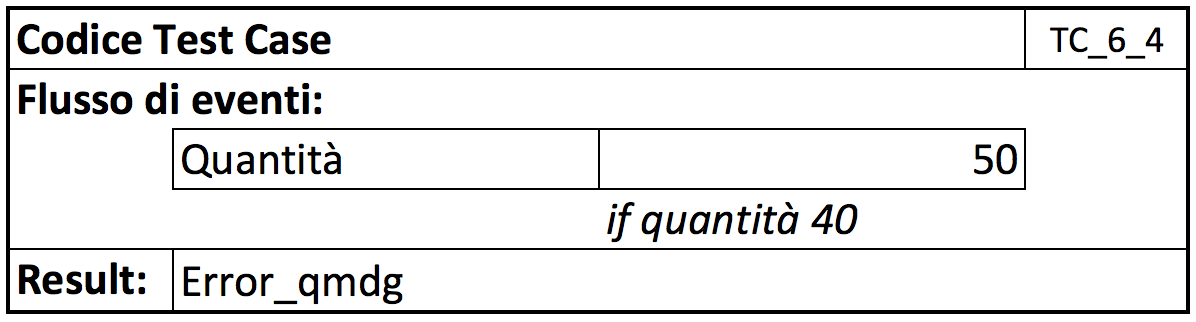
|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_6\_1 | l*q*\_a | Errore |
| Test\_case\_6\_2 | l*q*\_b, f*q*\_a | Errore |
| Test\_case\_6\_3 | l*q*\_b, f*q\_b*, ap*q*\_a | Errore |
| Test\_case\_6\_4 | l*q*\_b, f*q*\_b, ap*q*\_b, qmdq\_a | Errore |
| Test\_case\_6\_5 | l*q*\_b, f*q*\_b, ap*q*\_b, qmdq\_b | Corretto |

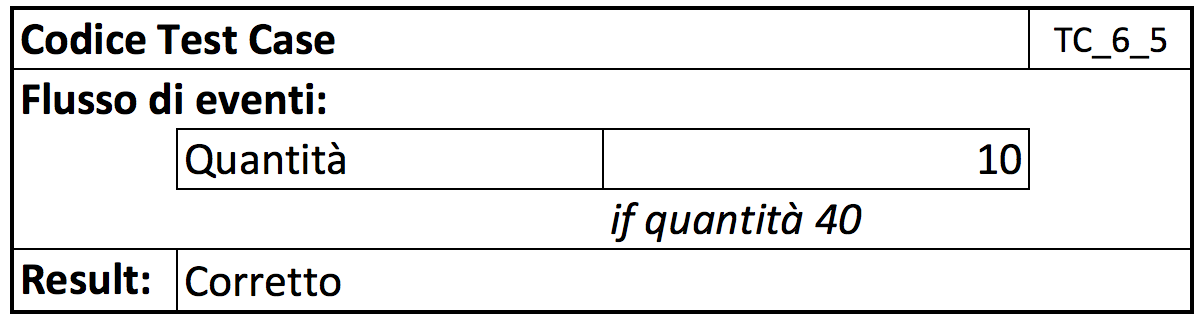
#### Specification











## Gestione magazzino

### Inserimento prodotti

#### Category partition

|  |  |
| --- | --- |
| **Parametro:** *smartphone*  **Formato:** *[checked - unchecked]* | |
| *Selezionato[ssm]* | 1. non selezionato 2. selezionato [se sco non è selezionato] |

|  |  |
| --- | --- |
| **Parametro:** *computer*  **Formato:** *[checked - unchecked]* | |
| *Selezionato[sco]* | 1. non selezionato 2. selezionato [se ssm non è selezionato] |

|  |  |
| --- | --- |
| **Parametro:** *modello*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[lmo]* | 1. 1< or >15 [error\_lmo] 2. 1<modello<15 [lmo\_pass] |
| *Formato[fmo]* | 1. formato errato [error\_mo] 2. formato giusto [fmo\_pass] |
| *Assenza parametri[apmo]* | 1. campo vuoto [error\_mo] 2. campo compilato [apmo\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *marca*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[lma]* | 1. 1< or >15 [error\_lma] 2. 1<marca<15 [lma\_pass] |
| *Formato[fma]* | 1. formato errato [error\_ma] 2. formato giusto [fma\_pass] |
| *Assenza parametri[apma]* | 1. campo vuoto [error\_ma] 2. campo compilato [apma\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *descrizione*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[lde]* | 1. 1< or >1500 [error\_lde] 2. 1<descrizione<1500 [lde\_pass] |
| *Formato[fde]* | 1. formato errato [error\_de] 2. formato giusto [fde\_pass] |
| *Assenza parametri[apde]* | 1. campo vuoto [error\_de] 2. campo compilato [apde\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *prezzo acquisto*  **Formato:** *[0-9]* | |
| *Lunghezza[lpa]* | 1. 1< or >5 [error\_lpa] 2. 1<prezzo acquisto<5 [lpa\_pass] |
| *Formato[fpa]* | 1. formato errato [error\_pa] 2. formato giusto [fpa\_pass] |
| *Assenza parametri[appa]* | 1. campo vuoto [error\_pa] 2. campo compilato [appa\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *prezzo vendita*  **Formato:** *[0-9]* | |
| *Lunghezza[lpv]* | 1. 1< or >5 [error\_lpv] 2. 1<prezzo vendita<5 [lpv\_pass] |
| *Formato[fpv]* | 1. formato errato [error\_pv] 2. formato giusto [fpv\_pass] |
| *Assenza parametri[appv]* | 1. campo vuoto [error\_pv] 2. campo compilato [appv\_pass] |

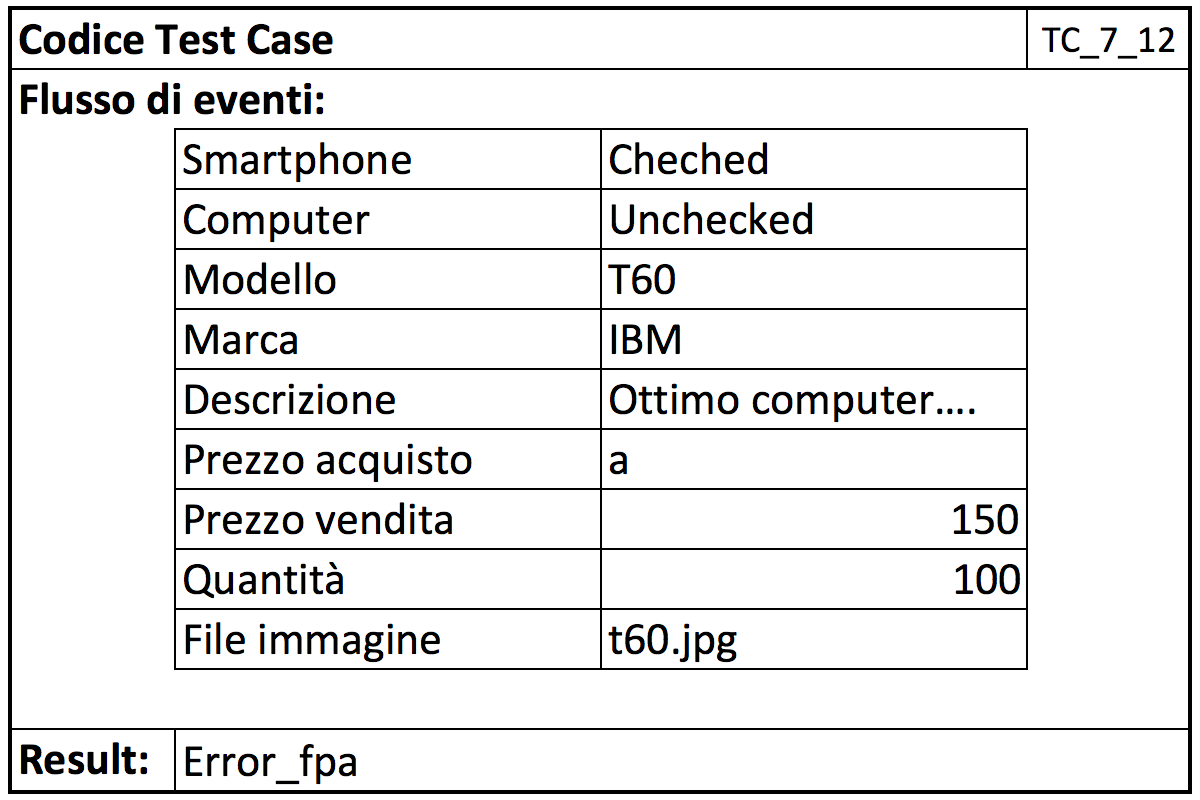
|  |  |
| --- | --- |
| **Parametro:** *quantità*  **Formato:** *[0-9]* | |
| *Lunghezza[lqu]* | 1. 1< or >5 [error\_lqu] 2. 1<quantità<5 [lqu\_pass] |
| *Formato[fqu]* | 1. formato errato [error\_qu] 2. formato giusto [fqu\_pass] |
| *Assenza parametri[apqu]* | 1. campo vuoto [error\_qu] 2. campo compilato [apqu\_pass] |

|  |  |
| --- | --- |
| **Parametro:** *file immagine*  **Formato:** *[.jpg, .jpeg, .png, .gif]* | |
| *Duplicata[di]* | 1. si[error\_di] 2. no[di\_pass] |
| *Formato[fi]* | 1. formato errato [error\_fi] 2. formato giusto [fi\_pass] |
| *Assenza file[afi]* | 1. campo vuoto [error\_afi] 2. formato giusto [afi\_pass] |
| *Grandezza[gi]* | (in byte)   1. 1< or >5000000 [error\_gi] 2. 1<immagine<5000000[gi\_pass] |

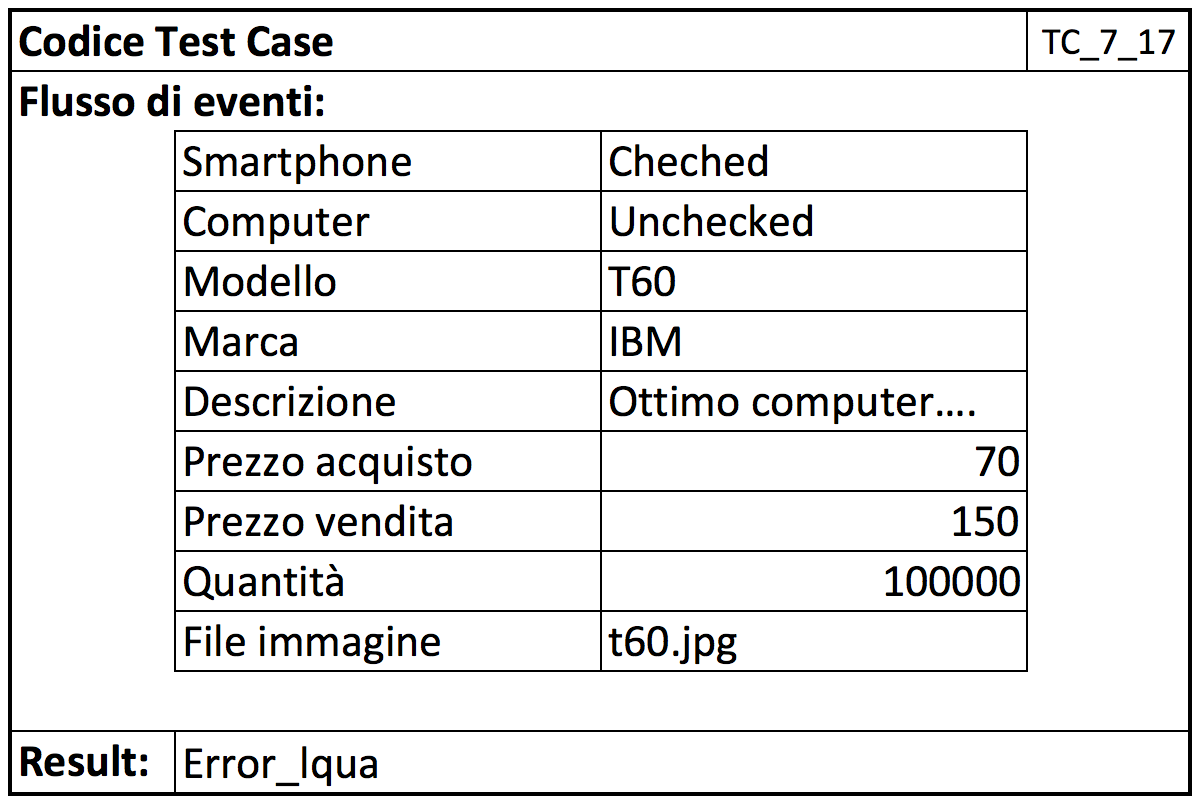
|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_7\_1 | ssm\_a, sco\_a | Errore |
| Test\_case\_7\_2 | ssm\_b, (or sco\_b), lmo\_a | Errore |
| Test\_case\_7\_3 | ssm\_b, (or sco\_b), lmo\_b, fmo\_a | Errore |
| Test\_case\_7\_4 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_a | Errore |
| Test\_case\_7\_5 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_a | Errore |
| Test\_case\_7\_6 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_a | Errore |
| Test\_case\_7\_7 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_a | Errore |
| Test\_case\_7\_8 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_a | Errore |
| Test\_case\_7\_9 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_a | Errore |
| Test\_case\_7\_10 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_a | Errore |
| Test\_case\_7\_11 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_a | Errore |
| Test\_case\_7\_12 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_a | Errore |
| Test\_case\_7\_13 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_a | Errore |
| Test\_case\_7\_14 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_a | Errore |
| Test\_case\_7\_15 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_a | Errore |
| Test\_case\_7\_16 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_b, appv\_a | Errore |
| Test\_case\_7\_17 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_b, appv\_b, lqu\_a | Errore |
| Test\_case\_7\_18 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_b, appv\_b, lqu\_b, fqu\_a | Errore |
| Test\_case\_7\_19 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_b, appv\_b, lqu\_b, fqu\_b, apqu\_a | Errore |
| Test\_case\_7\_20 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_b, appv\_b, lqu\_b, fqu\_b, apqu\_b, di\_a | Errore |
| Test\_case\_7\_21 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_b, appv\_b, lqu\_b, fqu\_b, apqu\_b, di\_b, fi\_a | Errore |
| Test\_case\_7\_22 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_b, appv\_b, lqu\_b, fqu\_b, apqu\_b, di\_b, fi\_b, afi\_a | Errore |
| Test\_case\_7\_23 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_b, appv\_b, lqu\_b, fqu\_b, apqu\_b, di\_b, fi\_b, afi\_b, gi\_a | Errore |
| Test\_case\_7\_24 | ssm\_b, (or sco\_b), lmo\_b, fmo\_b, apmo\_b, lma\_b, fma\_b, apma\_b, lde\_b, fde\_b, apde\_b, lpa\_b, fpa\_b, appa\_b, lpv\_b, fpv\_b, appv\_b, lqu\_b, fqu\_b, apqu\_b, di\_b, fi\_b, afi\_b, gi\_b | Corretto |

#### Specification

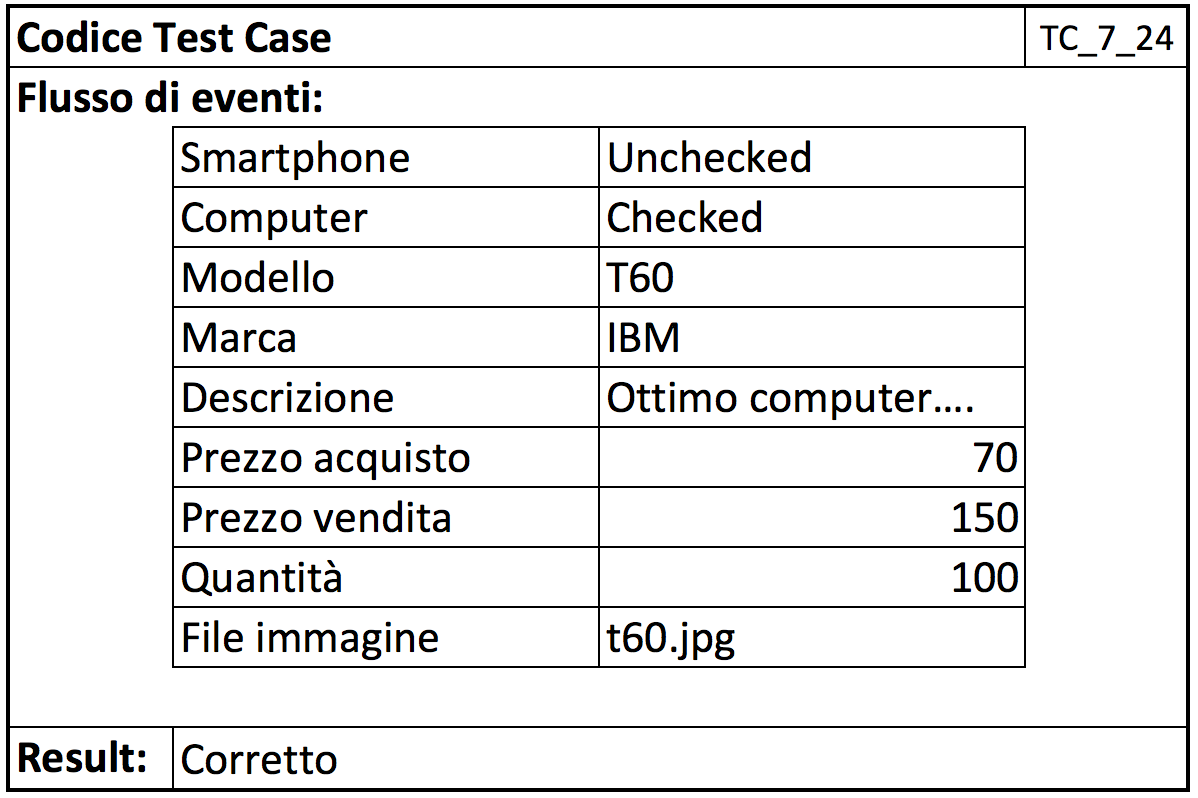
…



…



…



### Carico/scarico prodotti

#### Category partition

|  |  |
| --- | --- |
| **Parametro:** *quantità*  **Formato:** *[0-9]* | |
| *Lunghezza[lqn]* | 1. 1< or >5 [error\_lqn] 2. 1<quantità<5 [lqn\_pass] |
| *Formato[fqn]* | 1. formato errato [error\_fqn] 2. formato corretto [fqn\_pass] |
| *Assenza parametri[apqn]* | 1. campo vuoto [error\_apqn] 2. campo compilato [apqn\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_8\_1 | lqn\_a | Errato |
| Test\_case\_8\_2 | lqn\_b, fqn\_a | Errato |
| Test\_case\_8\_3 | lqn\_b, fqn\_b, apqn\_a | Errato |
| Test\_case\_8\_4 | lqn\_b, fqn\_b, apqn\_b | Corretto |

### Modifica prezzo

#### Category partition

|  |  |
| --- | --- |
| **Parametro:** *prezzo*  **Formato:** *[0-9]* | |
| *Lunghezza[lpz]* | 1. 1< or >5 [error\_lpz] 2. 1<quantità<5 [lpz\_pass] |
| *Formato[fpz]* | 1. formato errato [error\_fpz] 2. formato corretto [fpz\_pass] |
| *Assenza parametri[appz]* | 1. campo vuoto [error\_appz] 2. campo compilato [appz\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_9\_1 | lpz\_a | Errore |
| Test\_case\_9\_2 | lpz\_b, fpz\_a | Errore |
| Test\_case\_9\_3 | lpz\_b, fpz\_b, appz\_a | Errore |
| Test\_case\_9\_4 | lpz\_b, fpz\_b, appz\_b | Corretto |

## Gestione utenti

### Modifica dati [admin]

#### Category partition

##### Mail

|  |  |
| --- | --- |
| **Parametro:** *mail*  **Formato:** *[A-Za-z0-9]@[a-z]* | |
| *Lunghezza[lma]* | 1. 5< or >30 [error\_lma] 2. 5<mail<30 [lma\_pass] |
| *Formato[fma]* | 1. formato errato [error\_fma] 2. formato giusto [fma\_pass] |
| *Assenza parametri[apma]* | 1. campo vuoto [error\_apma] 2. campo compilato [apma\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_10\_1\_1\_1 | lma\_a | Errore |
| Test\_case\_10\_1\_1\_2 | lma\_b, fma\_a | Errore |
| Test\_case\_10\_1\_1\_3 | lma\_b, fma\_b, apma\_a | Errore |
| Test\_case\_10\_1\_1\_4 | lma\_b, fma\_b, apma\_b | Corretto |

##### Password

|  |  |
| --- | --- |
| **Parametro:** *password*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[lpsw]* | 1. 7< or >20 [error\_lpsw] 2. 7<password<20 [lpsw\_pass] |
| *Formato[fpsw]* | 1. formato errato [error\_fpsw] 2. formato giusto [fpsw\_pass] |
| *Assenza parametri[appsw]* | 1. campo vuoto [error\_appsw] 2. campo compilato [appsw\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_10\_1\_2\_1 | lps\_a | Errore |
| Test\_case\_10\_1\_2\_2 | lps\_b, fps\_a | Errore |
| Test\_case\_10\_1\_2\_3 | lps\_b, fps\_b, apps\_a | Errore |
| Test\_case\_10\_1\_2\_4 | lps\_b, fps\_b, apps\_b | Corretto |

### Modifica dati [clienti]

#### Category partition

##### Indirizzo

|  |  |
| --- | --- |
| **Parametro:** *indirizzo*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[lind]* | 1. <3 or >50 [error\_lind] 2. 3<indirizzo<50 [lind\_pass] |
| *Formato[find]* | 1. formato errato [error\_find] 2. formato giusto [find\_pass] |
| *Assenza parametri[apind]* | 1. campo vuoto [error\_apind] 2. campo compilato [apind\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_10\_2\_1\_1 | lind\_a | Errore |
| Test\_case\_10\_2\_1\_2 | lind\_b, find\_a | Errore |
| Test\_case\_10\_2\_1\_3 | lind\_b, find\_b, apind\_a | Errore |
| Test\_case\_10\_2\_1\_4 | lind\_b, find\_b, apind\_b | Corretto |

##### Telefono

|  |  |
| --- | --- |
| **Parametro:** *telefono*  **Formato:** *[0-9]* | |
| *Lunghezza[ltlf]* | 1. 10< or >11 [error\_ltlf] 2. 10<telefono<11 [ltlf\_pass] |
| *Formato[ftlf]* | 1. formato errato [error\_ftlf] 2. formato giusto [ftlf\_pass] |
| *Assenza parametri[aptlf]* | 1. campo vuoto [error\_aptlf] 2. campo compilato [aptlf\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_10\_2\_2\_1 | ltlf\_a | Errore |
| Test\_case\_10\_2\_2\_2 | ltlf \_b, ftlf\_a | Errore |
| Test\_case\_10\_2\_2\_3 | ltlf \_b, ftlf \_b, apltf\_a | Errore |
| Test\_case\_10\_2\_2\_4 | ltlf \_b, ftlf \_b, apltf\_b | Corretto |

##### Mail

|  |  |
| --- | --- |
| **Parametro:** *mail*  **Formato:** *[A-Za-z0-9]@[a-z]* | |
| *Lunghezza[lmail]* | 1. 5< or >30 [error\_lmail] 2. 5<mail<30 [lmail\_pass] |
| *Formato[fmail]* | 1. formato errato [error\_fmail] 2. formato giusto [fmail\_pass] |
| *Assenza parametri[apmail]* | 1. campo vuoto [error\_apmail] 2. campo compilato [apmail\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_10\_2\_3\_1 | lmail\_a | Errore |
| Test\_case\_10\_2\_3\_2 | lmail \_b, fmail\_a | Errore |
| Test\_case\_10\_2\_3\_3 | lmail \_b, fmail \_b, apmail\_a | Errore |
| Test\_case\_10\_2\_3\_4 | lmail \_b, fmail \_b, apmail\_b | Corretto |

##### Password

|  |  |
| --- | --- |
| **Parametro:** *password*  **Formato:** *[A-Za-z0-9]* | |
| *Lunghezza[lpass]* | 1. 7< or >20 [error\_lpass] 2. 7<password<20 [lpass\_pass] |
| *Formato[fpass]* | 1. formato errato [error\_fpass] 2. formato giusto [fpass\_pass] |
| *Assenza parametri[appass]* | 1. campo vuoto [error\_appass] 2. campo compilato [appass\_pass] |

|  |  |  |
| --- | --- | --- |
| **Codice** | **Combinazione** | **Esito** |
| Test\_case\_10\_2\_4\_1 | lpass\_a | Errore |
| Test\_case\_10\_2\_4\_2 | lpass \_b, fpass\_a | Errore |
| Test\_case\_10\_2\_4\_3 | lpass \_b, fpass \_b, appass\_a | Errore |
| Test\_case\_10\_2\_4\_4 | lpass \_b, fpass \_b, appass\_b | Corretto |

# Pianificazione del testing

Il team per il testing deve essere composto da persone che hanno una completa e approfondita conoscenza del sistema e delle tecniche di testing con i documenti associati, quali Test plan e Test case specification. Tali tecniche devono essere applicate nei tempi, nel budget e nei vincoli di qualità stabiliti. Il team dedicato al controllo della qualità è responsabile dell’attività di testing e quindi della ricerca di fault. Il sistema revisionato è poi testato nuovamente non solo per verificare se gli errori trovati in precedenza sono stati eliminati ma soprattutto per verificare che non ne siano stati introdotti dei nuovi. L’attività di testing è fondamentale nello sviluppo di un sistema software in quanto la mancanza di tale attività o una cattiva interpretazione di essa può portare al completo fallimento del sistema. Data l’importanza del testing ne risulta fondamentale la schedulazione.

## Determinazione dei ruoli

Il team di testing è composto da 4 persone esterne

* Giuseppe
* Umberto
* Pierpaolo
* Salvatore

I quali sono delegati al testing funzionale mentre i vari test di unità sono stati affidati agli sviluppatori per alleggerire il carico del team di testing e velocizzare eventuali modifiche.

## Determinazione dei rischi

I rischi di un completo fallimento verranno minimizzati effettuando una pianificazione verticale delle attività di testing funzionale. Questo permetterà in caso di ritardi, dovuti ad una grande quantità di failure trovati, di rilasciare meno funzionalità del previsto, ma completamente testate. Inoltre tale pianificazione ridurrà notevolmente la produzione di driver e stub, evitando l’introduzione di nuovi errori, dovuti all’implementazione di tali componenti.

## Organizzazione delle attività di testing

Le attività di testing verranno organizzate secondo uno schema che effettuerà una divisione funzionale di tipo verticale. In questo modo al termine di ogni attività si avrà una funzionalità completamente testata nei suoi livelli gerarchici. I vantaggi principali sono che in caso di ritardi dovuti al ritrovamento di numerosi failure il sistema verrà rilasciato con meno componenti, ma interamente testate e funzionanti.

# Stress and Time Testing

Lo stress testing è un particolare collaudo del software che permette di verificare l’efficienza del sistema. Attraverso l’utilizzo di un software (per questo test è stato utilizzato *Webserver Stress Tool* di Paessle) proveremo a far collassare il server che ospita “I-tech” sovraccaricando le sue risorse con una serie di richieste http.

## Test click - 250 utenti

Gli URL verranno cliccati in maniera random da parte dell’utente.

Le pagine sottoposte al test sono:

* Index
* Registrazione
* Info riparazione
* Prodotti
  + Computer
  + Smartphone

### 10 click al secondo

#### Run test until 10 click per user

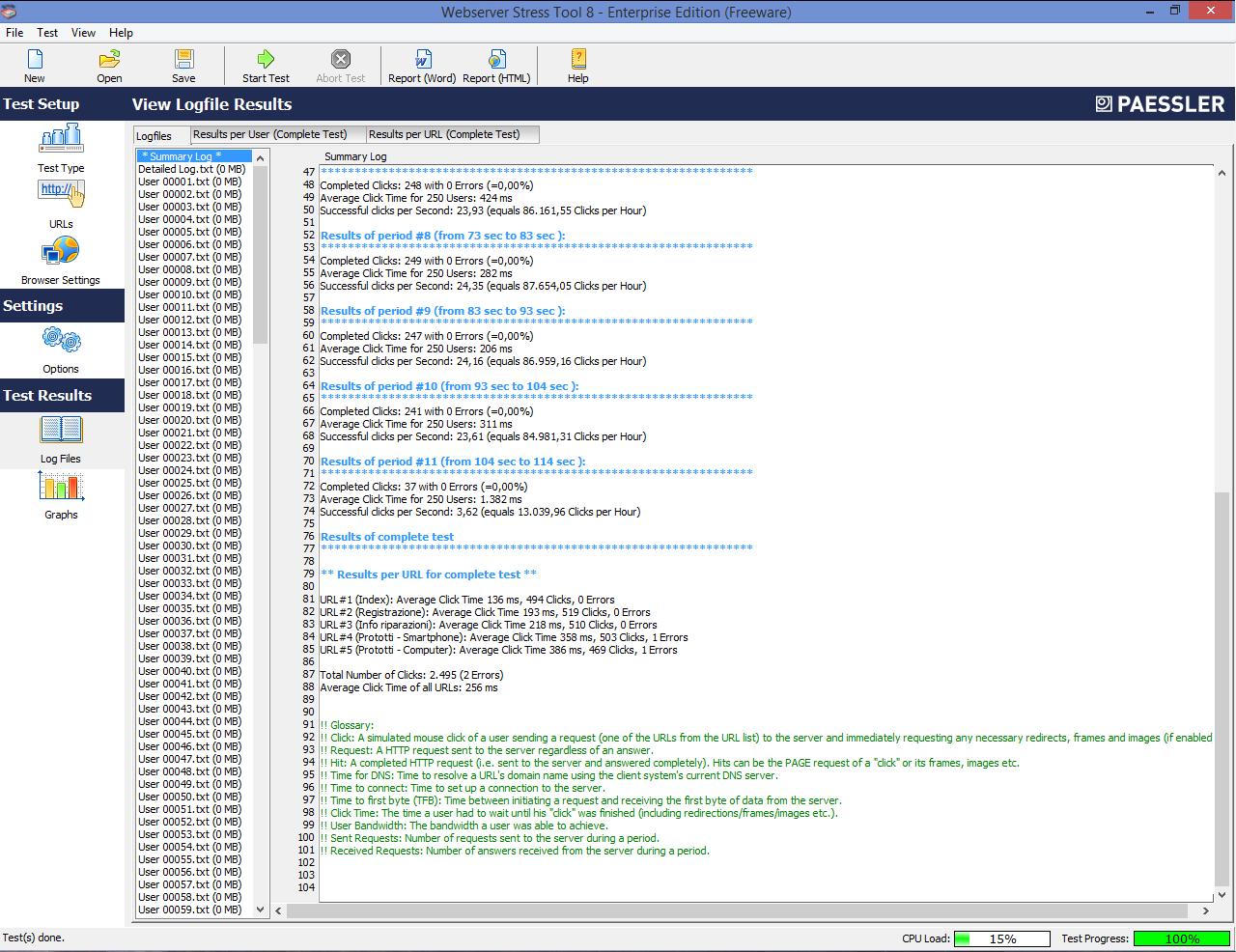
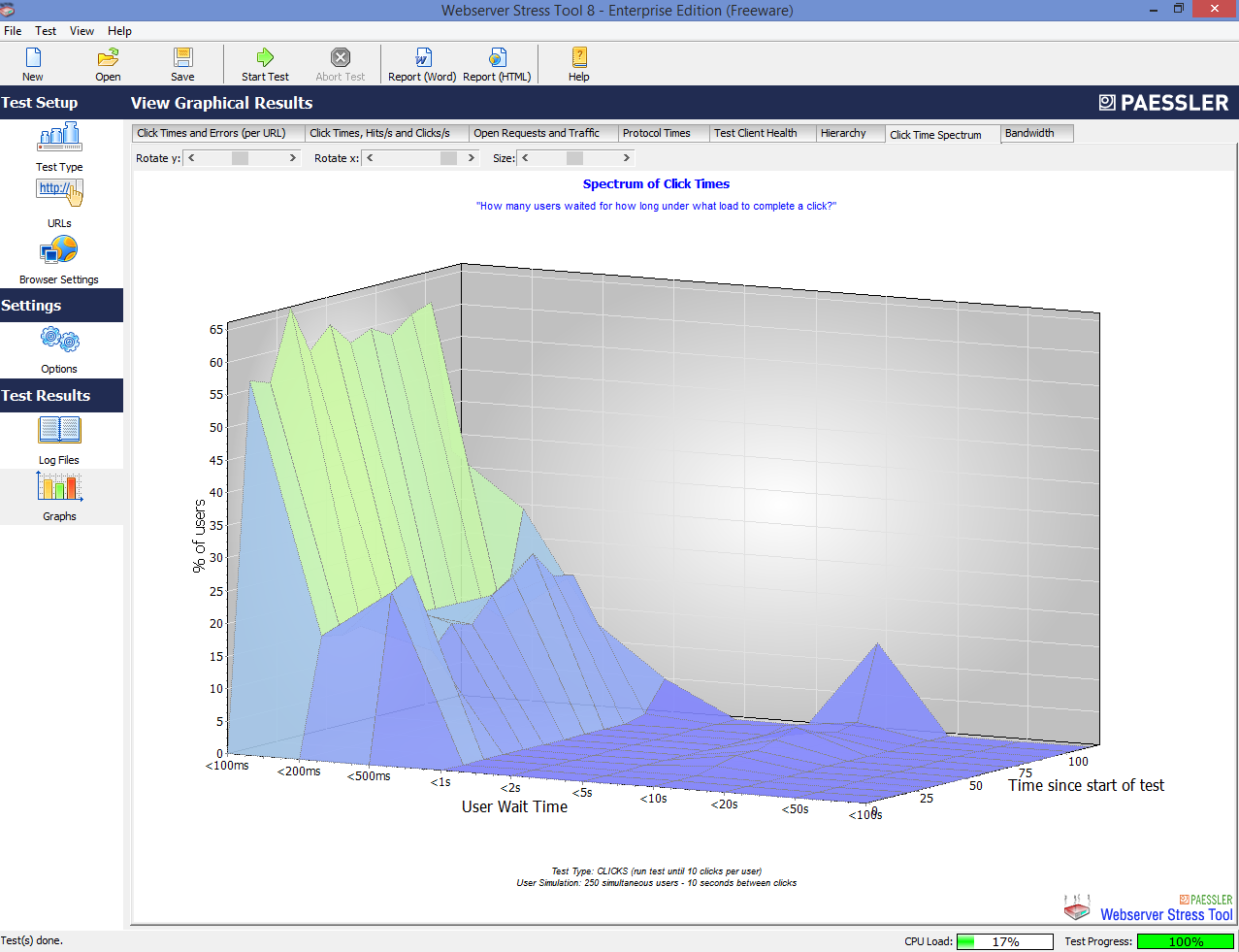


Grafico tridimensionale

I tre assi rappresentano:

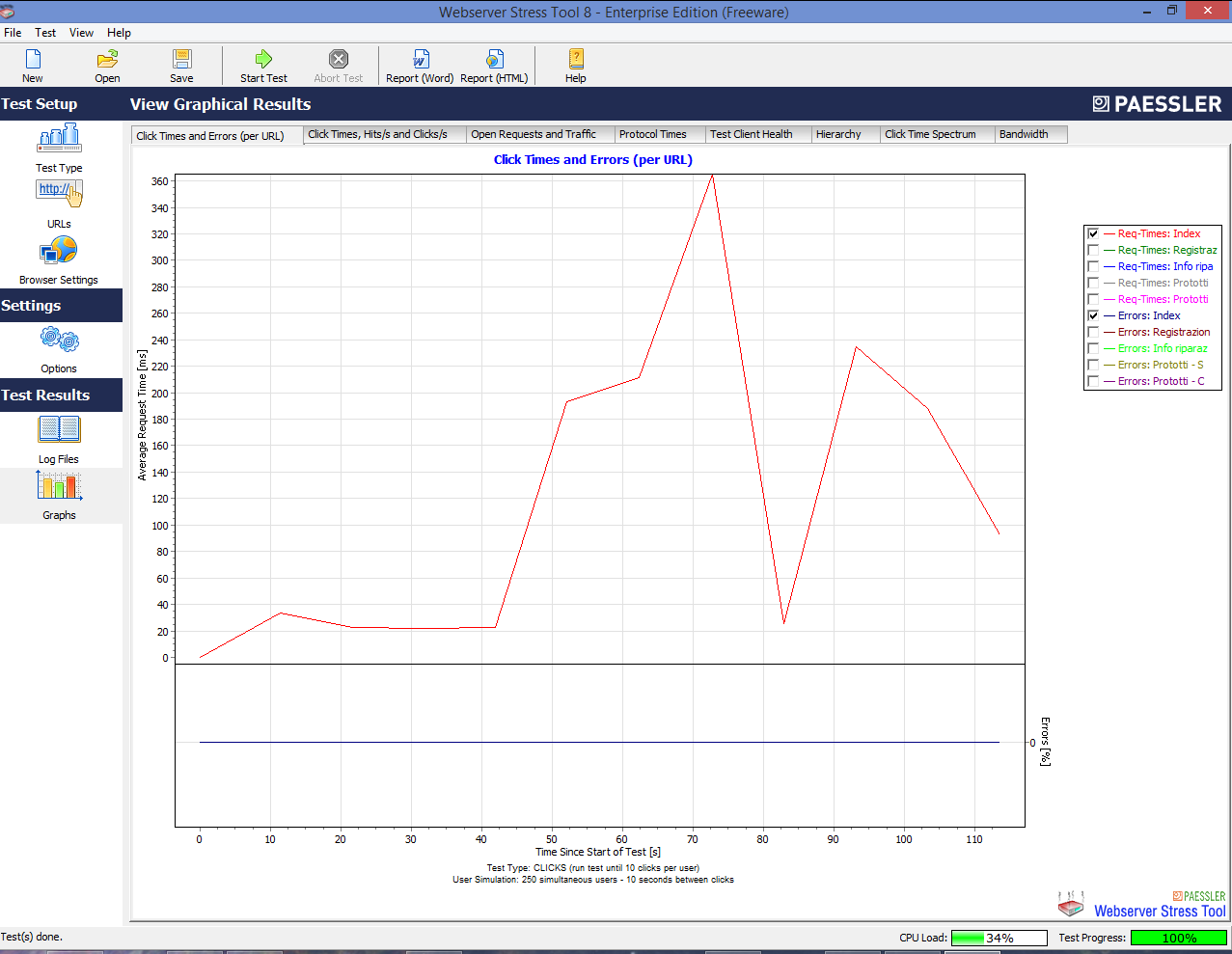
* percentuale di utenti che simulano le richieste
* tempo di attesa di ogni utente
* tempo di esecuzione del test



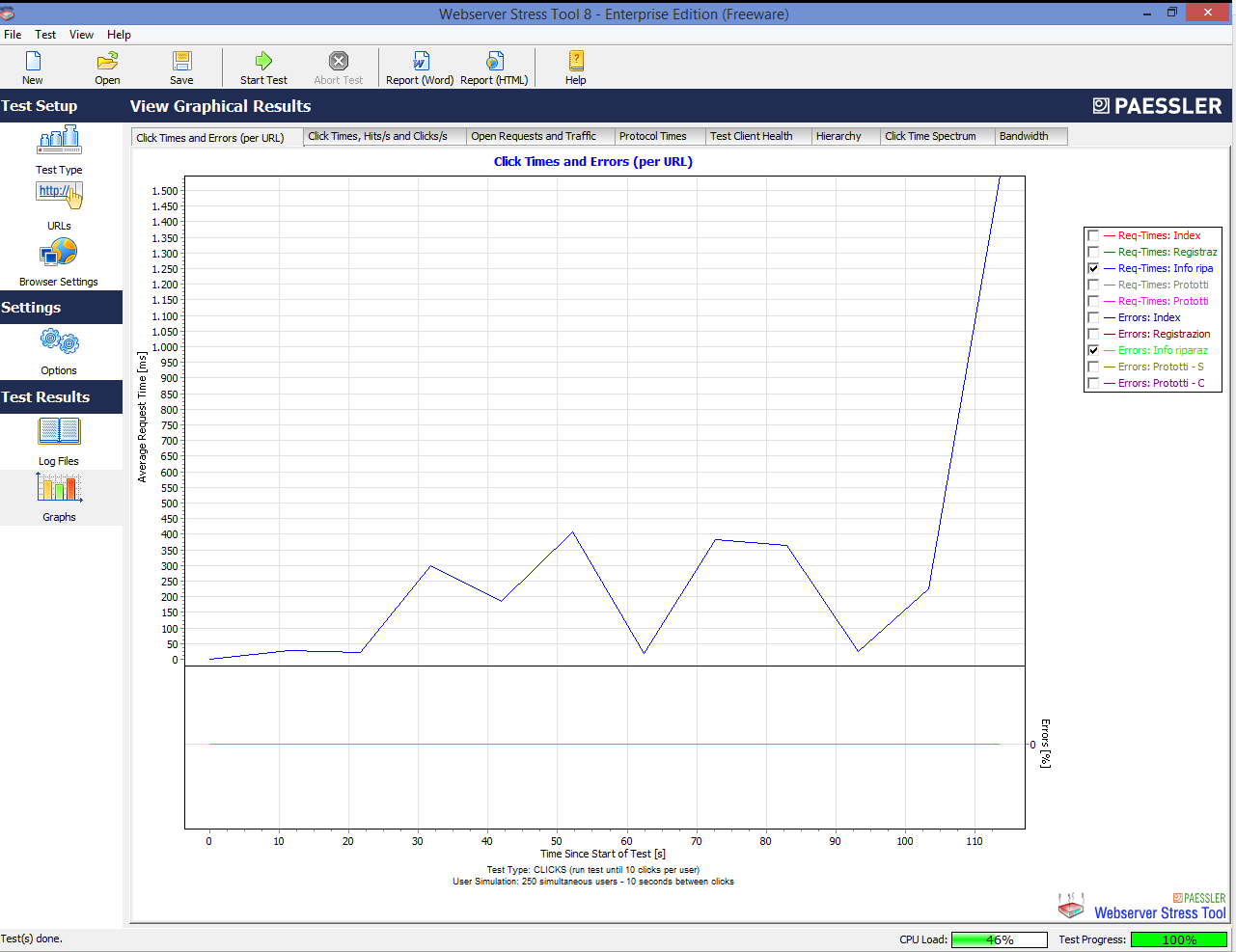
##### Grafico bidimensionale

Mostra l’andamento del tempo medio di richieste all’avanzare del test insieme ai possibili errori:

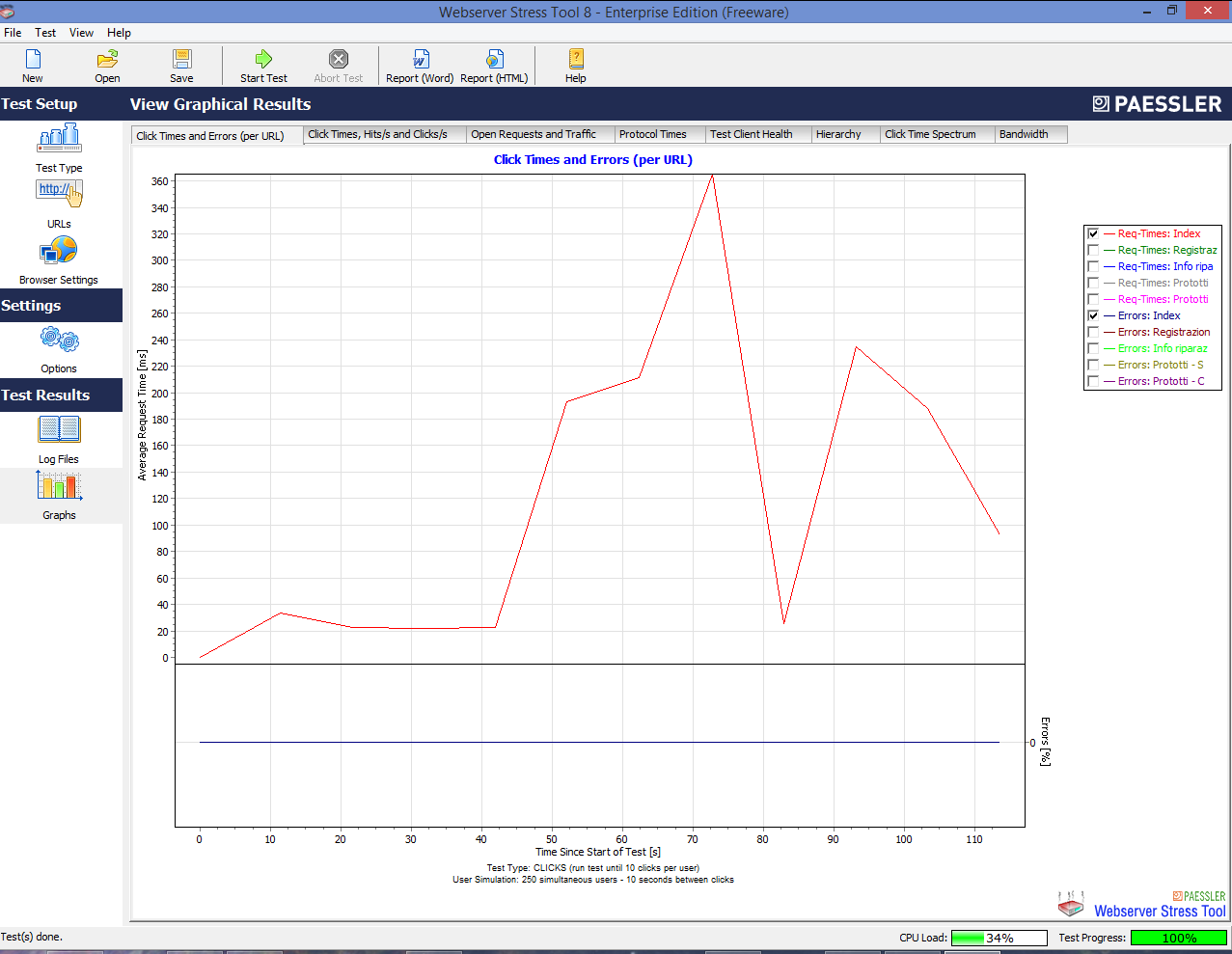
###### Index



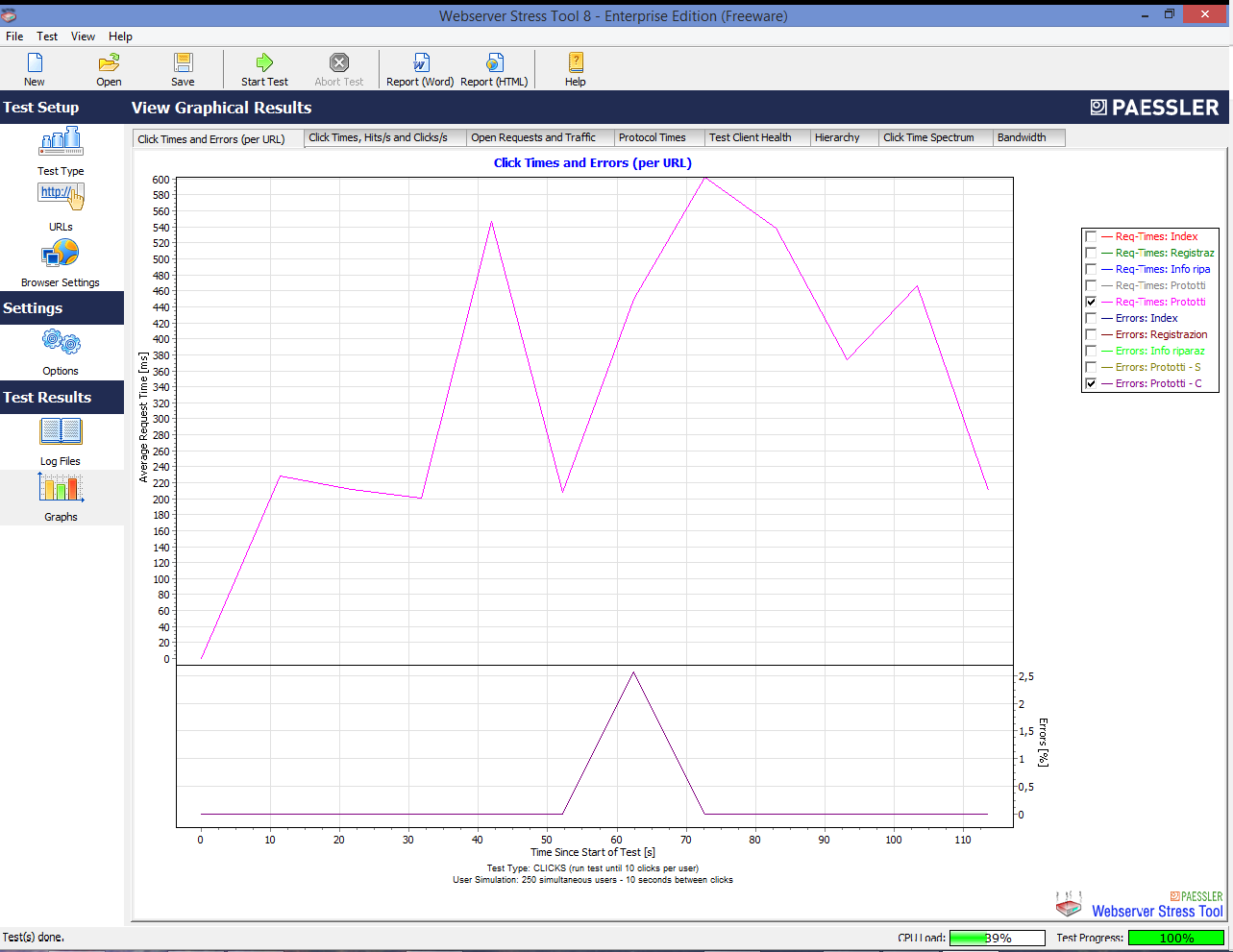
###### Info riparazioni



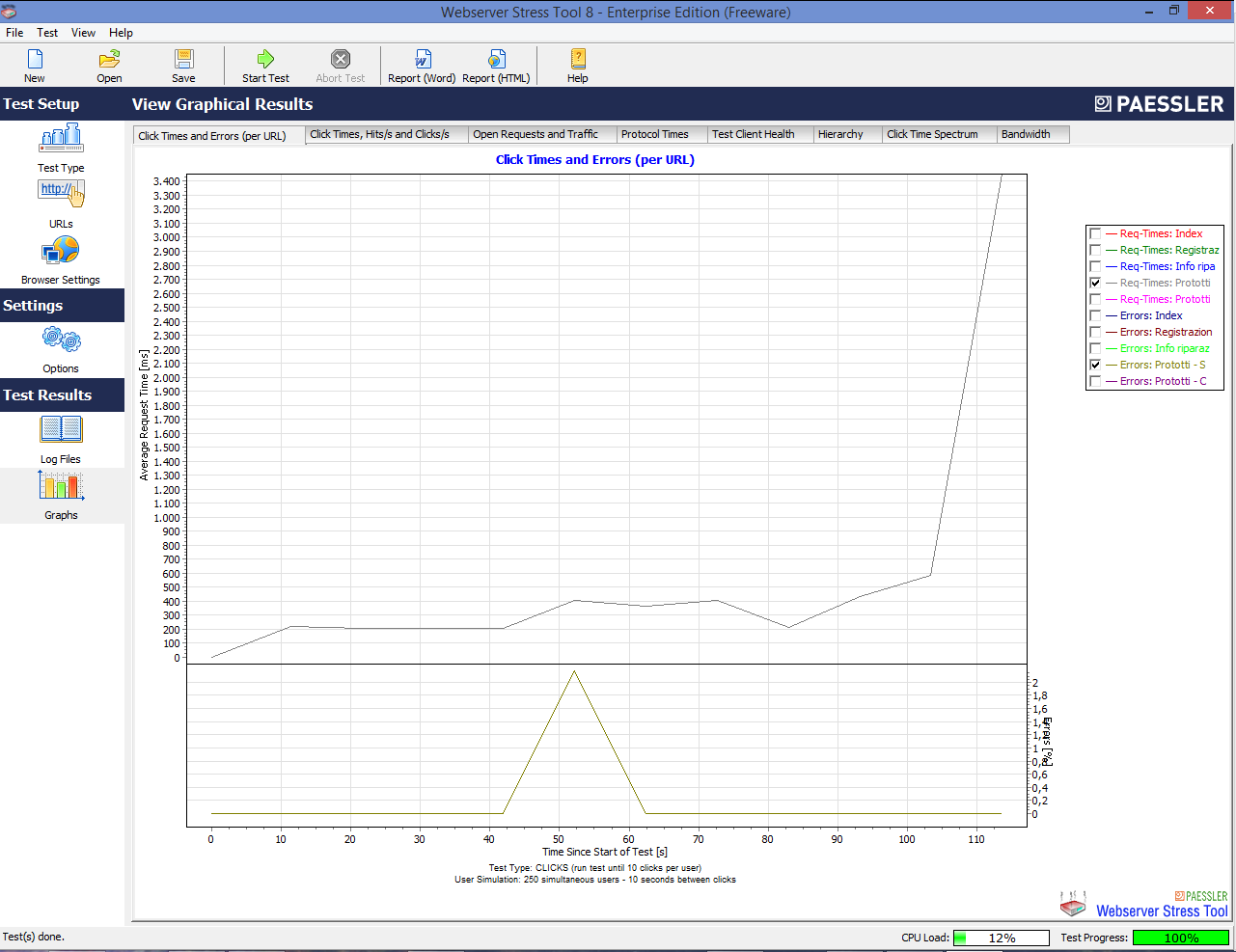
###### Registrazione



###### Prodotti – Computer



###### Prodotti – Smartphone



## Test click - 1000 utenti

### 100 click al secondo

#### Run test until 100 click per user

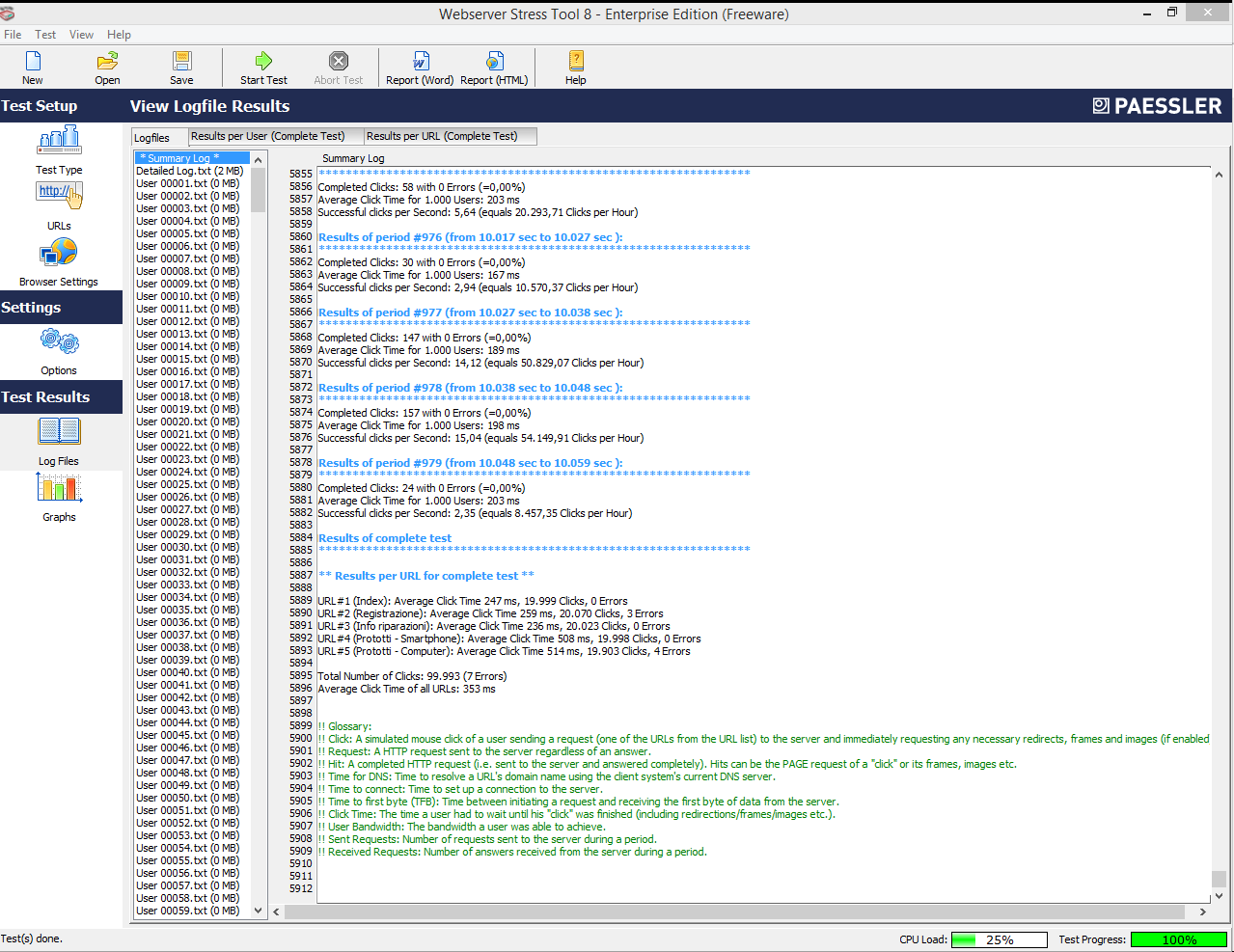


Grafico tridimensionale

I tre assi rappresentano:

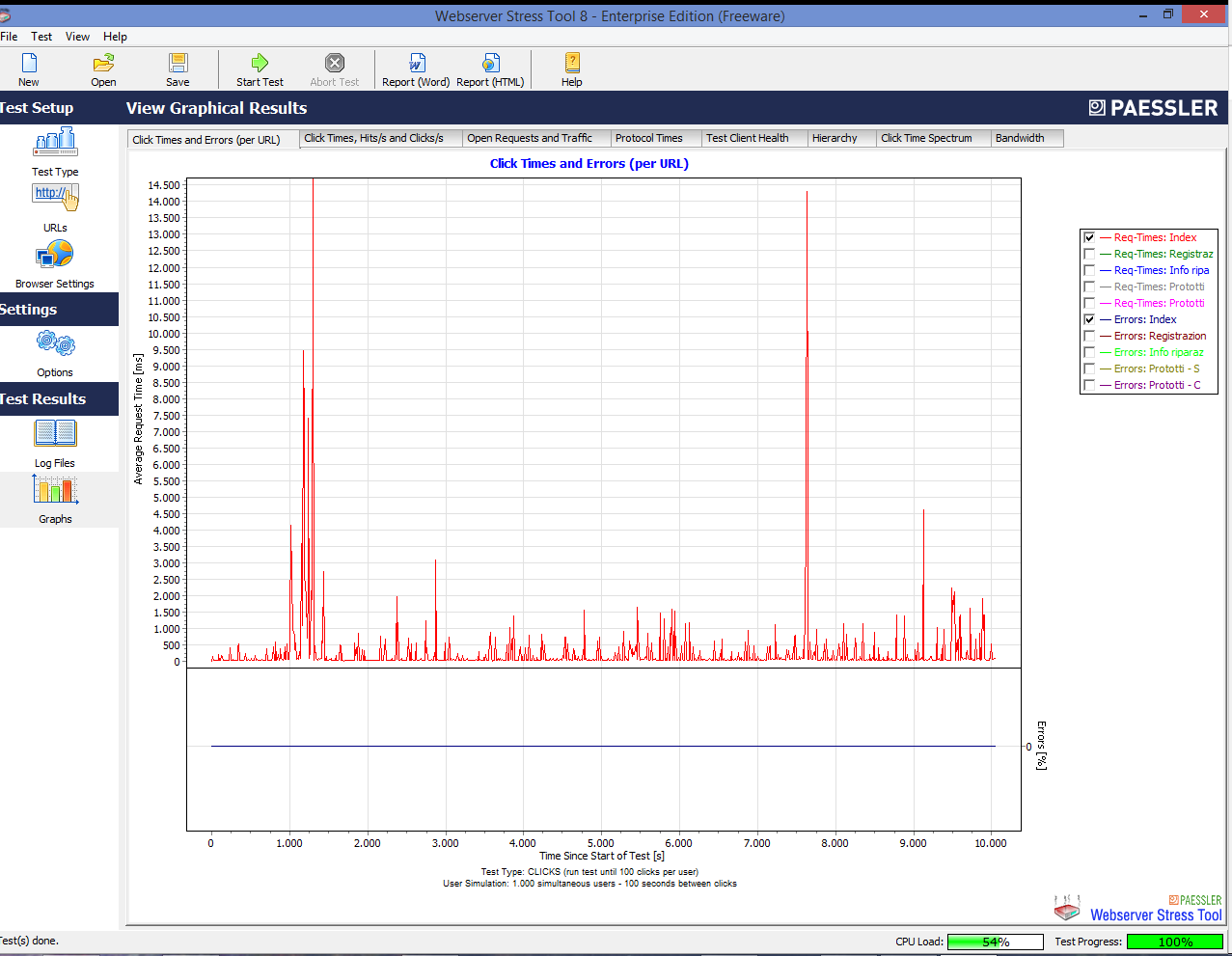
* percentuale di utenti che simulano le richieste
* tempo di attesa di ogni utente
* tempo di esecuzione del test



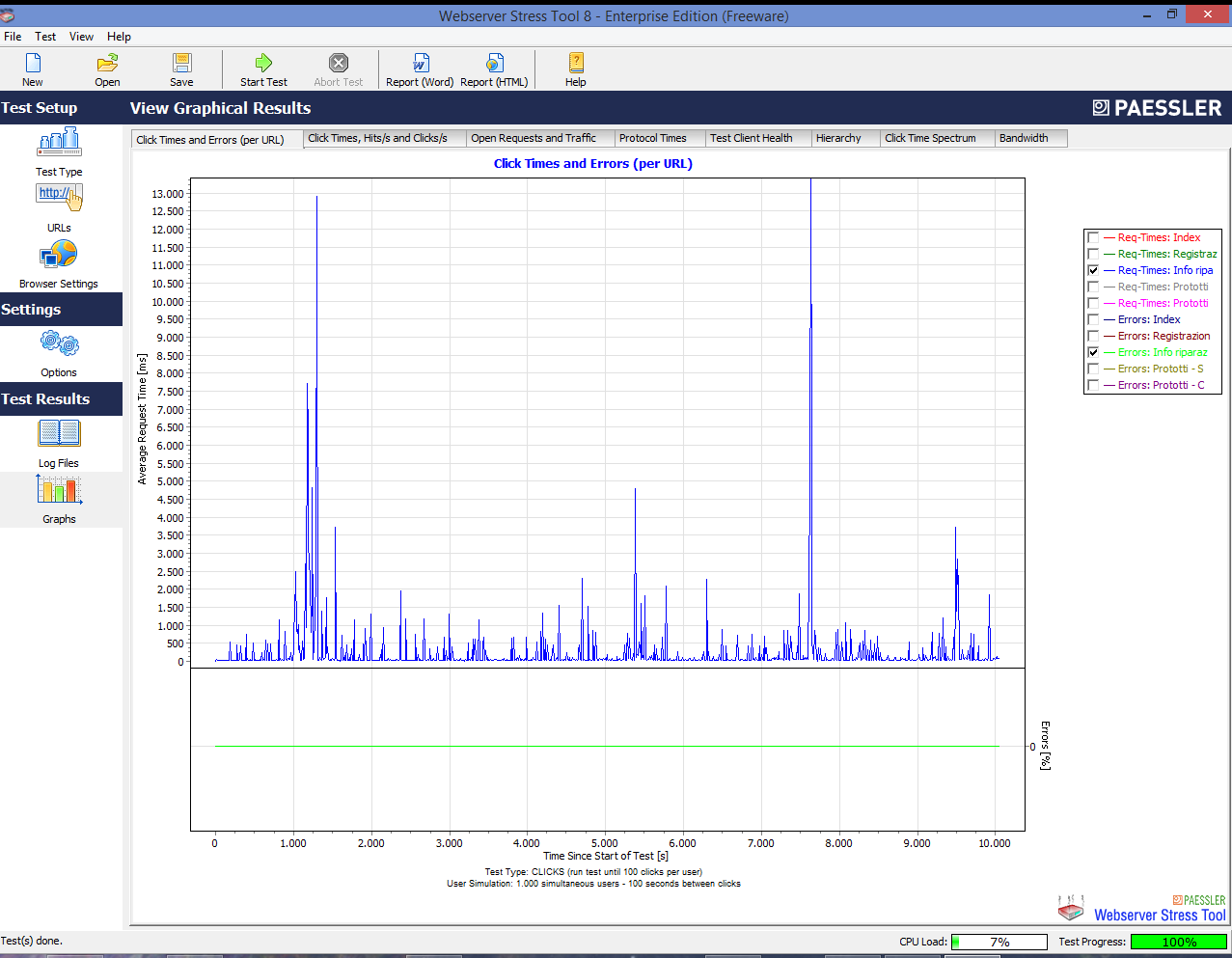
##### Grafico bidimensionale

Mostra l’andamento del tempo medio di richieste all’avanzare del test insieme ai possibili errori:

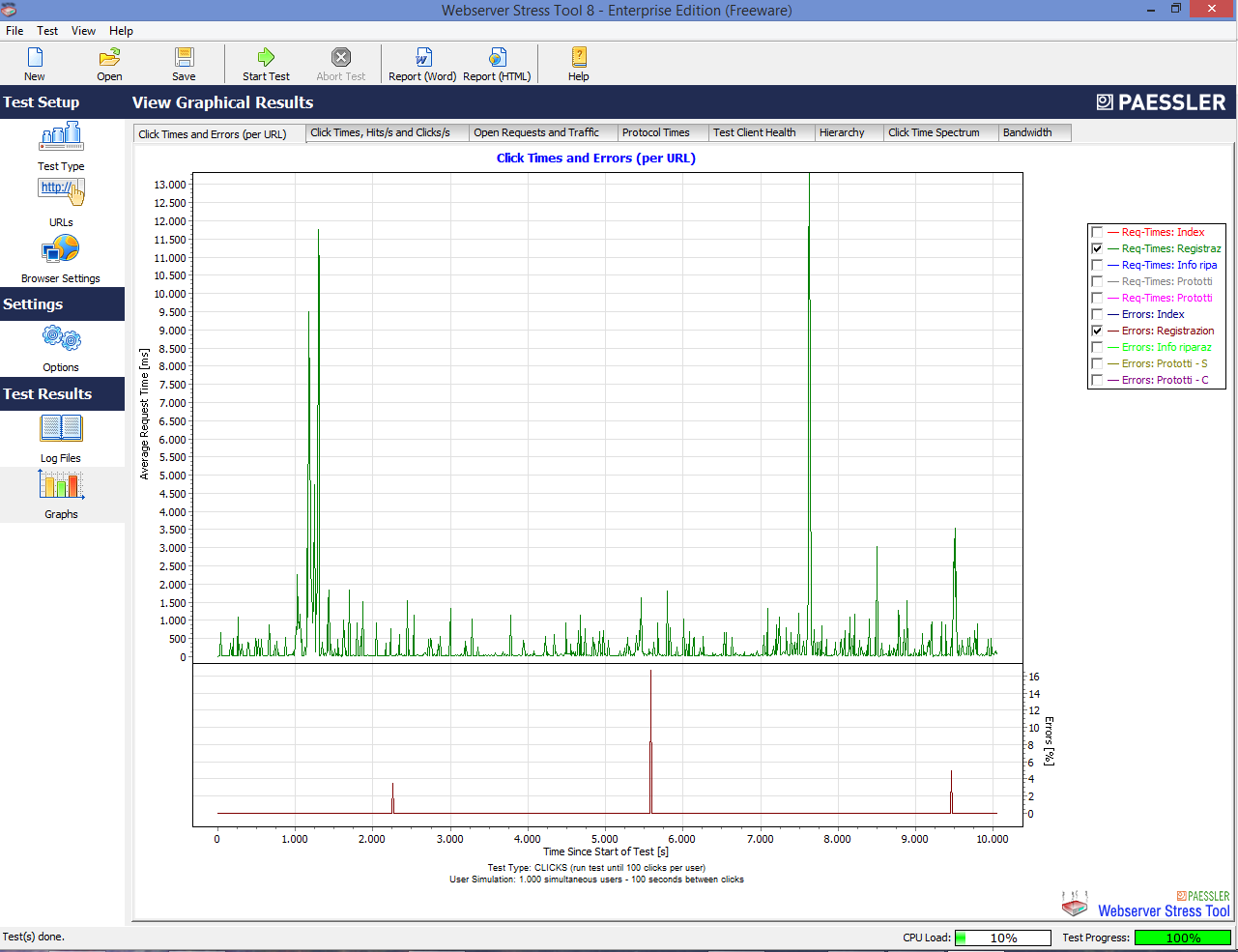
###### Index



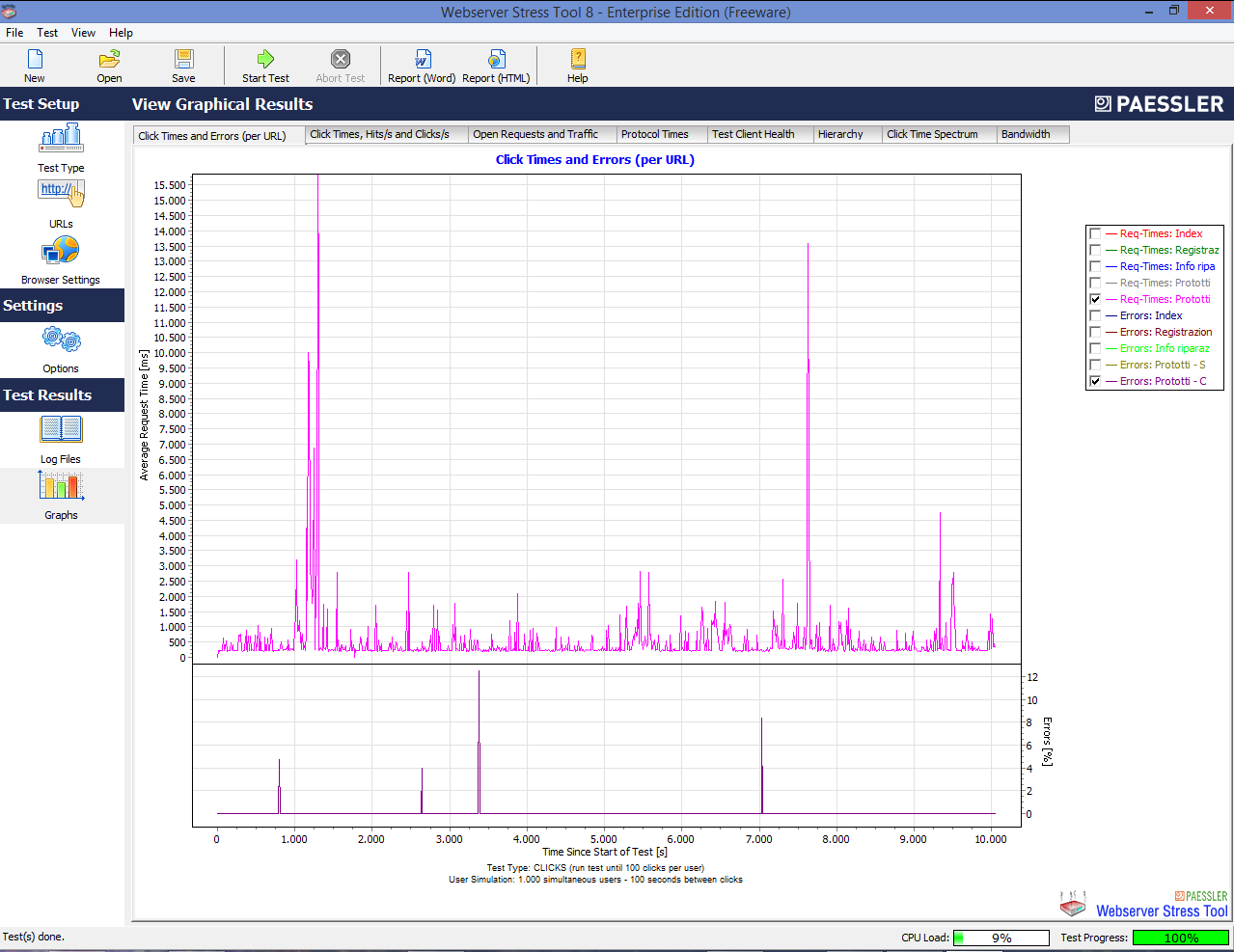
###### Info riparazioni



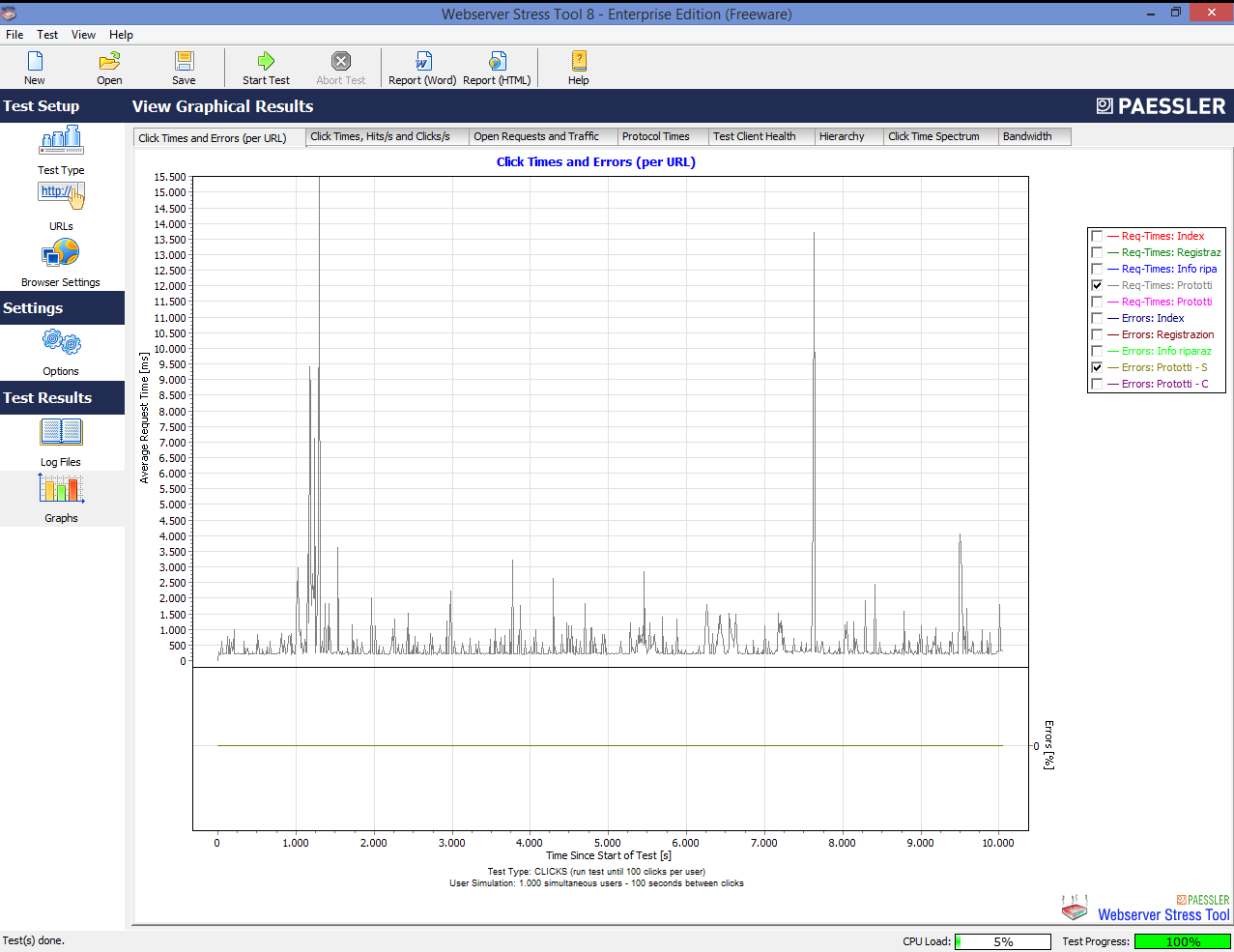
###### Registrazione



###### Prodotti – Computer



###### Prodotti – Smartphone



## Time test – 1500 utenti

### Time: 25 minuti

#### Click delay 5 secondi

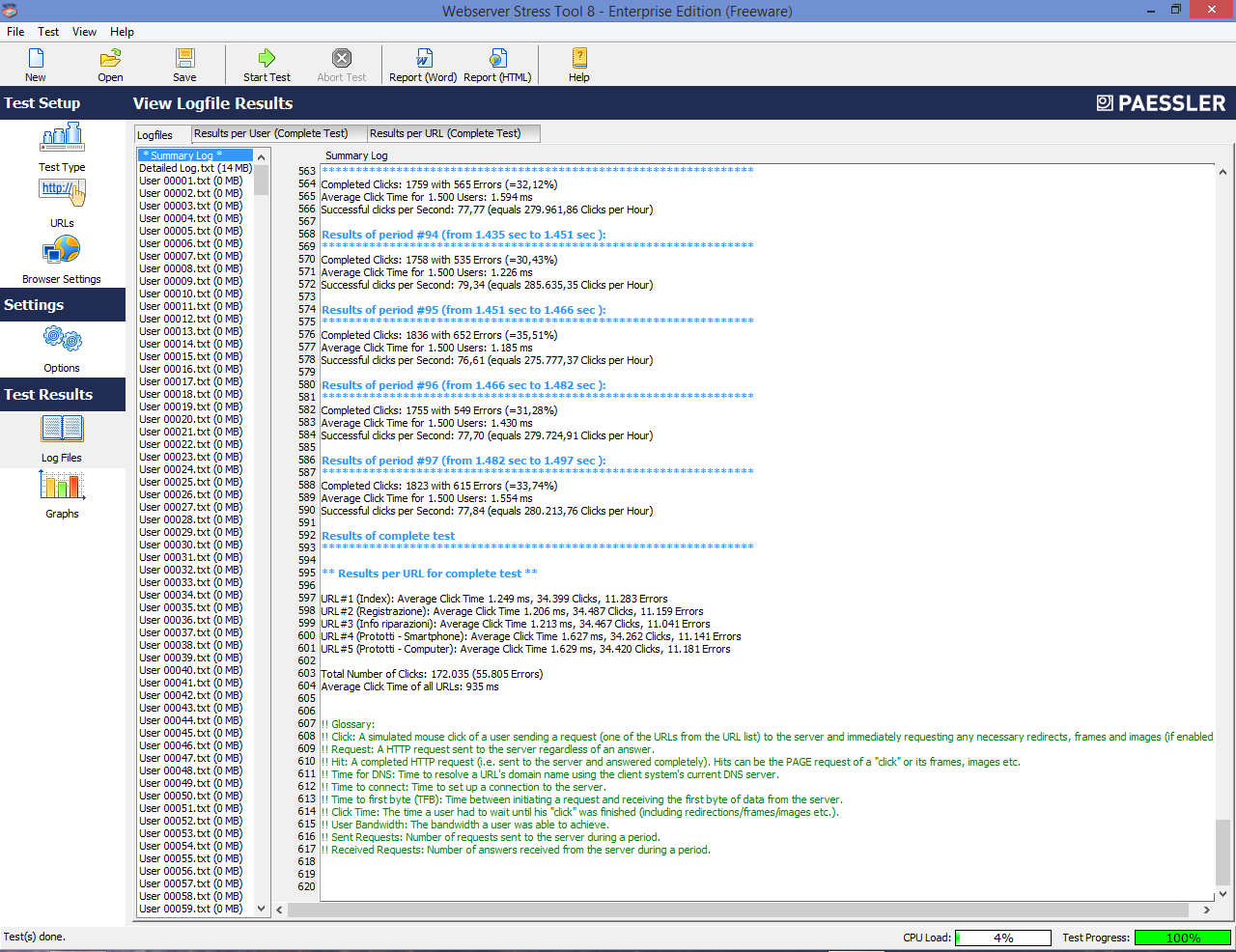
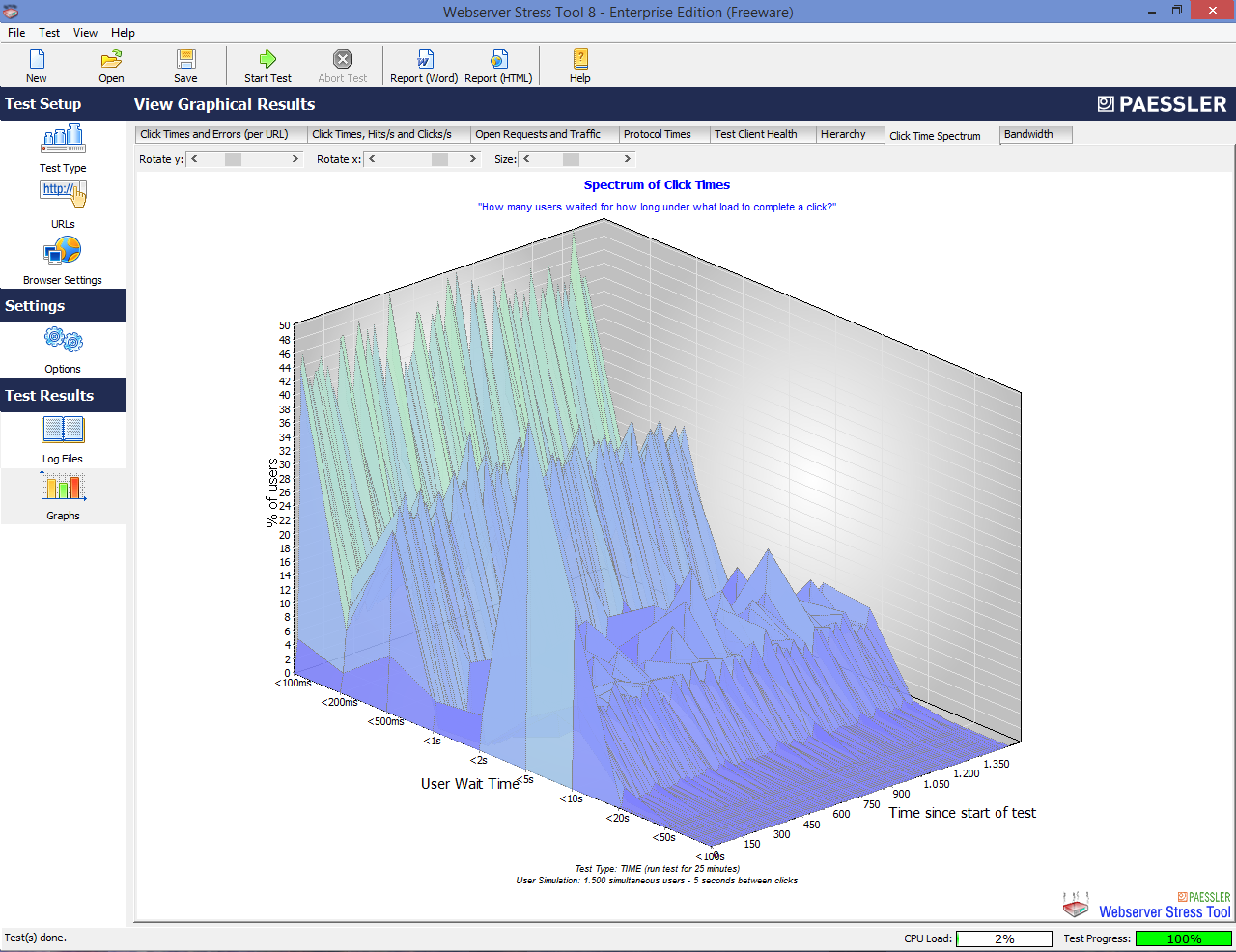


Grafico tridimensionale

I tre assi rappresentano:

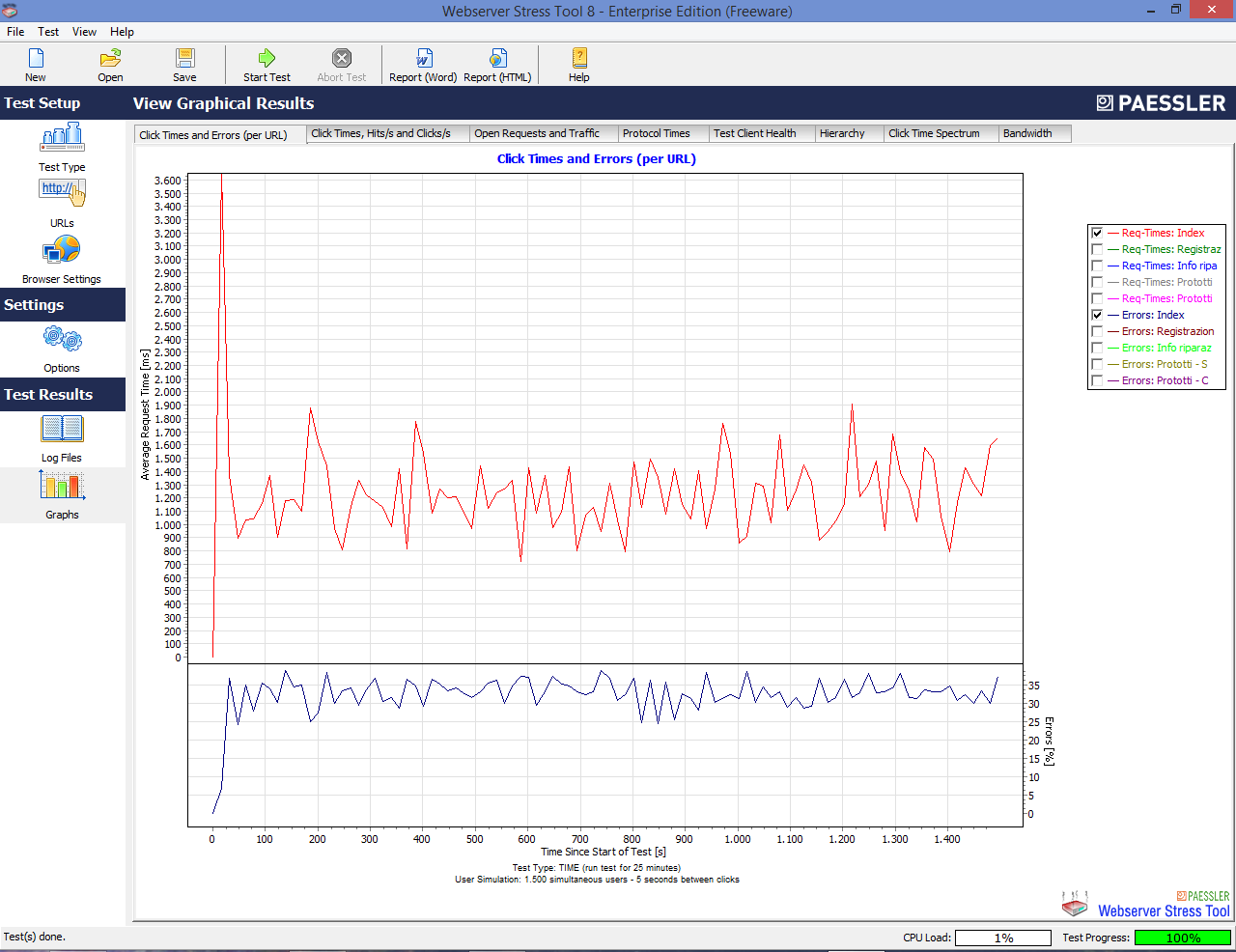
* percentuale di utenti che simulano le richieste
* tempo di attesa di ogni utente
* tempo di esecuzione del test



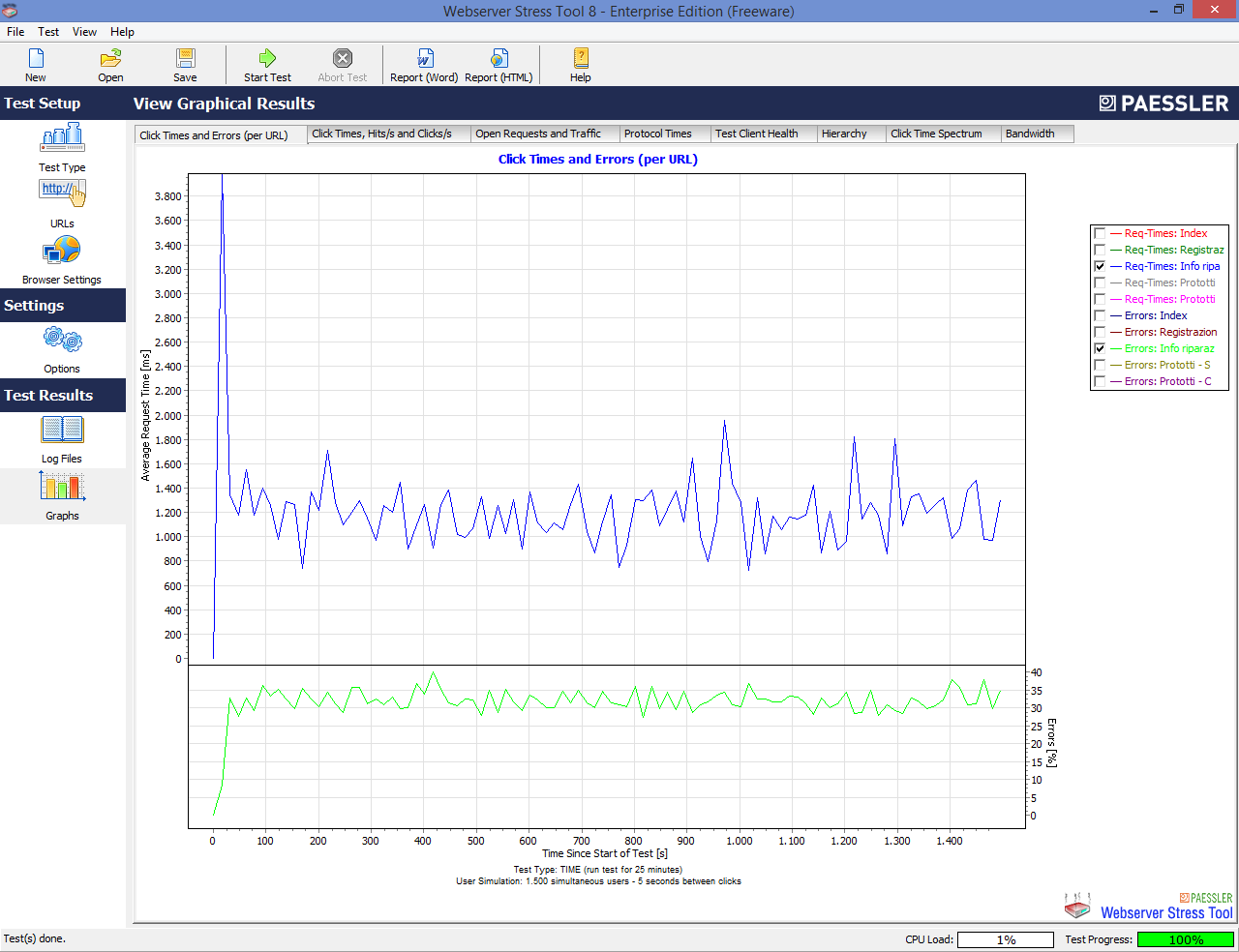
##### Grafico bidimensionale

Mostra l’andamento del tempo medio di richieste all’avanzare del test insieme ai possibili errori:

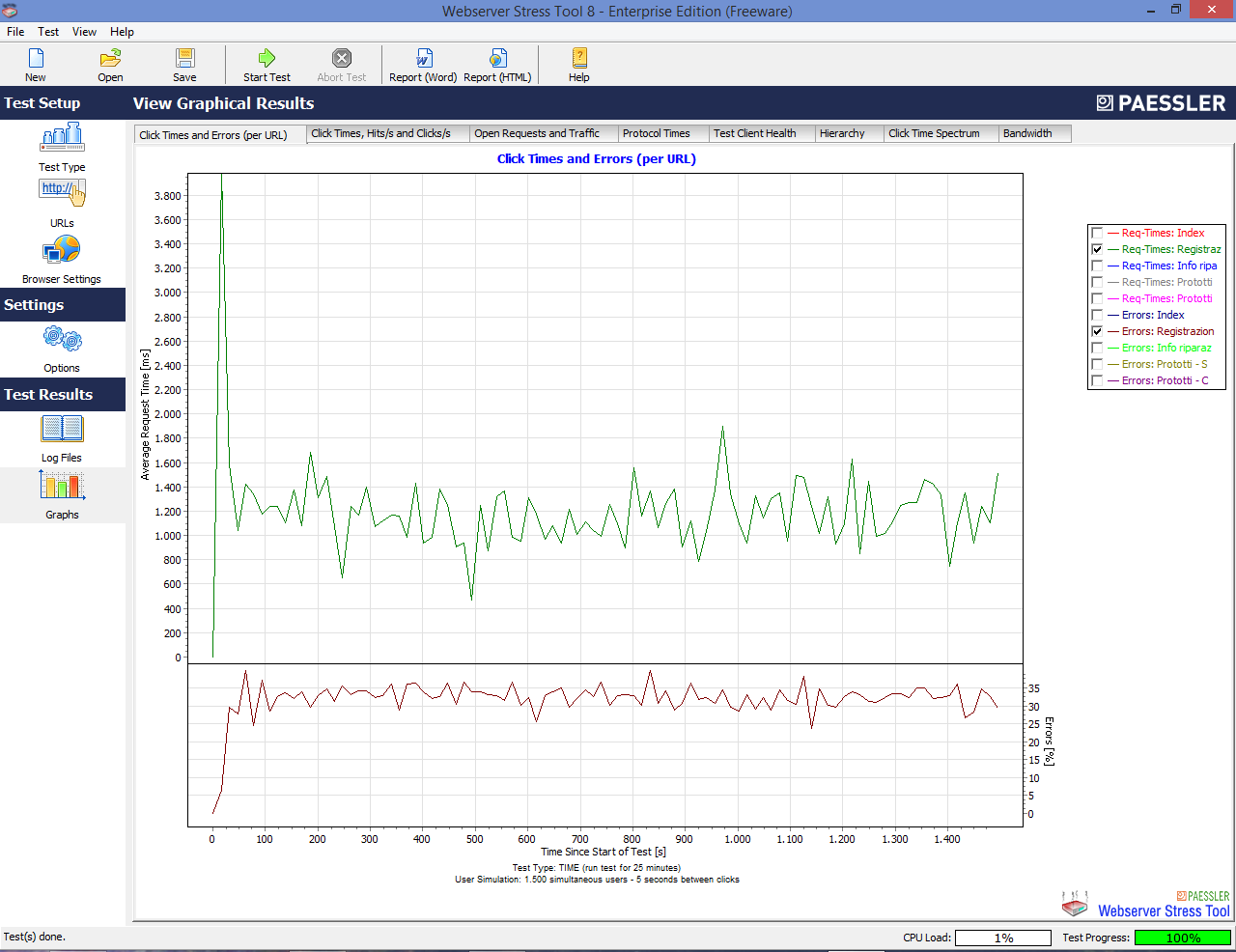
###### Index



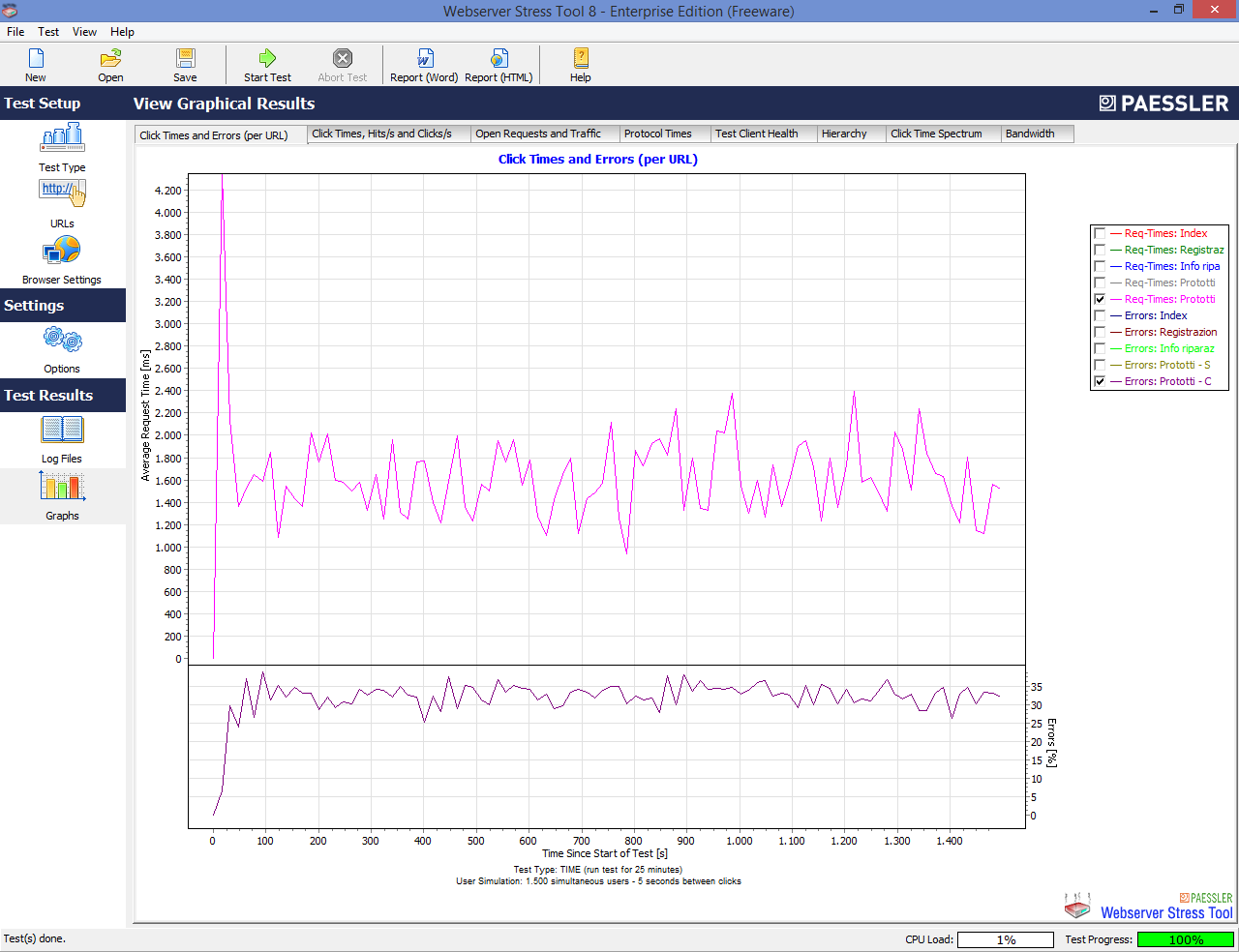
###### Info riparazioni



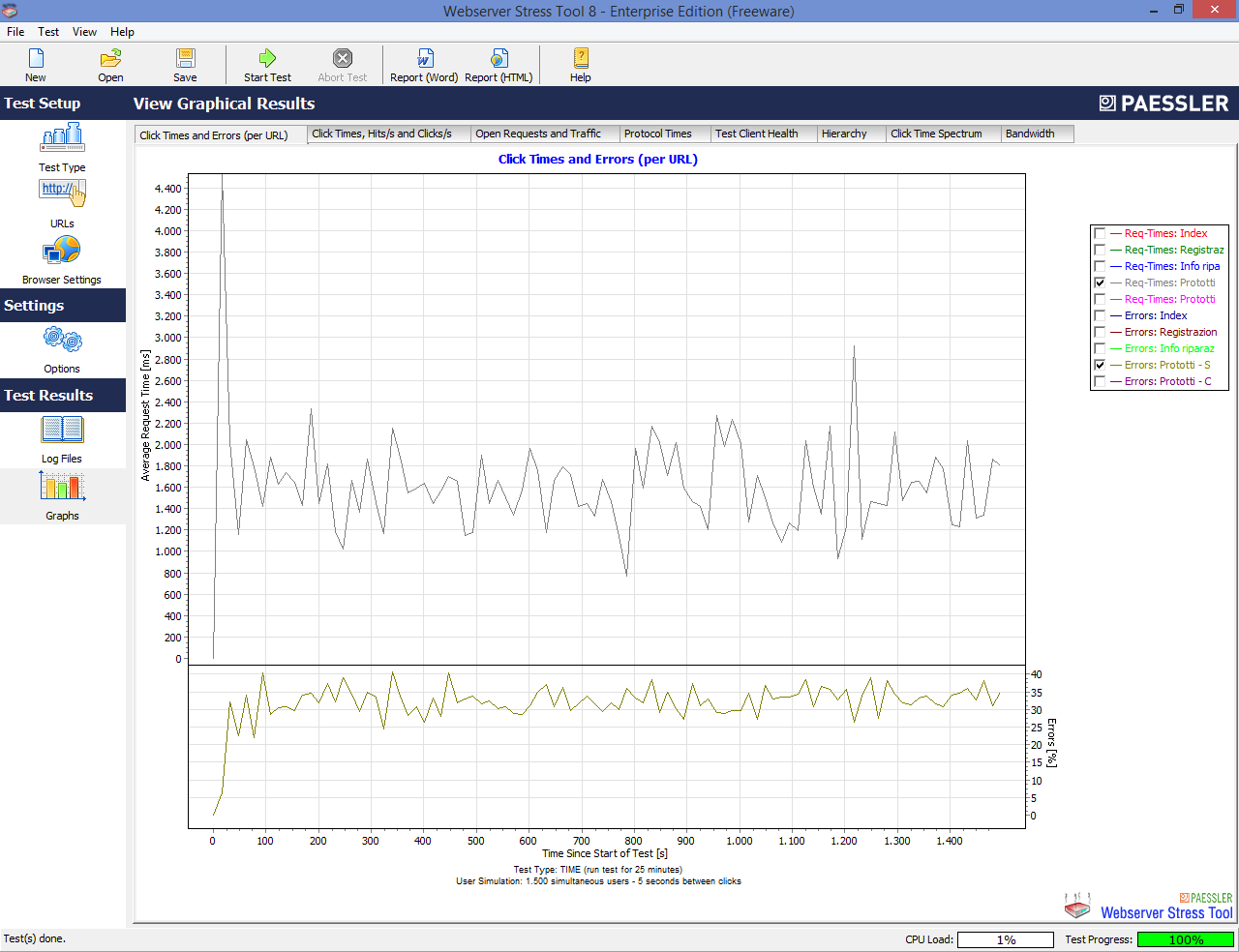
###### Registrazione



###### Prodotti – Computer



###### Prodotti – Smartphone



## Time test – 750 utenti

### Time: 50 minuti

#### Click delay 1 secondi

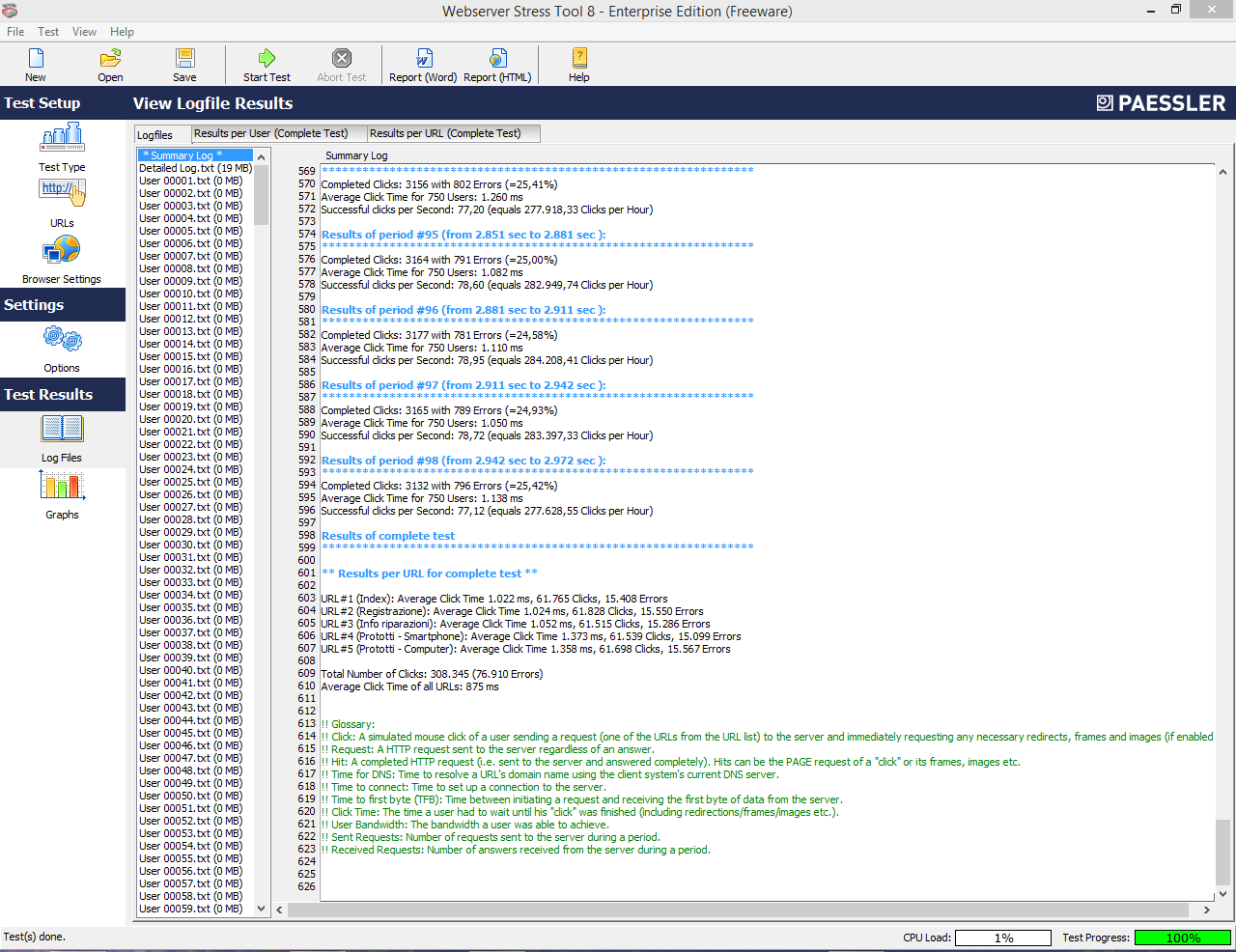
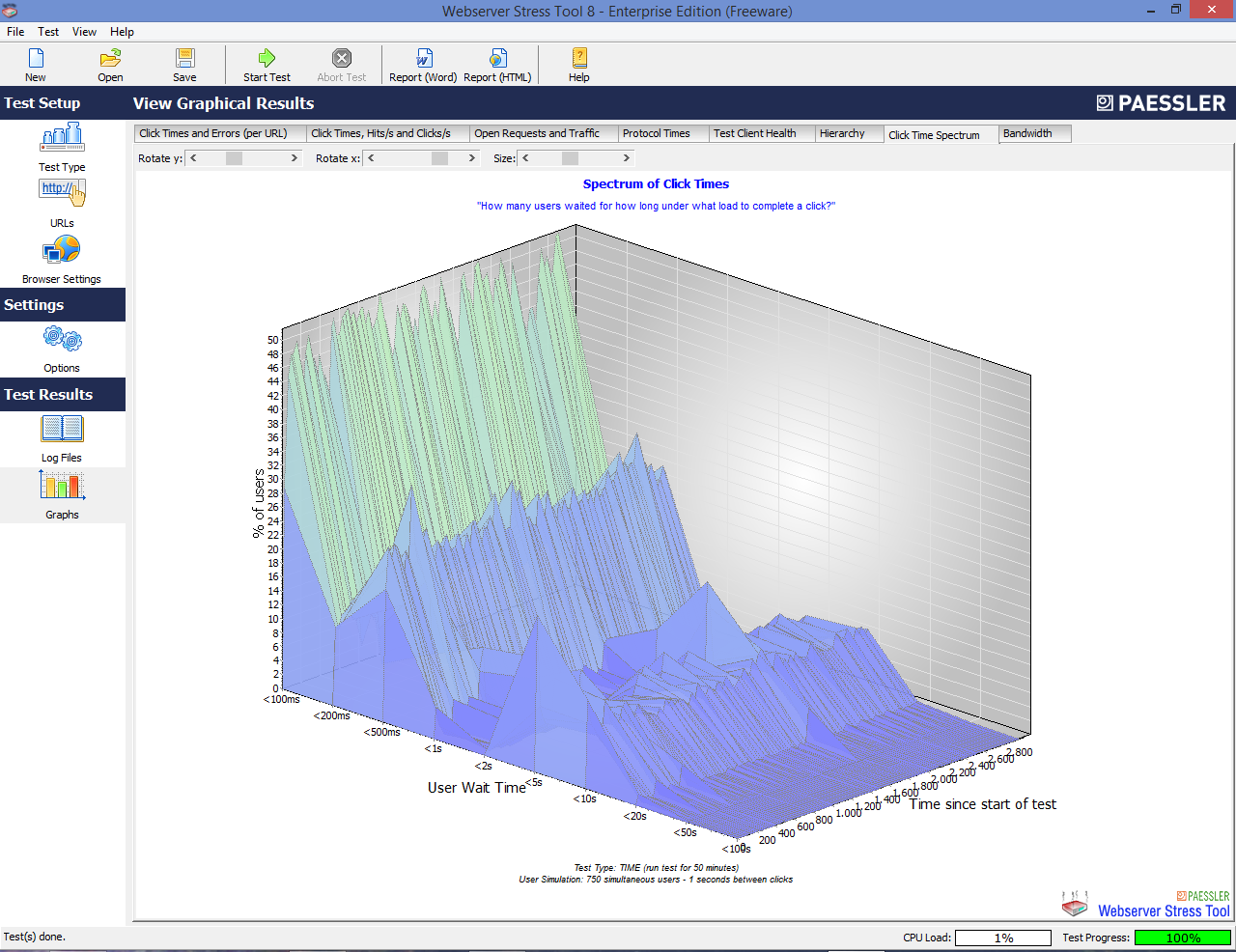


Grafico tridimensionale

I tre assi rappresentano:

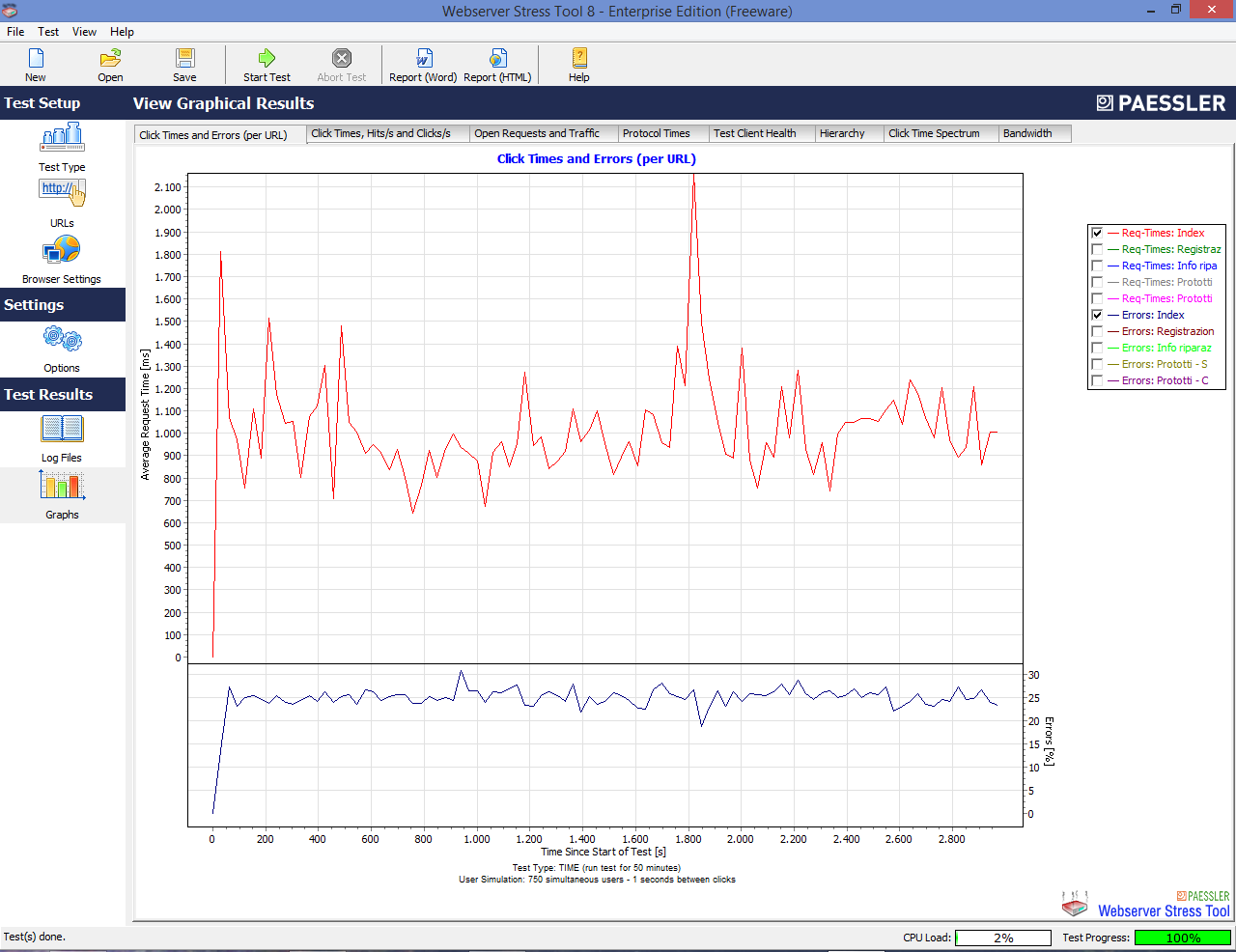
* percentuale di utenti che simulano le richieste
* tempo di attesa di ogni utente
* tempo di esecuzione del test



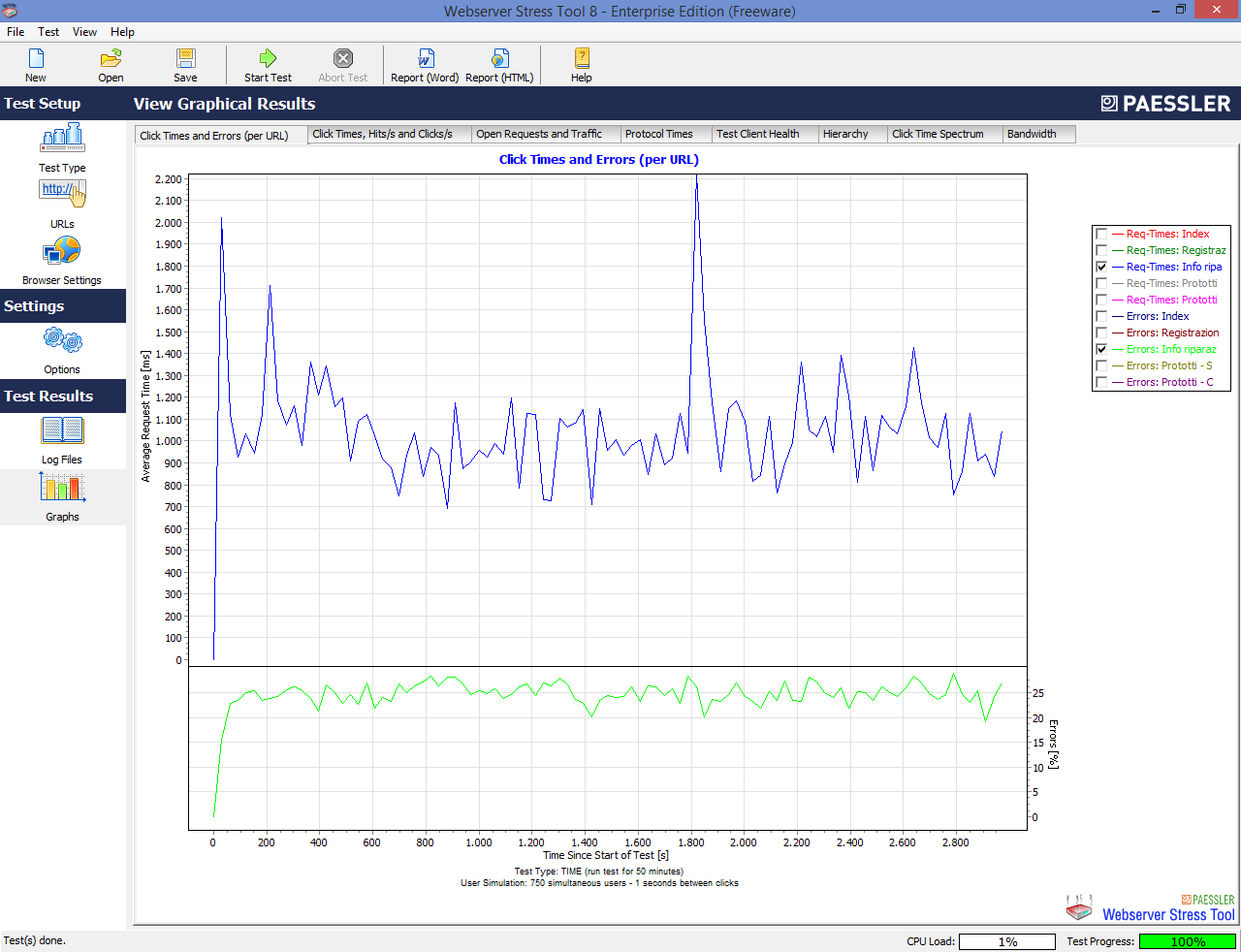
##### Grafico bidimensionale

Mostra l’andamento del tempo medio di richieste all’avanzare del test insieme ai possibili errori:

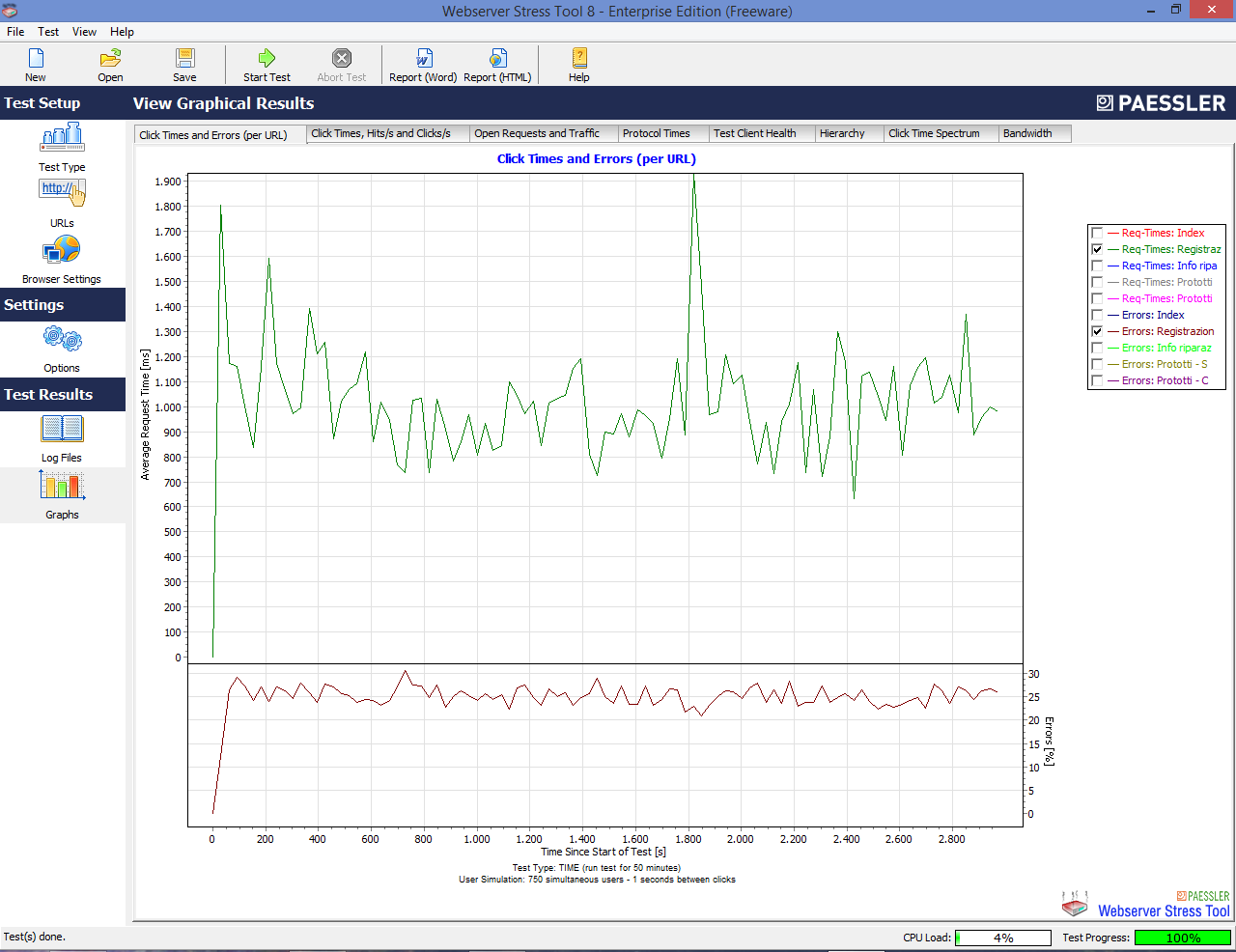
###### Index



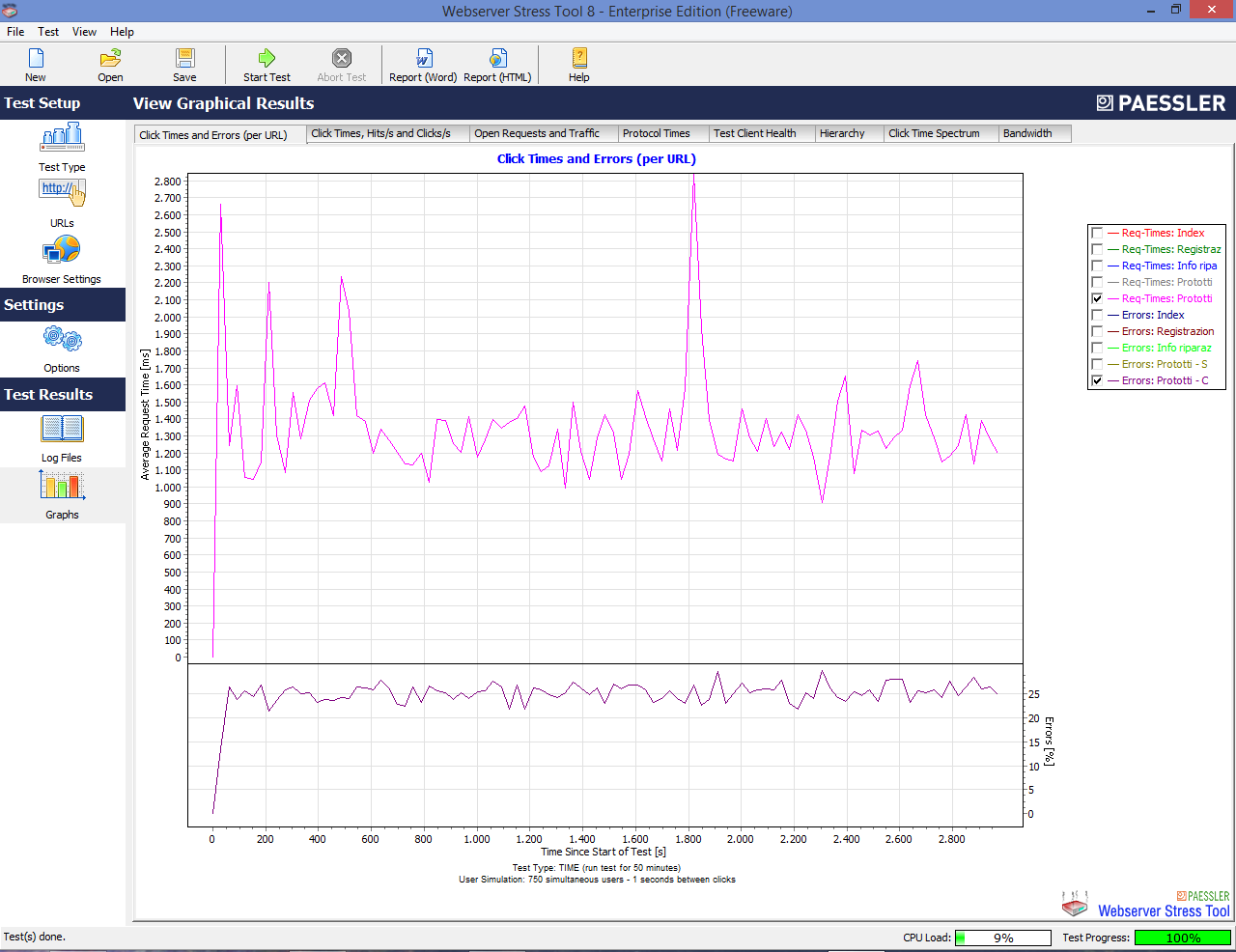
###### Info riparazioni



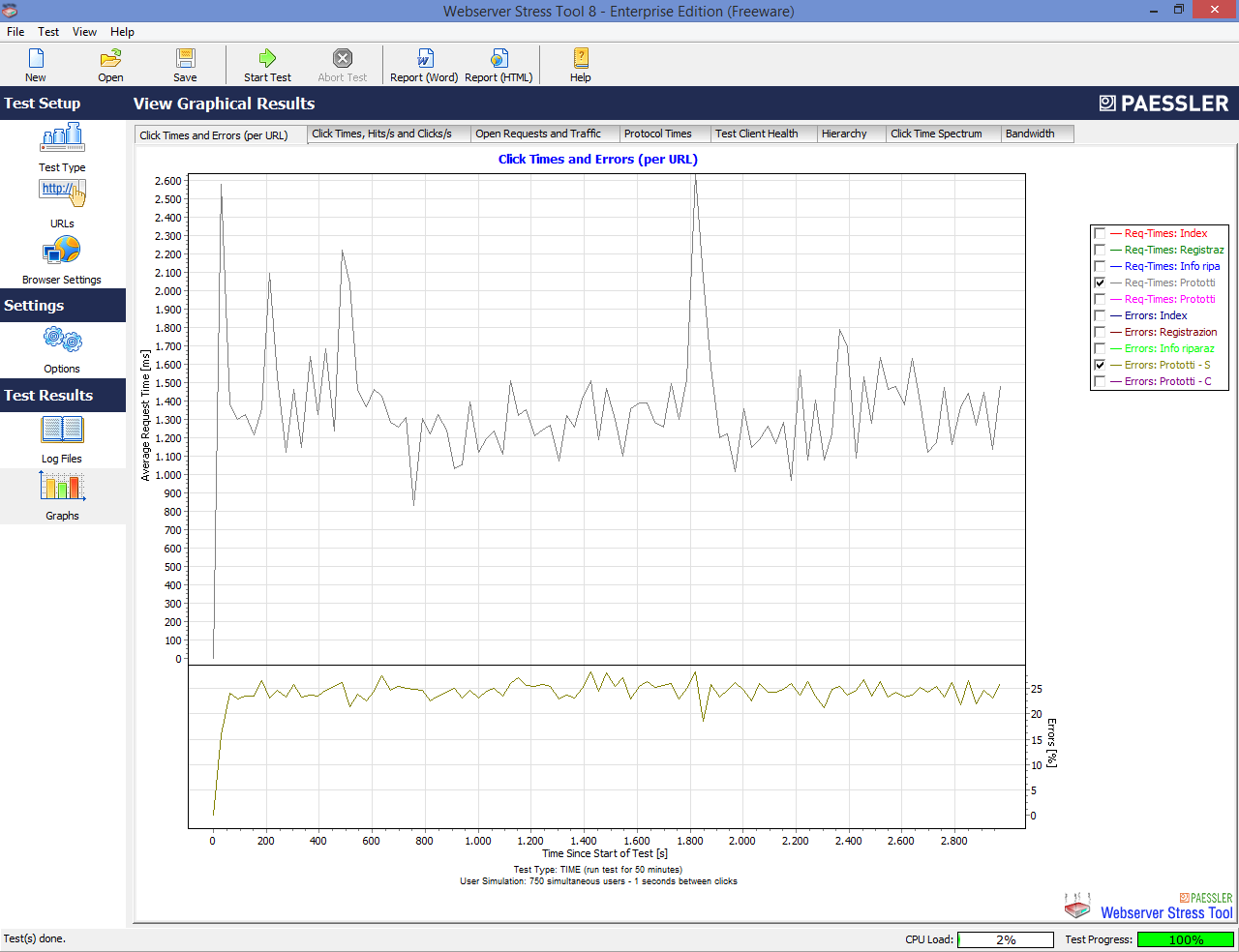
###### Registrazione



###### Prodotti – Computer



###### Prodotti – Smartphone



## RAMP test – 1000 utenti

### Time: 30 minuti

#### Click delay 3 secondi

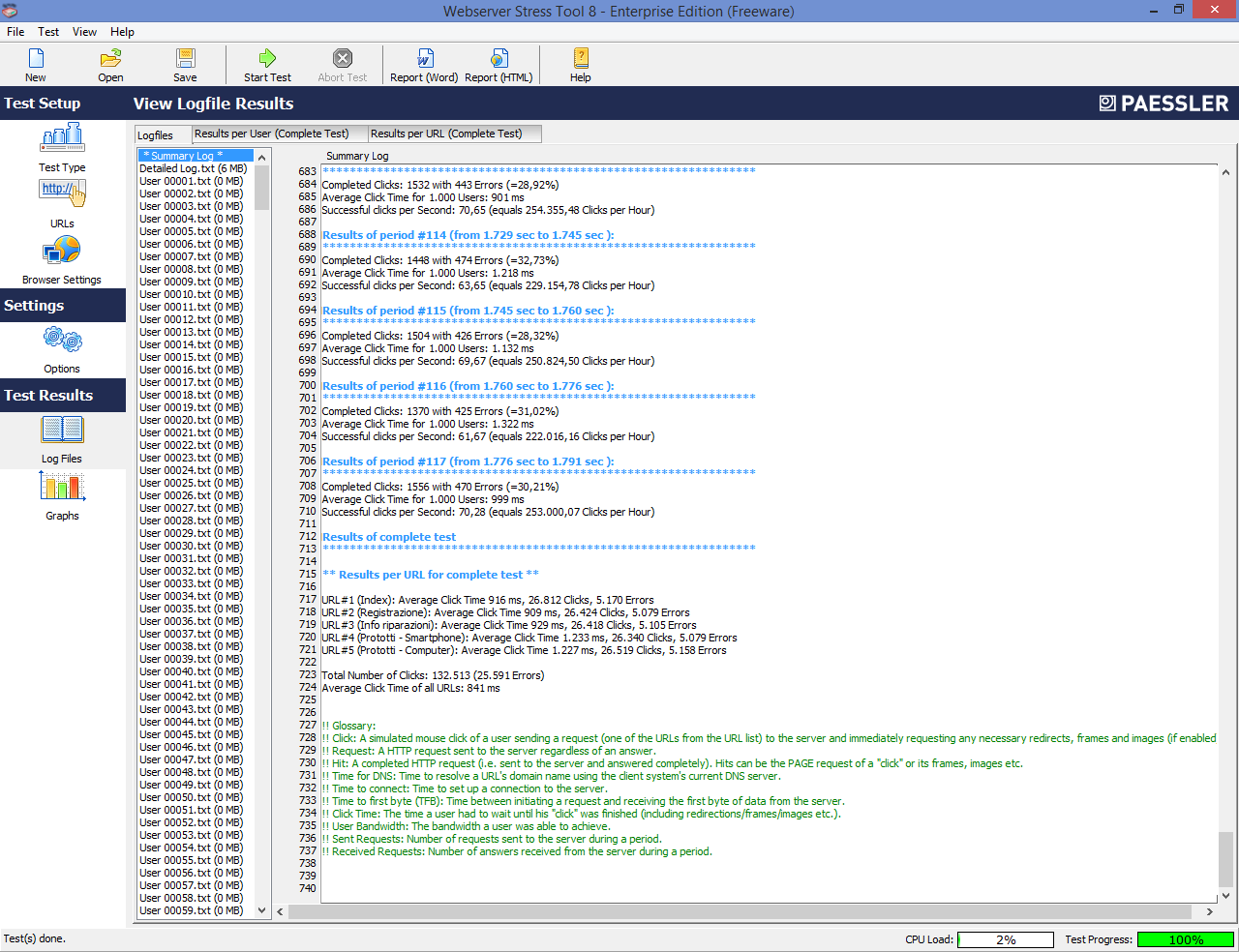
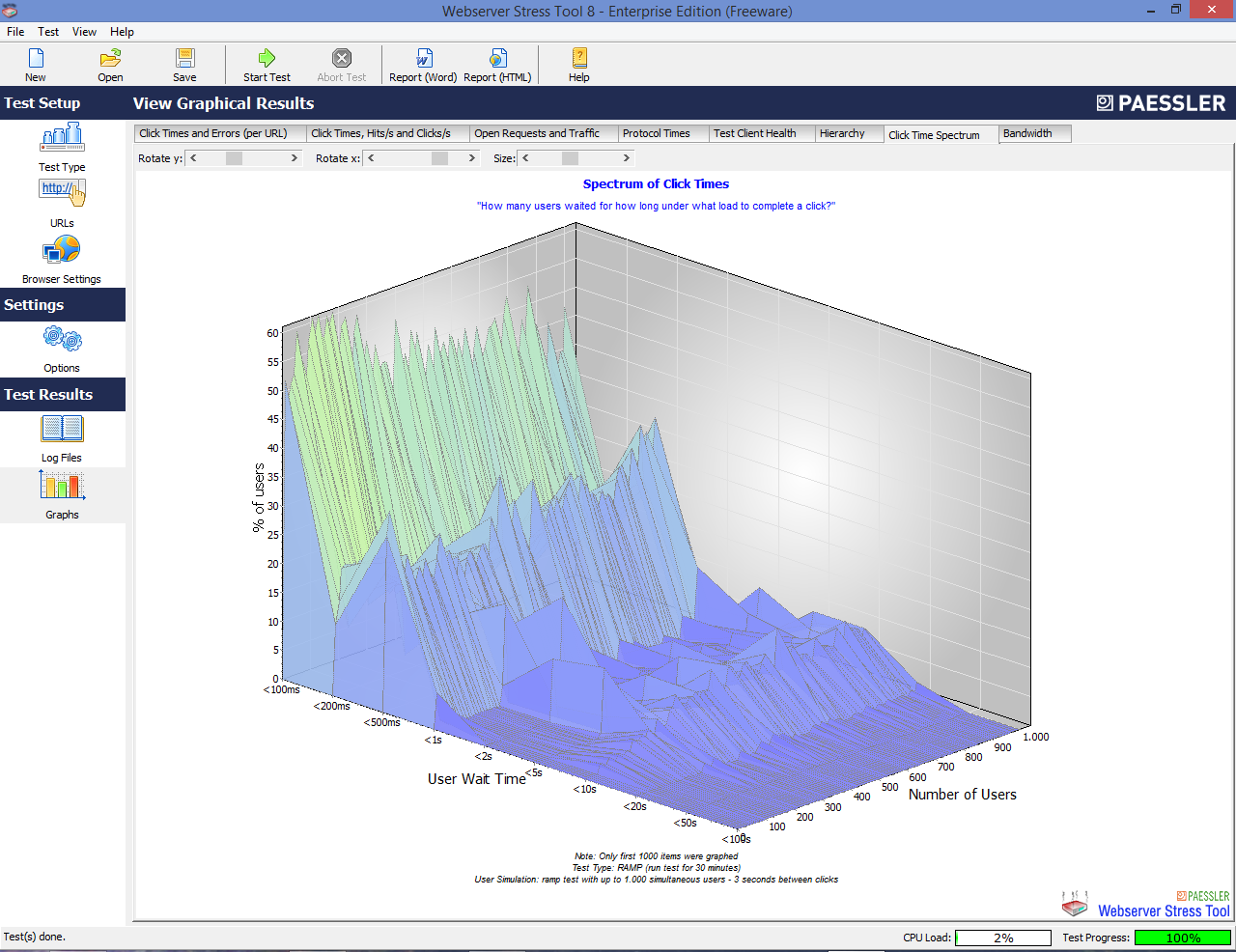


Grafico tridimensionale

I tre assi rappresentano:

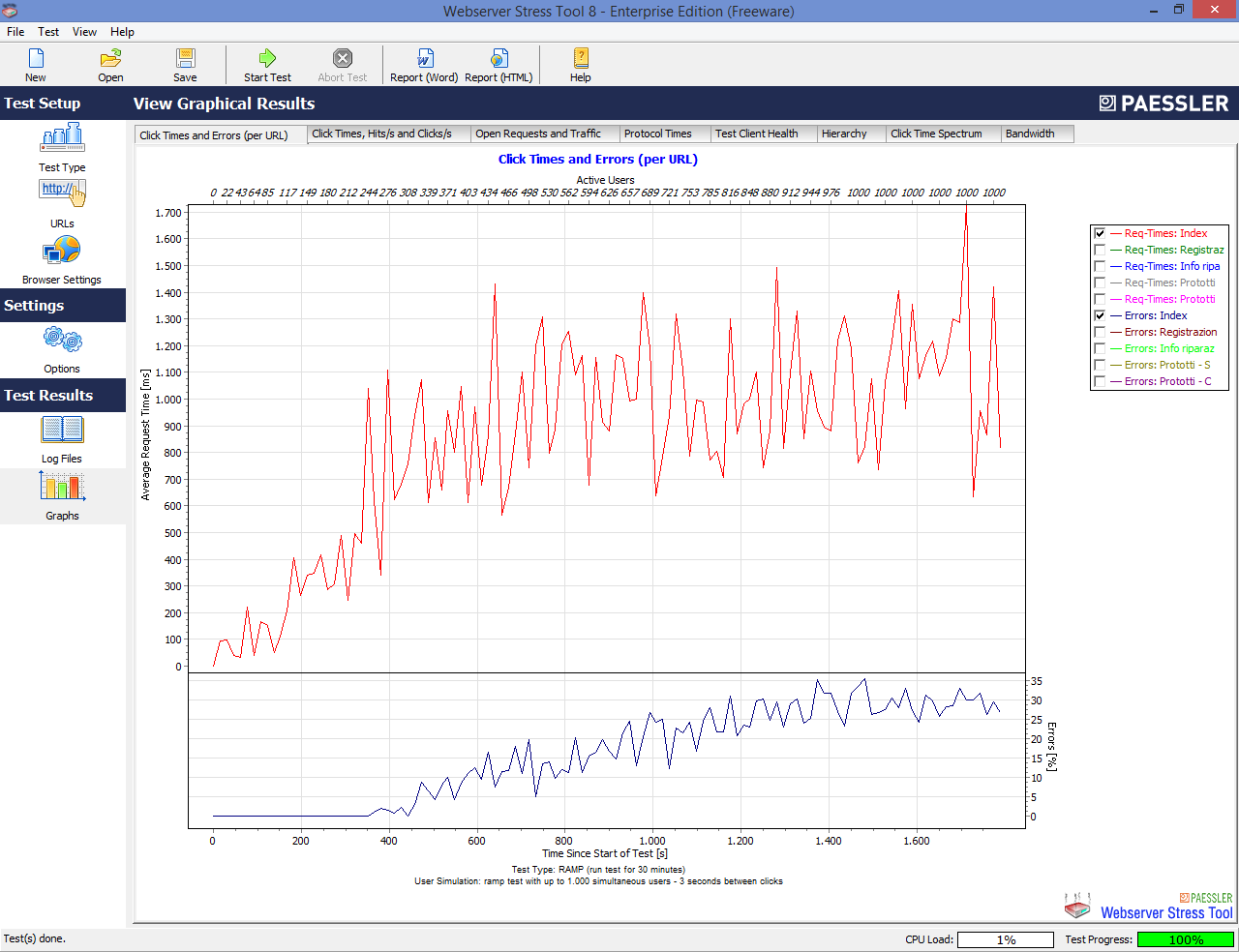
* percentuale di utenti che simulano le richieste
* tempo di attesa di ogni utente
* tempo di esecuzione del test



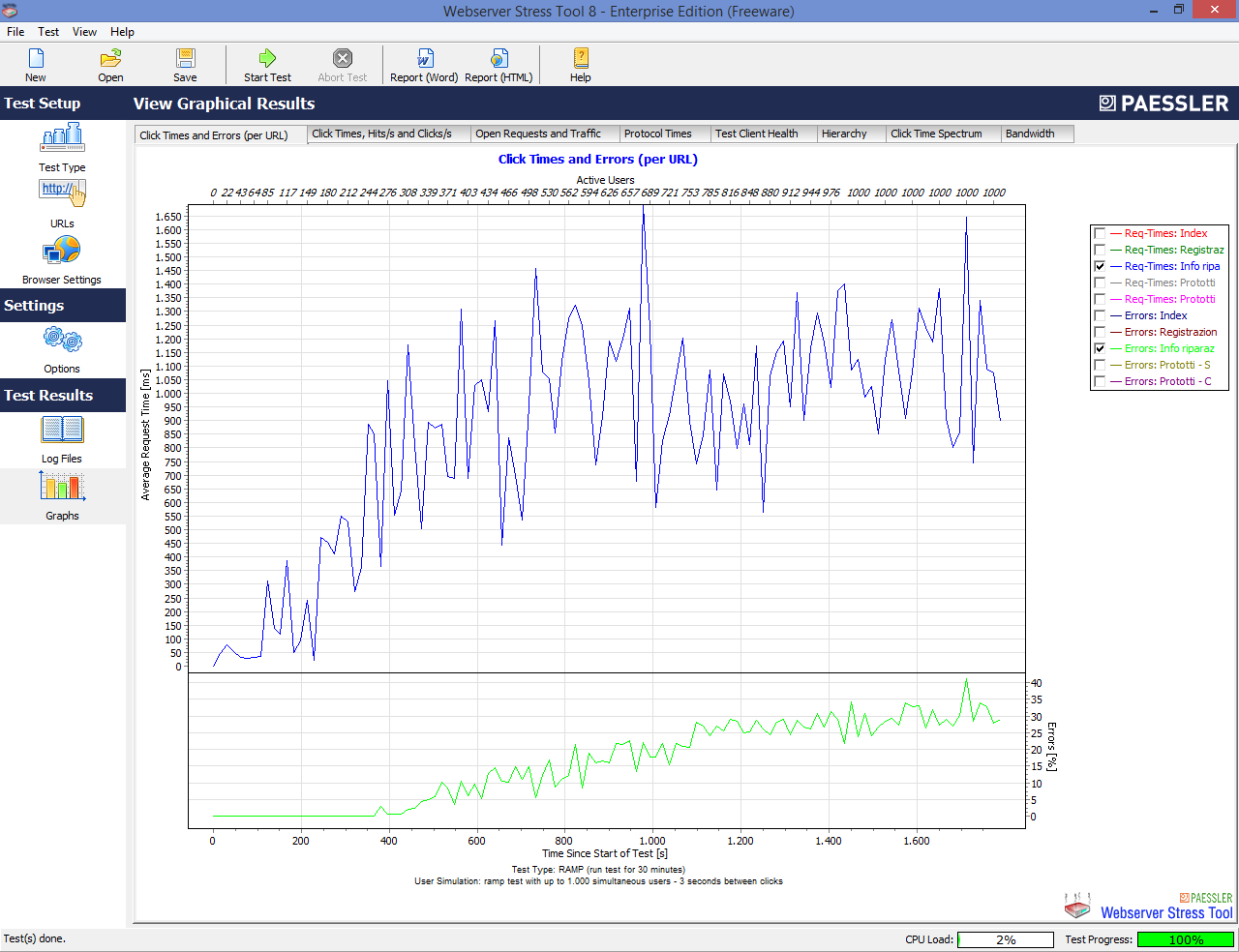
##### Grafico bidimensionale

Mostra l’andamento del tempo medio di richieste all’avanzare del test insieme ai possibili errori:

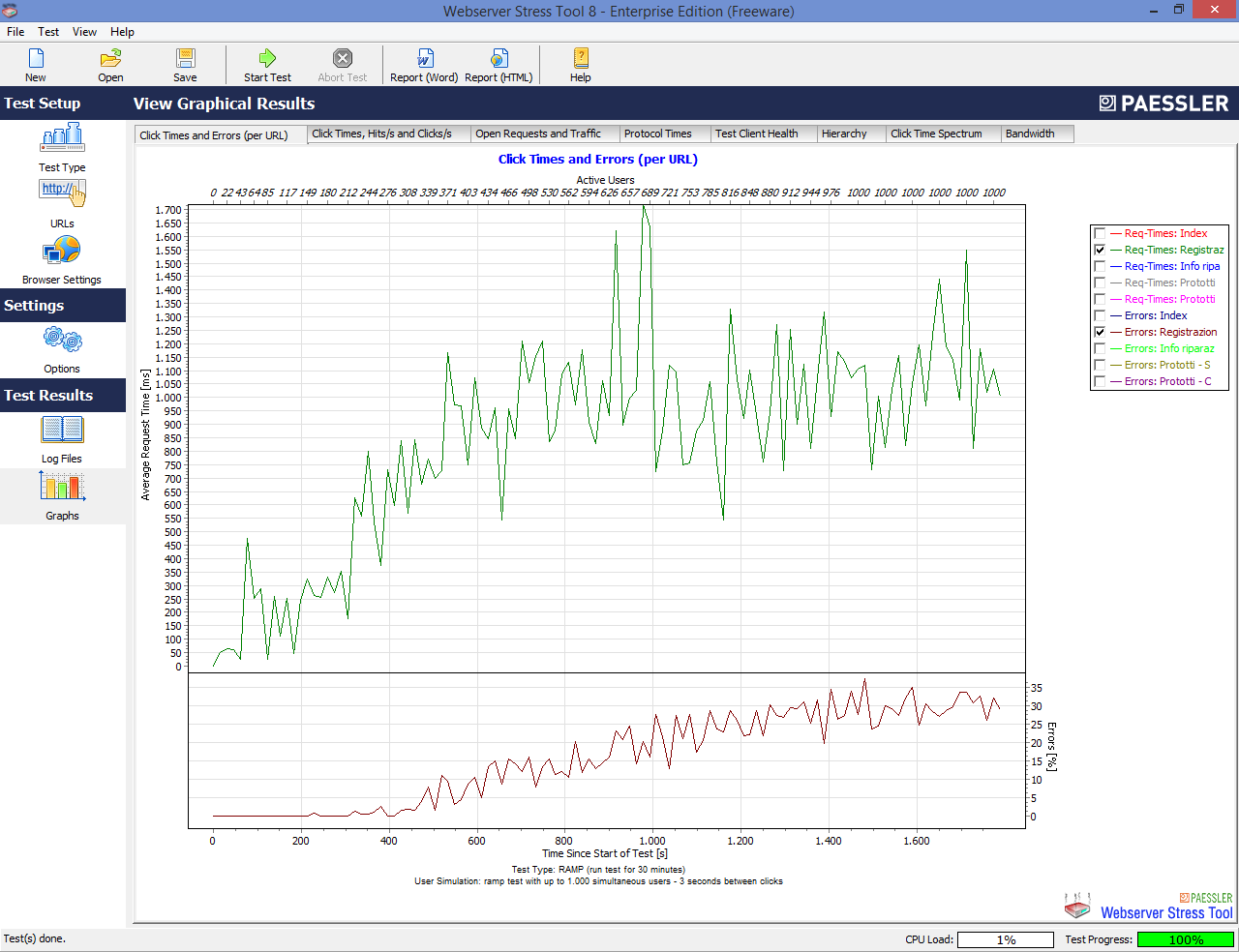
###### Index



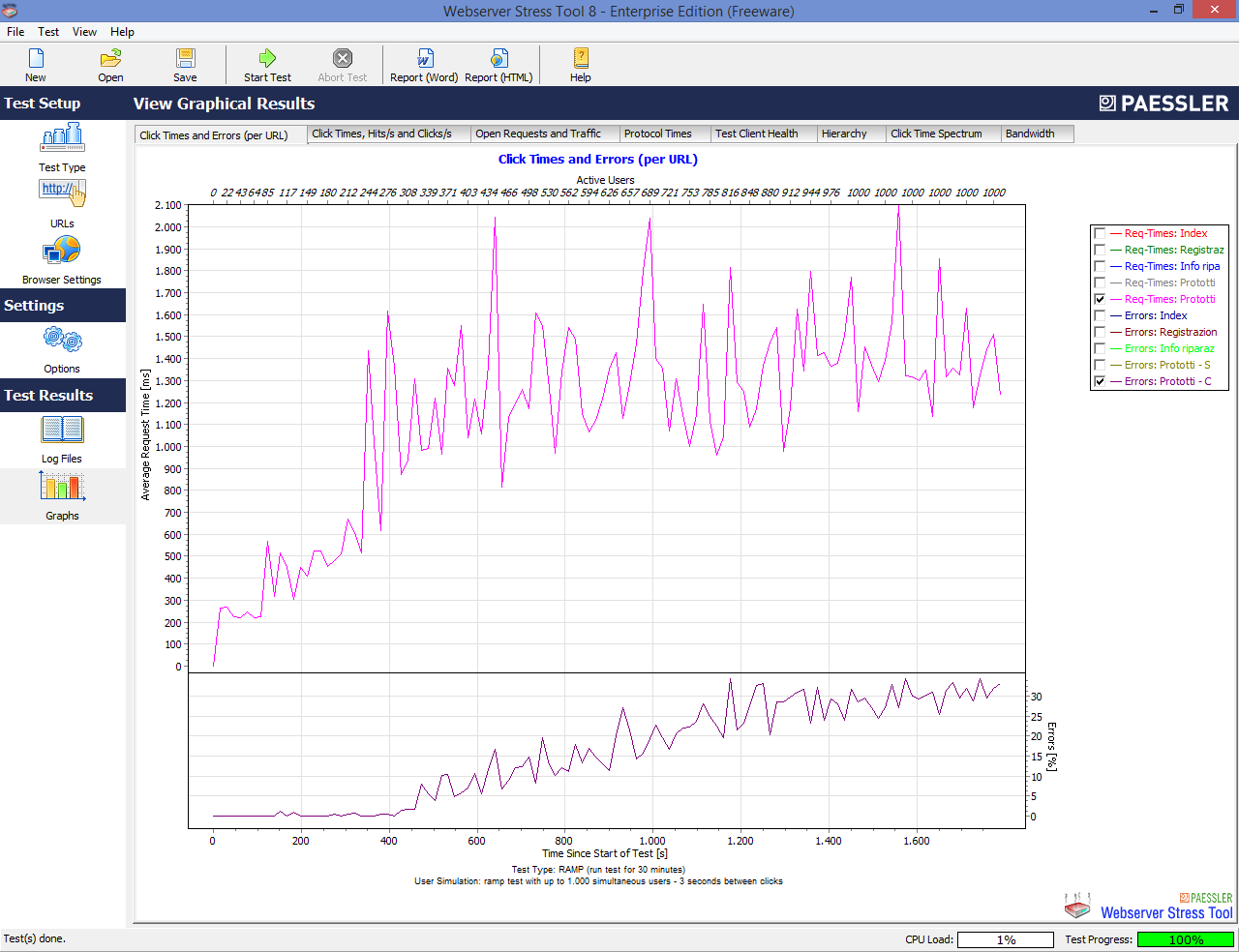
###### Info riparazioni



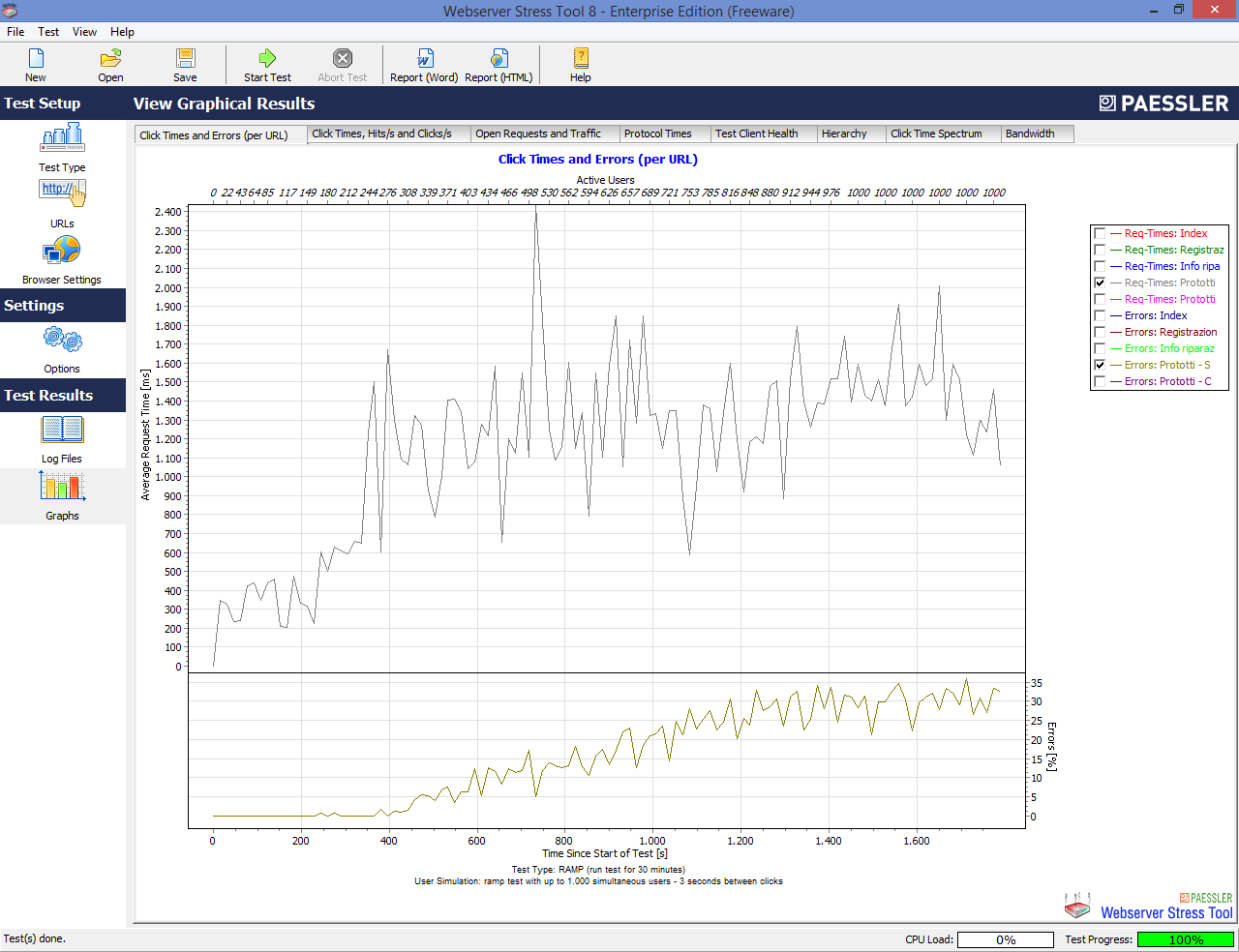
###### Registrazione



###### Prodotti – Computer



###### Prodotti – Smartphone



## RAMP test – 1000 utenti

### Time: 60 minuti

#### Click delay 1 secondi

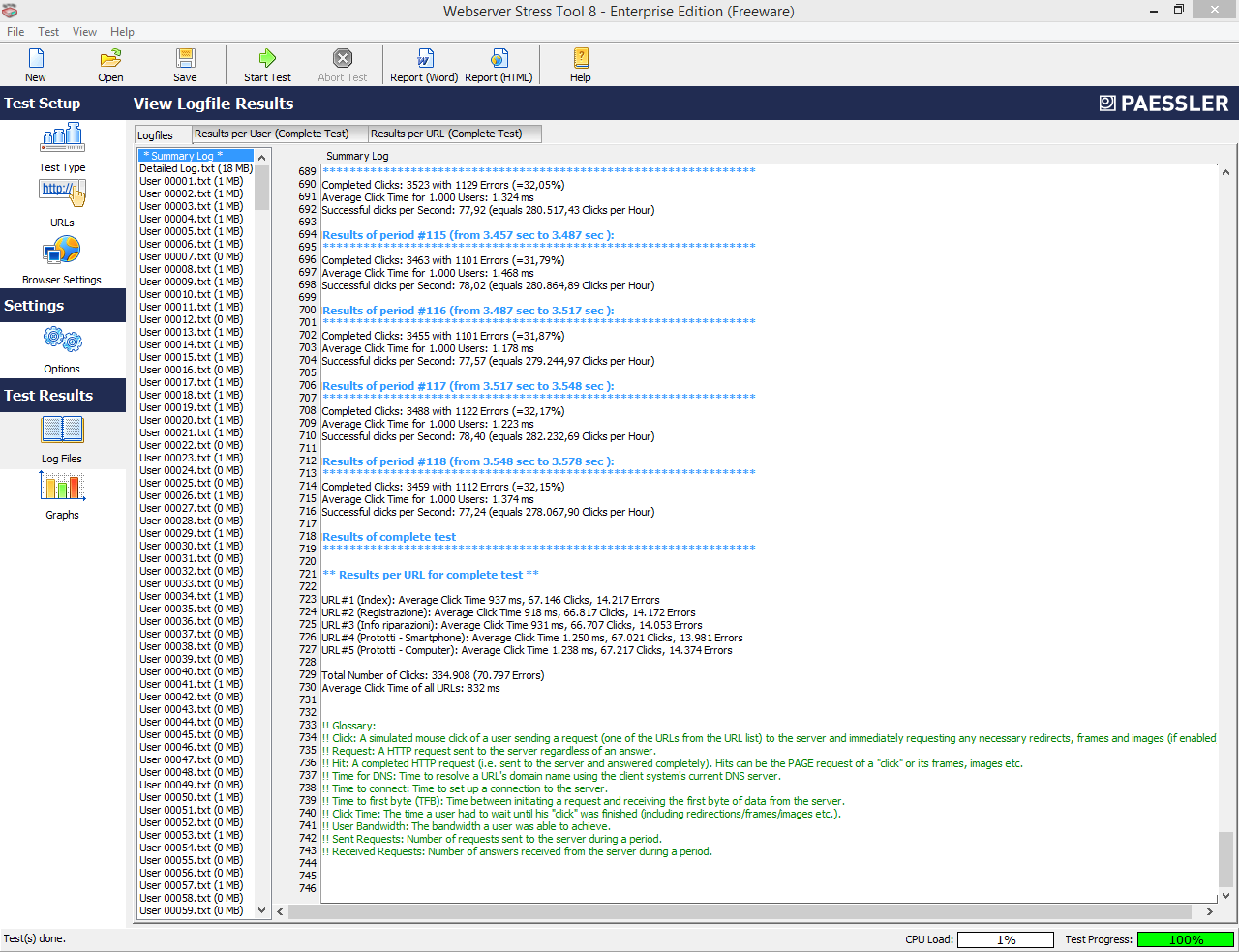
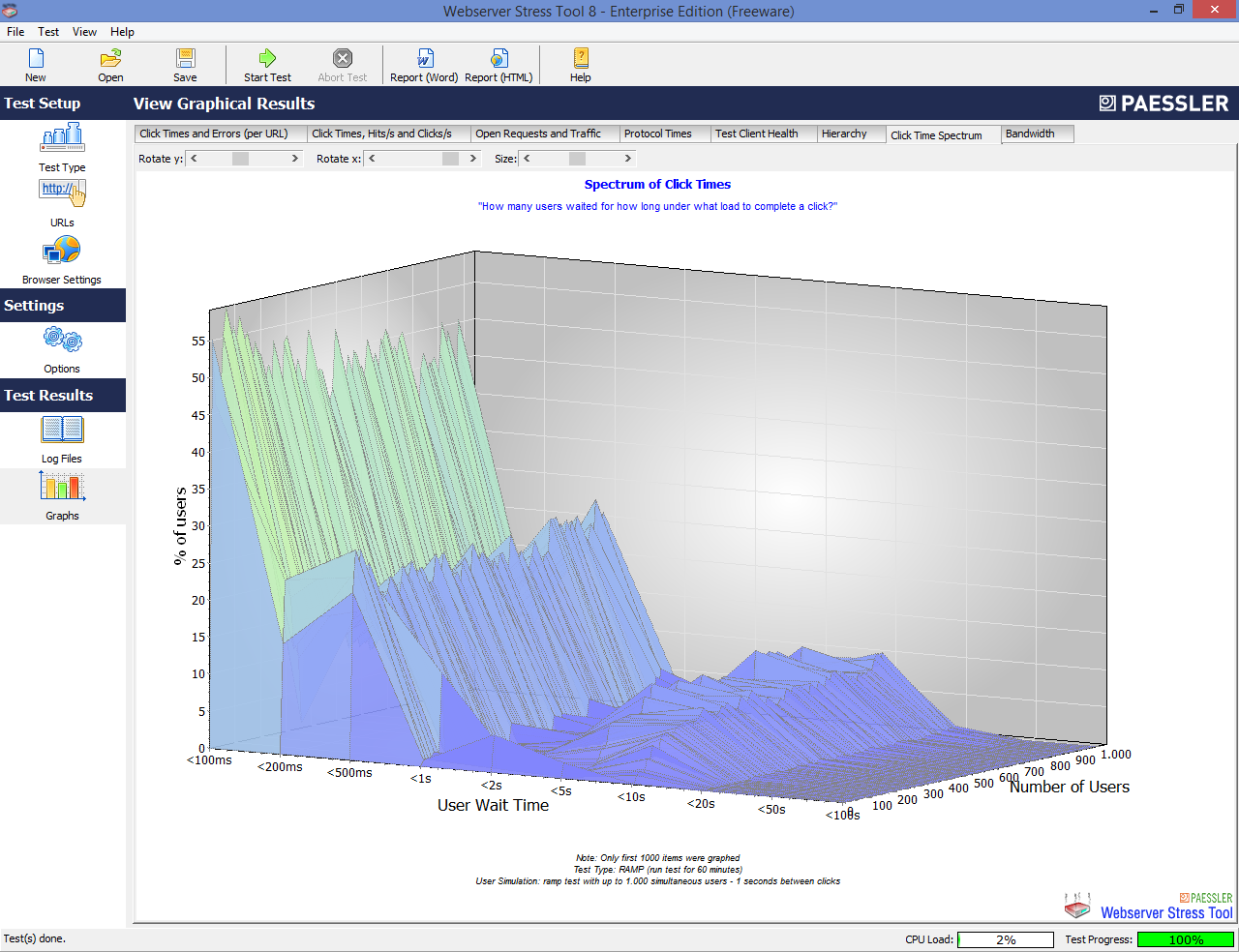


Grafico tridimensionale

I tre assi rappresentano:

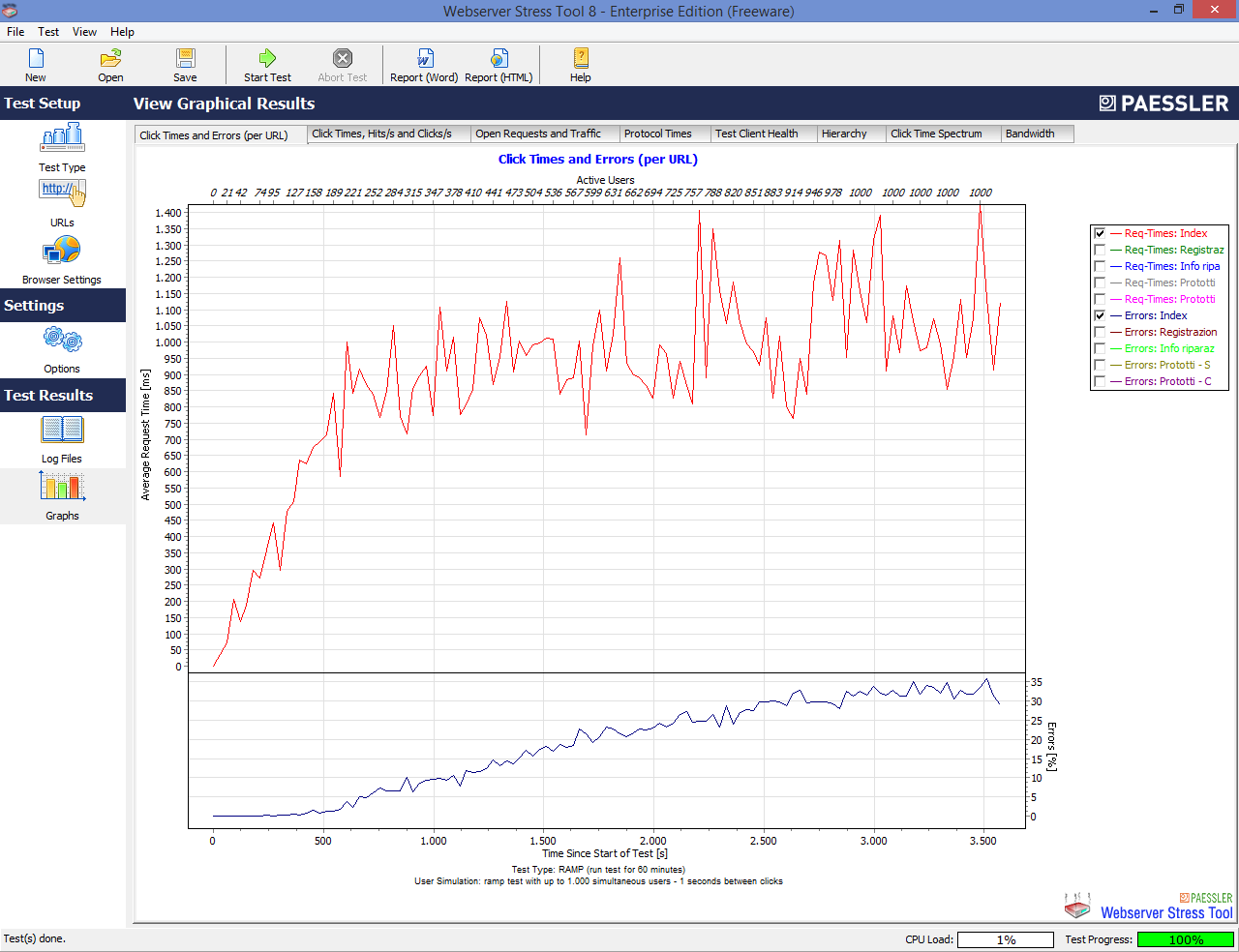
* percentuale di utenti che simulano le richieste
* tempo di attesa di ogni utente
* tempo di esecuzione del test



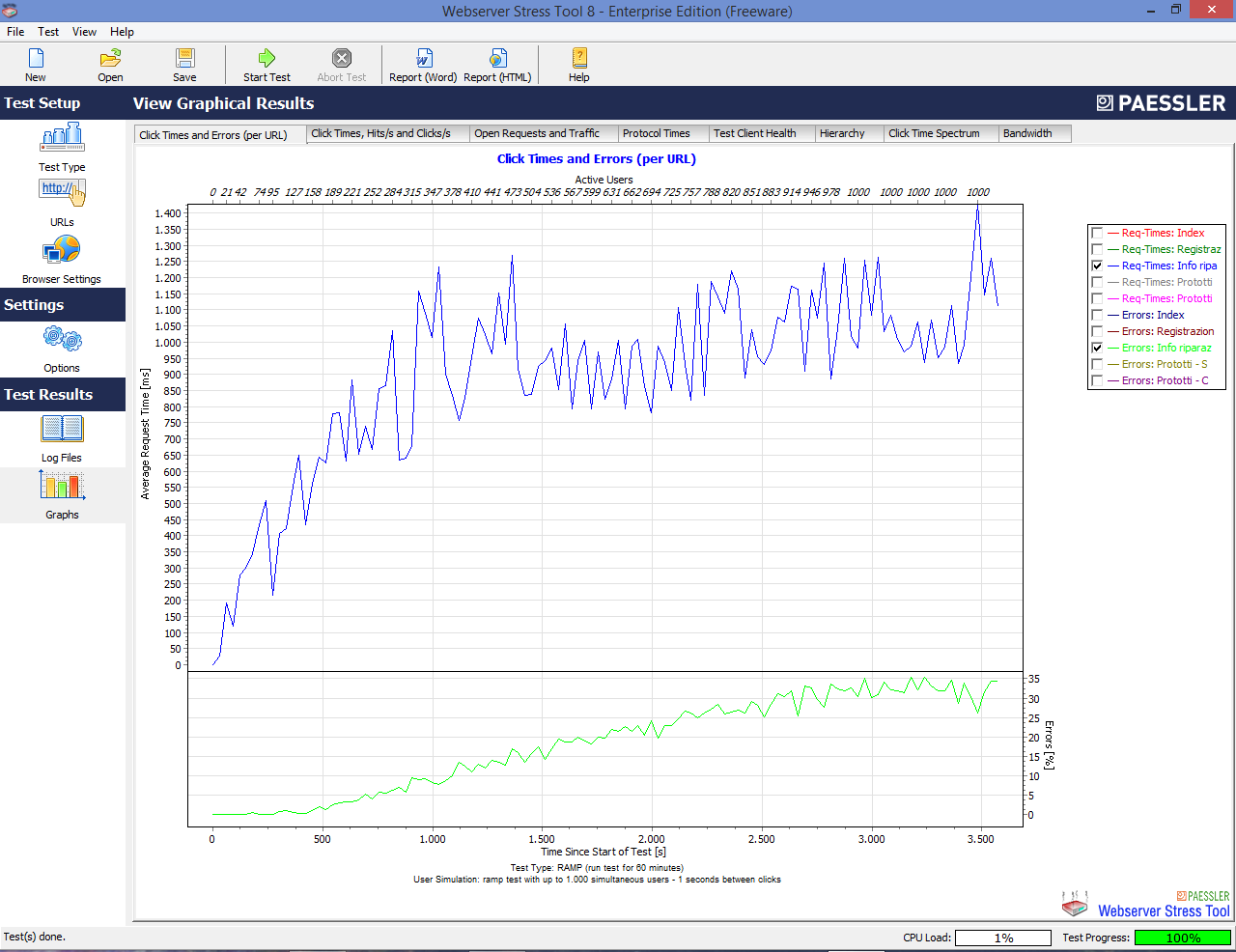
##### Grafico bidimensionale

Mostra l’andamento del tempo medio di richieste all’avanzare del test insieme ai possibili errori:

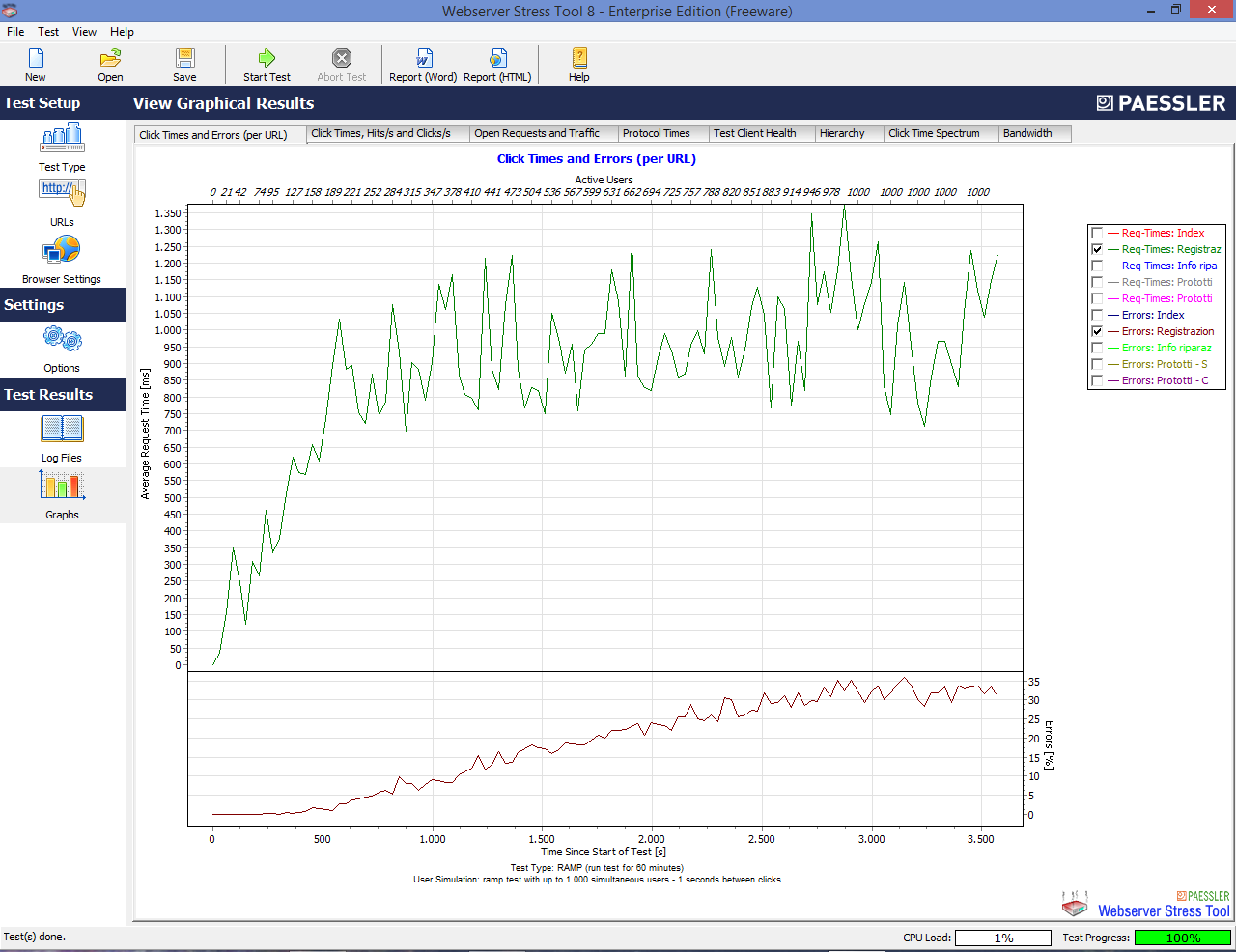
###### Index



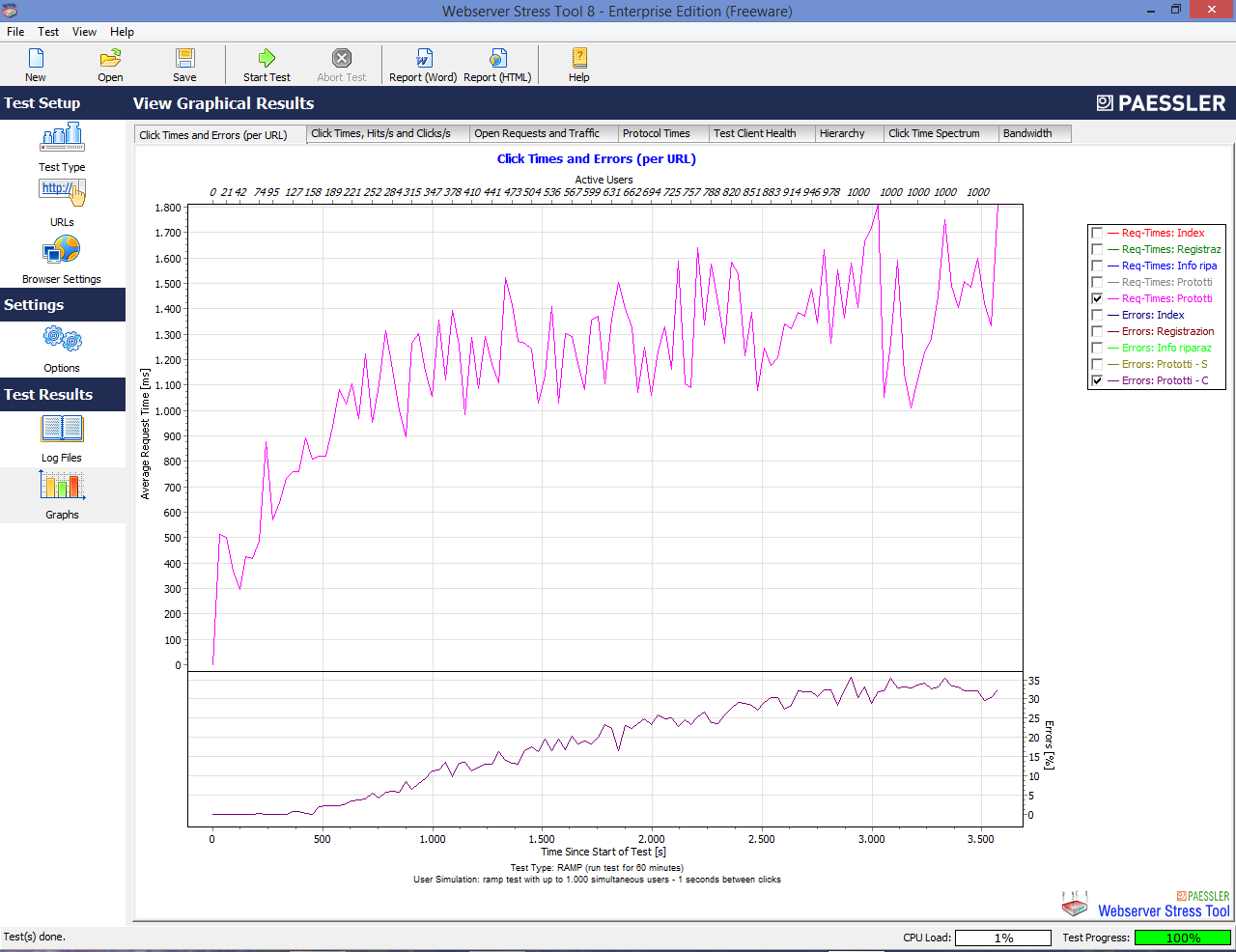
###### Info riparazioni



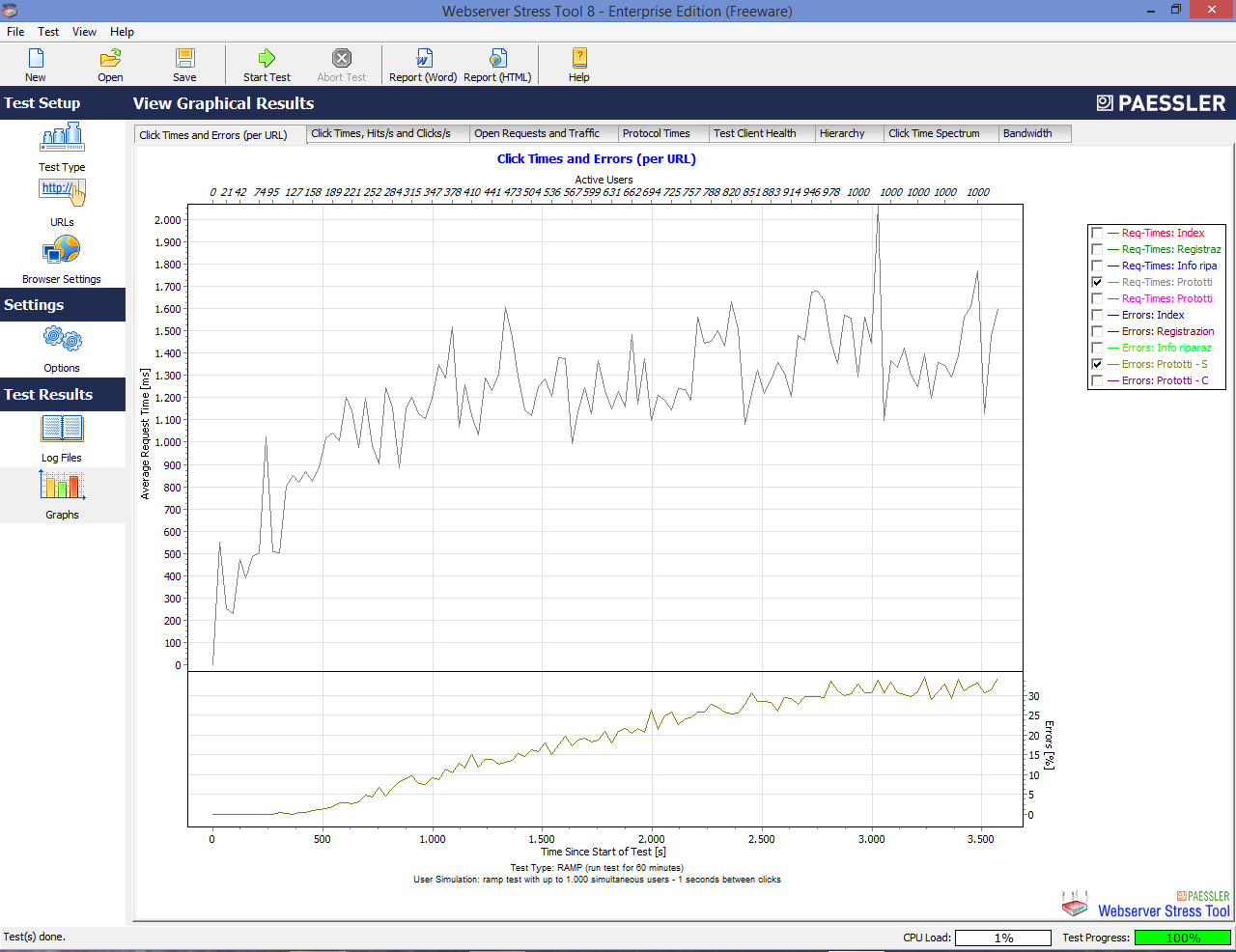
###### Registrazione



###### Prodotti – Computer



###### Prodotti – Smartphone



# Test incident

Il test incident riporta i vari bug riscontrati durante l’utilizzo del sistema.

## Bug Report

Un bug report riassume quanto si è riscontrato. Per ogni bug si indicherà:

* Codice progressivo per l’identificazione del bug
* Una descrizione sintetica dello stesso
* Priorità
  + Bassa: il fault non causa gravi malfunzionamenti al sistema e/o per la sua risoluzione  non si ha un forte impatto sul codice;
  + Media: il fault non permette all’utente alcune operazioni e/o per la sua risoluzione si  ha un discreto impatto sul codice
  + Alta: il fault causa gravi malfunzionamenti al sistema e/o per la sua risoluzione si ha un forte impatto sul codice.
* Stato:
  + In fase di risoluzione
  + Risolto

### Gestione registrazione

#### BR\_1.1

*Descrizione sintetica:*

Pur non avendo completato il form di registrazione viene creato un record nel database.

**Valore ottenuto**:“UTENTE REGISTRATO”

**Valore atteso**: “Per la registrazione sono obbligatori tutti i dati”

**Priorità:** Alta

**Stato:** Risolto

#### BR\_1.2

*Descrizione sintetica:*

Messaggio di errore relativo al campo codice fiscale inesatto.

**Valore ottenuto**:“Per la registrazione sono obbligatori tutti i dati”

**Valore atteso**: “Codice Fiscale non corretto”

**Priorità:** Bassa

**Stato:** Risolto

#### BR\_1.3

*Descrizione sintetica:*

Messaggio di errore relativo al campo email inesatto.

**Valore ottenuto**:“Per la registrazione sono obbligatori tutti i dati”

**Valore atteso**: “Email non corretta”

**Priorità:** Bassa

**Stato:** Risolto

### Gestione autenticazione

#### BR\_2.1

*Descrizione sintetica:*

Login con cliente reindirizza ad home admin.

**Valore ottenuto**:Reindirizzamento su home page admin

**Valore atteso**: Reindirizzamento su home page cliente

**Priorità:** Alta

**Stato:** Risolto

### Gestione utente

#### BR\_3.1

*Descrizione sintetica:*

Inserendo un valore non corretto in indirizzo il sistema va in crash riportando l’utente verso una pagina bianca

**Valore ottenuto**:Pagina bianca

**Valore atteso**: Messaggio di errore “Indirizzo non valido”

**Priorità**: Alta

**Stato**:Risolto

#### BR\_3.2

*Descrizione sintetica:*

Inserendo un valore non corretto in numero di telefono il sistema va in crash riportando l’utente verso una pagina bianca

**Valore ottenuto**:Pagina bianca

**Valore atteso**: Messaggio di errore “Numero di telefono non valido”

**Priorità**: Alta

**Stato**:Risolto

#### BR\_3.3

*Descrizione sintetica:*

Inserendo un valore corretto in Email il sistema da un messaggio non corretto

**Valore ottenuto**:Modifica OK

**Valore atteso**: Messaggio di errore “Mail modificata correttamente”

**Priorità**: Bassa

**Stato**:Risolto

### Gestione prodotti

#### BR\_4.1

*Descrizione sintetica:*

Nell’inserimento del prodotto se l’immagine è già presente viene visualizzato il messaggio di errore ma viene creata comunque l’istanza.

**Valore ottenuto**:Prodotto inserito correttamente

**Valore atteso**: Immagine non inserita, riprovare

**Priorità**: Media

**Stato**: Risolto

#### BR\_4.2

*Descrizione sintetica:*

Nell’inserimento del prodotto, pur non essendo presente in catalogo il prodotto che si vuole aggiungere e pur essendo validi i dati inseriti, il sistema restituisce il messaggio “Prodotto esistente”.

**Valore ottenuto**:Prodotto esistente

**Valore atteso**: Prodotto inserito correttamente

**Priorità**: Alta

**Stato**: Risolto

### Gestione riparazioni

*Non sono stati riscontrati bug*

### Gestione ordine

#### BR\_6.1

*Descrizione sintetica:*

Finalizzato l’acquisto il sistema non svuota il carrello.

**Valore ottenuto**:Il carrello è ancora esistente

**Valore atteso**: Il carrello dovrebbe essere distrutto

**Priorità**: Alta

**Stato**: Risolto

#### BR\_6.2

*Descrizione sintetica:*

Lo storico acquisti mostra gli acquisti di tutti i clienti e non solo quelli effettuati dal cliente loggato.

**Valore ottenuto**:Tutti gli acquisti effettuati

**Valore atteso**: Gli acquisti del cliente loggato

**Priorità**: Media

**Stato**: Risolto

### Gestione dati

*Non sono stati riscontrati bug*

### Gestione interfaccia

#### BR\_8.1

*Descrizione sintetica:*

Il campo newsletter non invia la richiesta di iscrizione

**Valore ottenuto**:Nessuno

**Valore atteso**: Iscrizione effettuata, invio mail

**Priorità**: Bassa

**Stato**: In fase di risoluzione

#### BR\_8.2

*Descrizione sintetica:*

Facendo il login come amministratore e cliccando sulla voce “Richieste di riparazione” del menù Riparazioni viene visualizzato il menù utenti.

**Valore ottenuto**:Cambio delle voci di menù

**Valore atteso**: Reindirizzamento a pagina riparazioni e menù inalterato

**Priorità**: Alta

**Stato**: Risolto

# Security and Penetration Test

Tale test è stato condotto per valutare la sicurezza del sistema sviluppato. L’obiettivo è quindi quello di cercare di trovare tutte le falle, in termini di sicurezza, del sistema e cercarle di isolarle e renderle note per uno sviluppo più accurato dei sistemi di sicurezza.

Il sistema, sviluppato per essere usufruito tramite browser, richiede l’utilizzo di username e password per l’accesso alle aree riservate. Questi parametri, passati tramite variabili $\_POST cioè non visibili in chiaro sul link, vengono confrontati con i dati presenti sul database che identificano in maniera univoca l’utente.

Il database si trova su una macchina esterna a quella dove risiede il web server per aumentare il livello di sicurezza del sistema.

Il web server è accessibile solo tramite la porta 8888 mentre il server MySQL solamente tramite la porta 3306.

L’accesso al database come utente non permette di avere privilegi di root quindi, pur conoscendo gli indirizzi dello stesso, non possono essere applicate modifiche strutturali al database.

## SQL Injection

Questo è un attacco di tipo “code Injection” usata per introdursi, attraverso l’utilizzo di stringhe di codice malevolo inserite nei campi di input, nei sistemi che hanno a che fare con database o che li utilizzano.

Tale attacco non può essere previsto in termini di conseguenze dal programmatore perché il malintenzionato, una volta che l’attacco è andato a buon fine, è in grado di autenticarsi con ampi privilegi in aree riservate del sistema pur non avendo i requisiti per navigarle. Con tali accessi potrebbe tranquillamente visualizzare e modificare i dati presenti nel database.

Si è deciso quindi di procedere all’inserimento, nelle varie text di input, di stringhe di codice malevolo testando così la resistenza a questo tipo di attacco.

Il sistema ha risposto a tali richieste in maniera impeccabile grazie ai controlli che vengono fatti sulle varie stringhe di input presenti nel sistema.

## JavaScript-HTML test

Questo test consiste nell’inserire, in campi dove è possibile introdurre tutti i tipi di caratteri, stringhe di codice malevolo in formato HTML e JavaScript.

Dopo la prima stesura del codice e le prove di questo tipo di attacco abbiamo notato la vulnerabilità del sistema che ci ha portato alla pulizia delle stringhe di input ogni volta che vengono sottomesse attraverso la funzione php *strip\_tags().* Questa funzione prende in input una stringa e la restituisce senza gli eventuali tag che possono essere stati inseriti dall’utente malintenzionato.

# Glossario

|  |  |
| --- | --- |
| Termine | Descrizione |
| *Test stub* | Implementazione parziale di un componente chiamata dalla componente testata da cui dipende quest’ultima. |
| *Test driver* | Implementazione parziale di un componente che chiama la componente testata che dipende da quest’ultima. |
| *Black-Box* | È incentrato sul comportamento I/O e non ha a che fare con la struttura interna della componente. |
| *Bottom-Up* | Testing di tipo incrementale che prevede il testing individuale delle varie componenti del layer più alto e poi il testing dell’integrazione di quest’ultime con il layer immediatamente più in basso. |
| *Test click* | Test che simula i clic del mouse e termina nel momento in cui ogni utente ha finito i click prestabiliti. |
| *Test time* | Test che, stabilito un tempo x, simula click da parte degli utenti. |
| *Test RAMP* | Test che, stabilito un tempo x, simula un crescendo di click partendo da un utente fino al massimo specificato. Si divide in due fasi:   * Fase 1: da 0% a 80% crescono i click; * Fase 2: da 81% a 100% tutti gli utenti cliccano.   NB = le percentuali si riferiscono al tempo. |