

**CNP Capitalia Vita**

**Debugging e stress Test per**  
**l'applicazione InVita**

***Allegato Tecnico***

Milano

<b>Hacking Team S.r.l.</b>	<a href="http://www.hackingteam.it">http://www.hackingteam.it</a>
<i>Via della Moscova, 13 20121 MILANO (MI) - Italy</i>	<a href="mailto:info@hackingteam.it">info@hackingteam.it</a>
<i>Tel. +39.02.29060603</i>	<i>Fax +39.02.63118946</i>

STORIA DEL DOCUMENTO		
Versione	Data	Modifiche Effettuate
1.0	09 Ottobre 2007	Prima emissione
//	//	//
//	//	//

INFORMAZIONI	
Data di Emissione	09 Ottobre 2007
Versione	1.0
Tipologia Documento	Allegato Tecnico
Numero Pagine	6
Numero Allegati	0
Redatto da	Federico Guerrini
Approvato da	Gianluca Vadruccio

## INDICE

1	Richiesta ed obiettivi del cliente.....	4
2	Ambiente di riferimento.....	4
3	Metodologia proposta .....	5
4	Assegnazione competenze e tempistiche .....	7

## 1 Richiesta ed obiettivi del cliente

CNP Capitalia Vita ha richiesto una attività di debugging per l'applicazione InVita. Attualmente l'applicazione presenta fenomeni di saturazione delle risorse che comportano la necessità di periodici riavvii dei server su cui essa viene eseguita (HTTP server di front-end, application server, database server).

L'obiettivo dell'attività consiste nell'identificazione delle cause di tali fenomeni di saturazione, analizzando il comportamento del sistema quando esso viene sottoposto a carichi elevati.

Le attività di debugging saranno condotte da un team costituito da Hacking Team, Capgemini e T-Systems, con la seguente ripartizione dei compiti:

- Hacking Team dovrà progettare ed eseguire i test di generazione del carico; raccogliere i risultati dei test "ai morsetti" (parametri misurabili osservando l'interfaccia HTTP dell'applicazione); fornire indicazioni sulle misure da effettuarsi sui componenti interni del sistema; suggerire ulteriori analisi e/o interventi correttivi;
- CapGemini dovrà fornire i dettagli sulle funzionalità, l'architettura logica e l'architettura fisica dell'applicazione necessari per l'esecuzione dei test; monitorare il comportamento dei singoli moduli dell'applicazione durante l'esecuzione dei test; implementare eventuali interventi correttivi identificati a valle dell'analisi dei risultati dei test;
- T-Systems dovrà fornire la piattaforma hardware per l'esecuzione dei test; monitorare il comportamento, durante l'esecuzione dei test, dei componenti infrastrutturali (http server, application server, DBMS).

## 2 Ambiente di riferimento

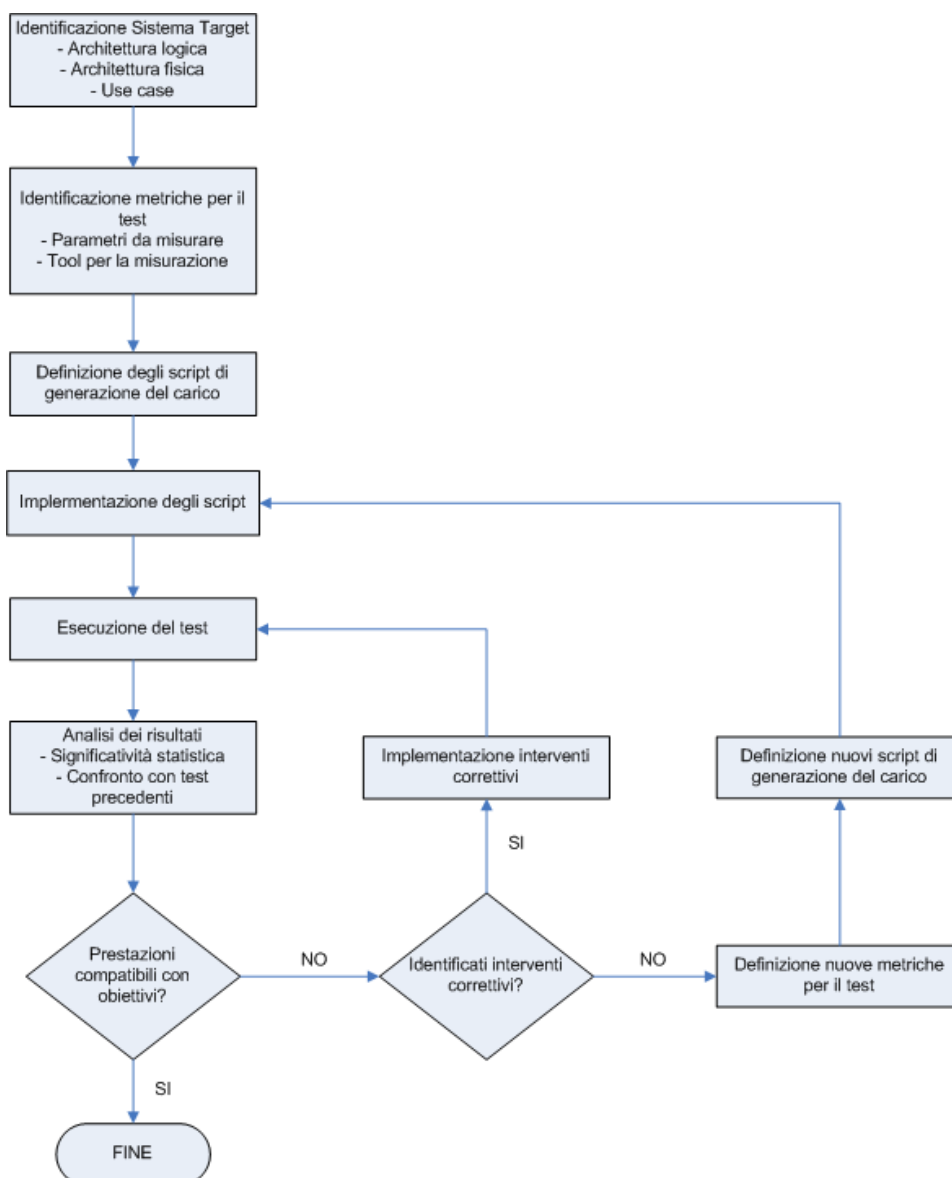
Il sistema target dell'attività oggetto del presente documento è costituito dai moduli software che compongono l'applicazione InVita e dagli elementi infrastrutturali su cui essi sono eseguiti. In particolare:

- applicazione InVita: è una applicazione J2EE che utilizza pagine JSP, servlet e componenti EJB. L'applicazione prevede, oltre alla normale operatività interattiva, anche l'attivazione di processi batch;
- HTTP server: il front-end web è costituito dal server HTTP della suite IBM WebSphere Application Server 5 (WAS5 nel seguito);
- Application server: il container J2EE per l'esecuzione di JSP, servlet, EJB è l'application server IBM WAS5.

- DBMS: il data tier è costituito da un DBMS IBM DB2.
- Infrastruttura hardware: l'ambiente su cui verranno effettuate le attività è costituito da un solo server su piattaforma AIX.

### 3 Metodologia proposta

La metodologia proposta per l'identificazione delle cause dei fenomeni di saturazione dell'applicativo InVita è descritta nel seguente diagramma di flusso.



**Figura 1. Strategia per il debugging**

E' importante sottolineare che le attività sintetizzate nel grafico sono relative alle operazioni di debugging e che con il blocco "fine" si intende la terminazione delle stesse e l'inizio della successiva e finale attività di stress test.

Come mostra la figura 1, la metodologia di debugging proposta prevede le seguenti fasi:

- Identificazione del sistema target: in questa fase vengono raccolte informazioni su architettura logica e fisica del sistema da testare e sui principali use case.
- Identificazione delle metriche: in questa fase vengono definiti i parametri che devono essere misurati allo scopo di rilevare l'insorgere, ed osservare l'evoluzione, delle anomalie di cui si vuole identificare la causa.
- Definizione degli script di generazione del carico: in questa fase viene definito l'insieme di richieste da sottoporre all'applicazione per simulare le condizioni di carico a cui compaiono le anomalie.
- Implementazione degli script: in questa fase si implementano, utilizzando opportuni tool, gli script che permettono la generazione delle sequenze di richieste definite nella fase precedente.
- Esecuzione del test: in questa fase si eseguono gli script implementati nella fase precedente, e si raccolgono i dati di misura secondo quanto stabilito nella fase di "Identificazione dei parametri"
- Analisi dei risultati: in questa fase si verifica la significatività statistica delle misure effettuate e si cercano modalità di aggregazione dei dati (tempo medio per l'esecuzione di uno use case; tempo medio di risposta di singole richieste http; ecc.).

Per la natura dell'attività, finalizzata all'identificazione delle cause di un malfunzionamento, sarà necessario applicare la questa metodologia in modo iterativo. In particolare:

- se in seguito all'analisi dei risultati di una iterazione non vengono identificati elementi utili per il debugging, si procederà alla modifica delle metriche e dei test da effettuare;
- se in seguito all'analisi dei risultati di una iterazione si identificano possibili cause del problema, si procederà all'implementazione di azioni correttive sull'applicativo, i cui effetti saranno valutati ripetendo i test.

## 4 Assegnazione competenze e tempistiche

Seguendo l'approccio proposto in figura, si sintetizzano di seguito i dati di ogni attività, fornendo per ognuna una stima dei tempi di esecuzione.

<b>Debugging di INVITA</b>			
<b>Attività</b>	<b>Descrizione sintetica</b>	<b>Owner</b>	<b>Tempistica</b>
Identificazione sistema target	Preparazione HW e SW ambiente di test	T-Systems CapGemini	Settimana del 8 Ottobre 2007
Use case	Identificazione flussi critici e percorsi applicativi da seguire nei test	CapGemini	
Metriche del test	Identificazione dei parametri da misurare e predisposizione degli eventuali tools di supporto	T-Systems CapGemini	Settimana del 8 Ottobre 2007
Script ed ambiente	Predisposizione ambiente di testing e di scripting	HT	Settimana del 8 Ottobre 2007
Kick-Off Testing	Incontro di controllo e predisposizione finale	T-Systems CapGemini HT	Un giorno della settimana del 15 Ottobre 2007 a conclusione di tutte le attività precedenti
Implementazione degli script	Rifinitura degli script in base all'ambiente predisposto	HT	Giornate a consumo previste fino all'orizzonte massimo del 15 Novembre 2007
Test execution	Esecuzione dei test e monitoring dei parametri stabiliti	T-Systems CapGemini HT	
Analisi dei risultati	Raccolta, analisi dei risultati ed eventuale rischedulazione dei test (dopo l'esecuzione degli interventi correttivi o la definizione dei nuovi script di generazione del carico)	T-Systems CapGemini HT	
<b>Stress Test di INVITA</b>			
<b>Attività</b>	<b>Descrizione sintetica</b>	<b>Owner</b>	<b>Tempistica</b>

Stress Test	<ul style="list-style-type: none"> <li>Tuning degli script esistenti e tarati durante la fase di debugging</li> <li>Stress test fuori orario lavorativo</li> <li>Analisi dei dati raccolti</li> <li>Report finale di capacity</li> </ul>	HT	Consegna entro la prima settimana di Dicembre 2007
-------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----	----------------------------------------------------

