



UNIVERSITÀ DEGLI STUDI DI SALERNO

Dipartimento di Informatica

Corso di Laurea Magistrale in Informatica

TEST CASE SPECIFICATION

INGEGNERIA, GESTIONE ED EVOLUZIONE DEL SOFTWARE

DOCENTI

Prof. Andrea De Lucia

TUTOR

Dott. Stefano Lambiase

Università degli Studi di Salerno

STUDENTI

Francesco Maria Torino

(0522501879)

Francesco Alessandro

Pinto (0522501981)

Stefano Guida

(0522502054)

Anno Accademico 2024-2025

Indice

Elenco delle Figure	iii
Elenco delle Tabelle	iv
1 Test case specification	1
1.1 System Test: Insert request functionality	1
1.1.1 Validazione URL GitHub (SYSTE1 - URL non valido)	1
1.1.2 Validazione URL GitHub (SYSTE2 - URL non valido)	3
1.1.3 Validazione URL GitHub (SYSTE3 - URL non valido)	5
1.1.4 Validazione URL GitHub (SYSTE4 - Data & URL Validi)	7
1.1.5 Validazione Data fine (SYSTE5 - Data non valida)	9
1.1.6 Validazione Data fine (SYSTE6 - Data non valida)	11
1.2 System Test: Get request functionality	13
1.2.1 Validazione Repository (SYSTE7 - Repository valida)	13
1.2.2 Validazione Repository (SYSTE8 - Repository con numero di commit non validi)	15
1.2.3 Validazione Repository (SYSTE9 - Repository con numero di membri non validi)	17
1.2.4 Validazione Repository (SYSTE10 - Repository con numero di milestones non validi)	19

1.2.5	Validazione Repository (SYSTE11 - Repository con geodispersione non valida)	21
1.2.6	Validazione Repository (SYSTE12 - Repository generic error) .	23
1.2.7	Validazione Repository (SYSTE13 - Richieste in stato pending)	25
1.2.8	Validazione Repository (SYSTE14 - Visualizzazione grafo community)	27
1.2.9	Validazione Repository (SYSTE15 - Visualizzazione tabella Smells)	29
1.2.10	Validazione Repository (SYSTE16 - Visualizzazione tabella Metrics)	31
1.2.11	Validazione Repository (SYSTE17 - Visualizzazione tabella Members)	33
	Bibliografia	35

Elenco delle figure

Elenco delle tabelle

1.1	Validazione URL GitHub (SYSTE1)	1
1.2	Validazione URL GitHub (SYSTE2)	3
1.3	Validazione URL GitHub (SYSTE3)	5
1.4	Validazione URL GitHub (SYSTE4)	7
1.5	Validazione Data fine (SYSTE5)	9
1.6	Validazione Data fine (SYSTE6)	11
1.7	Validazione Repository (SYSTE7)	13
1.8	Validazione Repository (SYSTE8)	15
1.9	Validazione Repository (SYSTE9)	17
1.10	Validazione Repository (SYSTE10)	19
1.11	Validazione Repository (SYSTE11)	21
1.12	Validazione Repository (SYSTE12)	23
1.13	Richieste pending (SYSTE13)	25
1.14	Visualizzazione grafo (SYSTE14)	27
1.15	Visualizzazione Smells (SYSTE15)	29
1.16	Visualizzazione Metrics (SYSTE16)	31
1.17	Visualizzazione Members (SYSTE17)	33

Test case specification

1.1 System Test: Insert request functionality

1.1.1 Validazione URL GitHub (SYSTE1 - URL non valido)

Tabella 1.1: Validazione URL GitHub (SYSTE1)

Test Case Id	SYSTE1
Obiettivo	Verificare che, in caso di URL GitHub malformato e data correttamente popolata nel passato, venga mostrato l'errore corretto nel form di richiesta.
Test Frame	Intent: insertRequest <ul style="list-style-type: none">• URL = Non valido• Data = Valida

Ambiente di test	<ul style="list-style-type: none">• Browser: Firefox• Sistema: localhost• Risoluzione: 1290x828• Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale
Input	<ul style="list-style-type: none">• Repository: <code>http://github.com/user</code>• Data fine: 29/06/2025
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"
Oracolo	Il sistema mostra il messaggio di errore: <i>"Please enter a valid GitHub repository URL (https://github.com/username/repo)"</i> , posizionato all'interno del form.

1.1.2 Validazione URL GitHub (SYSTE2 - URL non valido)

Tabella 1.2: Validazione URL GitHub (SYSTE2)

Test Case Id	SYSTE2
Obiettivo	Verificare che, in caso di URL GitHub malformato e data correttamente popolata nel passato, venga mostrato l'errore corretto nel form di richiesta.
Test Frame	Intent: insertRequest <ul style="list-style-type: none">• URL = Non valido• Data = Valida
Ambiente di test	<ul style="list-style-type: none">• Browser: Firefox• Sistema: localhost• Risoluzione: 1290x828• Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale
Input	<ul style="list-style-type: none">• Repository: <code>http://google.com/user/repo</code>• Data fine: 29/06/2025

Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"
Oracolo	Il sistema mostra il messaggio di errore: <i>"Please enter a valid GitHub repository URL (https://github.com/username/repo)"</i> , posizionato all'interno del form.

1.1.3 Validazione URL GitHub (SYSTE3 - URL non valido)

Tabella 1.3: Validazione URL GitHub (SYSTE3)

Test Case Id	SYSTE3
Obiettivo	Verificare che, in caso di URL GitHub malformato e data correttamente popolata nel passato, venga mostrato l'errore corretto nel form di richiesta.
Test Frame	Intent: insertRequest <ul style="list-style-type: none">• URL = Non valido• Data = Valida
Ambiente di test	<ul style="list-style-type: none">• Browser: Firefox• Sistema: localhost• Risoluzione: 1290x828• Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale
Input	<ul style="list-style-type: none">• Repository: €• Data fine: 29/06/2025

Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"
Oracolo	Il sistema mostra il messaggio di errore: <i>"Please enter a valid GitHub repository URL (https://github.com/username/repo)"</i> , posizionato all'interno del form.

1.1.4 Validazione URL GitHub (SYSTE4 - Data & URL Validi)

Tabella 1.4: Validazione URL GitHub (SYSTE4)

Test Case Id	SYSTE4
Obiettivo	Verificare che, in caso di URL GitHub ben formato e data correttamente popolata nel passato, la richiesta venga accettata.
Test Frame	Intent: insertRequest <ul style="list-style-type: none"> • URL = Valido • Data = Valida
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale
Input	<ul style="list-style-type: none"> • Repository: <code>https://github.com/rubygems/bundler</code> • Data fine: 01/01/2019

Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"
Oracolo	Il sistema mostra il messaggio di conferma di accettazione della richiesta: <i>"Request sent successfully!"</i> , posizionato in una modale di successo.

1.1.5 Validazione Data fine (SYSTE5 - Data non valida)

Tabella 1.5: Validazione Data fine (SYSTE5)

Test Case Id	SYSTE5
Obiettivo	Verificare che, in caso di URL GitHub ben formato e data malformata, venga mostrato l'errore corretto nel form di richiesta.
Test Frame	Intent: insertRequest <ul style="list-style-type: none"> • URL = Valido • Data = Non valida
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale
Input	<ul style="list-style-type: none"> • Repository: <code>https://github.com/rubygems/bundler</code> • Data fine: 29/06/2099

Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"
Oracolo	Il sistema mostra il messaggio di errore: <i>"The date must be today or a date in the past"</i> , posizionato all'interno del form.

1.1.6 Validazione Data fine (SYSTE6 - Data non valida)

Tabella 1.6: Validazione Data fine (SYSTE6)

Test Case Id	SYSTE6
Obiettivo	Verificare che, in caso di URL GitHub ben formato e data assente, venga mostrato l'errore corretto nel form di richiesta.
Test Frame	Intent: insertRequest <ul style="list-style-type: none"> • URL = Valido • Data = Non valida
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale
Input	<ul style="list-style-type: none"> • Repository: <code>https://github.com/rubygems/bundler</code> • Data fine: €

Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"
Oracolo	Il sistema mostra il messaggio di errore: <i>"The date must be today or a date in the past"</i> , posizionato all'interno del form.

1.2 System Test: Get request functionality

1.2.1 Validazione Repository (SYSTE7 - Repository valida)

Tabella 1.7: Validazione Repository (SYSTE7)

Test Case Id	SYSTE7
Obiettivo	Verificare che, in caso di richiesta accettata, nel caso in cui la repository rispetti tutti i criteri di correttezza il sistema mostri all'utente la richiesta elaborata con successo.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit non validi (Repository) [100] • Milestone validi (Repository) [> 0] • Membri non validi (Repository) [≥ 2] • Geodispersione community valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+

Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.
Input	<ul style="list-style-type: none">• Repository: <code>https://github.com/rubygems/bundler</code>• Data fine: 01/01/2019
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"
Oracolo	Il sistema mostra la richiesta conclusa con successo nell'apposita sezione.

1.2.2 Validazione Repository (SYSTE8 - Repository con numero di commit non validi)

Tabella 1.8: Validazione Repository (SYSTE8)

Test Case Id	SYSTE8
Obiettivo	Verificare che, in caso di richiesta accettata, nel caso in cui la repository abbia #commit ≤ 100 , venga mostrato l'errore corretto nell'apposita sezione.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit non validi (Repository) [≤ 100] • Milestone validi (Repository) [> 0] • Membri non validi (Repository) [≥ 2] • Geodispersione community valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.

Input	<ul style="list-style-type: none">• Repository: <code>https://github.com/StefanoLambiase/GUIDO</code>• Data fine: 01/01/2019
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"
Oracolo	Il sistema mostra il messaggio di errore: <i>"Invalid Repository: There must be at least 100 commits!"</i> , nell'apposita sezione.

1.2.3 Validazione Repository (SYSTE9 - Repository con numero di membri non validi)

Tabella 1.9: Validazione Repository (SYSTE9)

Test Case Id	SYSTE9
Obiettivo	Verificare che, in caso di richiesta accettata, nel caso in cui la repository abbia #membri < 2, venga mostrato l'errore corretto nell'apposita sezione.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit Validi (Repository) [> 100] • Milestone validi (Repository) [> 0] • Membri non validi (Repository) [< 2] • Geodispersione community valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.

Input	<ul style="list-style-type: none">• Repository: <code>https://github.com/legames/legames</code>• Data fine: 01/01/2019
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"
Oracolo	Il sistema mostra il messaggio di errore: <i>"Invalid Repository: Not enough members (min. 2)!"</i> , nell'apposita sezione.

1.2.4 Validazione Repository (SYSTE10 - Repository con numero di milestones non validi)

Tabella 1.10: Validazione Repository (SYSTE10)

Test Case Id	SYSTE10
Obiettivo	Verificare che, in caso di richiesta accettata, nel caso in cui la repository abbia #milestones < 2, venga mostrato l'errore corretto nell'apposita sezione.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit Validi (Repository) [> 100] • Milestone non validi (Repository) [< 1] • Membri validi (Repository) [> 2] • Geodispersione community valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.

Input	<ul style="list-style-type: none">Repository: <code>https://github.com/FrancescoTorino1999/angular</code>Data fine: 01/01/2019
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"
Oracolo	Il sistema mostra il messaggio di errore: <i>"Invalid Repository: No milestones found (min. 1)!"</i> , nell'apposita sezione.

1.2.5 Validazione Repository (SYSTE11 - Repository con geodispersione non valida)

Tabella 1.11: Validazione Repository (SYSTE11)

Test Case Id	SYSTE11
Obiettivo	Verificare che, in caso di richiesta accettata, nel caso in cui la repository non abbia informazioni riguardo la geodispersione sufficienti, venga mostrato l'errore corretto nell'apposita sezione.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit Validi (Repository) [> 100] • Milestone validi (Repository) [> 1] • Membri validi (Repository) [> 2] • Geodispersione community non valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+

Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.
Input	<ul style="list-style-type: none">• Repository: <code>https://github.com/SimpleMobileTools/Simple-Calculator</code>• Data fine: 01/03/2023
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"
Oracolo	Il sistema mostra il messaggio di errore: <i>"Invalid Repository: Insufficient geographical data!"</i> , nell'apposita sezione.

1.2.6 Validazione Repository (SYSTE12 - Repository generic error)

Tabella 1.12: Validazione Repository (SYSTE12)

Test Case Id	SYSTE12
Obiettivo	Verificare che, in caso di richiesta accettata, qualora la repository non esista o l'utente non abbia i permessi di accesso, venga mostrato il messaggio di errore corretto nell'apposita sezione.
Test Frame	Intent: getRequest <ul style="list-style-type: none"> • URL = Valido • Data = Valida
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.
Input	<ul style="list-style-type: none"> • Repository: <code>https://github.com/netty/netty</code> • Data fine: 01/06/2025

Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"
Oracolo	Il sistema mostra il messaggio di errore: <i>"An Error Occurred during the analysis... Please try again later!"</i> , nell'apposita sezione.

1.2.7 Validazione Repository (SYSTE13 - Richieste in stato pending)**Tabella 1.13:** Richieste pending (SYSTE13)

Test Case Id	SYSTE13
Obiettivo	Verificare che, in caso di richiesta accettata, nella sezione View your requests appaia la richiesta in stato di elaborazione.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit Validi (Repository) [> 100] • Milestone validi (Repository) [> 1] • Membri validi (Repository) [> 2] • Geodispersione community valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.

Input	<ul style="list-style-type: none">• Repository: <code>https://github.com/rubygems/bundler</code>• Data fine: 01/01/2019
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"
Oracolo	L'utente visualizza la richiesta fatta in stato PENDING.

1.2.8 Validazione Repository (SYSTE14 - Visualizzazione grafo community)

Tabella 1.14: Visualizzazione grafo (SYSTE14)

Test Case Id	SYSTE14
Obiettivo	Verificare che, in caso di richiesta accettata, nella sezione di dettaglio della richiesta (una volta conclusa l'elaborazione), venga visualizzato il grafo rappresentante la community costruito dal Tool TOAD.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit Validi (Repository) [> 100] • Milestone validi (Repository) [> 1] • Membri validi (Repository) [> 2] • Geodispersione community valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.

Input	<ul style="list-style-type: none">• Repository: <code>https://github.com/rubygems/bundler</code>• Data fine: 01/01/2019
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"7. L'utente clicca sulla richiesta in stato "SUCCESS"
Oracolo	L'utente visualizza il grafo della community riguardante la richiesta fatta ed elaborata dal Tool TOAD.

1.2.9 Validazione Repository (SYSTE15 - Visualizzazione tabella Smells)

Tabella 1.15: Visualizzazione Smells (SYSTE15)

Test Case Id	SYSTE15
Obiettivo	Verificare che, in caso di richiesta accettata, nella sezione di dettaglio della richiesta (una volta conclusa l'elaborazione), venga visualizzata la tabella degli smells trovati nella repository.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit Validi (Repository) [> 100] • Milestone validi (Repository) [> 1] • Membri validi (Repository) [> 2] • Geodispersione community valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.

Input	<ul style="list-style-type: none">• Repository: <code>https://github.com/rubygems/bundler</code>• Data fine: 01/01/2019
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"7. L'utente clicca sulla richiesta in stato "SUCCESS"8. L'utente clicca sul pulsante "Smells"
Oracolo	L'utente visualizza la tabella che riporta gli smells della repository trovati da TOAD.

1.2.10 Validazione Repository (SYSTE16 - Visualizzazione tabella Metrics)

Tabella 1.16: Visualizzazione Metrics (SYSTE16)

Test Case Id	SYSTE16
Obiettivo	Verificare che, in caso di richiesta accettata, nella sezione di dettaglio della richiesta (una volta conclusa l'elaborazione), venga visualizzata la tabella delle metriche calcolate sulla community che ha lavorato nella repository.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit Validi (Repository) [> 100] • Milestone validi (Repository) [> 1] • Membri validi (Repository) [> 2] • Geodispersione community valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+

Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.
Input	<ul style="list-style-type: none"> Repository: <code>https://github.com/rubygems/bundler</code> Data fine: 01/01/2019
Flusso di eventi	<ol style="list-style-type: none"> 1. L'utente clicca sulla sezione "Community Inspector" 2. L'utente clicca su "Insert Request" 3. L'utente inserisce in input la repository 4. L'utente inserisce in input la data fine 5. Clicca su "Aggiungi richiesta" 6. L'utente entra nell'area "View your requests" 7. L'utente clicca sulla richiesta in stato "SUCCESS" 8. L'utente clicca sul pulsante "Metrics"
Oracolo	L'utente visualizza la tabella che riporta le metriche della community che ha lavorato alla repository trovate da TOAD.

1.2.11 Validazione Repository (SYSTE17 - Visualizzazione tabella Members)

Tabella 1.17: Visualizzazione Members (SYSTE17)

Test Case Id	SYSTE17
Obiettivo	Verificare che, in caso di richiesta accettata, nella sezione di dettaglio della richiesta (una volta conclusa l'elaborazione), venga visualizzata la tabella dei membri della community.
Test Frame	<p>Intent: getRequest</p> <ul style="list-style-type: none"> • URL = Valido • Data = Valida • Commit Validi (Repository) [> 100] • Milestone validi (Repository) [> 1] • Membri validi (Repository) [> 2] • Geodispersione community valida (Repository)
Ambiente di test	<ul style="list-style-type: none"> • Browser: Firefox • Sistema: localhost • Risoluzione: 1290x828 • Versione Selenium: v4+
Pre-Condizione	L'utente ha installato GUIDO e si trova sulla pagina iniziale.

Input	<ul style="list-style-type: none">• Repository: <code>https://github.com/rubygems/bundler</code>• Data fine: 01/01/2019
Flusso di eventi	<ol style="list-style-type: none">1. L'utente clicca sulla sezione "Community Inspector"2. L'utente clicca su "Insert Request"3. L'utente inserisce in input la repository4. L'utente inserisce in input la data fine5. Clicca su "Aggiungi richiesta"6. L'utente entra nell'area "View your requests"7. L'utente clicca sulla richiesta in stato "SUCCESS"8. L'utente clicca sul pulsante "Members"
Oracolo	L'utente visualizza la tabella che riporta i membri della community.

Bibliografia
