

tesi di laurea magistrale

Un processo di sviluppo supportato da strumenti per la riduzione dei test breakage in ambito di applicazioni web template-based

Anno Accademico 2018/2019

relatore

Ch.mo Prof. Porfirio Tramontana

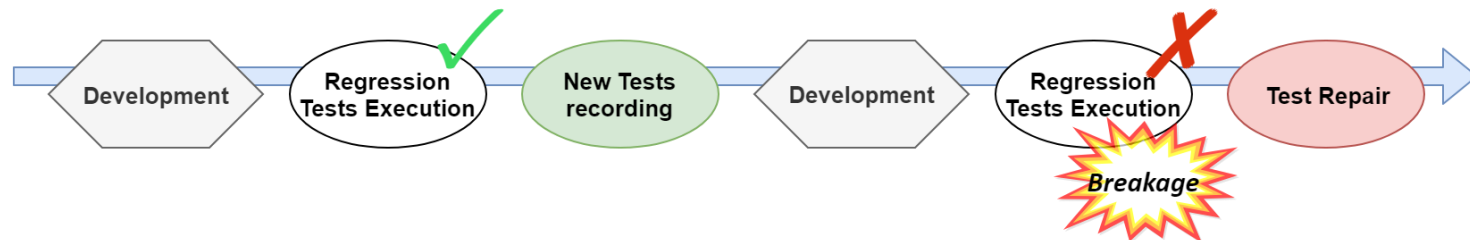
candidato

Pierantonio Cangianiello

Matr. M63/158

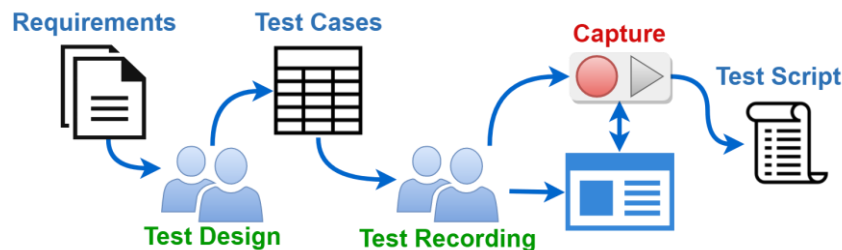
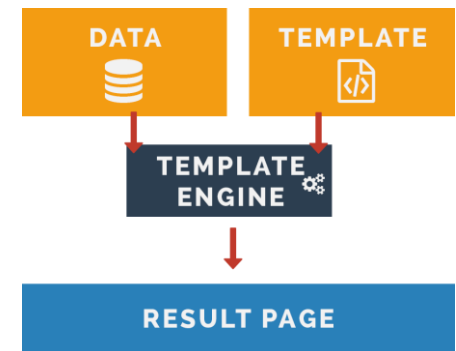
Applicazioni Web: Manutenzione e Test breakage

- **Manutenzione di interfacce grafiche di applicazioni web**
 - **Evolutiva:** cambio di stile, innovazioni UX, accessibilità, nuove *gesture*, cambio di livello di astrazione, nuovi fattori di forma, ecc.
 - **Correttiva:** risoluzione di bug grafici, di presentazione del dato, di internazionalizzazione, ecc.
- **GUI Test Automation**
 - **Test Breakage**



Ambito di riferimento e stato dell'arte

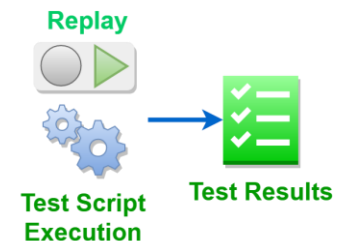
- **Applicazioni web template-based**
- **User Interface Testing: tecnica capture/replay**
 - **Katalon Recorder:** estensione per browser basata sul progetto Selenium
 - **Locatori:** istruzioni per identificare elementi delle pagine web coinvolti nei test
- **Stato dell'arte:** tecniche di test repair e/o breakage prevention



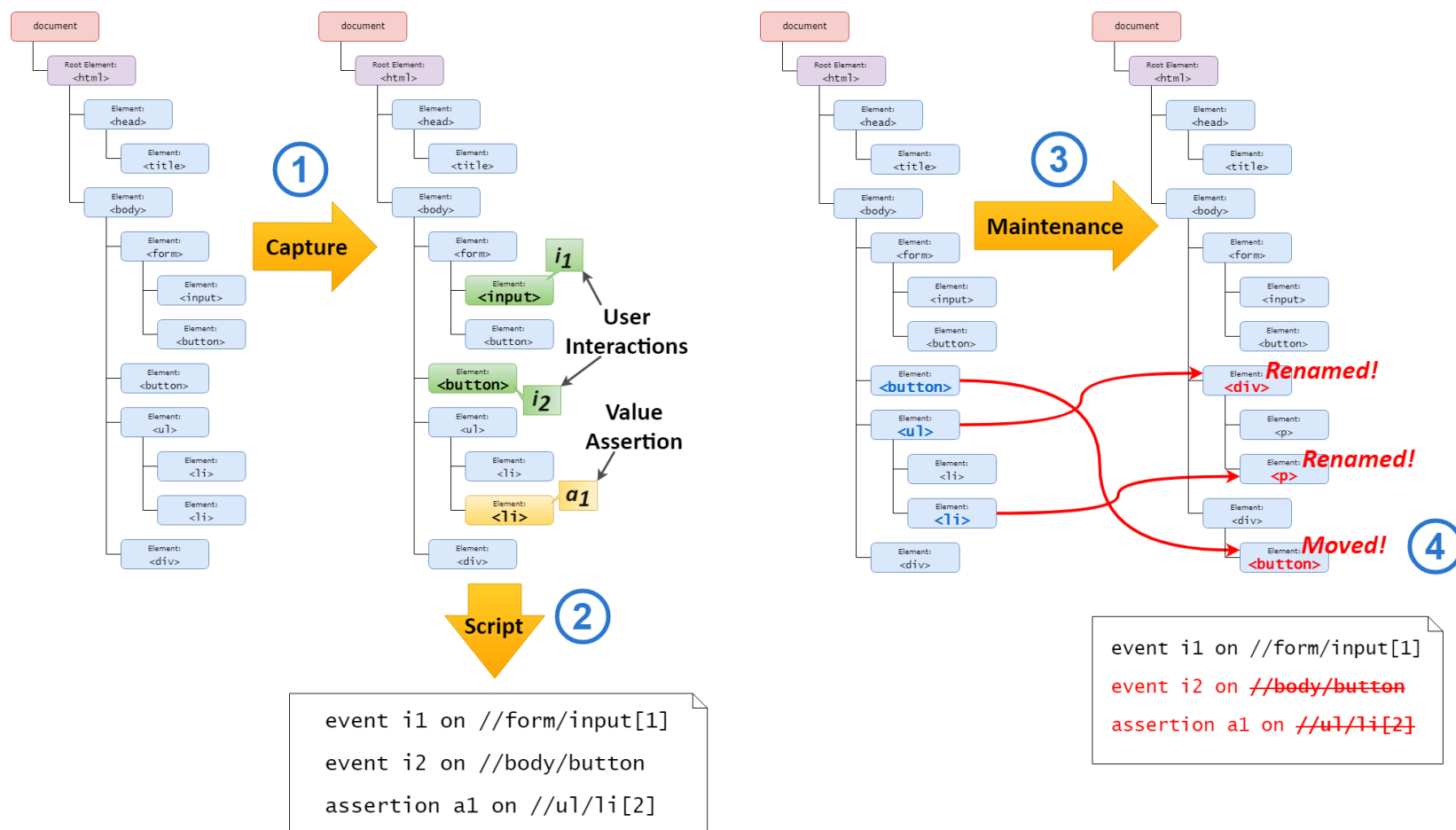
UI Maintenance

```

<input type="text" class="form-control" ng-model="task.description">
</td>
<td ng-if="task.status" class="text-align: center">
  <div class="progress">
    <div class="progress-bar" style="width: {{task.progress}}%;></div>
  </div>
  <div class="text">
    <span class="label label-default" ng-if="task.status === 'pending'>Pending</span>
    <span class="label label-success" ng-if="task.status === 'done'>Done</span>
  </div>
</td>
</tr>
</table>
  
```



Scenario di test breakage



Tecnica di rilevazione dei test breakage

■ Strategia

- Creazione automatica di **referimenti bidirezionali** tra template e test

■ Obiettivi

- Rilevazione dei breakage senza rieseguire i test
- Analisi di impatto della manutenzione dei template sui test
- Invasività minima nel codice dei template

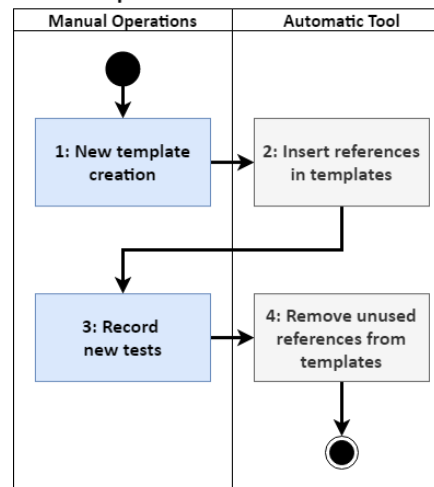
Test Suites		Command	Target
freemarker-demo-1*	⋮	open	http://localhost:8080/freemarker-sample/index
Test Case Freemarker		click	//*[@x-test-tpl-1]//*[@x-test-tpl-7]//*[@x-test-hook-15]
		click	//*[@x-test-tpl-1]//*[@x-test-tpl-7]//*[@x-test-tpl-30]//*[@x-test-hook-48]
		click	//*[@x-test-tpl-1]//*[@x-test-tpl-7]//*[@x-test-tpl-16]//*[@x-test-hook-20]
		type	//*[@x-test-tpl-1]//*[@x-test-tpl-7]//*[@x-test-tpl-16]//*[@x-test-hook-20]
		click	//*[@x-test-tpl-1]//*[@x-test-tpl-7]//*[@x-test-tpl-6]//*[@x-test-hook-23]
		select	//*[@x-test-tpl-1]//*[@x-test-tpl-7]//*[@x-test-tpl-16]//*[@x-test-hook-23]

```
<td>
  <input name="description" type="text" class="form-control" value=
    "${task.description!''}" x-test-hook-20>
</td>
<td>
  <select name="user" value="${task.user!''}" class="form-control"
    x-test-hook-23>
    <#list model.users as user>
      <option>${user.name}</option>
    </#list>
  </select>
</td>
```

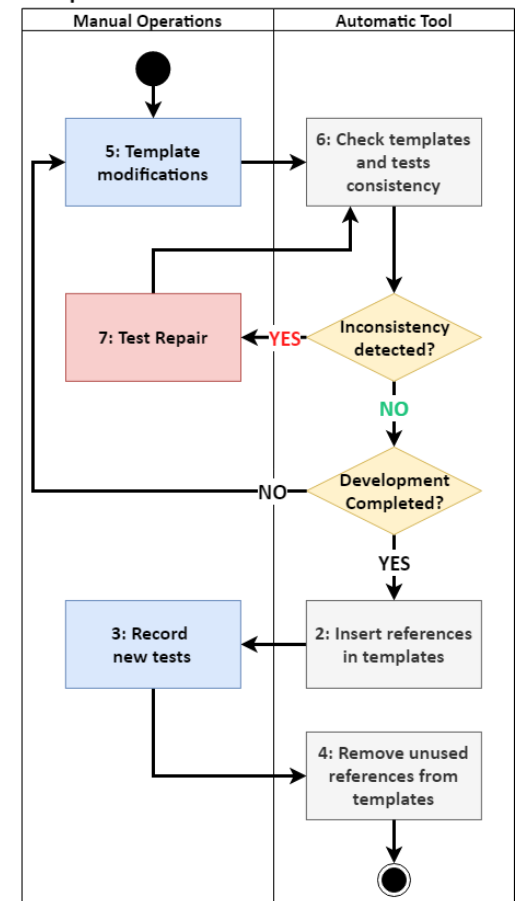
Processo di sviluppo e manutenzione

1. Implementazione nuovi template
2. Creazione riferimenti nei template (*instrumentation*)
3. Registrazione test
4. Rimozione riferimenti inutilizzati dai template
5. Modifica di template esistenti
6. Controllo coerenza tra template e test
7. Test repair

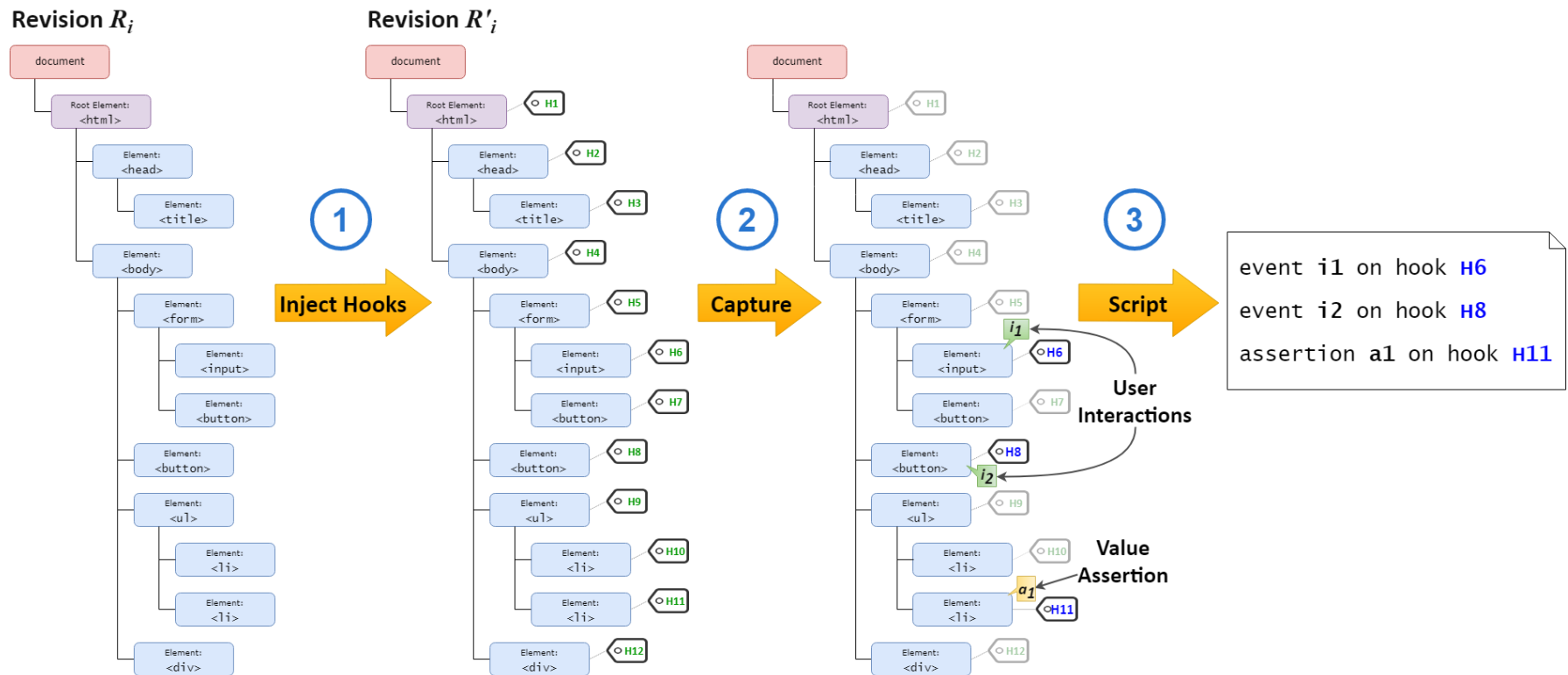
New Templates Process



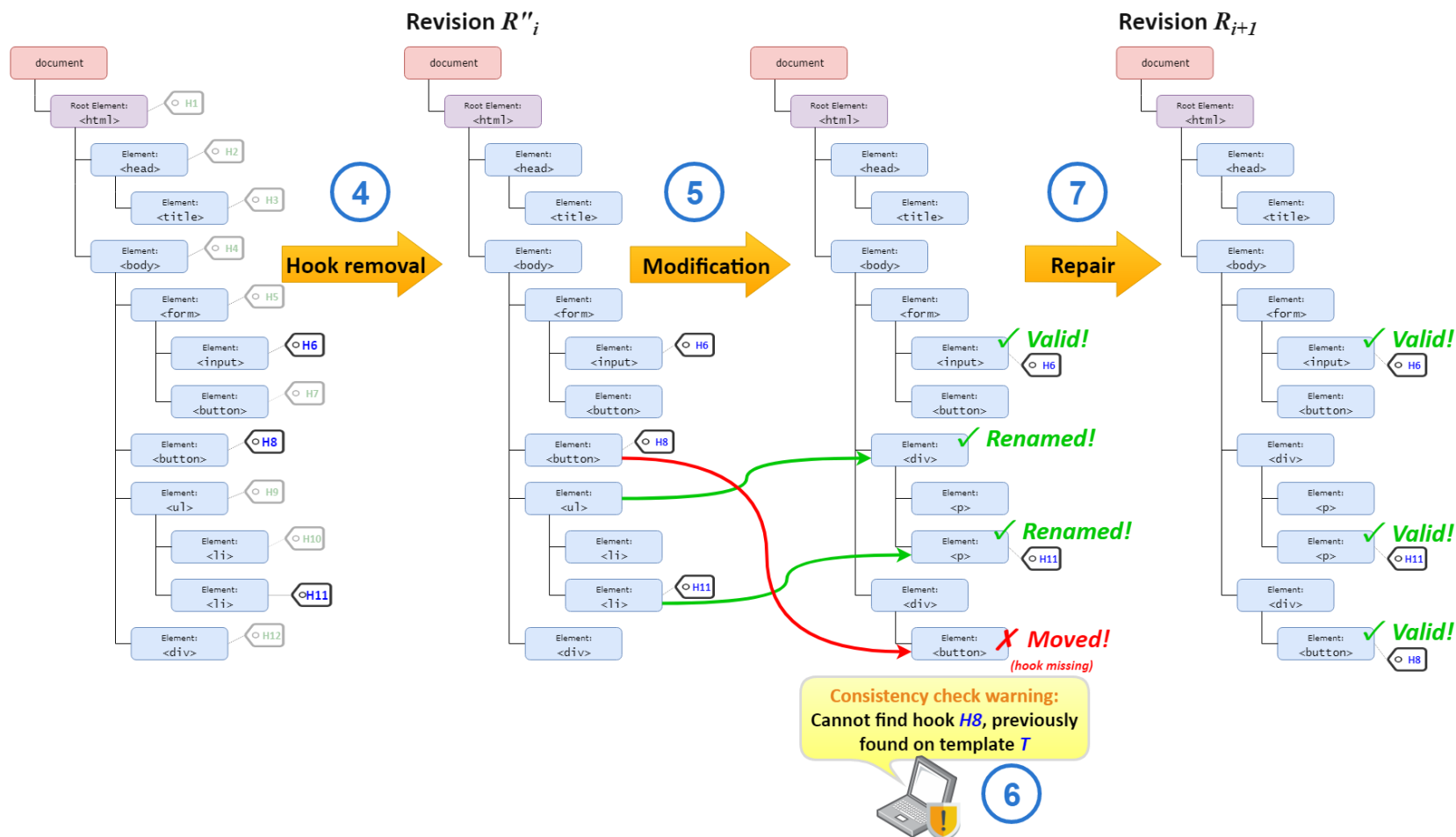
Templates Modification Process



Processo di sviluppo e manutenzione: esempio (1/2)



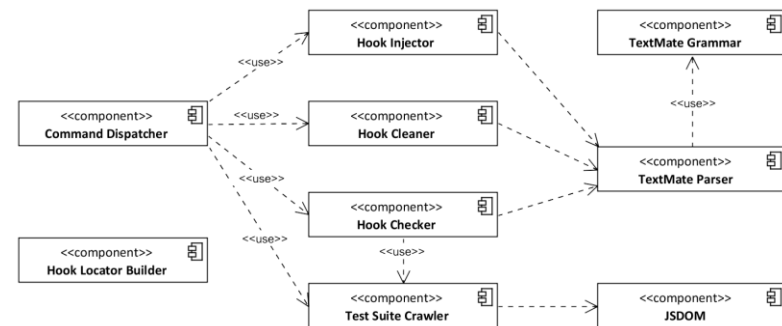
Processo di sviluppo e manutenzione: esempio (2/2)



Strumenti di supporto per il processo

■ Command Dispatcher

- Hook Injector
- Hook Cleaner
- Hook Consistency Checker



- **TextMate:** libreria per il parsing dei template basata su grammar definitions; consente l'indipendenza dalla tecnologia specifica di templating
- **Hook Locator Builder:** script di estensione per Katalon Recorder, crea i locatori basati sugli hook presenti nei template



```
<div class="{ div_style|default("btn-group") }">
  {% if default_item.icon %}
  <i class="material-icons">{{ default_item.icon }}</i>
  {% endif %}
</div>
```

Sperimentazione

1. Verifica compatibilità con varie tecnologie di templating

- Angular, FreeMarker, Twig, Smarty, ...

2. Sperimentazione empirica

- **Question:** Verificare **se** e di **quanto** si riduce il **numero di locatori broken** utilizzando la strategia hook-based rispetto a quella di Selenium
- **Object:** PrestaShop e-commerce
- **Factors:** 15 interventi di manutenzione sui template, selezionati dalla commit history del progetto, ordinando in maniera decrescente i template per dimensione e numero di modifiche
- **Independent Variable:** Strategia di generazione locatori (categorica, 2 livelli)
 - Locatori generati secondo la strategia di Selenium
 - Locatori hook-based
- **Dependent variable:** numero di breakage

Sperimentazione

- **Risultati e analisi quantitativa**
- **Analisi qualitativa: problemi risolti**
 - Locatori **attribute-based**: attributo non trovato, testo non corrispondente...
 - Locatori **hierarchy-based**: cambio nome tag, posizione elementi, innestamento template...
 - Locatori **index-based**: cambio di posizione relativa tra elementi simili.
- **Problematiche riscontrate**
- **Threats to validity**
 - **Internal**: selection, instrumentation
 - **External**: population

#	Maintenance description	Injected Locators	Selenium broken locators	Hook-based broken locators
1	Order Detail Template layout change	15	9	0
2	Order Detail Template layout change	15	8	0
3	Order Detail (no return) Template layout change	13	8	0
4	Stores Page Template layout change	11	4	0
5	Checkout Product Template layout change	12	2	0
6	Checkout Cart Totals Template layout change	7	3	0
7	Checkout Address Step Template layout change	8	1	0
8	Newsletter Block Component redesign	4	1	0
9	Currencies Block Component redesign	2	1	0
10	Blockcart Product Line Template layout change	5	2	0
11	Product (Catalog) Template layout improvements	14	3	0
12	Checkout Delivery Step Template redesign	10	2	0
13	Order Messages Component layout change	8	1	0
14	Order Followup Template layout change	5	1	0
15	Contact Form Template layout change	6	2	0
Totals:		135	48	0
		100%	35,56%	0%

Conclusioni e sviluppi futuri

■ Obiettivi raggiunti

- Rilevazione dei breakage senza eseguire i test
- Semplificazione del processo di repair guidato dal controllo consistenza
- Supporto di diverse tecnologie di templating

■ Sviluppi futuri

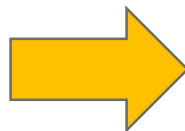
- Controllo di coerenza online (watchdog)
- Conversione locatori da script preesistenti
- Integrazione con IDE
- Sperimentazione sui tempi di sviluppo e test repair

TextMate parser: Scope Tree

```

text.html.twig
meta.tag.other.html
→ punctuation.definition.tag.begin.html '<'
entity.name.tag.other.html 'div'
meta.attribute-with-value.html
→ entity.other.attribute-name.html 'class'
punctuation.separator.key-value.html '='
punctuation.definition.string.begin.html '"'
punctuation.section.tag.twig '{{'
variable.other.twig 'div_style'
keyword.operator.other.twig '|'
support.function.twig 'default'
punctuation.definition.parameters.begin.twig '('
string.quoted.double.twig
punctuation.definition.string.begin.twig '"'
string.quoted.double.twig 'btn-group'
punctuation.definition.string.end.twig '"'
punctuation.definition.parameters.end.twig ')'
punctuation.section.tag.twig '}}'
→ punctuation.definition.string.end.html '"'
→ punctuation.definition.tag.end.html '>'
meta.tag.template.value.twig
punctuation.section.tag.twig '{%'
keyword.control.twig 'if'
variable.other.twig 'default_item'
punctuation.separator.property.twig '.'
variable.other.property.twig 'icon'
punctuation.section.tag.twig '%}'
meta.tag.other.html
→ punctuation.definition.tag.begin.html '<'
entity.name.tag.other.html 'i'
meta.attribute-with-value.html
→ entity.other.attribute-name.html 'class'
punctuation.separator.key-value.html '='
punctuation.definition.string.begin.html '"'
string.quoted.double.html 'material-icons'
punctuation.definition.string.end.html '"'
→ punctuation.definition.tag.end.html '>'
meta.tag.template.value.twig
punctuation.section.tag.twig '{%'
variable.other.twig 'default_item'
punctuation.separator.property.twig '.'
variable.other.property.twig 'icon'
punctuation.section.tag.twig '%}'
→ punctuation.definition.tag.begin.html '</'
entity.name.tag.other.html 'i'
→ punctuation.definition.tag.end.html '>'
meta.tag.template.value.twig
punctuation.section.tag.twig '{%'
keyword.control.twig 'endif'
punctuation.section.tag.twig '%}'
meta.tag.other.html
→ punctuation.definition.tag.begin.html '</'
entity.name.tag.other.html 'div'
→ punctuation.definition.tag.end.html '>'

```



```

<div class="{{ div_style|default("btn-group") }}">
  {% if default_item.icon %}
  → <i class="material-icons">{{ default_item.icon }}</i>
  {% endif %}
</div>

```

