



# Software process and product quality: Towards an integrated approach

Anno Accademico 2015/2016

Relatore: Danilo Caivano

Laureando: Cosoli Gabriele

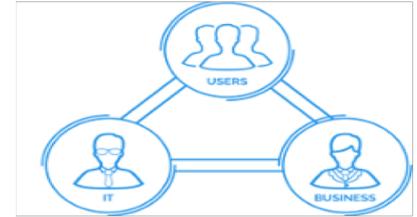
Dipartimento di Informatica - Università degli Studi di Bari

Via Orabona, 4 - 70125 - Bari

Tel: +39.080.5443270 | Fax: +39.080.5442536

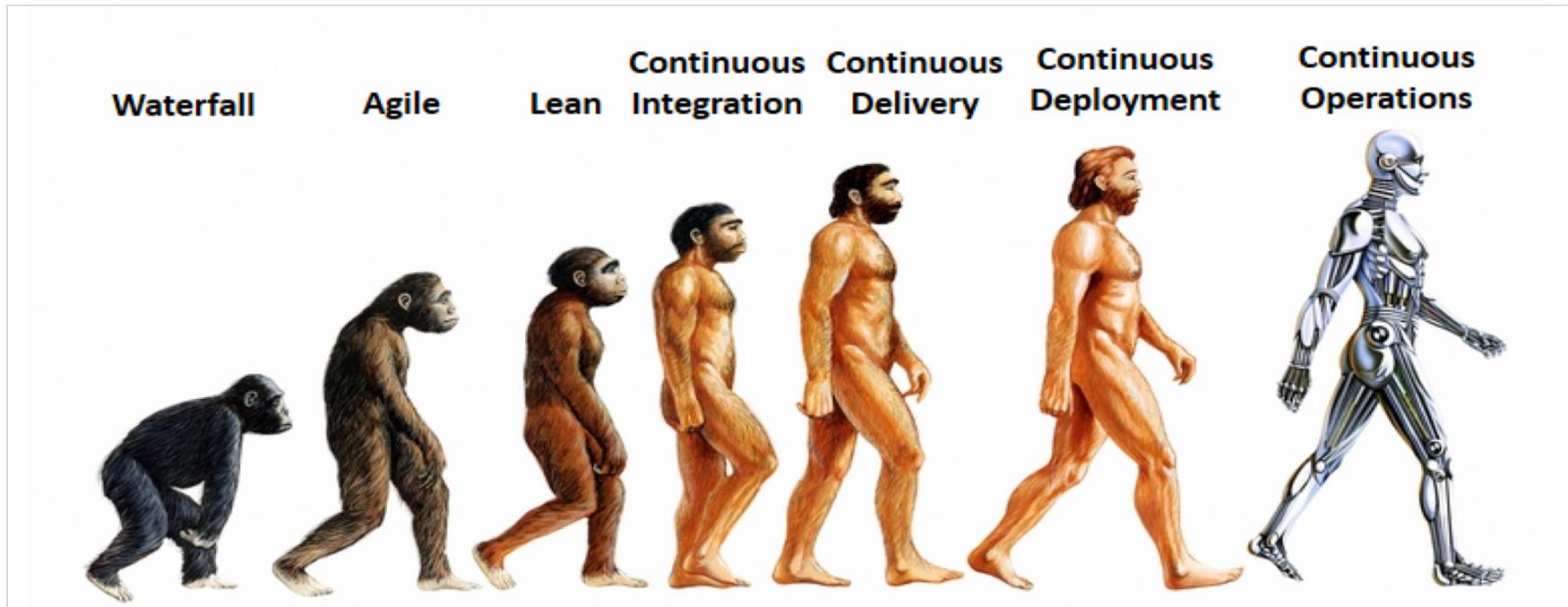
[serlab.di.uniba.it](http://serlab.di.uniba.it)

## Scenario di riferimento



- ⇒ Storicamente la qualità del prodotto software, a causa dell'intangibilità del software, è stata perseguita attraverso la qualità del processo.
- ⇒ Ciò in controtendenza rispetto ai settori industriali tradizionali che, grazie alla forte integrazione ed automazione dei processi in uso, puntavano invece sui materiali e sulla verifica delle caratteristiche di qualità operata direttamente sul prodotto finale
- ⇒ Oggigiorno la qualità finale di un sistema software è il risultato di tre fattori chiave:
  - ❑ Qualità Interna: che valuta l'insieme delle caratteristiche strutturali del sistema software sviluppato
  - ❑ Qualità Esterna: che valuta la conformità tra il sistema ed i requisiti che esso deve implementare e l'assenza di difetti
  - ❑ Qualità in Uso: che valuta la percezione dell'utilizzatore finale del sistema durante l'utilizzo
- ⇒ Anche il software oggi sta progressivamente trasformandosi in industria

# L'evoluzione dei processi di sviluppo software

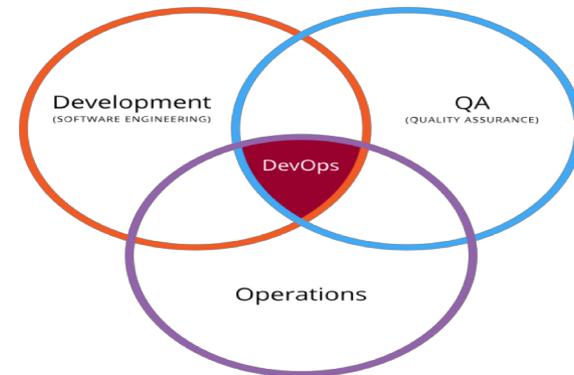


Waterfall	Evolutivo	Sviluppo per componenti	Prototipazione rapida	Spirale	Agile	<b>Continua integrazione</b>
-----------	-----------	-------------------------	-----------------------	---------	-------	------------------------------

# Scopo del lavoro di tesi

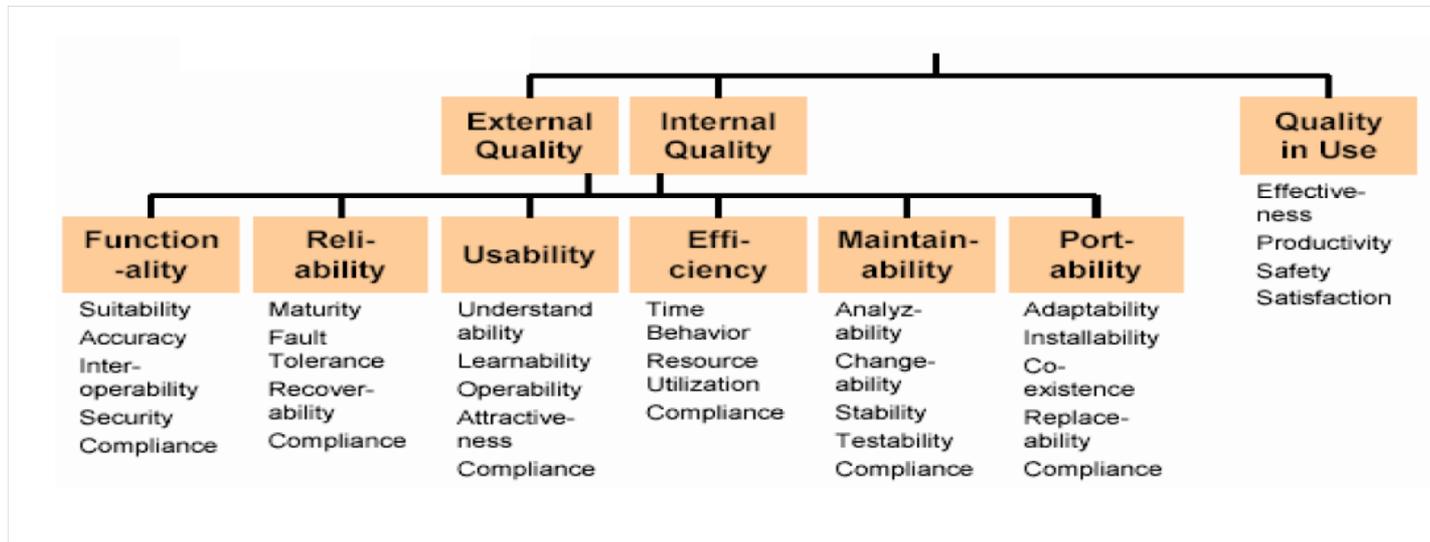
⇒ Lo scopo del lavoro di tesi è stato:

- ❑ La proposizione di un processo di sviluppo:
  - Orientato al customer, al fine di accrescere la qualità in uso
  - Orientato alla qualità interna del prodotto software
  - Fortemente integrato nelle sue fasi al fine di ridurre gli errori umani e favorire così la qualità esterna
- ❑ La definizione ed implementazione di una infrastruttura di supporto allo sviluppo, verifica della qualità interna, integrazione continua ed esercizio del software
- ❑ La sperimentazione dell'approccio proposto nell'ambito di due case studi industriali



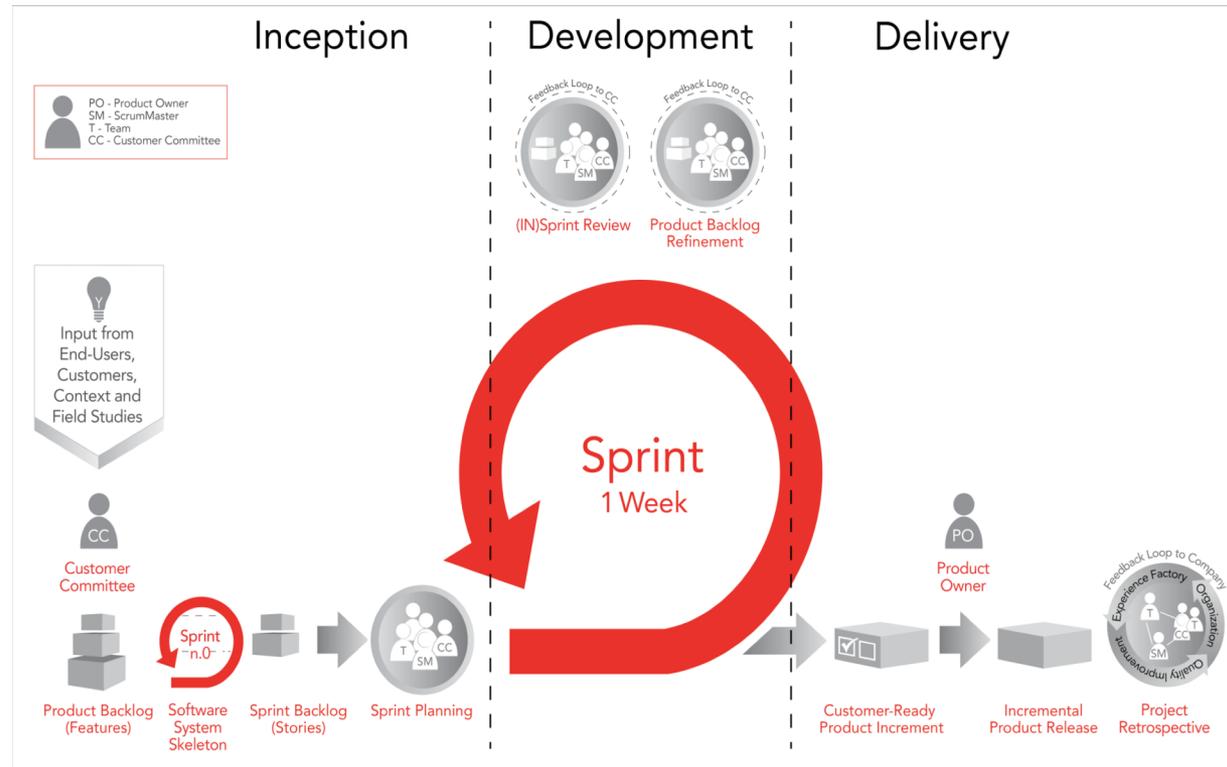
# Qualità del software - ISO 25000:

- ⇒ Qualità interna : Rappresenta le qualità intrinseche del prodotto .
- ⇒ Qualità esterna : Si riferisce alle prestazioni del prodotto e del suo comportamento dinamico in relazione ai requisiti richiesti.
- ⇒ Qualità in uso: Si identifica con il livello di soddisfazione dell'utente nell'utilizzo del software e con il grado di efficacia ed efficienza nello svolgere da esso i compiti attesi.

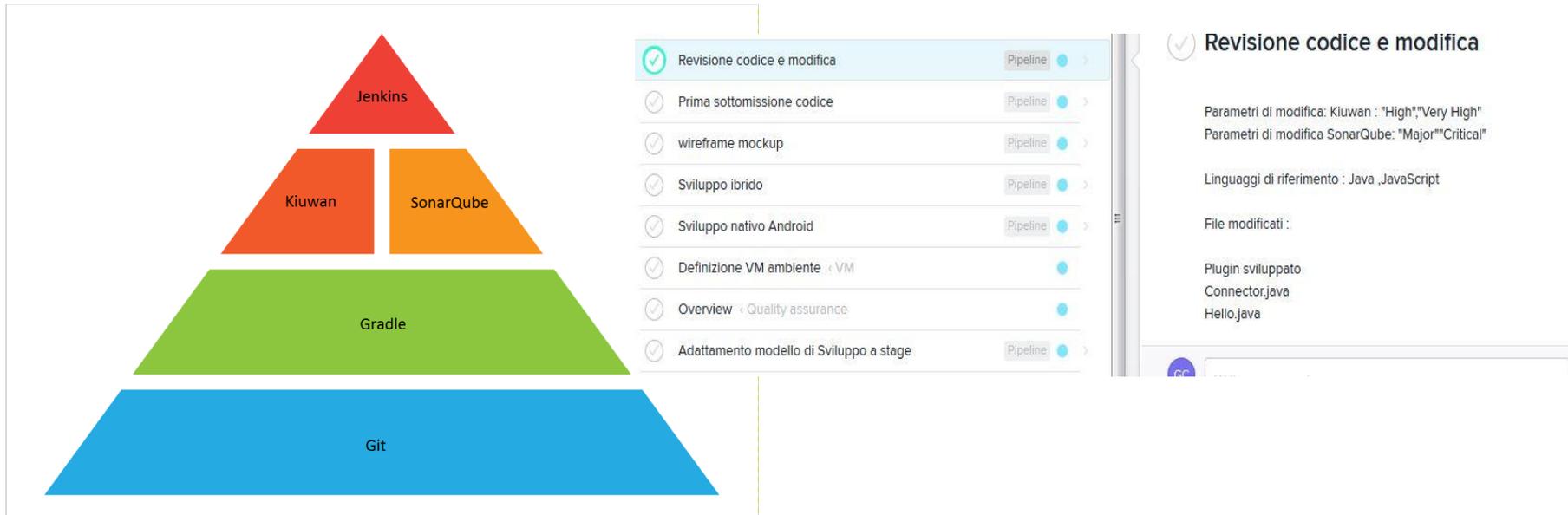


# Approccio Proposto – Processo di sviluppo

- ⇒ Customer Committee
- ⇒ Inception
- ⇒ Sprint n.0
- ⇒ Scrum Island
- ⇒ 1 Week Time Boxed Sprint
- ⇒ (IN)Sprint Review
- ⇒ Project retrospective

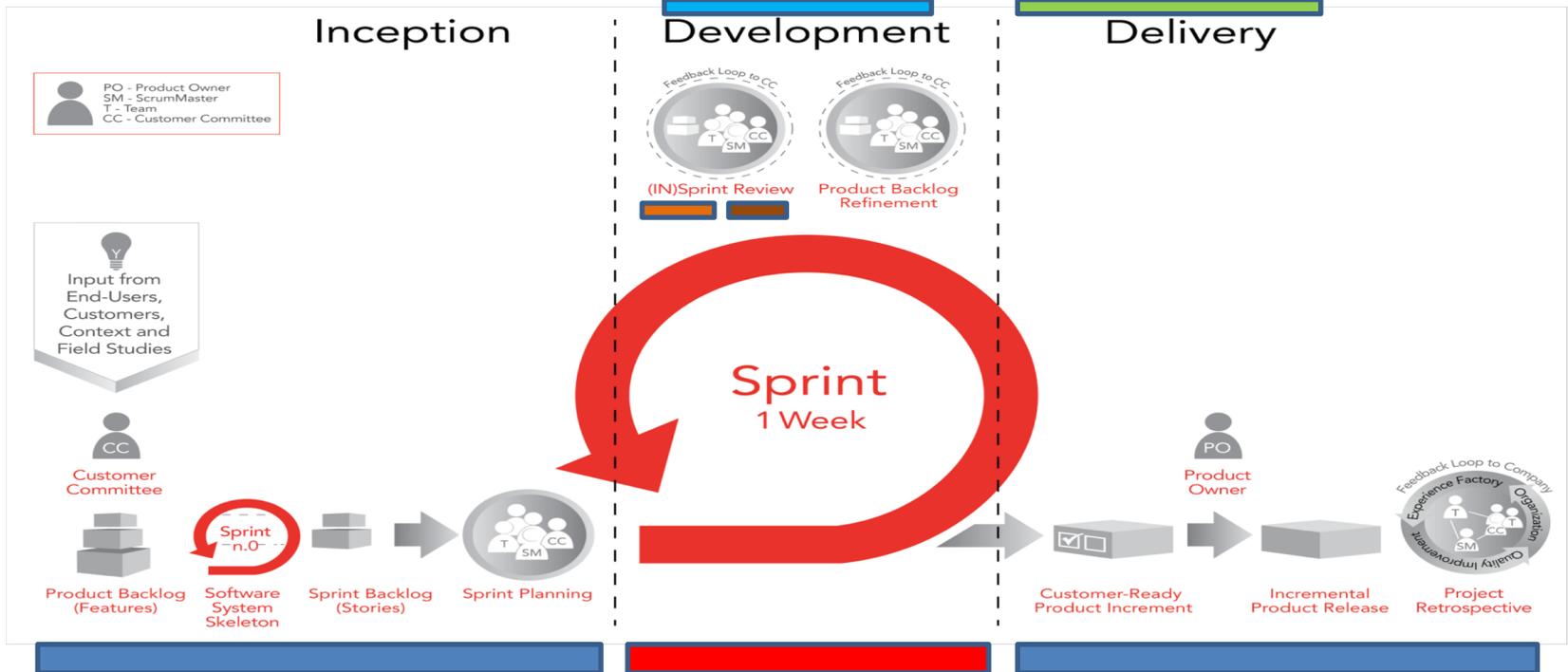


# Approccio proposto – Infrastruttura di supporto



E' stato individuato un tool per ciascuna delle fasi chiave del processo. Il tutto è stato realizzato in una architettura virtualizzata.

# Approccio proposto - Visione integrata

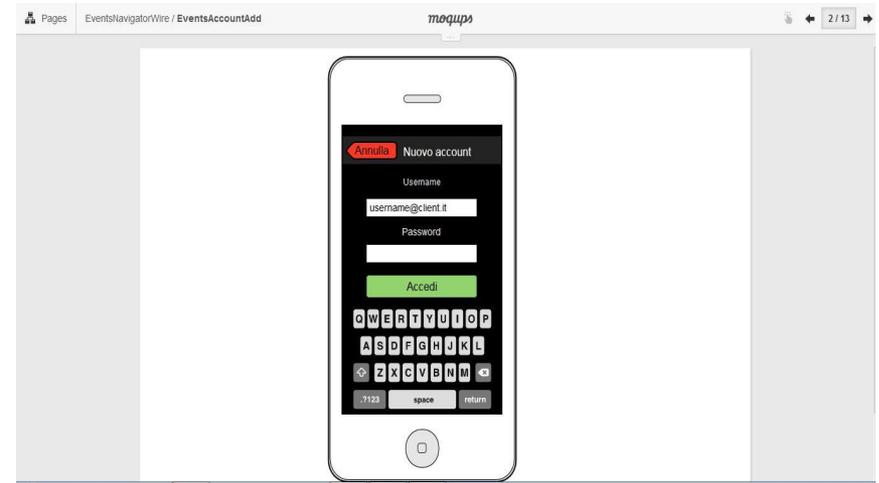


- Jenkins gestisce ed integra tutti gli strumenti
- Asana segue i backlog e permette la continua comunicazione tra le figure coinvolte nello sviluppo
- Git fornisce i codice sorgenti committati
- Gradle effettua la build
- Kiuwan e SonarQube effettuano le analisi dei codici

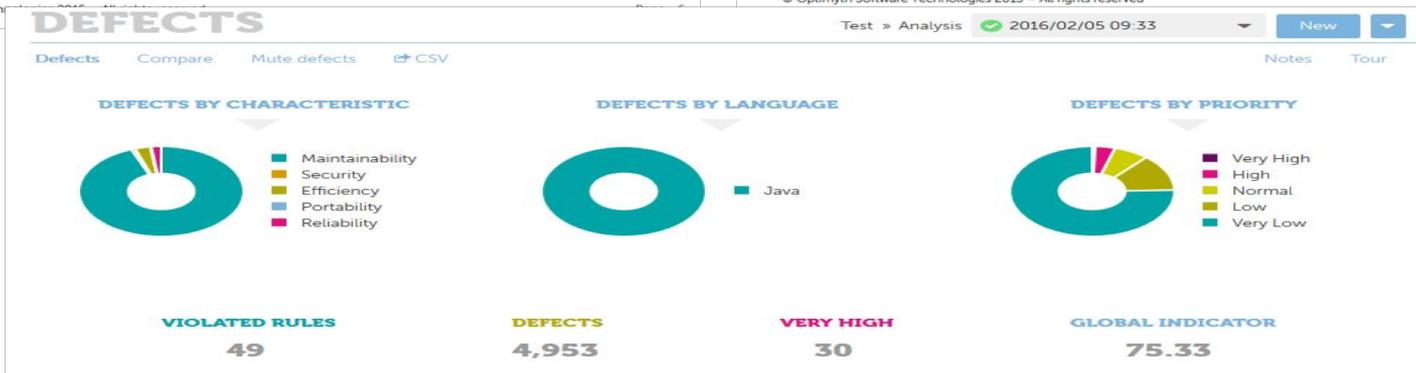
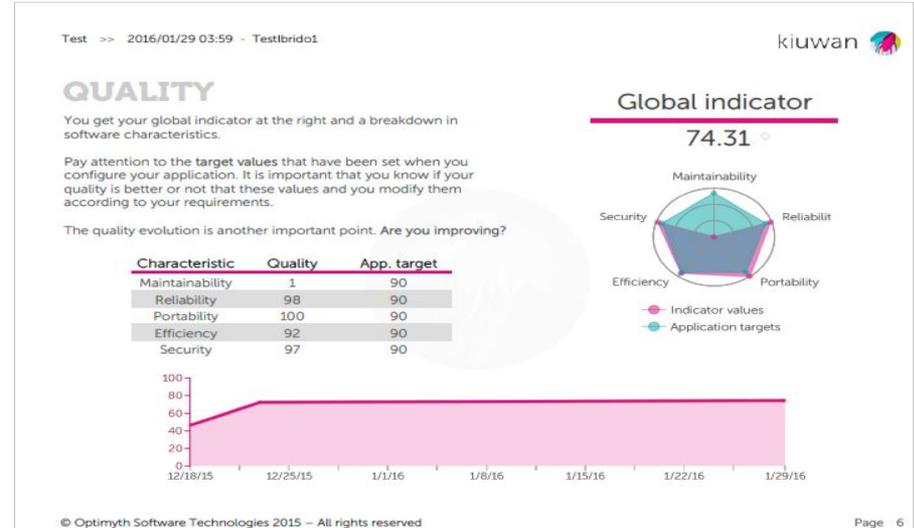
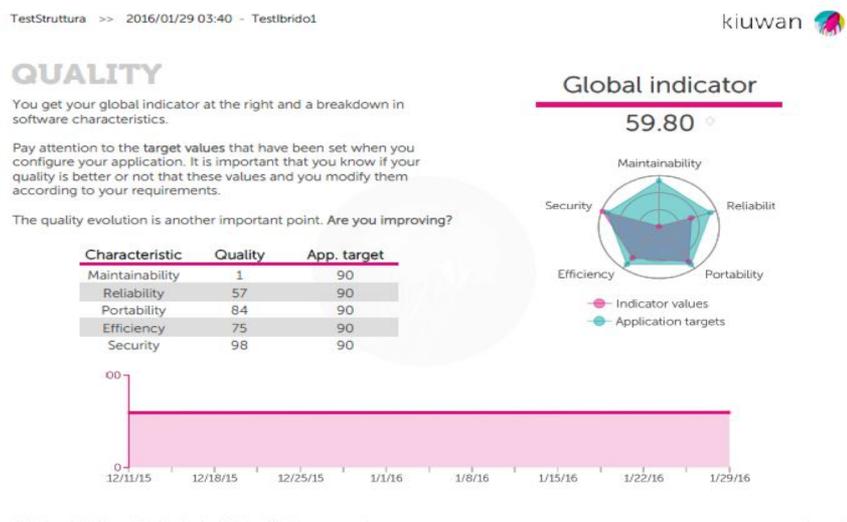


# Sperimentazione

- ⇒ Durante il tirocinio presso la multinazionale **Fincons Group** sono state sviluppate due app mobile :
  - Nativa Android
  - Ibrida
- ⇒ Durata del progetto :
  - Due mesi per l'app nativa Android da 8 sprint.
  - Un mese e mezzo da 6 sprint per l'app ibrida.
- ⇒ I team erano costituiti da tre persone
- ⇒ Requisiti principali :
  - Notificare ricezione email su client posta sincronizzato sul dispositivo con possibile evento da sincronizzare
  - In relazione al tipo (Webmail aziendale, Outlook) parserizzare l'evento.
  - Richiedere la sincronizzazione sul calendario del dispositivo
  - Sincronizzare evento sul calendario dispositivo



# Verifica qualità sprint - Kiuwan



In ogni fase di sprint review sono stati prodotti in maniera automatizzata i report sulla qualità interna dei codici prodotti.

# Verifica qualità - correzione errori

Vengono ispezionati gli errori presenti nella dashboard e corretti a seconda della gravità e alla aderenza ad una delle metriche .

Files	Defects	Rule	Priority	Characteristic	Language	Effort
Σ	223					694h
3	30	Duplicated code: big block	High	Maintainability	Java	120h
3	79	Duplicated code: medium block	High	Maintainability	Java	39h 30
5	42	Provide Javadoc comments for public classes and interfaces.	High	Maintainability	Java	21h 00
3	37	Inner classes should be 'protected' or 'private'.	High	Security	Java	296h
3	21	Maximum allowed number of fields.	High	Maintainability	Java	168h
5	5	Provide Javadoc comments for public fields.	High	Maintainability	Java	30m
3	5	Avoid classes / interfaces with too many lines of code.	High	Maintainability	Java	40h 00
1	2	Avoid using method calls in a loop.	High	Efficiency	Java	1h 00
1	1	Avoid using repeated cast over the same object or variable (Max 3 cast of every type).	High	Efficiency	Java	30m
1	1	Avoid excessive import lines	High	Maintainability	Java	8h 00

# Verifica qualità sprint - post modifiche



Successivamente alle modifiche effettuate vengono analizzati nuovamente i codici e visionati i report.

## Verifica qualità in uso

- ⇒ la verifica è stata effettuata attraverso una ispezione al termine dello sviluppo delle soluzioni che non ha riscontrato problemi evidenti data la natura del modello di processo che integra sin da subito e dai primi passi di sviluppo il committente/customer.
- ⇒ I prodotti sono in esercizio e verranno ulteriormente verificati dall'utente in successive fasi di test.



# Conclusioni

- Durante il lavoro di tesi è stato proposto un approccio incentrato sul customer che associati a strumenti ad hoc riesca a verificare i tre fattori di qualità della ISO 25000: intena , esterna , in uso.
- Si è costituita una infrastruttura virtuale che integra tutti gli strumenti necessari per ogni fase di sviluppo.
- L'azienda ne ha beneficiato prendendo in considerazione la possibilità di adottare lo strumento su infratruttura virtuale configurata per i successivi sviluppi.
- L'azienda terminata la fase di test potrebbe adottare il software sviluppato
- Si è riscontrata la possibilità di ridurre i tempi e costi di sviluppo soprattutto nelle prime fasi tramite prototipi ad alto livello.



# Grazie per l'attenzione

Dipartimento di Informatica - Università degli Studi di Bari  
Via Orabona, 4 - 70125 - Bari  
Tel: +39.080.5443270 | Fax: +39.080.5442536  
[serlab.di.uniba.it](http://serlab.di.uniba.it)