



Piano di Qualifica

Gruppo SWEight - Progetto Colletta

SWEightGroup@gmail.com

Informazioni sul documento

Versione	1.0.0
Owner	Da definire
Redattori	Sebastiano Caccaro Da definire
Verificatori	Da definire
Uso	Esterno
Distribuzione	MIVOQ Prof. Vardanega Tullio Prof. Cardin Riccardo Gruppo SWEight

Descrizione

Questo documento si occupa di definire i modi tramite quali il gruppo *SWEight* intende garantire la qualità del progetto.

Registro delle modifiche

Versione	Data	Descrizione	Nominativo	Ruolo
0.0.3	2018-x-x	Correzione del documento	Damien Ciagola	<i>Analista</i>
0.0.2	2018-x-x	x	x	<i>Verificatore</i>
0.0.1	2018-x-x	Creazione scheletro del documento e sezione introduzione	Damien Ciagola	<i>Analista</i>

Indice

1	Introduzione	3
1.1	Scopo del documento	3
1.2	Scopo del prodotto	3
1.3	Glossario	3
1.4	Riferimenti	3
1.4.1	Riferimenti normativi	3
1.4.2	Riferimenti informativi	3
2	Strategie di verifica	5
2.1	Obiettivi di qualità	5
2.1.1	Qualità di processo	5
2.1.2	Qualità di prodotto	5
2.2	Organizzazione????	5
2.3	Pianificazione strategica temporale?????	5
2.4	Risorse	5
2.5	Responsabilità	5
2.6	Misure e metriche	5
2.6.1	Metriche processi	5
2.6.2	Metriche documenti	5
2.6.3	Metriche software	5
3	Gestione amministrativa della revisione	5
	Appendici	5
A	Qualità	6
A.1	SPICE	6
A.2	ISO/IEC 9126	7
B	Appendice - Esito Verifica	8

1 Introduzione

1.1 Scopo del documento

In questo documento è illustrata la strategia di verifica e validazione del gruppo *SWEight*. Tale strategia è fondamentale per dare una misurazione oggettiva e quantificabile del livello di qualità di quanto viene prodotto.

Ciò è vantaggioso sia per il gruppo *SWEight*, che può più facilmente individuare difetti durante lo svolgimento del progetto, sia per il committente_G, che può costantemente monitorare la qualità del prodotto in base a criteri oggettivi e prestabiliti.

1.2 Scopo del prodotto

Lo scopo del progetto è realizzare una piattaforma collaborativa di raccolta dati in cui gli utenti possano predisporre e/o svolgere piccoli esercizi di grammatica (per esempio esercizi di analisi grammaticale) e i dati raccolti siano relativi sia agli esercizi predisposti che al loro svolgimento da parte degli utenti. I dati raccolti devono essere utilizzabili da sviluppatori e ricercatori al fine di insegnare ad un elaboratore a svolgere i medesimi esercizi mediante tecniche di apprendimento automatico supervisionato.

1.3 Glossario

Nel documento è possibile incontrare termini tecnici il quale significato potrebbe non essere immediatamente chiaro al lettore. Per disambiguarne il significato, essi sono stati marcati con una _G a pedice e la loro definizione è reperibile nel glossario fornito separatamente.

1.4 Riferimenti

1.4.1 Riferimenti normativi

- **Norme di Progetto:** *NormeDiProgetto_v1.0.0*;
- **Capitolato d'appalto C2:** Colletta
<https://www.math.unipd.it/~tullio/IS-1/2018/Progetto/C2.pdf>.

1.4.2 Riferimenti informativi

- **Piano di Progetto:** *PianoDiProgetto_v1.0.0*;
- **Slide del corso di Ingegneria del Software:**
<https://www.math.unipd.it/~tullio/IS-1/2018>;
- **Ian Sommerville, Software Engineering, Nona edizione:**
 - Capitolo 24: Quality management;
 - Capitolo 26: Process improvement;
- **Standard ISO/IEC 9126:**
https://it.wikipedia.org/wiki/ISO/IEC_9126
https://en.wikipedia.org/wiki/ISO/IEC_9126;
- **Standard ISO/IEC 15504:**
https://en.wikipedia.org/wiki/ISO/IEC_15504
- **VARI;**

- **ED;**
- **EVENTUALI;**
- ;;
- ;;
- ;;
- ;;

2 Strategie di verifica

La strategia generale del gruppo *SWEight* mira ad automatizzare il più possibile il lavoro di verifica, facendo gestire tale processo a degli strumenti informatici opportunamente predisposti.

I vantaggi derivanti da tale scelta sono i seguenti:

- **Costo:** L'uso di tool automatizzati non impiega risorse umane, il che si traduce in un risparmio in ore di lavoro e risorse economiche. È quindi possibile eseguire più spesso attività di verifica.
- **Misurabilità:** Se eseguita da strumenti informatici, l'attività di verifica produce dei valori oggettivi e tracciabili nel tempo.
- **Precisione:** La tediosità e la ripetitività della verifica manuale rendono quest'ultima frustrante e prona a contenere imprecisioni. Un computer, invece, esegue tale compito velocemente e senza errori.

2.1 Obiettivi di qualità

2.1.1 Qualità di processo

Al fine di garantire un prodotto di qualità, è fondamentale avere un elevato standard qualitativo anche per i processi. Per garantire tutto ciò, è stato deciso di aderire allo standard SPICE_G¹ (ISO/IEC 15504): quest'ultimo permette di valutare il livello di maturità dei processi, al fine di apportare modifiche migliorative.

2.1.2 Qualità di prodotto

Per garantire la qualità dei prodotti, viene adottato lo standard ISO/IEC 9126². Quest'ultimo permette di monitorare la qualità del software, fornendo delle metriche per misurarla.

2.2 Organizzazione????

2.3 Pianificazione strategica temporale?????

2.4 Risorse

2.5 Responsabilità

2.6 Misure e metriche

2.6.1 Metriche processi

2.6.2 Metriche documenti

2.6.3 Metriche software

3 Gestione amministrativa della revisione

¹SPICE: Vedi appendice §A.1

²ISO/IEC 9126: Vedi appendice §A.2

A Qualità

A.1 SPICE

Il modello ISO/IEC 1554, meglio conosciuto come SPICE (Software Process Improvement and Capability Determination), è lo standard di riferimento per valutare in modo oggettivo la qualità dei processi nello sviluppo del software.

Sono definiti:

- Una serie di **attributi** per ogni processo, che ne vanno a determinare la sua capability (capacità):
 - Process performance;
 - Performance management;
 - Work product management;
 - Process definition;
 - Process deployment;
 - Process measurement;
 - Process control;
 - Process innovation;
 - Process optimization;

Ognuno di questi attributi riceve una valutazione nella seguente scala:

- **N:** Non raggiunto (0 - 15%);
 - **P:** Parzialmente raggiunto (>15% - 50%);
 - **L:** Largamente raggiunto (>50%- 85%);
 - **F:** Pienamente raggiunto (>85% - 100%);
- Sei **livelli di capacità** dei processi:
 0. Incompleto;
 1. Eseguito;
 2. Gestito;
 3. Stabilito;
 4. Predicibile;
 5. Ottimizzato;
 - Delle linee guida per effettuare delle **stime**, eseguite tramite:
 - **Processi di misurazione**, descritti nel *PianoDiProgetto_v1.0.0*;
 - **Modello di misurazione**, descritto in questo documento;
 - **Strumenti di misurazione**, descritti nelle *NormeDiProgetto_v1.0.0*;
 - Una serie di **competenze** che chi effettua misurazioni deve possedere. La mancanza di esperienza degli elementi del gruppo *SWEight*, fa sì che nessun membro possieda queste skill, rendendo così impossibile la piena adesione allo standard. Tuttavia, ogni componente è chiamato a studiare SPICE e a applicare al meglio le indicazioni descritte in questo documento e nelle *NormeDiProgetto_v1.0.0*, al fine di perseguire un livello di qualità accettabile.

A.2 ISO/IEC 9126

Lo standard ISO/IEC 9126 stabilisce una serie di linee guida mirate al miglioramento delle qualità del software sviluppato.

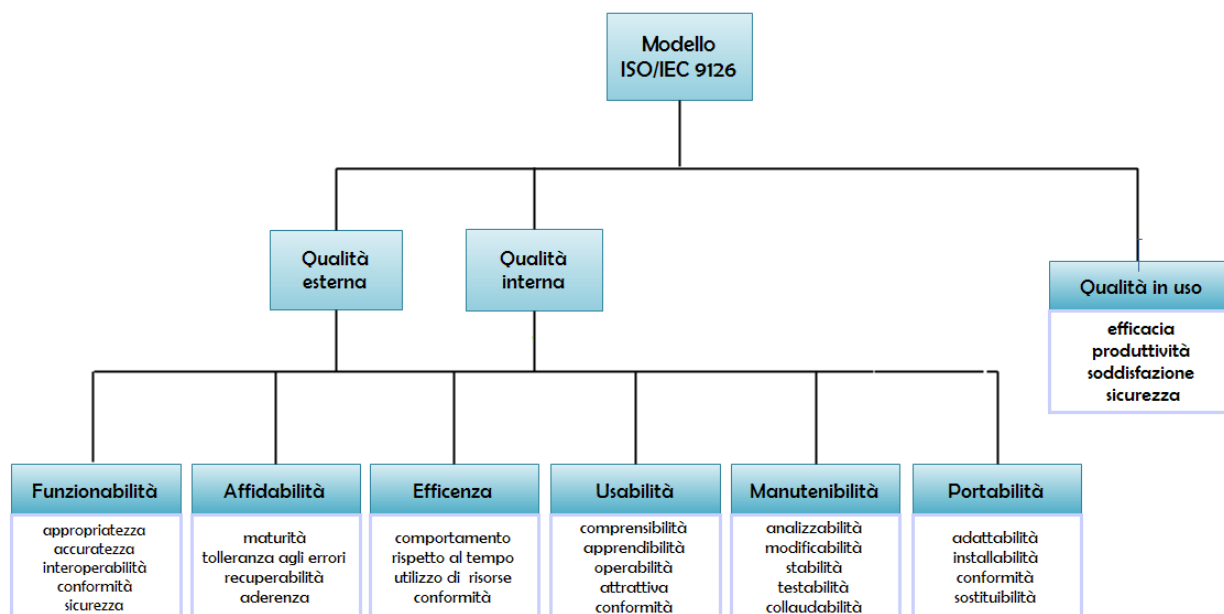


Figura 1: Rappresentazione grafica di ISO/IEC 9126 [Wikipedia]

Come presentato in Figura 1, ISO/IEC 9126 prescrive indicazioni su:

- **Qualità interna:** è misurata sul software non eseguibile, come, ad esempio il codice sorgente. Le misure effettuate (appendice §B) permettono di avere una buona previsione della qualità esterna. Le metriche usate dal gruppo *SWEight* sono reperibili in §2.6.3.
- **Qualità esterna:** è misurata tramite l'analisi dinamica su software eseguibile. Le misure effettuate (appendice §B) permettono di avere una buona previsione della qualità in uso prodotto. Le metriche usate dal gruppo *SWEight* sono le tecniche di analisi dinamica reperibili in §2.6.3.
- **Qualità in uso:** definita in base all'esperienza utente. Sono da perseguire i seguenti obiettivi:
 - Efficacia;
 - Produttività;
 - Soddisfazione;
 - Sicurezza.

ISO/IEC 9126 definisce inoltre una serie di requisiti da soddisfare per produrre software di qualità:

- **Funzionalità:** capacità di un prodotto software di soddisfare le esigenze stabilite (vedi *AnalisiDeiRequisiti_v1.0.0*);
- **Affidabilità:** capacità di un prodotto di mantenere un determinato livello di prestazioni in date condizioni d'uso per un certo periodo;
- **Efficienza:** capacità di un prodotto software di eseguire il proprio compito minimizzando il numero di risorse usate;

- **Usabilità:** capacità del prodotto software di essere utilizzato, capito e studiato dall'utente a cui è rivolto;
- **Manutenibilità:** capacità del prodotto software di evolvere mediante modifiche, correzioni e miglioramenti.
- **Portabilità:** capacità del prodotto software di funzionare ed essere installato su più ambienti hardware e software.

B Appendice - Esito Verifica