



MashUp

Software Engineering Group

info@mashup-unipd.it

Informazioni Documento

Nome documento	<i>Piano di Qualifica</i>
Versione	<i>v6.0.0</i>
Data redazione	2014-12-23
Redattori	Tesser Paolo Cusinato Giacomo
Verificatori	Carnovalini Filippo
Approvazione	Faccin Nicola
Lista distribuzione	<i>MashUp</i> <i>Prof. Tullio Vardanega</i> <i>Prof. Riccardo Cardin</i> <i>Dott. David Santucci - Zing Srl</i>
Uso	Esterno

Sommario

Documento che descrive le attività di verifica e validazione adottate dal gruppo
MashUp per il progetto BDSMApp.

Diario Revisioni

Versione	Data	Modifica	Autore & Ruolo
v6.0.0	2015-06-12	Approvazione documento	Faccin Nicola <i>Responsabile di Progetto</i>
v5.1.0	2015-06-10	Verifica documento	Carnovalini Filippo <i>Verificatore</i>
v5.0.6	2015-05-29	Stesura fase Validazione in appendice D.8	Cusinato Giacomo <i>Verificatore</i>
v5.0.5	2015-06-15	Incremento sezione Test in appendice B	Tesser Paolo <i>Verificatore</i>
v5.0.4	2015-06-11	Incremento appendice Ciclo PDCA in appendice A.3	Tesser Paolo <i>Verificatore</i>
v5.0.3	2015-06-10	Stesura sezione Esito delle Revisioni - Revisione di Qualifica in appendice E	Cusinato Giacomo <i>Verificatore</i>
v5.0.2	2015-06-08	Sistemazione history ordinata sulla base della versione	Tesser Paolo <i>Verificatore</i>
v5.0.1	2015-06-05	Incremento sezione Test in appendice B	Tesser Paolo <i>Verificatore</i>
v5.0.0	2015-05-29	Approvazione documento	Roetta Marco <i>Responsabile di progetto</i>
v4.1.0	2015-05-29	Verifica documento	Ceccon Lorenzo <i>Verificatore</i>
v4.0.2	2015-05-29	Stesura Consolidamento della progettazione e codifica dei requisiti desiderabili in appendice D.7	Cusinato Giacomo <i>Verificatore</i>
v4.0.1	2015-05-25	Incremento sezione Test in appendice B	Tesser Paolo <i>Verificatore</i>
v4.0.0	2015-05-21	Approvazione documento	Cusinato Giacomo <i>Responsabile di progetto</i>
v3.1.0	2015-05-21	Verifica documento	Ceccon Lorenzo <i>Verificatore</i>
v3.0.2	2015-05-20	Stesura Progettazione di dettaglio e codifica dei requisiti desiderabili in appendice D.6	Faccin Nicola <i>Verificatore</i>
v3.0.1	2015-05-18	Incremento sezione Test di unità in appendice B.5	Tesser Paolo <i>Verificatore</i>
v3.0.0	2015-05-13	Approvazione documento	Cusinato Giacomo <i>Responsabile di progetto</i>
v2.1.0	2015-05-13	Verifica documento	Tesser Paolo <i>Verificatore</i>
v2.0.6	2015-05-12	Stesura sezione Esito delle Revisioni - Revisione di Progettazione in appendice E	Carnovalini Filippo <i>Verificatore</i>
v2.0.5	2015-05-12	Incremento appendice Ciclo PDCA in appendice A.3	Santacatterina Luca <i>Verificatore</i>
v2.0.4	2015-05-11	Incremento sezione Gestione amministrativa della revisione in appendice 3	Carnovalini Filippo <i>Verificatore</i>

v2.0.3	2015-05-06	<i>Stesura Progettazione di dettaglio e codifica dei requisiti obbligatori in appendice D.5</i>	Santacatterina Luca Verificatore
v2.0.2	2015-05-04	<i>Terminata stesura Test di unità in appendice B.5</i>	Carnovalini Filippo Verificatore
v2.0.1	2015-04-22	<i>Iniziata stesura Test di unità in appendice B.5</i>	Santacatterina Luca Verificatore
v2.0.0	2015-03-30	<i>Approvazione documento</i>	Santacatterina Luca Responsabile di progetto
v1.2.0	2015-03-30	<i>Verifica documento</i>	Cusinato Giacomo Verificatore
v1.1.1	2015-03-26	<i>Correzione documento</i>	Roetta Marco Verificatore
v1.1.0	2015-03-25	<i>Verifica documento</i>	Cusinato Giacomo Verificatore
v1.0.11	2015-03-23	<i>Incremento sezione Dettaglio verifiche tramite analisi in appendice D</i>	Roetta Marco Verificatore
v1.0.10	2015-03-19	<i>Incremento sezione Resoconto attività di verifica in appendice C</i>	Ceccon Lorenzo Verificatore
v1.0.9	2015-03-18	<i>Terminata verifica Pianificazione dei test in appendice B</i>	Cusinato Giacomo Verificatore
v1.0.8	2015-03-11	<i>Iniziata verifica Pianificazione dei test in appendice B</i>	Cusinato Giacomo Verificatore
v1.0.7	2015-03-10	<i>Terminata stesura Pianificazione dei test in appendice B</i>	Roetta Marco Verificatore
v1.0.6	2015-02-26	<i>Iniziata stesura Pianificazione dei test in appendice B</i>	Ceccon Lorenzo Verificatore
v1.0.5	2015-02-23	<i>Incremento sezione Misure e metriche in sezione 2.8</i>	Roetta Marco Verificatore
v1.0.4	2015-02-23	<i>Rimozione sezione Tecniche</i>	Ceccon Lorenzo Verificatore
v1.0.3	2015-02-20	<i>Modifica sezione Procedure di controllo di qualità di processo 2.2</i>	Ceccon Lorenzo Verificatore
v1.0.2	2015-02-20	<i>Aggiunta sezione Definizione degli obiettivi di qualità 2.1</i>	Ceccon Lorenzo Verificatore
v1.0.1	2015-02-19	<i>Apportate modifiche alla struttura del documento a seguito della Revisione dei Requisiti</i>	Roetta Marco Verificatore
v1.0.0	2015-01-20	<i>Approvazione documento</i>	Tesser Paolo Responsabile di progetto
v0.2.0	2015-01-19	<i>Verifica documento</i>	Santacatterina Luca Verificatore
v0.1.1	2015-01-18	<i>Correzione documento</i>	Faccin Nicola Verificatore
v0.1.0	2015-01-16	<i>Verifica documento</i>	Santacatterina Luca Verificatore
v0.0.8	2015-01-13	<i>Iniziata stesura capitolo Resoconto delle attività di verifica C</i>	Ceccon Lorenzo Verificatore

v0.0.7	2015-01-12	<i>Terminata stesura capitolo Visione generale della strategia di verifica 2</i>	Faccin Nicola Verificatore
v0.0.6	2014-12-31	<i>Iniziata stesura capitolo Visione generale della strategia di verifica 2</i>	Ceccon Lorenzo Verificatore
v0.0.5	2014-12-30	<i>Terminata stesura appendici</i>	Faccin Nicola Verificatore
v0.0.4	2014-12-23	<i>Iniziata stesura appendici</i>	Ceccon Lorenzo Verificatore
v0.0.3	2014-12-23	<i>Stesura capitolo Gestione amministrativa della revisione 3</i>	Faccin Nicola Verificatore
v0.0.2	2014-12-15	<i>Stesura capitolo Introduzione 1</i>	Ceccon Lorenzo Verificatore
v0.0.1	2014-12-14	<i>Generazione struttura del documento</i>	Ceccon Lorenzo Verificatore

Indice

1	Introduzione	1
1.1	Scopo del documento	1
1.2	Scopo del prodotto	1
1.3	Glossario	1
1.4	Riferimenti	1
1.4.1	Normativi	1
1.4.2	Informativi	1
2	Visione generale della strategia di verifica	2
2.1	Definizione degli obiettivi di qualità	2
2.1.1	Qualità di processo	2
2.1.2	Qualità di prodotto	2
2.2	Procedure di controllo di qualità di processo	2
2.3	Procedure di controllo di qualità di prodotto	2
2.4	Organizzazione	3
2.5	Pianificazione strategica e temporale	3
2.6	Responsabilità	3
2.7	Risorse necessarie	4
2.8	Misure e metriche	4
2.8.1	Metriche per i processi	4
2.8.2	Metriche per i documenti	5
2.8.3	Metriche per il software	5
3	Gestione amministrativa della revisione	8
3.1	Gestione delle anomalie e delle discrepanze	8
A	Modelli e standard di qualità	9
A.1	Standard ISO/IEC 9126	9
A.2	Standard ISO/IEC 15504	11
A.3	Ciclo PDCA	13
B	Specifica dei test	14
B.1	Informazioni generali dei test	14
B.2	Test di validazione	14
B.2.1	TVF1	14
B.2.2	TVF1.1	14
B.2.3	TVF1.2	15
B.2.4	TVF1.3	15
B.2.5	TVF1.4	15
B.2.6	TVF1.8	15
B.2.7	TVF2	16
B.2.8	TVF2.4	16
B.2.9	TVF3	16
B.2.10	TVF3.1	17
B.2.11	TVF3.2	17
B.2.12	TVF3.3	17
B.2.13	TVF3.3.2	18
B.2.14	TVF3.3.3	18
B.2.15	TVF3.3.4	19

B.2.16 TVF3.3.7	19
B.2.17 TVF4	19
B.2.18 TVF4.1	20
B.2.19 TVF5	20
B.2.20 TVF5.1	20
B.2.21 TVF 5.1.3	21
B.2.22 TVF5.2	21
B.2.23 TVF5.3	21
B.2.24 TVF5.3.1	21
B.2.25 TVF5.4	22
B.2.26 TVF5.4.1	22
B.2.27 TVF5.5	23
B.2.28 TVF5.5.1	23
B.2.29 TVF5.6	24
B.2.30 TVF5.6.4	24
B.2.31 TVF6	25
B.2.32 TVF6.1	25
B.2.33 TVF6.2	25
B.2.34 TVF6.3	26
B.2.35 TVF7	26
B.2.36 TVF7.1	26
B.2.37 TVF7.1.1	27
B.2.38 TVF8	27
B.2.39 TVF8.1	27
B.2.40 TVF8.1.1	28
B.2.41 TVF8.2	28
B.2.42 TVF8.3	28
B.2.43 TVF8.4	29
B.2.44 TVF9	29
B.2.45 TVF9.1	29
B.2.46 TVF9.2	30
B.2.47 TVF9.3	30
B.2.48 TVF10	30
B.2.49 TVF10.1	31
B.2.50 TVF10.3	31
B.2.51 TVF11	31
B.2.52 TVF11.1	32
B.2.53 TVF11.2	32
B.2.54 TVF11.3	32
B.2.55 TVF12	32
B.2.56 TVQ1	32
B.2.57 TVQ3	33
B.2.58 TVV1	33
B.2.59 TVV1.1	33
B.2.60 TVV1.2	33
B.2.61 TVV2	33
B.2.62 TVV3	34
B.2.63 TVV4	34
B.2.64 TVV5	34
B.2.65 TVV6	34

B.2.66	TVV6.1	35
B.2.67	TVV6.2	35
B.2.68	TVV6.3	35
B.2.69	TVV7	35
B.2.70	TVV7.1	35
B.2.71	TVV7.2	36
B.2.72	TVV8	36
B.2.73	TVV8.1	36
B.2.74	TVV9	36
B.3	Test di sistema	36
B.3.1	Descrizione dei test di sistema	37
B.4	Test di integrazione	39
B.4.1	Descrizione dei test di integrazione	39
B.4.2	Tracciamento componenti - test di integrazione	43
B.5	Test di unità	45
B.5.1	Descrizione dei test di unità	45
B.5.2	Tracciamento test di unità - metodi	58
C	Resoconto delle attività di verifica	66
C.1	Revisione dei Requisiti	66
C.2	Revisione di Progettazione	66
D	Dettaglio delle verifiche tramite analisi	67
D.1	Ricerca ed implementazione degli strumenti	67
D.1.1	Processi	67
D.1.2	Documenti	67
D.2	Analisi dei requisiti	68
D.2.1	Processi	68
D.2.2	Documenti	68
D.3	Analisi di dettaglio	69
D.3.1	Processi	69
D.3.2	Documenti	69
D.4	Progettazione architetturale	70
D.4.1	Processi	70
D.4.2	Documenti	71
D.5	Progettazione di dettaglio e codifica dei requisiti obbligatori	71
D.5.1	Processi	71
D.5.2	Documenti	71
D.5.3	Progettazione	72
D.5.4	Metriche codice	74
D.6	Progettazione di dettaglio e codifica dei requisiti desiderabili	75
D.6.1	Processi	75
D.6.2	Documenti	76
D.7	Consolidamento della progettazione e codifica dei requisiti desiderabili	76
D.7.1	Processi	76
D.7.2	Documenti	77
D.8	Validazione	78
D.8.1	Processi	78
D.8.2	Documenti	78
D.8.3	Progettazione	78
D.8.4	Metriche codice	80

E	Esito delle revisioni	83
E.1	Revisione dei Requisiti	83
E.2	Revisione di Progettazione	83
E.3	Revisione di Qualifica	83

Elenco delle tabelle

1	Tracciamento test di sistema - requisiti	38
2	Tabella test di integrazione	43
3	Tracciamento componenti - test di integrazione	45
4	Tabella test di unità	58
5	Tabella tracciamento test di unità - metodi	65
6	Risultati indice Gulpease - Analisi dei requisiti	68
7	Risultati indice Gulpease - Analisi di dettaglio	70
8	Risultati indice Gulpease - Progettazione architetturale	71
9	Risultati indice Gulpease - Progettazione di dettaglio e codifica dei requisiti obbligatori	71
10	Indice di instabilità	73
11	Risultati indice Gulpease - Progettazione di dettaglio e codifica dei requisiti desiderabili	76
12	Risultati indice Gulpease - Consolidamento della progettazione e co- difica dei requisiti desiderabili (TODO cambiare versione documenti)	77
13	Risultati indice Gulpease - Validazione	79
14	Indice di instabilità	80

Elenco delle figure

1	Schema del modello di qualità ISO/IEC 9126	9
2	Ciclo di miglioramento della qualità PDCA	13
3	Grafico task Ricerca ed implementazione degli strumenti	67
4	Grafico task Analisi dei requisiti	68
5	Grafico task Analisi di dettaglio	69
6	Grafico task Progettazione architetturale	70
7	Grafico task Progettazione di dettaglio e codifica dei requisiti obbligatori	72
8	Grafico task Progettazione di dettaglio e codifica dei requisiti desiderabili	76
9	Grafico task Consolidamento della progettazione e codifica dei requisiti desiderabili	77
10	Grafico task Validazione	78

1 Introduzione

1.1 Scopo del documento

Questo documento ha lo scopo di definire la strategia e descrivere le modalità di verifica e validazione_G che il gruppo MashUp intende adottare per lo sviluppo del progetto, al fine di raggiungere gli obiettivi qualitativi prefissati. Per perseguire questi obiettivi è necessaria una costante attività di verifica in modo da permettere di rilevare e risolvere eventuali anomalie.

1.2 Scopo del prodotto

Lo scopo del prodotto è di creare una nuova infrastruttura che permetta di interrogare Big Data recuperati dai social network, quali: Facebook, Twitter, Instagram. L'applicazione sarà composta da due parti:

- consultazione e interrogazione con interfaccia web per utente;
- servizi web REST interrogabili.

1.3 Glossario

Al fine di evitare ogni ambiguità relativa al linguaggio usato nei documenti viene allegato il *Glossario v4.0.0*. Esso ha lo scopo di definire ed analizzare tutti i termini tecnici del progetto e di fugare eventuali ambiguità fornendo un'accurata descrizione. Tutte le occorrenze di tali termini nei documenti verranno contrassegnate con una "G" a pedice.

1.4 Riferimenti

1.4.1 Normativi

- **Norme di progetto:** *Norme di Progetto v6.0.0*;
- **Capitolato d'appalto C1:** *BDSMAApp: Big Data Social Monitoring App* <http://www.math.unipd.it/~tullio/IS-1/2014/Progetto/C1.pdf>;
- **Standard ISO/IEC 9126:** http://en.wikipedia.org/wiki/ISO/IEC_9126;
- **Standard ISO/IEC 15504:** http://en.wikipedia.org/wiki/ISO/IEC_15504.

1.4.2 Informativi

- **Piano di progetto:** *Piano di Progetto v7.0.0*;
- **Qualità di prodotto:** <http://www.math.unipd.it/~tullio/IS-1/2014/Dispense/P07a.pdf>;
- **Qualità di processo:** <http://www.math.unipd.it/~tullio/IS-1/2014/Dispense/P07b.pdf>;
- **SWEBOK 2004:** *Chapter 11 - Software Quality* <http://www.computer.org/portal/web/swebok/html/ch11>;
- **Software Engineering - Ian Sommerville - 9th Edition(2010):** *Chapters 8, 24, 26.*

2 Visione generale della strategia di verifica

2.1 Definizione degli obiettivi di qualità

2.1.1 Qualità di processo

Per garantire la qualità di prodotto è necessario, prima di tutto, ottenere la qualità dei processi che lo compongono. Per ottenere la qualità di processo si è quindi deciso di fare riferimento allo standard ISO_G/IEC_G 15504, noto anche come SPICE_G che fornisce degli strumenti per valutare i processi, assegnando a ciascuno di essi un grado di maturità definito dal modello CMM che lo standard incorpora.

Per ottenere un miglioramento di questi processi, si farà affidamento al ciclo di Deming, noto anche come PDCA_G, che definisce un metodo di gestione iterativo per il controllo e il miglioramento continuo dei processi.

2.1.2 Qualità di prodotto

Per ottenere la qualità di prodotto è necessario fissare degli obiettivi qualitativi al fine di massimizzare l'efficacia. A tale scopo, lo standard ISO_G/IEC_G 9126 definisce un modello di qualità software e le metriche per il raggiungimento di tali obiettivi.

2.2 Procedure di controllo di qualità di processo

Per garantire la qualità di processo sarà applicato il principio del PDCA_G ad ognuno dei processi. In questo modo si potrà ottenere un miglioramento continuo dei processi e quindi, di conseguenza, il miglioramento del prodotto.

Per ottenere la qualità è necessario attuare le seguenti procedure di controllo:

- la pianificazione deve essere dettagliata;
- le risorse necessarie devono essere definite in modo chiaro;
- le attività pianificate devono essere monitorate.

Una descrizione più dettagliata di tali attività viene riportata nel *Piano di Progetto v7.0.0*.

Mediante l'analisi costante della qualità di prodotto si ottengono informazioni anche sulla qualità dei processi. Un prodotto di bassa qualità indica la presenza di qualche processo migliorabile.

2.3 Procedure di controllo di qualità di prodotto

Per garantire la qualità del prodotto software si fa affidamento a due modalità di controllo:

- **Software Quality Assurance (SQA):** è un insieme di attività che assicurano che il software sviluppato sia conforme alle specifiche di qualità standard o definite dal gruppo. L'SQA è un processo appartenente al ciclo di vita del software che controlla regolarmente e preventivamente il software sviluppato per assicurare il rispetto degli obbiettivi di qualità prefissati;
- **Verifica e Validazione_G (V&V):** sono due processi che controllano che il sistema software soddisfi le specifiche e che raggiunga appieno il suo scopo. Per verifica si intende il processo attuo a valutare che il software in una determinata fase di sviluppo soddisfi le condizioni imposte all'inizio di tale fase. Per

validazione_G si intende il processo attuo a valutare se al termine del processo di sviluppo questo soddisfa i requisiti specificati. In altre parole, la verifica garantisce che il software è stato creato correttamente, mentre la validazione assicura che si è creato il giusto prodotto.

2.4 Organizzazione

L'attività di verifica accompagnerà l'intero ciclo di vita del software e sarà effettuata su tutti i processi realizzati e sugli output prodotti da questi ultimi. La verifica sarà applicata solamente ai cambiamenti effettuati dall'ultima versione approvata del prodotto.

Il team ha scelto di adottare un modello di ciclo di vita di tipo incrementale suddiviso in diverse fasi e riportate dettagliatamente nel *Piano di Progetto v7.0.0*. Per ciascuna di queste fasi verranno utilizzate specifiche attività di verifica.

Il processo di verifica sarà così composto:

- **Ricerca e implementazione degli strumenti:** in questa fase l'attività di verifica consente di verificare che tutti gli script creati siano corretti e che l'output prodotto dagli stessi sia uguale alle aspettative;
- **Analisi dei requisiti e di dettaglio:** in questa fase si verificherà che ogni requisito abbia corrispondenza in un caso d'uso e si effettueranno dei test sulla documentazione prodotta per verificare che rispetti le *Norme di Progetto v6.0.0*;
- **Progettazione architetturale:** l'attività di verifica in questa fase consiste nell'analizzare che la soluzione generale ad alto livello soddisfa i requisiti richiesti oltre a verificare i processi utilizzati per ottenere questa soluzione;
- **Progettazione di dettaglio e codifica dei requisiti obbligatori, desiderabili e opzionali:** si eseguiranno attività di verifica sui processi di progettazione e codifica del codice prodotto dai programmatori facendo uso di tecniche di analisi statica e dinamica;
- **Validazione_G:** in quest'ultima fase verrà effettuato il collaudo del prodotto che garantirà il corretto funzionamento del prodotto realizzato.

2.5 Pianificazione strategica e temporale

L'attività di verifica necessaria, per il miglioramento della qualità dei processi e del prodotto, deve essere sistematica ed organizzata. Ciò permetterà l'individuazione e la correzione degli errori il prima possibile evitando la propagazione di questi ultimi in larga scala.

Ciascuna attività che riguarda la documentazione o la codifica dovrà essere preceduta da uno studio preliminare che ci permetta di rendere chiaro la struttura degli stessi. Questo studio preventivo ci consentirà di ottenere un maggiore livello di qualità e una minore possibilità di fallimento.

Per quanto riguarda le tempistiche, l'obiettivo primario è quello di rispettare le scadenze forniteci dal committente_G e riportate nel *Piano di Progetto v7.0.0*.

2.6 Responsabilità

Le responsabilità relative all'assegnazione degli incarichi appartengono al *Responsabile di Progetto*, mentre le responsabilità relative all'adeguamento dell'ambiente di

lavoro per lo svolgimento di tutti i compiti necessari alla realizzazione del progetto appartengono all'*Amministratore di Progetto*.

2.7 Risorse necessarie

Le risorse necessarie alla verifica della qualità dei processi e del prodotto sono:

- **Risorse umane:** il *Responsabile di Progetto* controlla la qualità dei processi interni, l'*Amministratore di Progetto* definisce le norme e i piani per le attività di verifica, il *Programmatore* esegue le prove di verifica e validazione_G del codice, il *Verificatore* esegue la verifica dei documenti e fornisce i risultati delle prove effettuate. Per una descrizione più dettagliata dei ruoli consultare il *Piano di Progetto v7.0.0*;
- **Risorse software:** sono necessari strumenti per il tracciamento dei requisiti, per la stesura dei documenti in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, per la creazione di diagrammi UML_G , per lo sviluppo del prodotto e per il supporto e la verifica del codice;
- **Risorse hardware:** sono necessari computer per scrivere documenti e creare il prodotto software finale. È richiesto un ambiente di lavoro stabile in cui poter lavorare al progetto.

2.8 Misure e metriche

Il processo di verifica, per essere utile ed informativo, deve essere quantificabile. Vanno quindi stabilite a priori delle metriche. Sulla base di queste, saranno basate le misure rilevate dal processo di verifica. Nel caso in cui vi fossero metriche approssimate ed incerte, esse miglioreranno in modo incrementale. Questo è permesso grazie al ciclo di vita adottato descritto nel *Piano di Progetto v7.0.0*. Vi possono essere due tipologie di range:

- **range di accettazione:** termini entro i quali il prodotto è accettato
- **range ottimale:** termini entro i quali il prodotto dovrebbe collocarsi, questi non sono vincolanti, ma caldamente consigliati, il mancato rispetto di tali limiti deve seguire una verifica approfondita.

2.8.1 Metriche per i processi

- **Schedule Variance:** questo indice di efficacia mette in relazione il lavoro pianificato con quello realmente svolto. Se il confronto è maggiore di zero, allora si è svolto più lavoro rispetto a quello pianificato, viceversa se negativo. L'utilizzo di periodi di slack aumenta la possibilità che la schedule variance assumi valori positivi.
 - **range di accettazione:** $\geq -$ (preventivo fase *5%);
 - **range ottimale:** ≥ 0 .
- **Budget Variance:** l'avanzamento dei processi verrà valutato anche tramite la budget variance. Questo indice di efficienza mette in relazione il preventivo pianificato con il consuntivo. Se tale indice è maggiore di zero, allora si è riusciti ad essere più efficienti di quello pianificato, viceversa se negativo.
 - **range di accettazione:** $\geq -$ (preventivo fase*10%);
 - **range ottimale:** ≥ 0 .

2.8.2 Metriche per i documenti

Indice Gulepease_G: questo indice, tarato specificatamente per la lingua italiana, ha anche il vantaggio di utilizzare la lunghezza delle parole in lettere e non delle sillabe, semplificandone il calcolo.

$$89 + \frac{300 * (\text{Numero delle frasi}) - 10 * (\text{Numero delle lettere})}{\text{Numero delle parole}}$$

100 indica la leggibilità più alta mentre 0 quella più bassa, sono presenti dei range così da poter quantificare meglio la complessità del documento in analisi:

- inferiori a 80 sono difficili da leggere per chi ha la licenza elementare;
- inferiori a 60 sono difficili da leggere per chi ha la licenza media;
- inferiori a 40 difficili da leggere per chi possiede un diploma superiore;
- **range di accettazione:** [40 - 100];
- **range ottimale:** [50 - 100].

È stato considerato che questo indice tuttavia non indica se il testo sia comprensibile o meno, infatti contenuto delle frasi potrebbe essere totalmente non comprensibile e/o stravolto, ma avere lo stesso un ottimo indice Gulepease. Visto lo scopo dei documenti e la loro natura di documenti formali, capiterà spesso che vengano impiegati termini tecnici che non si possono sostituire, interrompere la frase a favore di un indice più alto potrebbe spezzarne il ragionamento o, storpiarne il contenuto e usare frasi troppo dirette potrebbe risultare poco professionale ai fini del documento. Perciò, i documenti saranno valutati precedentemente da un essere umano per stabilire se e come il testo potrebbe essere semplificato. I limiti imposti da tale indice saranno sufficientemente rilassati per accettare frasi un po' più articolate.

2.8.3 Metriche per il software

- **Complessità Ciclomatica:** è utilizzata per misurare la complessità di un metodo, attraverso il grafo di controllo di flusso che misura direttamente il numero di cammini linearmente indipendenti. I nodi di questo grafo rappresentano gruppi indivisibili di istruzioni e gli archi connettono due nodi solamente se le istruzioni di un nodo possono essere eseguite immediatamente dopo le istruzioni dell'altro nodo.

In questo progetto si cercherà di rispettare la raccomandazione di *McCabe*, che sviluppò tale teoria, ossia quella di non superare una complessità di 10. Rispettando questo vincolo si aumentano le possibilità di riuso del codice, manutenibilità, coesione e correttezza di quest'ultimo. Il vincolo presentato sarà di tipo lasco, ossia potrà essere portato a valori maggiori nell'eventualità porti a notevoli benefici in termini di velocità di esecuzione.

- **range di accettazione:** <15;
- **range ottimale:** <10.

- **Numero di metodi:** metrica_G utilizzata per calcolare una media delle occorrenze dei metodi per package_G; valori alti potrebbero indicare la necessità di scomporlo.

- **range di accettazione:** [3 - 10];

- **range ottimale:** [3 - 8].
- **Numero di parametri:** $metrica_G$ utilizzata per calcolare il numero di parametri formali di un metodo. Un valore basso è indice di maggior manutenibilità e astrazione del codice.
 - **range di accettazione:** [0 - 8];
 - **range ottimale:** [0 - 4].
- **Linee di codice per linee di commento:** $metrica_G$ atta a migliorare la manutenibilità del codice attraverso il monitoraggio del rapporto tra questi valori.
 - **range di accettazione:** >0.20 ;
 - **range ottimale:** >0.35 .
- **Numero di bug_G per linee di codice:** $metrica_G$ per la misura dei bug trovati per un certo quantitativo di linee di codice. Questa metrica è utile in quanto all'aumentare dell'ampiezza del codice si aumenta la probabilità di nascondere degli errori. Presupponendo che nessuno del gruppo avrà conoscenze sufficienti dello stack tecnologico che si andrà ad utilizzare si partirà con un valore di accettazione alto per poi cercare di ridurlo in modo incrementale. L'obiettivo fissato è quello di raggiungere valori compresi tra 0 e 20. Difficoltà particolari verranno gestite dal responsabile di progetto;
- **Numero di livelli di annidamento:** $metrica_G$ per misurare il livello di annidamento dei metodi. Un numero elevato comporta eccessiva complessità del codice e ne riduce il livello di astrazione.
 - **range di accettazione:** [1 - 6];
 - **range ottimale:** [1 - 4].
- **Grado di accoppiamento:** viene derivato da due singoli indici:
 - **Accoppiamento afferente:** questo indice viene calcolato dal numero di classi esterne al package_G che hanno dipendenze con classi interne. Se il numero è basso significa che il package fornirà poche funzionalità, se è molto alto avremo molte dipendenze da tracciare in caso di modifiche.
 - **Accoppiamento efferente:** questo indice calcolato dal numero di classi interne al package_G che hanno dipendenze con classi esterne ad esso. Un numero alto può significare una scarsa progettazione.
- **Grado di instabilità:** $metrica_G$ per la misura dell'instabilità delle componenti del sistema. Questa è molto legata all'indice di accoppiamento afferente e quello efferente, in quanto viene calcolato con questa formula:

$$I = \frac{C_e}{C_a + C_e}$$

dove C_e rappresenta l'accoppiamento efferente e C_a quello afferente.

- **range di accettazione:** [0 - 0.8];
- **range ottimale:** [0.3 - 0.7].

- **Logical SLOC:** misura le dimensioni del software basandosi sul numero di linee di codice sorgente. Ci sono due tipi di SLOC, quello *physical* e quello *logical* : si userà il secondo, che conta solamente gli statement, ignorando le righe vuote, quelle bianche e i commenti.

- **range di accettazione:** [1 - 80];
- **range ottimale:** [1 - 50]

- **Copertura del codice:** metrica_G che misura la copertura dei test. Maggiore sarà questo valore più significativi saranno i test eseguiti. Questa metrica può essere raffinata tralasciando nell'analisi i metodi semplici che non richiedono test. La copertura del codice può essere di due tipologie:

- **Statement coverage:**

$$\frac{\text{number of statements exercised}}{\text{total number of statements}} * 100$$

Questo rappresenta il valore in percentuale del numero di linee di codice esaminate dai test.

- * **range di accettazione:** [80% - 100%];
- * **range ottimale:** [90% - 100%].

- **Branch coverage:**

$$\frac{\text{number of decision outcomes exercised}}{\text{total number of decision outcomes}} * 100$$

Verifica che ciascun ramo della logica del flusso di controllo sia eseguito.

- * **range di accettazione:** [80% - 100%];
- * **range ottimale:** [90% - 100%].

- **Validazione_G W3C:** l'applicativo web deve superare correttamente i test di validazione_G offerti dal W3C. Qui si controlla il numero di avvisi o inesattezze per pagina che il test segnala.

- **range di accettazione:** [0 - 10];
- **range ottimale:** [0].

3 Gestione amministrativa della revisione

3.1 Gestione delle anomalie e delle discrepanze

La fase di verifica porta alla ricerca di eventuali difetti, i quali possono essere errori logici oppure anomalie presenti nel codice. Il *Verificatore* ha il compito di esaminare scrupolosamente il codice del prodotto e sottolineare le eventuali anomalie e problemi per essere risolti successivamente.

Per anomalia si intende una deviazione del prodotto dalla sue aspettative, quindi, causa un malfunzionamento del sistema. Si possono dividere in:

- **Computational Error:** differenza tra un valore calcolato, osservato o misurato e il suo valore teoricamente corretto;
- **Error:** azione umana che produce un risultato errato;
- **Defect:** imperfezione o carenza all'interno di un prodotto, il quale non soddisfa le sue esigenze o specifiche prefissate e necessità di essere sistemato;
- **Fault:** difetto all'interno del codice sorgente. Può essere inteso come la codifica di un errore umano nel codice sorgente;
- **Failure:** evento nel quale un sistema, o un suo componente, non esegue una funzione richiesta entro limiti specificati. Si verifica quando, sotto specifiche condizioni, viene rilevato un *Fault*.

Per ottenere il miglioramento continuo del prodotto, la gestione di queste anomalie assume un ruolo di primaria importanza all'interno del progetto e dovranno essere gestite in maniera più rapida possibile.

Per discrepanza invece, si intende una divergenza tra il prodotto sviluppato e quello atteso. La presenza di una discrepanza fa sì che il prodotto funzioni senza che si verifichino *Failure* ma, rende il prodotto realizzato errato rispetto alle attese. Una discrepanza sarà, quindi, trattata come un'anomalia di bassa priorità.

Dopo aver pianificato ed eseguito un task_G , il *Verificatore* dovrà verificare quanto prodotto, nel caso incontrasse un'anomalia, dovrà seguire una procedura standard sia per quanto riguarda le anomalie che riguardano i documenti, sia quelle relative al software. Il *Verificatore* dovrà quindi aprire un ticket_G seguendo le regole riportate nel *Norme di Progetto v6.0.0*.

Il *Responsabile di Progetto*, in seguito, dovrà approvare il ticket_G ; se approvato, il membro a cui è stato assegnato tale ticket dovrà pianificare una soluzione all'anomalia riscontrata per poi procedere a correggerla entro i termini stabiliti.

A Modelli e standard di qualità

A.1 Standard ISO/IEC 9126

Lo standard ISO_G/IEC_G 9126 è uno standard creato per delineare delle normative utili a descrivere un modello di qualità del software. Lo standard propone un approccio in cui viene posta attenzione al miglioramento dell'organizzazione e dei processi di una società di software, in modo da migliorare di conseguenza la qualità del prodotto software.

Lo standard ISO_G/IEC_G 9126 è suddiviso in quattro parti:

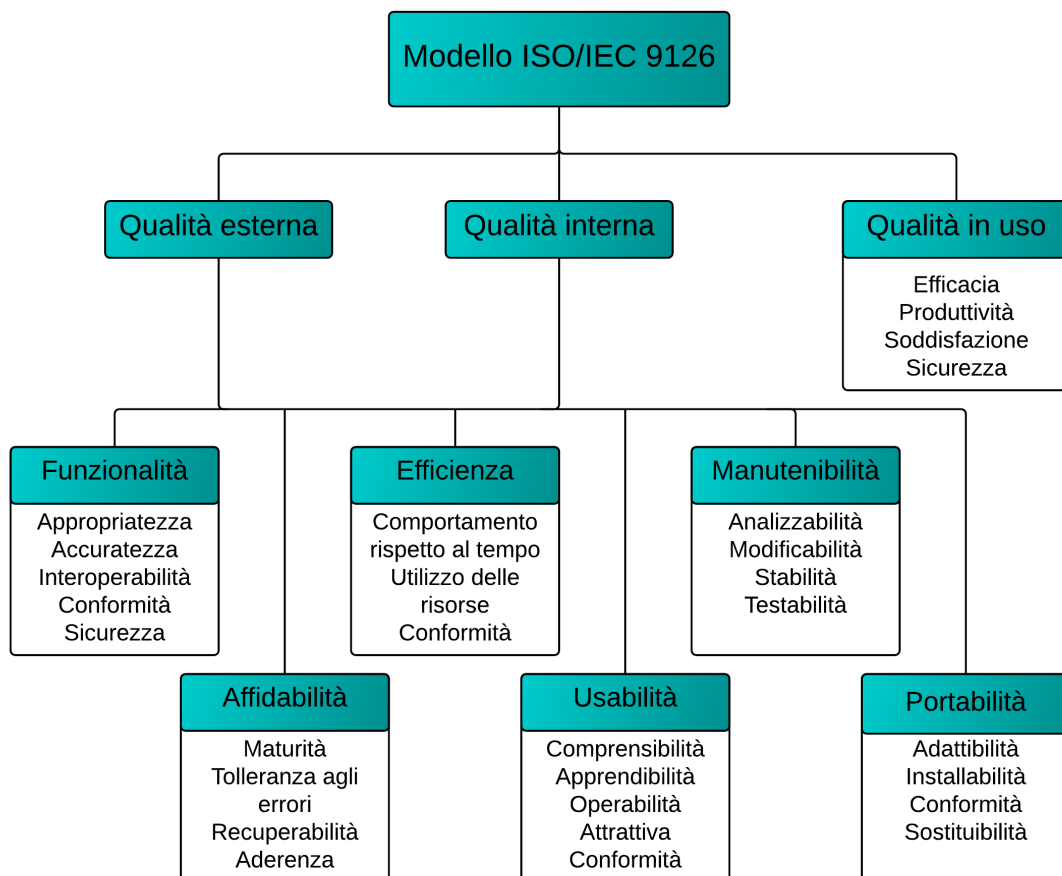


Figura 1: Schema del modello di qualità ISO/IEC 9126

- **Modello di qualità:** la prima parte dello standard classifica il modello di qualità in sei caratteristiche generali e in varie sotto caratteristiche misurabili tramite l'utilizzo di metriche.

Le sei caratteristiche generali e le relative sotto caratteristiche sono:

- **Funzionalità:** capacità di un prodotto software di fornire funzioni che soddisfano esigenze stabilite
 - * **Appropriatezza:** capacità del prodotto software di fornire un appropriato insieme di funzioni per i specifici compiti ed obiettivi prefissati all'utente;
 - * **Accuratezza:** capacità del prodotto software di fornire i risultati richiesti;

- * **Interoperabilità:** capacità del prodotto software di interagire con i diversi sistemi specificati;
- * **Conformità:** capacità del prodotto software di aderire agli standard e alle convenzioni appartenenti al settore in cui vengono applicati;
- * **Sicurezza:** capacità del prodotto software di consentire l'accesso a dati e informazioni solamente alle persone autorizzate.
- **Affidabilità:** capacità del prodotto software di mantenere uno specificato livello di prestazioni
 - * **Maturità:** capacità di un software di evitare che si verificano errori, malfunzionamenti o siano prodotti risultati non corretti;
 - * **Tolleranza agli errori:** capacità del software di mantenere un adeguato livello di prestazioni in presenza di malfunzionamenti;
 - * **Recuperabilità:** capacità di un prodotto di ripristinare il livello appropriato di prestazioni in seguito a un malfunzionamento;
 - * **Aderenza:** capacità di aderire a standard, regole e convenzioni inerenti all'affidabilità.
- **Usabilità:** capacità del software di essere capito, appreso e usato dall'utente
 - * **Comprensibilità:** esprime la facilità di comprensione delle funzionalità del prodotto;
 - * **Apprendibilità:** capacità del software di essere appreso in tempo brevi;
 - * **Operabilità:** capacità di permettere agli utenti di utilizzare al software al fine di raggiungere i propri scopi;
 - * **Attrattiva:** capacità del prodotto di risultare interessante all'utente;
 - * **Conformità:** capacità del software di aderire a standard, regole e convenzioni relativi all'usabilità.
- **Efficienza:** capacità di fornire prestazioni relativamente alla quantità di risorse usate
 - * **Comportamento rispetto al tempo:** capacità di fornire tempi di risposta, elaborazione e velocità di attraversamento ottimali in relazione alla funzione utilizzata;
 - * **Utilizzo delle risorse:** capacità del software di utilizzare adeguate quantità di risorse;
 - * **Conformità:** capacità del software di aderire a standard, regole e convenzioni relativi all'efficienza.
- **Manutenibilità:** capacità del software di essere modificato apportando correzioni, miglioramenti o adattamenti
 - * **Analizzabilità:** esprime la facilità nell'analizzare il codice sorgente per ricercare errori;
 - * **Modificabilità:** capacità del software di permettere l'implementazione di nuove modifiche;
 - * **Stabilità:** capacità del software di evitare effetti indesiderati a seguito di modifiche errate;
 - * **Testabilità:** capacità del software di eseguire facilmente la validazione delle modifiche apportate al software.

- **Portabilità:** capacità del software di lavorare in diversi ambienti di lavoro
 - * **Adattabilità:** capacità del software di essere adattato a diversi ambienti senza dover applicare modifiche diverse da quelle fornite;
 - * **Installabilità:** capacità del software di essere installato in uno specifico ambiente;
 - * **Conformità:** capacità del prodotto software di aderire a standard, regole e convenzioni relativi alla portabilità;
 - * **Sostituibilità:** capacità del software di sostituire un altro software analogo per svolgere certi compiti.
- **Qualità esterne:** le metriche esterne applicabili al software, e quindi rilevabili tramite l'analisi dinamica, misurano il comportamento del prodotto sulla base dei test, dall'operatività e dall'osservazione durante la sua esecuzione;
- **Qualità interne:** le metriche interne, misurabili attraverso l'analisi statica, sono utili per prevedere il livello della qualità esterna ed in uso, poiché i suoi attributi interni influiscono su quelli esterni ed in uso. Permettono così di individuare anomalie prima che queste ultime possano influenzare la qualità del prodotto finale;
- **Qualità in uso:** la qualità in uso, raggiungibile solo dopo aver ottenuto la qualità interna ed esterna, fornisce metriche per misurare il grado di utilizzabilità del prodotto da parte dell'utente finale.

A.2 Standard ISO/IEC 15504

Lo standard ISO_G/IEC_G 15504, conosciuto anche come SPICE_G, è un insieme di documenti tecnici per lo sviluppo di processi software, utili a valutare la dimensione dei processi tramite l'utilizzo di specifiche metriche. È derivato dallo standard ISO/IEC 12207 e da modelli di maturità quali Bootstrap, Trillium e il CMM.

Lo standard definisce la dimensione del processo e la suddivide nelle seguenti cinque categorie:

- **Customer/Supplier;**
- **Engineering;**
- **Support;**
- **Management;**
- **Organization.**

Per ogni processo, viene definito un livello di capacità dei processi definito da una scala di sei livelli e da nove attributi suddivisi nei vari livelli:

- **Level 5. Optimizing process:** il processo è predicibile ed in grado di adattarsi per raggiungere obiettivi specifici
 - **Process Innovation:** le modifiche ad un processo sono identificate ed implementate al fine di ottenere il miglioramento continuo nel raggiungimento degli obiettivi;
 - **Process Optimization:** le modifiche alla definizione, gestione, attuazione di un processo sono controllate.

- **Level 4. Predictable process:** il processo è stabilizzato ed è attuato all'interno di definiti limiti di controllo
 - **Process Measurement:** i risultati raggiunti e le misure rilevate durante l'attuazione di un processo sono utilizzati per garantire il raggiungimento di specifici obiettivi;
 - **Process Control:** un processo è controllato attraverso le misure di prodotto e di processo rilevate, al fine di migliorare le modalità di attuazione del processo stesso.
- **Level 3. Established process:** il processo è attuato, pianificato e controllato sulla base di procedure standard basate sui principi dell'ingegneria del software
 - **Process Definition:** l'attuazione di un processo, per raggiungere gli obiettivi, si basa sull'adozione di approcci standard;
 - **Process Deployment:** l'attuazione di un processo, per raggiungere gli obiettivi, fa uso di risorse umane e tecniche appropriate.
- **Level 2. Managed process:** il processo è attuato ma anche pianificato, tracciato, verificato ed aggiustato se necessario, sulla base di obiettivi ben definiti
 - **Performance Management:** l'implementazione di un processo è pianificata e controllata al fine di produrre risultati coerenti agli obiettivi attesi;
 - **Work Product Management:** l'implementazione di un processo è pianificata e controllata al fine di produrre risultati documentati, controllati e verificati in modo appropriato.
- **Level 1. Performed process:** il processo viene messo in atto e raggiunge i suoi obiettivi. Il risultato potrebbe non essere stato pianificato e tracciato rigorosamente
 - **Process Performance:** capacità di un processo di raggiungere i suoi obiettivi trasformando input identificabili in output identificabili.
- **Level 0. Incomplete process:** il processo non è stato implementato oppure non raggiunge gli obiettivi.

Ogni attributo è misurabile tramite l'utilizzo di una scala di valutazione divisa in quattro punti:

- **Not achieved (0-15%);**
- **Partially achieved (15-50%);**
- **Largely achieved (50-85%);**
- **Fully achieved (85-100%).**

Lo standard fornisce una guida per l'effettuazione di una valutazione formata da:

- Processo di valutazione;
- Modello per la valutazione;
- Strumenti per la valutazione.

Lo standard infine, stabilisce che per una corretta valutazione i verificatori debbano avere un buon livello di competenza e di esperienza.

A.3 Ciclo PDCA

Il ciclo PDCA_G, noto anche come ciclo di Deming, è un metodo di gestione iterativo a quattro fasi per il controllo e il miglioramento continuo dei processi. Tutti i processi saranno gestiti facendo riferimento a tale modello che permette un incremento della qualità dei processi ad ogni ciclo d'iterazione.

Le quattro fasi che lo compongono sono:

- **Plan:** vengono stabiliti gli obiettivi e i processi necessari a raggiungere la qualità attesa, più in particolare, la fase sarà caratterizzata dalla valutazione della situazione attuale, dalla comprensione della natura del problema da risolvere e dalla ricerca ed analisi delle possibili soluzioni al problema;
- **Do:** fase composta dall'attuazione del piano applicando le soluzioni al problema trovate nella fase precedente;
- **Check:** si analizzano i risultati ottenuti nella fase precedente. Si confrontano i nuovi dati ottenuti con quelli attesi per verificare che quanto fatto sia corretto. Si producono grafici e tabelle per avere un riassunto chiaro su quanto rilevato. In caso di obiettivo raggiunto si potrà procedere passando all'ultima fase del ciclo PDCA_G, altrimenti, si ripeterà nuovamente il ciclo PDCA analizzando le cause che non hanno permesso di raggiungere l'obiettivo stabilito;
- **Act:** la soluzione al problema individuata e considerata adeguata nella fase precedente viene standardizzata e i membri del gruppo verranno informati. Terminata questa fase si procederà con una nuova iterazione del ciclo partendo dalla prima fase.

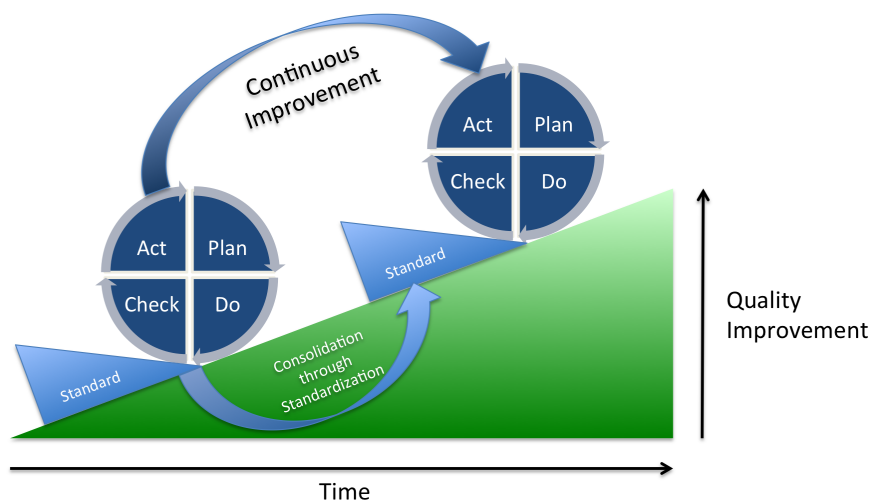


Figura 2: Ciclo di miglioramento della qualità PDCA

B Specifica dei test

B.1 Informazioni generali dei test

In seguito sono descritti tutti i test di validazione_G, sistema, di integrazione e di unità pianificati. Lo stato **N.I.** presente nelle tabelle sottostanti è da intendersi come non applicato, tali test verranno infatti svolti successivamente come descritto in *Piano di Progetto v7.0.0*.

Per una descrizione più dettagliata sui test si rimanda alla sezione dei test del documento *Norme di Progetto v6.0.0*.

B.2 Test di validazione

I test di validazione_G vengono utilizzati per accertarsi che il prodotto finale sviluppato sia conforme alle attese.

Per ogni test viene riportata una descrizione contenente i passi che l'utente deve seguire per verificare che i requisiti siano soddisfatti. Il tracciamento tra test di validazione_G e i requisiti correlati viene riportato nel documento *Analisi dei Requisiti v6.0.0*.

B.2.1 TVF1

L'utente vuole verificare che si possa registrare al sistema. All'utente è richiesto di:

- inserire un username;
- inserire una email;
- inserire una password;
- inserire una password di conferma;
- verificare che il sistema avverta l'utente in fase di compilazione nel caso in cui i dati inseriti non siano conformi alle norme del sistema;
- confermare la registrazione;
- verificare che il sistema avverta l'utente nel caso in cui i dati inseriti non siano conformi alle norme del sistema;
- verificare che, nel caso la registrazione abbia avuto esito positivo, venga inviata un email di conferma.

B.2.2 TVF1.1

L'utente vuole verificare che l'inserimento di un username avvenga correttamente. All'utente è richiesto di:

- inserire un username;
- verificare che, se lo username è già presente nel database_G, venga visualizzato un errore;
- verificare che, nel caso lo username non sia conforme alle norme del sistema, venga visualizzato un errore;
- verificare che, nel caso in cui lo username sia conforme con le norme di sistema, non vengano visualizzati errori.

B.2.3 TVF1.2

L'utente vuole verificare che l'inserimento di una email avvenga correttamente. All'utente è richiesto di:

- inserire una email;
- verificare che, se la email è già presente nel database_G, venga visualizzato un errore;
- verificare che, nel caso la email non sia conforme alle norme del sistema, venga visualizzato un errore;
- verificare che, nel caso la email sia conforme con le norme di sistema, non vengano visualizzati errori.

B.2.4 TVF1.3

L'utente vuole verificare che l'inserimento di una password avvenga correttamente. All'utente è richiesto di:

- inserire una password;
- verificare che, nel caso la password contenga lo username al suo interno, venga visualizzato un errore;
- verificare che, nel caso la password non sia conforme alle norme del sistema, venga visualizzato un errore;
- verificare che, nel caso la password sia conforme alle norme del sistema, non vengano visualizzati errori.

B.2.5 TVF1.4

L'utente vuole verificare che l'inserimento della password di conferma avvenga correttamente. All'utente è richiesto di:

- inserire una password di conferma;
- verificare che, nel caso in cui la password di conferma non sia uguale alla password precedentemente inserita, venga visualizzato un errore;
- verificare che, nel caso in cui la password di conferma coincida con la password precedentemente inserita, non vengano visualizzati errori.

B.2.6 TVF1.8

L'utente vuole verificare che, nel caso la registrazione abbia avuto esito positivo, venga inviata un email di conferma. All'utente è richiesto di:

- completare la registrazione;
- accedere alla propria casella email;
- verificare che sia arrivata l'email di conferma dal sistema.

B.2.7 TVF2

L'utente vuole verificare che si possa autenticarsi al sistema. All'utente è richiesto di:

- inserire un username;
- inserire una password;
- confermare l'autenticazione;
- verificare che il sistema avverta l'utente nel caso in cui i dati inseriti dall'utente risultino errati;
- verificare che il sistema, nel caso in cui la form_G sia compilata correttamente, non mostri errori;
- verificare che il sistema, in caso di autenticazione avvenuta con successo, aggiorni la home screen dell'utente autenticato.

B.2.8 TVF2.4

L'utente vuole verificare che il sistema avverta l'utente nel caso in cui i dati inseriti dall'utente risultino errati. All'utente è richiesto di:

- confermare l'autenticazione;
- verificare che il sistema avverta l'utente nel caso in cui il username inserito non sia presente nel sistema;
- verificare che il sistema avverta l'utente nel caso in cui la password inserita non corrisponda a quella relativa allo username inserito.

erificare che, nel caso la nuova password sia conforme alle norme del sistema, non venga visualizzati errori.

B.2.9 TVF3

L'utente vuole verificare che si possa visualizzare le proprie informazioni personali. All'utente è richiesto di:

- effettuare l'autenticazione;
- aprire il menù delle informazioni personali;
- verificare che sia possibile visualizzare i propri dati;
- verificare che sia possibile visualizzare le proprie statistiche;
- verificare che sia possibile modificare i propri dettagli personali.

B.2.10 TVF3.1

L'utente vuole verificare che si possa visualizzare i propri dati. All'utente è richiesto di:

- effettuare l'autenticazione;
- aprire il menù delle informazioni personali;
- verificare che l'utente non visualizzi i propri dati personali prima di aver premuto l'apposito pulsante;
- premere il pulsante di visualizzazione dei dati personali;
- verificare che il sistema reperisca i dati personali dell'utente;
- verificare che l'utente visualizzi il proprio username;
- verificare che l'utente visualizzi la propria email;
- verificare che l'utente visualizzi l'ultimo accesso effettuato.

B.2.11 TVF3.2

L'utente vuole verificare che si possa visualizzare le proprie statistiche. All'utente è richiesto di:

- effettuare l'autenticazione;
- aprire il menù delle informazioni personali;
- verificare che l'utente non visualizzi le proprie statistiche prima di aver premuto l'apposito pulsante;
- premere il pulsante di visualizzazione delle statistiche;
- verificare che il sistema reperisca le informazioni dell'utente;
- verificare che si visualizzi il numero di View_G attive;
- verificare che si visualizzi il numero di Recipe_G disponibili.

B.2.12 TVF3.3

L'utente vuole verificare che si possa modificare i propri dettagli personali. All'utente è richiesto di:

- effettuare l'autenticazione;
- aprire il menù delle informazioni personali;
- selezionare il pulsante di modifica dei dati personali;
- verificare che si apra una finestra contenente la form_G per modificare i propri dati personali;
- verificare che l'utente possa inserire un nuovo username;
- verificare che l'utente possa inserire un nuovo indirizzo email;

- verificare che l'utente possa cambiare la propria password;
- verificare che il sistema avverta l'utente in fase di compilazione nel caso in cui i dati inseriti non siano conformi alle norme del sistema;
- premere il pulsante di conferma delle modifiche;
- verificare che, alla pressione del pulsante di conferma delle modifiche, il sistema analizzi le conformità dei dati inseriti;
- verificare che l'utente possa eliminare il proprio account.

B.2.13 TVF3.3.2

L'utente vuole verificare che si possa modificare il proprio username. All'utente è richiesto di:

- effettuare l'autenticazione;
- aprire il menù delle informazioni personali;
- selezionare il pulsante di modifica dei dati personali;
- inserire un nuovo username nell'apposito campo;
- verificare che, nel caso il nuovo username coincida con quello vecchio, venga visualizzato un errore;
- verificare che, se lo username è già presente nel database_G, venga visualizzato un errore;
- verificare che, nel caso lo username non sia conforme alle norme del sistema, venga visualizzato un errore;
- verificare che, nel caso in cui lo username sia conforme con le norme di sistema, non vengano visualizzati errori.

B.2.14 TVF3.3.3

L'utente vuole verificare che si possa modificare la propria email. All'utente è richiesto di:

- effettuare l'autenticazione;
- aprire il menù delle informazioni personali;
- selezionare il pulsante di modifica dei dati personali;
- inserire una nuova email nell'apposito campo;
- verificare che, nel caso la nuova email coincida con quella vecchia, venga visualizzato un errore;
- verificare che, se la email è già presente nel database_G, venga visualizzato un errore;
- verificare che, nel caso la email non sia conforme alle norme del sistema, venga visualizzato un errore;
- verificare che, nel caso la email sia conforme con le norme di sistema, non vengano visualizzati errori.

B.2.15 TVF3.3.4

L'utente vuole verificare che si possa modificare la propria password. All'utente è richiesto di:

- effettuare l'autenticazione;
- aprire il menù delle informazioni personali;
- selezionare il pulsante di modifica dei dati personali;
- inserire la password corrente nell'apposito campo;
- inserire la nuova password nell'apposito campo;
- inserire la password di conferma nell'apposito campo;
- verificare che, nel caso la nuova password contenga lo username al suo interno, venga visualizzato un errore;
- verificare che, nel caso la nuova password non sia conforme alle norme del sistema, venga visualizzato un errore;
- verificare che, nel caso la nuova password non coincida con quella di conferma, venga visualizzato un errore;
- verificare che, nel caso la nuova password sia conforme alle norme del sistema e la vecchia password sia stata inserita correttamente, non vengano visualizzati errori.

B.2.16 TVF3.3.7

L'utente vuole verificare che l'utente possa eliminare il proprio account. All'utente è richiesto di:

- effettuare l'autenticazione;
- aprire il menù delle informazioni personali;
- premere il pulsante di eliminazione dell'account;
- premere il pulsante di conferma dell'eliminazione dell'account;
- verificare che, alla conferma dell'eliminazione dell'account, il sistema rimuova tutti i dati associati a quell'utente;
- verificare che il sistema, una volta confermata l'eliminazione dell'account, riporti l'utente alla home page di default del sistema per gli utenti non registrati.

B.2.17 TVF4

L'utente vuole verificare che si possa effettuare la deautenticazione dal sistema. All'utente è richiesto di:

- effettuare l'autenticazione nel caso non si fosse attualmente autenticati;
- accedere alla home screen;
- premere il pulsante di deautenticazione;
- verificare che, alla pressione del pulsante di deautenticazione, la sessione termini.

B.2.18 TVF4.1

L'utente vuole verificare che, alla pressione del pulsante di deautenticazione, la sessione termini. All'utente è richiesto di:

- effettuare l'autenticazione nel caso non si fosse attualmente autenticati;
- accedere alla home screen;
- premere il pulsante di deautenticazione;
- verificare che il sistema chieda la conferma di logout se la sessione è valida;
- verificare che, se non si conferma il logout, il sistema non effettui la deautenticazione;
- verificare che, se si conferma il logout, il sistema stampi un messaggio d'errore se la sessione è scaduta;
- verificare che il sistema reindirizzi l'utente alla schermata di login.

B.2.19 TVF5

L'utente vuole verificare che nella home screen dell'utente autenticato siano mostrate tutte le Recipe_G presenti nel sistema. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- verificare che il sistema recuperi l'elenco delle Recipe_G dal database_G ;
- verificare che sia possibile visualizzare tutte le metriche di una Recipe_G ;
- verificare che sia possibile effettuare un confronto.

B.2.20 TVF5.1

L'utente vuole verificare che il sistema recuperi l'elenco delle Recipe_G dal database_G . All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- verificare che il sistema fornisca per ogni Recipe_G il titolo e, se presente, la descrizione;
- verificare che il sistema fornisca per ogni Recipe_G due pulsanti per scegliere se visualizzare le metriche della Recipe o effettuare un confronto tra le metriche;
- verificare che il sistema permetta di dare un voto alla Recipe_G tramite un selettore.

B.2.21 TVF 5.1.3

L'utente vuole verificare che il sistema permetta di dare un voto alla Recipe_G tramite un selettore. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- selezionare dal selettore posto di fianco ad una recipe_G da una a cinque stelle;
- verificare che il voto dato ad una Recipe_G sia un intero compreso tra 1 e 5.

B.2.22 TVF5.2

L'utente vuole verificare che sia possibile visualizzare tutte le metriche di una Recipe_G . All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- verificare che la visualizzazione delle metriche sia suddivisa per categoria_G ;
- verificare che il sistema, per ogni metrica_G , fornisca il nome, la descrizione se presente e la tipologia della metrica;
- verificare che il sistema, per ogni metrica_G , fornisca il pulsante per accedere alle visualizzazioni delle View_G presenti per quella metrica.

B.2.23 TVF5.3

L'utente vuole verificare che il sistema fornisca tutte le View_G , di una metrica_G della categoria_G Facebook, predisposte dal sistema. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il pulsante per la visualizzazione delle View_G di una metrica_G della categoria_G Facebook;
- verificare che siano mostrati i grafici relativi a tale metrica_G .

B.2.24 TVF5.3.1

L'utente vuole verificare che vengano mostrati i grafici relativi ad una metrica_G della categoria_G Facebook. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il pulsante per la visualizzazione delle View_G di una metrica_G della categoria_G Facebook;
- verificare che sia visualizzato un Line Chart contenente l'andamento dei "likes" e dei "talking about" di una pagina;

- verificare che sia visualizzato un Pie Chart che per tutti gli eventi di una pagina mostri le percentuali di persone che hanno messo partecipa, forse o rifiuta;
- verificare che sia visualizzato un Pie Chart che mostri la percentuale di commenti di terzi a tutti i post di una pagina rispetto la percentuale di commenti della pagina stessa;
- verificare che sia visualizzato un Pie Chart che mostri la percentuale di commenti della pagina e di terzi rispetto a tutti i post di terzi sulla pagina in questione;
- verificare che sia visualizzato un Map Chart che illustri le zone nella quale si sono creati degli eventi di una pagina e mostra dei cerchi di diverse dimensioni a seconda della media di partecipanti agli eventi di quella zona;
- verificare che sia visualizzato un Bar Chart che mostri la media di commenti delle pagina per ogni post effettuato dalla stessa;
- verificare che sia visualizzato un Line Chart che mostri l'andamento del numero di post giornalieri di una pagina.

B.2.25 TVF5.4

L'utente vuole verificare che il sistema fornisca tutte le $View_G$, di una $metrica_G$ della categoria $_G$ Twitter, predisposte dal sistema. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il pulsante per la visualizzazione delle $View_G$ di una $metrica_G$ della categoria $_G$ Twitter;
- verificare che siano mostrati i grafici relativi a tale $metrica_G$.

B.2.26 TVF5.4.1

L'utente vuole verificare che vengano mostrati i grafici relativi ad una $metrica_G$ della categoria $_G$ Twitter. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il pulsante per la visualizzazione delle $View_G$ di una $metrica_G$ della categoria $_G$ Twitter;
- verificare che sia visualizzato un Bar Chart orizzontale che mostri un riepilogo istantaneo dei Tweet e dei Follower di un singolo utente;
- verificare che sia visualizzato un Line Chart che indichi il numero di Tweet e il numero di Follower dell'utente;
- verificare che sia visualizzato un Map Chart che illustri l'area geografica di appartenenza dei Tweet con un particolare hashtag;

- verificare che sia visualizzato un Line Chart che mostri il numero di Tweet con un determinato hashtag;
- verificare che sia visualizzato un Pie Chart che illustri quanti Tweet, quanti Preferiti e quanti Retweet ha un utente;
- verificare che sia visualizzato un Radar Chart che illustri in che momento della giornata vengono fatti i Tweet;
- verificare che sia visualizzato un Pie Chart che indichi su che tipo di piattaforma gli utenti hanno twittato;

B.2.27 TVF5.5

L'utente vuole verificare che il sistema fornisca tutte le $View_G$, di una $metrica_G$ della categoria $_G$ Instagram, predisposte dal sistema. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il pulsante per la visualizzazione delle $View_G$ di una $metrica_G$ della categoria $_G$ Instagram;
- verificare che siano mostrati i grafici relativi a tale $metrica_G$.

B.2.28 TVF5.5.1

L'utente vuole verificare che vengano mostrati i grafici relativi ad una $metrica_G$ della categoria $_G$ Instagram. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il pulsante per la visualizzazione delle $View_G$ di una $metrica_G$ della categoria $_G$ Instagram;
- verificare che sia visualizzato un Bar Chart che mostri il numero di like e il numero di commenti per ogni post di un utente;
- verificare che sia visualizzato un Line Chart che mostri i nuovi follower di un utente per giorno e mostri i momenti in cui l'utente ha pubblicato un post tramite una linea verticale;
- verificare che sia visualizzato un Line Chart che mostri il numero di like e i nuovi commenti ricevuti da ogni utente e mostri i momenti in cui l'utente ha pubblicato un post tramite una linea verticale;
- verificare che sia visualizzato un Line Chart che illustri l'andamento dei like e quello dei commenti ricevuti da un utente fratto il numero di follower che possiede;
- verificare che sia visualizzato un Line Chart che mostri il numero di like ricevuti al giorno da un utente fratto il numero di post giornalieri dello stesso utente;

- verificare che sia visualizzato un Line Chart che mostri il numero di post e il numero di utenti che hanno utilizzato un determinato hashtag;
- verificare che sia visualizzato un Line Chart che mostri il numero di like e il numero di commenti ricevuto dai post che contengono un determinato hashtag;
- verificare che sia visualizzato un Map Chart che mostri le zone geografiche dove sono stati fatti i post contenenti un determinato hashtag;
- verificare che sia visualizzato un Radar Chart che illustri, per i vari momenti della giornata, la frequenza con cui un utente posta.

B.2.29 TVF5.6

L'utente vuole verificare che il sistema permetta all'utente di effettuare il confronto tra le metriche. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il tasto di confronto;
- selezionare una categoria_G tra Facebook, Twitter e Instagram;
- selezionare un tipo di metrica_G tra quelle presenti nella categoria_G selezionata;
- verificare che vengano visualizzate tutte le metriche del tipo e della categoria_G selezionati;
- selezionare due o tre metriche tra quelle visualizzate;
- verificare che sia possibile generare un confronto tra le metriche selezionate.

B.2.30 TVF5.6.4

L'utente vuole verificare che il sistema generi il confronto tra le metriche selezionate. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il tasto di confronto;
- selezionare una categoria_G tra Facebook, Twitter e Instagram;
- selezionare un tipo di metrica_G tra quelle presenti nella categoria_G selezionata;
- selezionare due o tre metriche tra quelle visualizzate;
- verificare che venga visualizzate la View_G di confronto;
- verificare che i dati vengano illustrati sul grafico con un colore diverso per ciascuna delle metriche selezionate;
- verificare che, nel caso non siano presenti dati sufficienti per generare la View_G di confronto, venga visualizzato un messaggio di avvertimento e si ritorni alla pagina di selezione delle metriche.

B.2.31 TVF6

L'utente vuole verificare che il sistema permetta all'utente di gestire le View_G che più preferisce. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- verificare che il sistema permetta all'utente di visualizzare tutte le View_G che sono state marcate da lui come preferite;
- verificare che il sistema permetta all'utente di aggiungere qualsiasi View_G a quelle preferite cliccando il pulsante vicino alla View che vuole aggiungere;
- verificare che il sistema permetta all'utente di rimuovere le View_G che ha aggiunto tra le preferite cliccando il pulsante vicino alla View che vuole rimuovere;
- verificare che l'utente possa inserire come preferite un numero illimitato di View_G.

B.2.32 TVF6.1

L'utente vuole verificare che il sistema permetta all'utente di visualizzare tutte le View_G che sono state marcate da lui come preferite. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- accedere al menù delle View_G preferite;
- verificare che il sistema avvisi l'utente nel caso non ci fossero View_G salvate come preferite;
- verificare che il sistema mostri tutte le View_G preferite dell'utente nel caso ce ne sia almeno una.

B.2.33 TVF6.2

L'utente vuole verificare che il sistema permetta all'utente di aggiungere qualsiasi View_G a quelle preferite cliccando il pulsante vicino alla View che vuole aggiungere. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il pulsante di aggiunta della View_G tra le preferite;
- verificare che il sistema mostri un errore nel caso non sia stata aggiunta correttamente una View_G tra quelle preferite;
- verificare che il sistema abbia aggiunto correttamente la View_G tra i preferiti.

B.2.34 TVF6.3

L'utente vuole verificare che il sistema permetta all'utente di rimuovere le $View_G$ che ha aggiunto tra le preferite cliccando il pulsante vicino alla View che vuole rimuovere. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- accedere al menù delle $View_G$ preferite;
- premere il pulsante di rimozione della $View_G$ dalle preferite;
- verificare che il sistema mostri un errore nel caso non sia stata rimossa correttamente una $View_G$ tra quelle preferite;
- verificare che il sistema mostri abbia rimosso correttamente una $View_G$ da quelle preferite.

B.2.35 TVF7

L'utente vuole verificare che il sistema permetta all'utente di richiedere l'inserimento di una nuova $Recipe_G$. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla sezione per la richiesta di una nuova $Recipe_G$;
- verificare che l'utente possa generare la richiesta dall'apposito $form_G$;
- verificare che il sistema invii una notifica agli amministratori che è stata inserita una nuova richiesta di $Recipe_G$.

B.2.36 TVF7.1

L'utente vuole verificare che il sistema permetta all'utente di generare la richiesta dall'apposito $form_G$. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla sezione per la richiesta di una nuova $Recipe_G$;
- verificare che il sistema controlli che l'utente inserisca tutti i parametri richiesti dalla $form_G$ correttamente;
- premere il pulsante di invio della richiesta;
- verificare che la richiesta fallisca se i dati inseriti sono errati o si sono lasciati campi vuoti.

B.2.37 TVF7.1.1

L'utente vuole verificare che il sistema controlli che l'utente inserisca tutti i parametri richiesti dalla form_G correttamente. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla sezione per la richiesta di una nuova Recipe_G;
- inserire il titolo della Recipe_G;
- inserire almeno una metrica_G nella richiesta della Recipe_G;
- selezionare la categoria_G della metrica_G tra Facebook, Twitter e Instagram;
- selezionare la tipologia di metrica_G in base alla categoria_G scelta;
- compilare i campi in base alla tipologia e alla categoria_G di metrica_G scelta;
- verificare che il sistema avvisi l'utente in modo istantaneo se un campo è stato compilato correttamente;
- verificare che il sistema avvisi l'utente in modo istantaneo se un campo obbligatorio è stato lasciato vuoto.

B.2.38 TVF8

L'utente vuole verificare che il sistema permetta ad un utente con privilegi di amministratore di usufruire di tutte le funzioni offerte agli amministratori. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere all'area riservata;
- verificare che un amministratore può accedere al pannello per inserire una nuova Recipe_G;
- verificare che il sistema richieda la conferma dei dati inseriti prima di creare la nuova Recipe_G;
- verificare che l'amministratore possa eliminare una Recipe_G memorizzata nel sistema dalla schermata di elenco delle Recipe;
- verificare che l'amministratore possa visualizzare la classifica delle Recipe_G valutate dagli utenti.

B.2.39 TVF8.1

L'utente vuole verificare che il sistema permetta ad un utente con privilegi di amministratore di accedere al pannello per inserire una nuova Recipe_G. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere all'area riservata;
- verificare che sia richiesto l'inserimento di determinati parametri per individuare il contesto dei dati richiesti dalla Recipe_G.

B.2.40 TVF8.1.1

L'utente vuole verificare che il sistema richieda l'inserimento di determinati parametri per individuare il contesto dei dati richiesti dalla Recipe_G . All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere all'area riservata;
- inserire il titolo della Recipe_G ;
- inserire almeno una metrica_G nella Recipe_G ;
- selezionare la categoria_G della metrica_G tra Facebook, Twitter e Instagram;
- selezionare la tipologia di metrica_G in base alla categoria_G scelta;
- compilare i campi in base alla tipologia e alla categoria_G di metrica_G scelta.

B.2.41 TVF8.2

L'utente vuole verificare che il sistema richieda la conferma dei dati inseriti prima di creare la nuova Recipe_G . All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere all'area riservata;
- inserire i dati della nuova Recipe_G nel form_G ;
- verificare che sia visualizzato un messaggio d'errore nel caso in cui venga confermata una Recipe_G che non contiene almeno una metrica_G per categoria_G ;
- verificare che sia visualizzato un messaggio d'errore nel caso in cui venga confermata una Recipe_G in cui l'utente non ha inserito dei parametri specificati nel requisito ROF8.1.1.4;
- verificare che nel caso venga confermata una nuova Recipe_G , il sistema salvi i suoi parametri nel database_G e reindirizzi l'amministratore all'elenco delle Recipe .

B.2.42 TVF8.3

L'utente vuole verificare che l'amministratore possa eliminare una Recipe_G memorizzata nel sistema dalla schermata di elenco delle Recipe . All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere all'area riservata;
- premere il pulsante di eliminazione posto di fianco alla Recipe_G che si desidera eliminare;
- verificare che il sistema non provvedi all'eliminazione di tutte le informazioni legate alla Recipe_G senza aver confermato prima l'eliminazione;

- verificare che il sistema provvedi all'eliminazione di tutte le informazioni legate alla Recipe_G che si desidera eliminare alla pressione del tasto di conferma eliminazione;
- verificare che il sistema provvedi ad eliminare tutti i dati grezzi dal database_G legati alla Recipe_G .

B.2.43 TVF8.4

L'utente vuole verificare che l'amministratore possa visualizzare la classifica delle Recipe_G valutate dagli utenti. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere all'area riservata;
- premere il pulsante per visualizzare la classifica delle Recipe_G ;
- verificare che se venga visualizzata la lista con le ricette ordinate in base al voto;
- verificare che se nessuna Recipe_G è stata valutata venga visualizzato un messaggio.

B.2.44 TVF9

L'utente vuole verificare che l'amministratore possa gestire gli utenti. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- accedere all'elenco degli utenti;
- verificare che il sistema permetta all'utente amministratore di modificare i permessi di un utente;
- verificare che il sistema permetta all'utente amministratore di eliminare un utente dal sistema;
- verificare che il sistema permetta all'utente amministratore di confermare le modifiche applicate.

B.2.45 TVF9.1

L'utente vuole verificare che il sistema permetta all'utente amministratore di modificare i permessi di un utente. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- accedere all'elenco degli utenti;
- verificare che, alla pressione del tasto di modifica dei permessi, le modifiche vengano applicate solo localmente in attesa che vengano confermate.

B.2.46 TVF9.2

L'utente vuole verificare che il sistema permetta all'utente amministratore di eliminare un utente dal sistema. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- accedere all'elenco degli utenti;
- verificare che, alla pressione del tasto di modifica dei permessi, le modifiche vengano applicate solo localmente in attesa che vengano confermate.

B.2.47 TVF9.3

L'utente vuole verificare che il sistema permetta all'utente amministratore di confermare le modifiche applicate. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- accedere all'elenco degli utenti;
- verificare che il sistema mostri un errore e le modifiche vengano annullate nel caso in cui un utente cambi i permessi a se stesso;
- verificare che il sistema mostri un errore e le modifiche vengano annullate nel caso in cui si provi ad eliminare un utente attualmente autenticato;
- verificare che il sistema mostri un messaggio di conferma nel caso le modifiche abbiano avuto esito positivo.

B.2.48 TVF10

L'utente vuole verificare che l'amministratore possa gestire le richieste di nuove Recipe_G ricevute dagli utenti. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- accedere all'elenco delle richieste di nuove Recipe_G ;
- verificare che l'utente amministratore possa vedere l'elenco delle nuove Recipe_G ;
- verificare che il sistema mostri un messaggio nel caso non siano presenti richieste di nuove Recipe_G ;
- verificare che, cliccando sulla visualizzazione dei dettagli di una Recipe_G , si acceda ad un pannello contenente il form_G per la creazione di una nuova Recipe precompilata con i dati richiesti;
- verificare che, cliccando sul pulsante di accettazione, venga inserita la Recipe_G richiesta nel sistema;
- verificare che, cliccando sul pulsante di rifiuto, venga rifiutato l'inserimento della Recipe_G richiesta.

B.2.49 TVF10.1

L'utente vuole verificare che l'utente amministratore possa vedere l'elenco delle nuove Recipe_G . All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- accedere all'elenco degli utenti;
- verificare che, per ogni voce dell'elenco, sia fornito il titolo della Recipe_G richiesta, lo username dell'utente che l'ha richiesta e la descrizione nel caso sia presente;
- verificare che, per ogni voce dell'elenco, sia presente un pulsante per accedere alla visualizzazione dei dettagli della richiesta.

B.2.50 TVF10.3

L'utente vuole verificare che, cliccando sulla visualizzazione dei dettagli di una Recipe_G , si acceda ad un pannello contenente il form_G per la creazione di una nuova Recipe precompilata con i dati richiesti. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- accedere all'elenco degli utenti;
- premere il tasto per visualizzare i dettagli della Recipe_G ;
- verificare che il sistema permetta all'utente amministratore di modificare il titolo e la descrizione della Recipe_G richiesta;
- verificare che il sistema non permetta di modificare altri campi della richiesta della Recipe_G al di fuori del titolo e della descrizione.

B.2.51 TVF11

L'utente vuole verificare che il sistema fornisca una serie di servizi REST_G . All'utente è richiesto di:

- verificare che il sistema fornisca la possibilità di richiedere un token di accesso per utilizzare i servizi REST_G ;
- verificare che il sistema permetta all'utente di richiedere l'annullamento di un token;
- verificare che il sistema gestisca la richiesta dei dati di una View_G contenente un token di accesso.

B.2.52 TVF11.1

L'utente vuole verificare che il sistema fornisca la possibilità di richiedere un token di accesso per utilizzare i servizi REST_G. All'utente è richiesto di:

- verificare che il sistema restituisca un errore se il sistema ha avuto problemi nel generare il token;
- verificare che il sistemi generi e restituisca un token di accesso se l'utente è autenticato.

B.2.53 TVF11.2

L'utente vuole verificare che il sistema permetta all'utente di richiedere l'annullamento di un token. All'utente è richiesto di:

- verificare che il sistema gestisca una richiesta di chiusura sessione contenente un token di accesso valido rendendolo non valido;
- verificare che il sistema restituisca un errore se non si è riuscito ad invalidare il token;
- verificare che il sistema restituisca una conferma dell'avvenuta invalidazione del token.

B.2.54 TVF11.3

L'utente vuole verificare che il sistema gestisca la richiesta dei dati di una View_G contenente un token di accesso. All'utente è richiesto di:

- verificare che il sistema restituisca i dati richiesti se il token utilizzato è valido;
- verificare che il sistema restituisca un errore se il token utilizzato non è valido.

B.2.55 TVF12

L'utente vuole verificare che il sistema permetta all'utente di visualizzare la specifica dei servizi REST_G offerti. All'utente è richiesto di:

- effettuare l'autenticazione;
- accedere alla home screen;
- premere il tasto per visualizzare la specifica dei servizi REST_G;
- verificare che siano elencati tutti i servizi offerti;
- verificare che sia presente un link al manuale completo.

B.2.56 TVQ1

L'utente vuole verificare che venga fornito un manuale per l'utente. All'utente è richiesto di:

- accedere al sito;
- verificare che il sito fornisca il manuale utente attraverso un link;
- verificare che, cliccando il link, si apra il manuale utente correttamente.

B.2.57 TVQ3

L'utente vuole verificare che venga fornito un manuale per l'uso dei servizi REST_G. All'utente è richiesto di:

- accedere al sito;
- verificare che il sito fornisca il manuale d'uso dei servizi REST_G attraverso un link;
- verificare che, cliccando il link, si apra il manuale d'uso dei servizi REST_G correttamente.

B.2.58 TVV1

L'utente vuole verificare che il sistema sfrutti gli strumenti offerti da Google Cloud Platform. All'utente è richiesto di:

- effettuare l'autenticazione su Google Cloud Platform;
- verificare che il sistema sfrutti le Google App Engine_G;
- verificare che il sistema sfrutti il Google Cloud Datastore;
- verificare che il sistema sfrutti gli strumenti offerti da Google Cloud Platform.

B.2.59 TVV1.1

L'utente vuole verificare che il sistema sfrutti le Google App Engine_G. All'utente è richiesto di:

- effettuare l'autenticazione su Google Cloud Platform;
- accedere alla console della Google Cloud Platform;
- verificare che sia presente il progetto;

B.2.60 TVV1.2

L'utente vuole verificare che il sistema sfrutti il Google Cloud Datastore. All'utente è richiesto di:

- effettuare l'autenticazione su Google Cloud Platform;
- accedere alla console della Google Cloud Platform;
- verificare che nella sezione Cloud Datastore sia presente il database_G.

B.2.61 TVV2

L'utente vuole verificare che il linguaggio di programmazione principalmente usato per il back-end sia Python. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che il codice sorgente per il back-end sia scritto in Python.

B.2.62 TVV3

L'utente vuole verificare che il codice sorgente sia soggetto a versionamento tramite il modello di branching descritto nelle *Norme di Progetto v6.0.0*. All'utente è richiesto di:

- accedere al repository_G;
- entrare all'interno del progetto contenente il codice sorgente;
- verificare che il modello di branching utilizzato sia lo stesso di quello descritto nelle *Norme di Progetto v6.0.0*.

B.2.63 TVV4

L'utente vuole verificare che l'interfaccia web sia di tipo single-page. All'utente è richiesto di:

- accedere al sito;
- effettuare l'autenticazione;
- verificare, navigando nel sito, che l'interfaccia sia di tipo single-page.

B.2.64 TVV5

L'utente vuole verificare che l'interfaccia web funzioni con i principali browser attualmente sul mercato. All'utente è richiesto di:

- verificare il corretto funzionamento dell'interfaccia web con il browser Google Chrome v. 39.0 e successive;
- verificare il corretto funzionamento dell'interfaccia web con il browser Firefox v. 35.0 e successive;
- verificare il corretto funzionamento dell'interfaccia web con il browser Safari v. 8.0 e successive.

B.2.65 TVV6

L'utente vuole verificare che il sistema utilizzi le librerie esterne per effettuare le chiamate alle API_G dei diversi social network. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che sia utilizzata la libreria facebook-sdk_G per le chiamate alle API_G di Facebook;
- verificare che sia utilizzata la libreria tweepy per le chiamate alle API_G di Twitter;
- verificare che sia utilizzata la libreria python-instagram per le chiamate alle API_G di Instagram.

B.2.66 TVV6.1

L'utente vuole verificare che sia utilizzata la libreria facebook-sdk_G per le chiamate alle API_G di Facebook. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che sia importata la libreria facebook-sdk_G;
- verificare che siano effettuate le chiamate tramite la libreria facebook-sdk_G.

B.2.67 TVV6.2

L'utente vuole verificare che sia utilizzata la libreria tweepy per le chiamate alle API_G di Twitter. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che sia importata la libreria tweepy;
- verificare che siano effettuate le chiamate tramite la libreria tweepy.

B.2.68 TVV6.3

L'utente vuole verificare che sia utilizzata la libreria python-instagram per le chiamate alle API_G di Instagram. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che sia importata la libreria python-instagram;
- verificare che siano effettuate le chiamate tramite la libreria python-instagram.

B.2.69 TVV7

L'utente vuole verificare che siano utilizzate librerie esterne per generare i grafici necessari alle View_G. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che sia utilizzata la libreria Chart.js per generare i Line Chart, Bar Chart, Pie Chart e Radar Chart;
- verificare che sia utilizzata la libreria Google Geo Chart per generare i Map Chart.

B.2.70 TVV7.1

L'utente vuole verificare che sia utilizzata la libreria Chart.js per generare i Line Chart, Bar Chart, Pie Chart e Radar Chart. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che sia importata la libreria Chart.js;
- verificare che siano utilizzata la libreria Chart.js per la generazione dei Line Chart;

- verificare che siano utilizzata la libreria Chart.js per la generazione dei Bar Chart;
- verificare che siano utilizzata la libreria Chart.js per la generazione dei Pie Chart;
- verificare che siano utilizzata la libreria Chart.js per la generazione dei Radar Chart.

B.2.71 TVV7.2

L'utente vuole verificare che sia utilizzata la libreria Google Chart per generare i Map Chart. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che sia importata la libreria Google Chart;
- verificare che siano utilizzata la libreria Google Chart per la generazione dei Map Chart.

B.2.72 TVV8

L'utente vuole verificare che il linguaggio di programmazione principalmente usato per il front-end sia Javascript. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che il codice sorgente per il front-end sia scritto in Javascript.

B.2.73 TVV8.1

L'utente vuole verificare che il venga utilizzato AngularJS_G come framework_G per il front-end. All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare che sia importato il framework_G AngularJS_G per il front-end;
- verificare che il codice sorgente per il front-end utilizzi AngularJS_G.

B.2.74 TVV9

L'utente vuole verificare che il token di accesso sia nel formato: "id utente" + "data creazione token" + "valore random". All'utente è richiesto di:

- avere accesso al codice sorgente;
- verificare nel codice sorgente che la creazione del token di accesso sia programmato in modo da rispettare la combinazione seguente: "id utente" + "data creazione token" + "valore random".

B.3 Test di sistema

I test di sistema servono per verificare che il sistema software completamente integrato soddisfi tutti i requisiti software individuati e descritti nel documento *Analisi dei Requisiti v6.0.0*.

B.3.1 Descrizione dei test di sistema

Test	Descrizione	Stato	Requisito
TSF1	Viene verificato che il sistema permetta, all'utente non autenticato, la registrazione al servizio	Superato	ROF1
TSF2	Viene verificato che il sistema permetta all'utente di effettuare l'autenticazione	Superato	ROF2
TSF3	Viene verificato che il sistema permetta, all'utente autenticato, di accedere al menù delle informazioni personali e delle statistiche del sistema	Superato	ROF3
TSF4	Viene verificato che il sistema permetta, all'utente autenticato, di effettuare la deautenticazione	Superato	ROF4
TSF5	Viene verificato che il sistema permetta, all'utente autenticato, di visualizzare tutte le Recipe_G presenti nel sistema	Superato	ROF5
TSF6	Viene verificato che il sistema permetta all'utente di gestire le proprie View_G	Superato	RDF6
TSF7	Viene verificato che il sistema permetta all'utente di richiedere l'inserimento di una nuova Recipe_G	Superato	RFF7
TSF8	Viene verificato che il sistema permetta, all'utente amministratore autenticato, di accedere all'area riservata del sistema	Superato	ROF8
TSF9	Viene verificato che il sistema permetta, all'utente amministratore autenticato, di accedere all'elenco degli utenti registrati	Superato	ROF9
TSF10	Viene verificato che il sistema permetta, all'utente amministratore, di gestire la richiesta di nuove Recipe_G	N.I.	RFF10
TSF11	Viene verificato che il sistema fornisca una serie di servizi REST_G	Superato	ROF11
TSF12	Viene verificato che il sistema permetta all'utente di visualizzare la specifica dei servizi REST_G offerti	Superato	ROF12
TSP1	Viene verificato che l'utente visualizzi le proprie View_G entro 10 secondi	Superato	RDP1

TSP2	Viene verificato che l'interfaccia web utilizzi un design di tipo responsive _G	Superato	RFP2
TSQ1	Viene verificato che sia disponibile un manuale per l'utente	Superato	ROQ1
TSQ2	Viene verificato che tutto il codice rispetti le norme e le metriche descritte nel <i>Piano di Qualifica v6.0.0</i> e <i>Norme di Progetto v6.0.0</i>	Superato	ROQ2
TSQ3	Viene verificato che sia disponibile un manuale per l'uso dei servizi REST _G	Superato	ROQ3
TSV1	Viene verificato che il sistema utilizzi gli strumenti offerti da Google Cloud Platform	Superato	RDV1
TSV2	Viene verificato che il linguaggio di programmazione utilizzato per il back-end sia Python	Superato	RDV2
TSV3	Viene verificato che il codice sia soggetto a versionamento tramite il modello di branching descritto nelle <i>Norme di Progetto v6.0.0</i>	Superato	RDV3
TSV4	Viene verificato che l'interfaccia web sia di tipo single-page	Superato	RDV4
TSV5	Viene verificato che l'interfaccia web funzioni con i principali browser attualmente sul mercato	Superato	ROV5
TSV6	Viene verificato che il sistema utilizzi librerie esterne per effettuare le chiamate alle API _G dei diversi social network	Superato	ROV6
TSV7	Viene verificato che il sistema utilizzi librerie esterne per generare i grafici necessari alle View _G	Superato	ROV7
TSV8	Viene verificato che il linguaggio di programmazione utilizzato per il front-end sia Javascript	Superato	ROV8
TSV9	Viene verificato che il token di accesso sia formato dalla combinazione: "id utente" + "data creazione token" + "valore random"	Superato	ROV9

Tabella 1: Tracciamento test di sistema - requisiti

B.4 Test di integrazione

I test di integrazione vengono utilizzati per verificare che tutti i componenti del sistema siano integrati correttamente tra di loro e che il flusso dati all'interno del sistema sia corretto.

Si è deciso di fare affidamento ad una strategia di integrazione di tipo incrementale in modo da permetterci di sviluppare e verificare più componenti in parallelo.

La strategia incrementale ci permette anche di effettuare la ricerca dei difetti in maniera più precisa, infatti nel caso si presenti un errore, questo sarà probabilmente causato dall'ultima componente inserita e permettendoci di ritornare ad uno stato del sistema corretto.

Viene utilizzato il metodo meet-in-the-middle, cioè vengono integrate le componenti partendo sia dall'alto che dal basso fino ad integrare il tutto insieme. In questo modo è possibile ottenere in breve tempo delle integrazioni corrette in entrambe le direzioni (top-down e bottom-up). Sarà, quindi, necessario testare più volte le componenti per assicurarci che il prodotto software finale non contenga difetti.

Successivamente si procederà ad aggiungere le componenti che corrispondono ai requisiti desiderabili e opzionali.

B.4.1 Descrizione dei test di integrazione

Test	Descrizione	Componente	Stato
TIadmin_controller	Viene verificato che il sistema gestisca correttamente le decisioni su quali pagine HTML mostrare all'utente amministratore e su quali operazioni può interagire quest'ultimo.	bdsms_app::client::controller::admin	Superato
TIadmin_view	Viene verificato che il sistema permetta una corretta visualizzazione di tutte le pagine HTML disponibili per l'utente amministratore.	bdsms_app::client::view _G ::admin	Superato
TIapi	Viene verificato che il sistema fornisca delle API _G funzionanti al client che le utilizzerà.	bdsms_app::server::endpoints::api _G	Superato
TIapp_data	Viene verificato che il sistema gestisca correttamente il modello di dati degli utenti, delle Recipe _G , dei preferiti e delle richieste di nuove Recipe.	bdsms_app::server::db::app_data	Superato

TIbdsms_app	Viene verificata la correttezza generale del sistema integrante i livelli client, database _G e server.	bdsms_app	Superato
TIclient	Viene verificato che il sistema gestisca correttamente il front-end del prodotto.	bdsms_app::client	Superato
TIcommands	Viene verificato che il sistema gestisca correttamente le chiamate provenienti dal client.	bdsms_app::server::processor _G ::commands	Superato
TIdata	Viene verificato che il sistema gestisca correttamente la struttura di memorizzazione dei dati utilizzata.	bdsms_app::client::model::data	N.I.
TIdb	Viene verificato che il sistema gestisca correttamente l'interazione con il database _G .	bdsms_app::server::db	Superato
TIendpoints	Viene verificato che il sistema gestisca correttamente la componente endpoints del server. Più precisamente viene verificato che il sistema gestisca correttamente le chiamate provenienti dal client e le relative risposte.	bdsms_app::server::endpoints	Superato
TIfb_miner	Viene verificato che il sistema prelevi, elabori e salvi correttamente i dati dal social network Facebook.	bdsms_app::server::miner _G ::fb	Superato
TIfb_resp	Viene verificato che il sistema gestisca correttamente le risposte relative ai servizi REST _G pubblici che riguardano i dati provenienti dal social network Facebook.	bdsms_app::server::endpoints::resp::public::fb	Superato

Tlig_miner	Viene verificato che il sistema prelevi, elabori e salvi correttamente i dati dal social network Instagram.	bdsms_app::server::miner _G ::ig	Superato
Tlig_resp	Viene verificato che il sistema gestisca correttamente le risposte relative ai servizi REST _G pubblici che riguardano i dati provenienti dal social network Instagram.	bdsms_app::server::endpoints::resp::public::ig	Superato
Tlminer	Viene verificato che il sistema prelevi, elabori e salvi correttamente i dati dai diversi servizi.	bdsms_app::server::miner _G	Superato
Tlprivate_api	Viene verificato che il sistema effettui correttamente le chiamate ai servizi REST _G privati provenienti dal client.	bdsms_app::server::endpoints::api _G ::private	Superato
Tlprivate_resp	Viene verificato che il sistema gestisca correttamente le risposte relative ai servizi REST _G privati verso il client.	bdsms_app::server::endpoints::resp::private	Superato
Tlprocessor	Viene verificato che il sistema gestisca correttamente le chiamate provenienti dai servizi REST _G utilizzati dal client.	bdsms_app::server::processor _G	Superato
Tlpublic_api	Viene verificato che il sistema effettui correttamente le chiamate ai servizi REST _G pubblici provenienti dal client.	bdsms_app::server::endpoints::api _G ::public	Superato
Tlpublic_controller	Viene verificato che il sistema gestisca correttamente le decisioni su quali pagine HTML mostrare all'utente non autenticato e su quali operazioni può interagire quest'ultimo.	bdsms_app::client::controller::public	Superato

TIpública_resp	Viene verificato che il sistema gestisca correttamente le risposte relative ai servizi REST _G pubblici verso il client.	bdsms_app::server::endpoints::resp::public	Superato
TIpública_view	Viene verificato che il sistema permetta una corretta visualizzazione di tutte le pagine HTML disponibili per l'utente non autenticato.	bdsms_app::client::view _G ::public	Superato
TIdraw_data	Viene verificato che il sistema gestisca correttamente il modello di dati grezzi ricavati dai vari social network.	bdsms_app::server::db::raw_data	Superato
TIdrecipe_commands	Viene verificato che il sistema gestisca correttamente i comandi relativi alla gestione delle Recipe _G .	bdsms_app::server::processor _G ::commands::recipe _G	Superato
TIdrequests_commands	Viene verificato che il sistema gestisca correttamente i comandi relativi alla gestione delle richieste di aggiunta delle Recipe _G .	bdsms_app::server::processor _G ::commands::requests	Superato
TIdresp	Viene verificato che il sistema gestisca correttamente le risposte verso il client.	bdsms_app::server::endpoints::resp	Superato
TIdserver	Viene verificato che il sistema gestisca correttamente il back-end del prodotto.	bdsms_app::server	Superato
TIdservices	Viene verificato che il sistema gestisca correttamente l'interazione con il back-end.	bdsms_app::client::model::services	Superato
TIdsocial_commands	Viene verificato che il sistema gestisca correttamente i comandi relativi alla raccolta dei dati grezzi ricavati dai social network.	bdsms_app::server::processor _G ::commands::social	Superato

TI _{tw_miner}	Viene verificato che il sistema prelevi, elabori e salvi correttamente i dati dal social network Twitter.	bds _{m_app} ::-server::miner _G ::-tw	Superato
TI _{tw_resp}	Viene verificato che il sistema gestisca correttamente le risposte relative ai servizi REST _G pubblici che riguardano i dati provenienti dal social network Twitter.	bds _{m_app} ::-server::-endpoints::-resp::public::tw	Superato
TI _{user_commands}	Viene verificato che il sistema gestisca correttamente i comandi relativi alla gestione degli utenti.	bds _{m_app} ::-server::-processor _G ::-commands::user	Superato
TI _{user_controller}	Viene verificato che il sistema gestisca correttamente le decisioni su quali pagine HTML mostrare all'utente autenticato e su quali operazioni può interagire quest'ultimo.	bds _{m_app} ::-client::-controller::user	Superato
TI _{user_view}	Viene verificato che il sistema permetta una corretta visualizzazione di tutte le pagine HTML disponibili per l'utente autenticato.	bds _{m_app} ::-client::view _G ::-user	Superato

Tabella 2: Tabella test di integrazione

B.4.2 Tracciamento componenti - test di integrazione

Componente	Test
bds _{m_app}	TI _{bds_{m_app}}
bds _{m_app} ::client	TI _{client}
bds _{m_app} ::client::model	Architettura del sistema
bds _{m_app} ::client::model::data	TI _{data}
bds _{m_app} ::client::model::services	TI _{services}
bds _{m_app} ::client::view _G	Architettura del sistema

bdsms_app::client::view _G ::public	TIpublic_view
bdsms_app::client::view _G ::user	TIuser_view
bdsms_app::client::view _G ::admin	TIadmin_view
bdsms_app::client::controller	Architettura del sistema
bdsms_app::client::controller::public	TIcontroller_public
bdsms_app::client::controller::user	TIcontroller_user
bdsms_app::client::controller::admin	TIcontroller_admin
bdsms_app::server	TIserver
bdsms_app::server::db	TIdb
bdsms_app::server::db::raw_data	TIraw_data
bdsms_app::server::db::app_data	TIapp_data
bdsms_app::server::processor _G	TIprocessor
bdsms_app::server::processor _G ::- commands	TIcommands
bdsms_app::server::processor _G ::- commands::user	TIuser_commands
bdsms_app::server::processor _G ::- commands::recipe _G	TIrecipe_commands
bdsms_app::server::processor _G ::- commands::requests	TIrequests_commands
bdsms_app::server::processor _G ::- commands::social	TIsocial_commands
bdsms_app::server::miner _G	TIminer
bdsms_app::server::miner _G ::fb	TIfb_miner
bdsms_app::server::miner _G ::tw	TItw_miner
bdsms_app::server::miner _G ::ig	TIig_miner
bdsms_app::server::endpoints	TIendpoints
bdsms_app::server::endpoints::api _G	TIapi
bdsms_app::server::endpoints::- api _G ::public	TIpublic_api
bdsms_app::server::endpoints::- api _G ::private	TIprivate_api
bdsms_app::server::endpoints::resp	TIresp
bdsms_app::server::endpoints::- resp::public	TIpublic_resp

bdsm_app::server::endpoints::-resp::public::fb	TIfb_resp
bdsm_app::server::endpoints::-resp::public::tw	TItw_resp
bdsm_app::server::endpoints::-resp::public::ig	TIig_resp
bdsm_app::server::endpoints::-resp::private	TIprivate_resp

Tabella 3: Tracciamento componenti - test di integrazione

B.5 Test di unità

B.5.1 Descrizione dei test di unità

Test	Descrizione	Esito
TU1	Viene verificato che, passando l'id di una pagina Facebook, vengano estratti correttamente tutti i dati relativi a tale pagina utilizzando le API _G fornite da Facebook.	Superato
TU2	Viene verificato che, passando l'id di un evento di Facebook, vengano estratti correttamente tutti i dati relativi a tale evento utilizzando le API _G fornite da Facebook.	Superato.
TU3	Viene verificato che, passando l'id di una pagina Facebook, vengano estratti correttamente tutti i dati dei trend relativi a tale pagina utilizzando le API _G fornite da Facebook.	Superato
TU4	Viene verificato che, passando l'id di una pagina Facebook, vengano estratti correttamente tutti i dati dei post presenti su tale pagina utilizzando le API _G fornite da Facebook.	Superato
TU5	Viene verificato che, passando l'id di un evento di Facebook, vengano estratti correttamente tutti i dati dei trend relativi a tale evento utilizzando le API _G fornite da Facebook.	Superato
TU6	Viene verificato che, passando l'id di un evento di Facebook, vengano estratti correttamente tutti i dati dei post presenti su tale evento utilizzando le API _G fornite da Facebook.	Superato

TU7	Viene verificato che, passando il nome di un utente di Twitter, vengano estratti correttamente tutti i dati relativi a tale utente utilizzando le API _G fornite da Twitter.	Superato
TU8	Viene verificato che, passando il nome di un utente di Twitter, vengano estratti correttamente tutti i dati dei trend relativi a tale utente utilizzando le API _G fornite da Twitter.	Superato
TU9	Viene verificato che, passando il nome di un utente di Twitter, vengano estratti correttamente tutti i dati dei tweet relativi a tale utente utilizzando le API _G fornite da Twitter.	Superato
TU10	Viene verificato che, passando il nome di un hashtag di Twitter, vengano estratti correttamente tutti i dati relativi a tale hashtag utilizzando le API _G fornite da Twitter.	Superato
TU11	Viene verificato che, passando il nome di un hashtag di Twitter, vengano estratti correttamente tutti i dati dei tweet relativi a tale hashtag utilizzando le API _G fornite da Twitter.	Superato
TU12	Viene verificato che, passando il nome di un utente di Instagram, vengano estratti correttamente tutti i dati relativi a tale utente utilizzando le API _G fornite da Instagram.	Superato
TU13	Viene verificato che, passando il nome di un utente di Instagram, vengano estratti correttamente tutti i dati relativi a tale utente utilizzando le API _G fornite da Instagram.	Superato
TU14	Viene verificato che, passando il nome di un utente di Instagram, vengano estratti correttamente tutti i dati dei media relativi a tale utente utilizzando le API _G fornite da Instagram.	Superato
TU15	Viene verificato che, passando il nome di un hashtag di Instagram, vengano estratti correttamente tutti i dati relativi a tale hashtag utilizzando le API _G fornite da Instagram.	Superato
TU16	Viene verificato che, passando il nome di un hashtag di Instagram, vengano estratti correttamente tutti i dati dei trend relativi a tale hashtag utilizzando le API _G fornite da Instagram.	Superato

TU17	Viene verificato che, passando il nome di un hashtag di Instagram, vengano estratti correttamente tutti i dati dei media relativi a tale hashtag utilizzando le API _G fornite da Instagram.	Superato
TU18	Viene verificato che, passando il nome di una pagina di Facebook, non vengano prelevati i dati statici di tale pagina se già presente nel database _G .	Superato
TU19	Viene verificato che, passando il nome di un evento di Facebook, non vengano prelevati i dati statici di tale evento se già presente nel database _G .	Superato
TU20	Viene verificato che, passando il nome di un utente di Twitter, non vengano prelevati i dati statici di tale utente se già presente nel database _G .	Superato
TU21	Viene verificato che, passando il nome di un utente di Instagram, non vengano prelevati i dati statici di tale utente se già presente nel database _G .	Superato
TU22	Viene verificato che, passando una ricetta contenente delle metriche, vengano prelevati tutti i dati di tale metriche.	Superato
TU23	Viene verificato il corretto funzionamento del singleton del processor _G .	Superato
TU24	Viene verificato che, dato un dizionario contenente i dati di registrazione di un utente, l'aggiunta di un nuovo utente al sistema avvenga correttamente.	Superato
TU25	Viene verificato che, dato un dizionario contenente i dati modificati di un utente già registrato al sistema, la modifica dei dati dell'utente avvenga correttamente.	Superato
TU26	Viene verificato che, dato il nome utente di un utente registrato al sistema, la cancellazione di tale utente dal sistema avvenga correttamente.	Superato
TU27	Viene verificato che, dato il nome utente di un utente registrato al sistema, il metodo restituisca i dati relativi a tale utente.	Superato
TU28	Viene verificato che il metodo restituisca correttamente la lista di tutti gli utenti registrati al sistema.	Superato

TU29	Viene verificato che, dato il nome utente di un utente registrato al sistema, il metodo permetta la modifica dei permessi di tale utente facendolo diventare un utente amministratore.	Superato
TU30	Viene verificato che, dati i dati di una $View_G$, il metodo permetta all'utente di aggiungere tale View tra quelle preferite.	Superato
TU31	Viene verificato che, dati i dati di una $View_G$ tra quelle preferite, il metodo permetta all'utente la rimozione di una View dalle preferite.	Superato
TU32	Viene verificato che il metodo permetta il recupero di tutte le $View_G$ impostate come preferite da un determinato utente.	Superato
TU33	Viene verificato che, dati i dati di una $Recipe_G$ e delle sue metriche associate, il metodo permetta l'aggiunta di tale Recipe nel sistema.	Superato
TU34	Viene verificato che, dato il titolo di una $Recipe_G$, il metodo permetta l'eliminazione della Recipe, avente tale titolo, dal sistema.	Superato
TU35	Viene verificato che, dato il titolo di una $Recipe_G$, il metodo ricavi correttamente i dati della Recipe presente nel sistema e avente tale titolo.	Superato
TU36	Viene verificato che il metodo restituisca correttamente la lista di tutte le $Recipe_G$ presenti nel sistema.	Superato
TU37	Viene verificato che, dato il titolo della $Recipe_G$ e un voto, il metodo permetta di valutare tale Recipe assegnandoli tale voto.	Superato
TU38	Viene verificato che, passati i dati di una $Recipe_G$ e delle sue metriche associate, il metodo permetta ad un utente non amministratore di richiedere l'inserimento di una nuova Recipe e delle metriche ad essa associata.	Superato
TU39	Viene verificato che, passato il titolo di una richiesta di $Recipe_G$ effettuata precedentemente, il metodo permetta ad un utente di eliminare tale richiesta di nuova Recipe.	Superato
TU40	Viene verificato che, passato il titolo di una richiesta di $Recipe_G$ effettuata precedentemente, il metodo permetta di visualizzare tale richiesta di nuova Recipe.	Superato

TU41	Viene verificato che il metodo permetta di visualizzare la lista delle richieste di nuove Recipe _G presenti nel sistema.	Superato
TU42	Viene verificato che, dato il titolo di una richiesta di Recipe _G il metodo permetta di visualizzare la lista di tutte le metriche presenti in una richiesta di nuova Recipe.	N.I.
TU43	Viene verificato che, dato l'id di una pagina di Facebook, il metodo restituisca correttamente i dati statici relativi a tale pagina Facebook presenti nel database _G .	Superato
TU44	Viene verificato che, dato l'id di una pagina di Facebook, il metodo restituisca correttamente tutti i dati dinamici relativi ai trend di tale pagina Facebook presenti nel database _G .	Superato
TU45	Viene verificato che, dato l'id di una pagina di Facebook, il metodo restituisca correttamente tutti i dati dinamici relativi ai post di tale pagina Facebook presenti nel database _G .	Superato
TU46	Viene verificato che, dato l'id di un evento di Facebook, il metodo restituisca correttamente i dati statici relativi a tale evento di Facebook presenti nel database _G .	Superato
TU47	Viene verificato che, dato l'id di un evento di Facebook, il metodo restituisca correttamente tutti i dati dinamici relativi a tale evento di Facebook presenti nel database _G .	Superato
TU48	Viene verificato che, dato l'id di un evento di Facebook, il metodo restituisca correttamente tutti i dati dinamici relativi a tale evento di Facebook presenti nel database _G .	Superato
TU49	Viene verificato che, dato il nome di un utente di Twitter, il metodo restituisca correttamente i dati statici relativi a tale utente di Twitter presenti nel database _G .	Superato
TU50	Viene verificato che, dato il nome di un utente di Twitter, il metodo restituisca correttamente tutti i dati dinamici relativi ai trend di tale utente di Twitter presenti nel database _G .	Superato
TU51	Viene verificato che, dato il nome di un utente di Twitter, il metodo restituisca correttamente tutti i dati dinamici relativi ai tweet di tale utente di Twitter presenti nel database _G .	Superato

TU52	Viene verificato che, dato il nome di un hashtag di Twitter, il metodo restituisca correttamente tutti i dati dinamici relativi ai tweet di tale hashtag di Twitter presenti nel database _G .	Superato
TU53	Viene verificato che, dato il nome di un utente di Instagram, il metodo restituisca correttamente i dati statici relativi a tale un utente di Instagram presenti nel database _G .	Superato
TU54	Viene verificato che, dato il nome di un utente di Instagram, il metodo restituisca correttamente tutti i dati dinamici relativi ai trend di tale utente di Instagram presenti nel database _G .	Superato
TU55	Viene verificato che, dato il nome di un utente di Instagram, il metodo restituisca correttamente tutti i dati dinamici relativi ai media di tale utente di Instagram presenti nel database _G .	Superato
TU56	Viene verificato che, dato il nome di un hashtag di Instagram, il metodo restituisca correttamente tutti i dati dinamici relativi ai trend di tale hashtag di Instagram presenti nel database _G .	Superato
TU57	Viene verificato che, dato il nome di un hashtag di Instagram, il metodo restituisca correttamente tutti i dati dinamici relativi ai media di tale hashtag di Instagram presenti nel database _G .	Superato
TU58	Viene verificato che, dopo aver recuperato i dati statici di una pagina Facebook tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati statici di tale pagina.	Superato
TU59	Viene verificato che, dopo aver recuperato i dati dinamici relativi ai trend di una pagina Facebook tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati dinamici dei trend relativi a tale pagina.	Superato
TU60	Viene verificato che, dopo aver recuperato i dati dinamici relativi ai post di una pagina Facebook tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati dinamici dei post relativi a tale pagina.	Superato

TU61	Viene verificato che, dopo aver recuperato i dati statici di un evento di Facebook tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati statici di tale evento.	Superato
TU62	Viene verificato che, dopo aver recuperato i dati dinamici relativi ai trend di un evento di Facebook tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati dinamici dei trend relativi a tale evento.	Superato
TU63	Viene verificato che, dopo aver recuperato i dati statici di un utente di Twitter tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati statici di tale utente.	Superato
TU64	Viene verificato che, dopo aver recuperato i dati dinamici relativi ai trend di un utente di Twitter tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati dinamici dei trend relativi a tale utente.	Superato
TU65	Viene verificato che, dopo aver recuperato i dati dinamici relativi ai tweet di un utente di Twitter tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati dinamici dei tweet relativi a tale utente.	Superato
TU66	Viene verificato che, dopo aver recuperato i dati statici di un hashtag di Twitter tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati statici di tale hashtag.	Superato
TU67	Viene verificato che, dopo aver recuperato i dati dinamici relativi ai tweet di un hashtag di Twitter tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati dinamici dei tweet relativi a tale hashtag.	Superato
TU68	Viene verificato che, dopo aver recuperato i dati statici di un utente di Instagram tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati statici di tale utente.	Superato

TU69	Viene verificato che, dopo aver recuperato i dati dinamici relativi ai trend di un utente di Instagram tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati dinamici dei trend relativi a tale utente.	Superato
TU70	Viene verificato che, dopo aver recuperato i dati dinamici relativi ai media di un utente di Instagram tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati dinamici dei media relativi a tale utente.	Superato
TU71	Viene verificato che, dopo aver recuperato i dati statici di un hashtag di Instagram tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati statici di tale hashtag.	Superato
TU72	Viene verificato che, dopo aver recuperato i dati dinamici relativi ai trend di un hashtag di Instagram tramite le API _G e averli salvati nel database _G , il metodo permetta di interrogare il database e recuperare i dati dinamici dei trend relativi a tale hashtag.	Superato
TU73	Viene verificato che l'API _G ritorni una lista contenente tante Recipe _G quante quelle presenti nel sistema.	Superato
TU74	Viene verificato che l'API _G , una volta ricevuta una Recipe _G come parametro, ritorni una lista contenente tante metriche quante quelle presenti nel sistema relative a quella determinata Recipe.	Superato
TU75	Viene verificato che l'API _G , una volta ricevuta il nome di una pagina Facebook come parametro, ritorni i dati statici di quella pagina se già presente nel sistema.	Superato
TU76	Viene verificato che l'API _G , una volta ricevuta il nome di una pagina Facebook come parametro, ritorni i dati dinamici dei trend di quella pagina se già presenti nel sistema.	Superato
TU77	Viene verificato che l'API _G , una volta ricevuta il nome di una pagina Facebook come parametro, ritorni i dati dinamici dei post di quella pagina se già presenti nel sistema.	Superato

TU78	Viene verificato che l'API _G , una volta ricevuta il nome di un evento Facebook come parametro, ritorni i dati statici di quel evento se già presente nel sistema.	Superato
TU79	Viene verificato che l'API _G , una volta ricevuta il nome di un evento Facebook come parametro, ritorni i dati dinamici dei trend di quel evento se già presenti nel sistema.	Superato
TU80	Viene verificato che l'API _G , una volta ricevuta il nome di un evento Facebook come parametro, ritorni i dati dinamici dei post di quel evento se già presenti nel sistema.	Superato
TU81	Viene verificato che l'API _G , una volta ricevuta il nome di un utente Twitter come parametro, ritorni i dati statici di quel utente se già presente nel sistema.	Superato
TU82	Viene verificato che l'API _G , una volta ricevuta il nome di un utente Twitter come parametro, ritorni i dati dinamici dei trend di quel utente se già presenti nel sistema.	Superato
TU83	Viene verificato che l'API _G , una volta ricevuta il nome di un utente Twitter come parametro, ritorni i dati dinamici dei tweet di quel utente se già presenti nel sistema.	Superato
TU84	Viene verificato che l'API _G , una volta ricevuta il nome di un hashtag di Twitter come parametro, ritorni i dati dinamici dei tweet di quel hashtag se già presenti nel sistema.	Superato
TU85	Viene verificato che l'API _G , una volta ricevuta il nome di un utente di Instagram come parametro, ritorni i dati statici di quel utente se già presente nel sistema.	Superato
TU86	Viene verificato che l'API _G , una volta ricevuta il nome di un utente di Instagram come parametro, ritorni i dati dinamici dei trend di quel utente se già presenti nel sistema.	Superato
TU87	Viene verificato che l'API _G , una volta ricevuta il nome di un utente di Instagram come parametro, ritorni i dati dinamici dei media di quel utente se già presenti nel sistema.	Superato

TU88	Viene verificato che l'API _G , una volta ricevuta il nome di un hashtag di Instagram come parametro, ritorni i dati dinamici dei trend di quel hashtag se già presenti nel sistema.	Superato
TU89	Viene verificato che l'API _G , una volta ricevuta il nome di un hashtag di Instagram come parametro, ritorni i dati dinamici dei media di quel hashtag se già presenti nel sistema.	Superato
TU90	Viene verificato che l'API _G , una volta ricevuti dal client i dettagli di una nuova Recipe _G da inserire, chiami correttamente la classe del processor _G che si occuperà di inserire la nuova Recipe nel sistema.	Superato
TU91	Viene verificato che l'API _G , una volta ricevuto dal client il nome di una Recipe _G , chiami correttamente la classe del processor _G che si occuperà di rimuovere la Recipe presente nel sistema.	Superato
TU92	Viene verificato che l'API _G , una volta ricevuti il nome di una Recipe _G e il voto da assegnarli, chiami correttamente la classe del processor _G che si occuperà di assegnare tale voto alla Recipe presente nel sistema.	Superato
TU93	Viene verificato che l'API _G ritorni una lista contenente tanti utenti tanti quelli registrati nel sistema.	Superato
TU94	Viene verificato che l'API _G , una volta ricevuto il nome di un utente registrato nel sistema, ritorni i dati relativi a tale utente.	Superato
TU95	Viene verificato che l'API _G , una volta ricevuto il nome di un utente registrato nel sistema e i dati da modificare di tale utente, modifichi correttamente i dati di tale utente.	Superato
TU96	Viene verificato che l'API _G , una volta ricevuto il nome di un utente registrato nel sistema, cambi i permessi dell'utente registrato al sistema facendolo diventare amministratore .	Superato
TU97	Viene verificato che l'API _G , una volta ricevuto il nome di un utente registrato nel sistema, rimuova correttamente l'utente dal sistema.	Superato
TU98	Viene verificato che l'API _G , una volta ricevuto i dati di registrazione di un nuovo utente, registri correttamente l'utente al sistema.	Superato

TU99	Viene verificato che l'API _G , una volta ricevuto il nome e la password di un utente registrato nel sistema, effettui correttamente l'autenticazione di tale utente se i dati inseriti risultano esatti.	Superato
TU100	Viene verificato che l'API _G , una volta ricevuto il nome di un utente registrato nel sistema, deautentichi correttamente l'utente dal sistema.	Superato
TU101	Viene verificato che l'API _G ritorni una lista contenente tante richieste di nuove Recipe _G tante quelle presenti nel sistema.	Superato
TU102	Viene verificato che l'API _G , una volta ricevuto i dettagli di una nuova richiesta di Recipe _G , salvi correttamente la richiesta al sistema.	Superato
TU103	Viene verificato che l'API _G , una volta ricevuto il nome di una richiesta di Recipe _G precedentemente effettuata, elimini tale richiesta dal sistema.	Superato
TU104	Viene verificato che l'API _G ritorni una lista contenente tante View _G quante quelle impostate dall'utente come preferite nel sistema.	Superato
TU105	Viene verificato che l'API _G , una volta ricevuto il nome di una View _G , inserisca tale View tra le preferite dell'utente che ha effettuato l'operazione.	Superato
TU106	Viene verificato che l'API _G , una volta ricevuto il nome di una View _G impostata come preferita da un utente, elimini dai preferiti tale View.	Superato
TU107	Viene verificato che il controller ApiDocs esista e sia associato alla pagina ApiDocs.	Superato
TU108	Viene verificato che, alla fine dell'esecuzione del metodo getRestService, la variabile contenga una lista di servizi REST _G aventi un determinato formato.	Superato
TU109	Viene verificato che il controller Login esista e sia associato alla pagina Login.	Superato
TU110	Viene verificato che, alla fine dell'esecuzione del metodo login, la variabile contenente le credenziali d'accesso sia vuota se l'autenticazione è avvenuta con successo.	Superato
TU111	Viene verificato che il controller Register esista e sia associato alla pagina Register.	Superato

TU112	Viene verificato che, nel metodo checkMatchPwd, se la password e la password di conferma inserite coincidono venga ritornato true.	Superato
TU113	Viene verificato che, alla fine dell'esecuzione del metodo register, la variabile contenente le credenziali sia vuota se la registrazione è avvenuta con successo.	Superato
TU114	Viene verificato che la lunghezza dell'access token sia di lunghezza uguale a quanto descritto nel documento d'analisi.	Superato
TU115	Viene verificato che, se la chiamata al metodo deleteAccessToken è avvenuta con successo, il valore della variabile accessToken sia vuoto.	Superato
TU116	Viene verificato che il metodo matchNewPassword restituisca true nel caso la password nuova e quella di conferma coincidano.	Superato
TU117	Viene verificato che il metodo checkModifyFields restituisca true nel caso siano avvenute modifiche da salvare nel form _G .	Superato
TU118	Viene verificato che il metodo saveEdit restituisca true nel caso in cui il salvataggio delle modifiche sia avvenuto correttamente.	Superato
TU119	Viene verificato che il controller Recipe _G esista e sia associato alla pagina Recipe.	Superato
TU120	Viene verificato che, se la chiamata al metodo è avvenuta con successo, la variabile contenente la lista delle Recipe _G non sia vuota e gli oggetti in essa siano nel formato previsto.	Superato
TU121	Viene verificato che il controller Metrics esista e sia associato alla pagina Metrics.	Superato
TU122	Viene verificato che il metodo getMetricType ritorni un oggetto nel formato corretto.	Superato
TU123	Viene verificato che il metodo getMetricList ritorni una lista di metriche nel formato corretto.	Superato
TU124	Viene verificato che il controller UsersConfig esista e sia associato alla pagina UsersConfig.	Superato
TU125	Viene verificato che, se la chiamata al metodo è avvenuta con successo, la variabile contenente la lista degli utenti non sia vuota e gli oggetti in essa siano nel formato previsto.	Superato

TU126	Viene verificato che l'esecuzione del metodo editPermission modifichi correttamente il valore del permesso di un determinato utente facendolo diventare amministratore.	Superato
TU127	Viene verificato che, a seguito dell'esecuzione del metodo deleteUserAccount, la lista degli utenti iscritti al sistema sia diminuito di uno.	Superato
TU128	Viene verificato che il metodo checkMetricQuantity restituisca true se il numero di metriche inserito sia maggiore o uguale a due.	Superato
TU129	Viene verificato che, a seguito della chiamata al metodo addMetric, il numero di metriche presenti nella ricetta sia aumentato di uno.	Superato
TU130	Viene verificato che, se la chiamata al metodo insertRecipe ha avuto successo, allora i valori contenuti nella form _G siano resettati correttamente.	Superato
TU131	Viene verificato che il metodo getRequestList ritorni una lista contenente tante richieste quante quelle presenti nel sistema.	Superato
TU132	Viene verificato che il metodo discardRequest ritorni una lista contenente un numero di richieste pari al precedente meno uno.	N.I.
TU133	Viene verificato che il metodo approveRequest ritorni una lista contenente un numero di richieste pari al precedente più uno.	N.I.
TU134	Viene verificato che il metodo getListOfRecipe ritorni una lista contenente tante ricette quante quelle presenti nel sistema.	Superato
TU135	Viene verificato che venga restituita una promise.	Superato
TU136	Viene verificato che venga restituita una promise.	Superato
TU137	Viene verificato che venga restituita una promise.	Superato
TU138	Viene verificato che il numero di dati passato al metodo sia uguale al numero di entries nel grafico.	Superato
TU139	Viene verificato che nella generazione del grafico PieChart le percentuali vengano rispettate.	Superato

TU140	Viene verificato che nella generazione del grafico LineChart le date disposte nell'ascissa siano ordinate in modo crescente.	Superato
TU141	Viene verificato che nel datastorage sia stata inserita la coppia chiave-valore di cui si voleva effettuare l'inserimento.	Superato
TU142	Viene verificato che nel datastorage sia stata rimossa la coppia chiave-valore di cui si voleva effettuare la rimozione.	Superato

Tabella 4: Tabella test di unità

B.5.2 Tracciamento test di unità - metodi

Test	Classe.Metodo
TU1	FbPageFetcher.__init__() FbPageFetcher.run() FbPageFetcher.get_data()
TU2	FbEventFetcher.__init__() FbEventFetcher.run() FbEventFetcher.get_data()
TU3	FbPageFetcher.__init__() FbPageFetcher.run() FbPageFetcher.get_data() FbPageFetcher.get_trend_data()
TU4	FbPageFetcher.run() FbPageFetcher.get_trend_data() PostCounter.__init__() AbsFbCounter.count() PostCounter.action() PostCounter.get_likes_count() CommentsCounter.__init__() CommentsCounter.action() AbsFbCounter.break_condition()
TU5	FbEventFetcher.__init__() FbEventFetcher.run() FbEventFetcher.get_trend_data() FbEventFetcher.get_trend_data()

TU6	FbEventFetcher.run() FbEventFetcher.get_trend_data() FbEventFetcher.get_invited_stats() FbEventFetcher.parse_location() PostCounter.__init__() AbsFbCounter.count() PostCounter.action() PostCounter.get_likes_count() CommentsCounter.__init__() CommentsCounter.action() AbsFbCounter.break_condition()
TU7	TwUserFetcher.__init__() TwUserFetcher.run() TwUserFetcher.get_data()
TU8	TwUserFetcher.__init__() TwUserFetcher.run() TwUserFetcher.get_trend_data()
TU9	TwUserFetcher.get_trend_data() UserTweetCounter.__init__() UserTweetCounter.action() AbsTwCounter.count() UserTweetCounter.break_condition()
TU10	TwHashtagFetcher.__init__() TwHashtagFetcher.run() TwHashtagFetcher.get_data()
TU11	TwHashtagFetcher.get_trend_data() HashtagTweetCounter.__init__() HashtagTweetCounter.action() AbsTwCounter.count() HashtagTweetCounter.break_condition()
TU12	IgUserFetcher.__init__() IgUserFetcher.run() IgUserFetcher.get_data()
TU13	IgUserFetcher.__init__() IgUserFetcher.run() IgUserFetcher.get_data() IgUserFetcher.get_trend_data()
TU14	IgUserFetcher.get_trend_data() MediaCounter.__init__() MediaCounter.count_user() MediaCounter.action() MediaCounter.break_condition()

TU15	IgHashtagFetcher.__init__() IgHashtagFetcher.run() IgHashtagFetcher.get_data()
TU16	IgHashtagFetcher.__init__() IgHashtagFetcher.run() IgHashtagFetcher.get_data() IgHashtagFetcher.get_trend_data()
TU17	IgHashtagFetcher.get_trend_data() MediaCounter.__init__() MediaCounter.count_hashtag() MediaCounter.action() MediaCounter.break_condition()
TU18	FbPageFetcher.__init__() FbPageFetcher.run() FbPageFetcher.get_data()
TU19	FbEventFetcher.__init__() FbEventFetcher.run() FbEventFetcher.get_data()
TU20	TwUserFetcher.__init__() TwUserFetcher.run() TwUserFetcher.get_data()
TU21	IgUserFetcher.__init__() IgUserFetcher.run() IgUserFetcher.get_data()
TU22	MinerScheduler.__init__() MinerScheduler.schedule_fetchers() MinerScheduler.update_fb_metrics() MinerScheduler.update_ig_metrics() MinerScheduler.update_tw_metrics()
TU23	RequestHandler.get_instance()
TU24	AddUserCommand.__init__() AddUserCommand.execute()
TU25	EditUserCommand.__init__() EditUserCommand.execute()
TU26	DeleteUserCommand.__init__() DeleteUserCommand.execute()
TU27	GetUserCommand.__init__() GetUserCommand.execute()
TU28	GetUserListCommand.__init__() GetUserListCommand.execute()

TU29	EditPermissionCommand.__init__() EditPermissionCommand.execute()
TU30	AddFavoriteCommand.__init__() AddFavoriteCommand.execute()
TU31	DeleteFavoriteCommand.__init__() DeleteFavoriteCommand.execute()
TU32	GetFavoritesCommand.__init__() GetFavoritesCommand.execute()
TU33	AddRecipeCommand.__init__() AddRecipeCommand.execute()
TU34	DeleteRecipeCommand.__init__() DeleteRecipeCommand.execute()
TU35	GetRecipeCommand.__init__() GetRecipeCommand.execute()
TU36	GetRecipeListCommand.__init__() GetRecipeListCommand.execute()
TU37	RateRecipeCommand.__init__() RateRecipeCommand.execute()
TU38	AddRequestCommand.__init__() AddRequestCommand.execute()
TU39	DeleteRequestCommand.__init__() DeleteRequestCommand.execute()
TU40	GetRequestCommand.__init__() GetRequestCommand.execute()
TU41	GetRequestListCommand.__init__() GetRequestListCommand.execute()
TU42	GetMetricListCommand.__init__() GetMetricListCommand.execute()
TU43	FbPageCommand.__init__() FbPageCommand.execute()
TU44	FbPageTrendCommand.__init__() FbPageTrendCommand.execute()
TU45	FbPagePostsCommand.__init__() FbPagePostsCommand.execute()
TU46	FbEventCommand.__init__() FbEventCommand.execute()
TU47	FbEventTrendCommand.__init__() FbEventTrendCommand.execute()

TU48	FbEventPostsCommand.__init__() FbEventPostsCommand.execute()
TU49	TwUserCommand.__init__() TwUserCommand.execute()
TU50	TwUserTrendCommand.__init__() TwUserTrendCommand.execute()
TU51	TwUserTweetsCommand.__init__() TwUserTweetsCommand.execute()
TU52	TwHashtagTweetsCommand.__init__() TwHashtagTweetsCommand.execute()
TU53	IgUserCommand.__init__() IgUserCommand.execute()
TU54	IgUserTrendCommand.__init__() IgUserTrendCommand.execute()
TU55	IgUserMediaCommand.__init__() IgUserMediaCommand.execute()
TU56	IgHashtagTrendCommand.__init__() IgHashtagTrendCommand.execute()
TU57	IgUserMediaCommand.__init__() IgUserMediaCommand.execute()
TU58	RawFbPage.query().fetch()
TU59	RawFbPageTrend.query().fetch()
TU60	RawFbPostTrend.query().fetch()
TU61	RawFbEvent.query().fetch()
TU62	RawFbEventTrend.query().fetch()
TU63	RawTwUser.query().fetch()
TU64	RawTwUserTrend.query().fetch()
TU65	RawTwUserTweet.query().fetch()
TU66	RawTwHashtag.query().fetch()
TU67	RawTwHashtagTweet.query().fetch()
TU68	RawIgUser.query().fetch()
TU69	RawIgUserTrend.query().fetch()
TU70	RawIgMedia.query().fetch()
TU71	RawIgHashtag.query().fetch()
TU72	RawIgHashtagTrend.query().fetch()
TU73	MetricApi.recipeG_list()

TU74	MetricApi.metric_list()
TU75	FbMetricsApi.get_page()
TU76	FbMetricsApi.get_page_trend()
TU77	FbMetricsApi.get_page_posts()
TU78	FbMetricsApi.get_event()
TU79	FbMetricsApi.get_event_trend()
TU80	FbMetricsApi.get_event_posts()
TU81	TwMetricsApi.get_user()
TU82	TwMetricsApi.get_user_trend()
TU83	TwMetricsApi.get_user_tweets()
TU84	TwMetricsApi.get_hashtag_tweets()
TU85	IgMetricsApi.get_user()
TU86	IgMetricsApi.get_user_trend()
TU87	IgMetricsApi.get_media()
TU88	IgMetricsApi.get_hashtag()
TU89	IgMetricsApi.get_hashtag_trend()
TU90	GetRecipe.add_recipe()
TU91	GetRecipe.remove_recipe()
TU92	GetRecipe.rate_recipe()
TU93	UserApi.user_list()
TU94	UserApi.user_details()
TU95	UserApi.edit_user()
TU96	UserApi.edit_permission()
TU97	UserApi.remove_user()
TU98	OAuthApi.add_user()
TU99	OAuthApi.login()
TU100	OAuthApi.logout()
TU101	RecipeRequestApi.request_list()
TU102	RecipeRequestApi.add_request()
TU103	RecipeRequestApi.remove_request()
TU104	FavouritesApi.favourites_list()
TU105	FavouritesApi.add_favourite()

TU106	FavouritesApi.remove_favourite()
TU107	ApiDocs
TU108	ApiDocsCtrl.getRestService()
TU109	LoginCtrl
TU110	LoginCtrl.login()
TU111	RegisterCtrl
TU112	RegisterCtrl.checkMatchPwd()
TU113	RegisterCtrl.register()
TU114	TokenConfigCtrl.getAccessToken()
TU115	TokenConfigCtrl.deleteAccessToken()
TU116	SettingsCtrl.matchNewPassword()
TU117	SettingsCtrl.checkModifyFields()
TU118	SettingsCtrl.saveEdit()
TU119	RecipeCtrl
TU120	RecipeCtrl.getListOfRecipes()
TU121	MetricsCtrl
TU122	MetricsCtrl.getMetricType()
TU123	MetricsCtrl.getMetricList()
TU124	UserConfigCtrl
TU125	UserConfigCtrl.getUsers()
TU126	UserConfigCtrl.editPermissions()
TU127	UserConfigCtrl.deletUserAccount()
TU128	InsertRecipeCtrl.checkMetricsQuantity()
TU129	InsertRecipeCtrl.addMetric()
TU130	InsertRecipeCtrl.insertRecipe()
TU131	RequestListCtrl.getRequestList()
TU132	RequestListCtrl.discardRequest()
TU133	RequestListCtrl.approveRequest()
TU134	RatingsCtrl.getListOfRecipes()

TU135	UserService.getFavourites() UserService.addFavourite() UserService.getAccessToken() UserService.deleteAccessToken() UserAdminService.getListOfUsers() UserAdminService.editUserPermissions() UserAdminService.deleteUserAccount()
TU136	RecipeService.getRecipesList() RecipeService.getMetricsList() RecipeService.getMetricType() RecipeService.getMetricTypeNode() RecipeAdminService.getRecipesListAll() RecipeAdminService.getListOfRecipesRequest() RecipeAdminService.createRecipe() RecipeAdminService.setRecipeRating() RecipeAdminService.discardRecipeRequest() RecipeAdminService.approveRecipeRequest()
TU137	AuthService.login() AuthService.register() AuthService.deleteAccount() AuthService.isLogged() AuthService.updateSettingsAccount()
TU138	ChartCreator.dataFormat()
TU139	PieChartCreator.dataFormat() ChartCreator.chartGeneration()
TU140	LineChartCreator.dataFormat() ChartCreator.chartGeneration()
TU141	DataManagerService.postRestCall()
TU142	DataManagerService.deleteRestCall()

Tabella 5: Tabella tracciamento test di unità - metodi

C Resoconto delle attività di verifica

C.1 Revisione dei Requisiti

Nel periodo precedente alla consegna della *Revisione dei Requisiti* sono state effettuate delle attività di verifica sia per i documenti sia per i processi.

Per quanto riguarda la verifica dei documenti si è effettuata attività di analisi statica descritta nelle *Norme di Progetto v6.0.0*. Inizialmente si è fatto uso della tecnica walkthrough per individuare gli errori. Dopodiché sono state avviate le procedure per la segnalazione e la gestione degli errori rilevati descritte nelle *Norme di Progetto v6.0.0*. In seguito si è proceduto a:

- correggere gli errori rilevati;
- compilare la lista di controllo utilizzando l'apposita sezione *Bug_G Tracking Document* di Asana_G.

Una volta compilata la lista di controllo si è cominciato ad utilizzare la tecnica inspection.

Si è quindi applicato il ciclo PDCA_G per migliorare i processi che hanno generato gli errori. Infine si sono applicate le metriche per i documenti descritte nella sezione 2.8.2 riportando i risultati nella sezione D dell'appendice di questo documento. Per quanto riguarda la verifica dei processi si sono svolte le attività descritte nelle *Norme di Progetto v6.0.0*. Si sono quindi applicate le metriche per i processi descritte nella sezione 2.8.1 di questo documento riportando i risultati nella sezione C dell'appendice di questo documento.

C.2 Revisione di Progettazione

Nel periodo precedente alla consegna della *Revisione di Progettazione* sono state effettuate delle attività di verifica sia per i documenti sia per i processi.

Per quanto riguarda la verifica dei documenti si è effettuata attività di analisi statica descritta nelle *Norme di Progetto v6.0.0*. Inizialmente si è fatto uso della tecnica walkthrough per individuare gli errori. Dopodiché sono state avviate le procedure per la segnalazione e la gestione degli errori rilevati descritte nelle *Norme di Progetto v6.0.0*. In seguito si è proceduto a:

- correggere gli errori rilevati;
- compilare la lista di controllo utilizzando l'apposita sezione *Bug_G Tracking Document* di Asana_G.

Una volta compilata la lista di controllo si è cominciato ad utilizzare la tecnica inspection ponendo attenzione ai grafici riportati nella *Specifica Tecnica*.

Si è quindi applicato il ciclo PDCA_G per migliorare i processi che hanno generato gli errori. Infine si sono applicate le metriche per i documenti descritte nella sezione 2.8.2 riportando i risultati nella sezione D dell'appendice di questo documento. Per quanto riguarda la verifica dei processi si sono svolte le attività descritte nelle *Norme di Progetto v6.0.0*. Si sono quindi applicate le metriche per i processi descritte nella sezione 2.8.1 di questo documento riportando i risultati nella sezione C dell'appendice di questo documento.

D Dettaglio delle verifiche tramite analisi

D.1 Ricerca ed implementazione degli strumenti

D.1.1 Processi

Il seguente grafico deriva dall'analisi dei ticket_G pianificati, svolti, approvati e verificati dal gruppo in relazione alle date in cui hanno cambiato stato.

Come si può notare in figura 3, c'è una rapida crescita di task_G pianificati nelle date dei verbali interni che corrispondono alle riunioni del gruppo, nonché ai verbali interni. Verso la fine della fase si osserva un aumento dei task approvati e verificati, questo per tentare di completare tutti quelli pianificati entro la data di fine fase. Si nota inoltre che non tutti i task sono stati completati entro la fine della fase, questo andrà ad appesantire dunque la fase successiva che si verrà a trovare i task della fase precedente da completare.

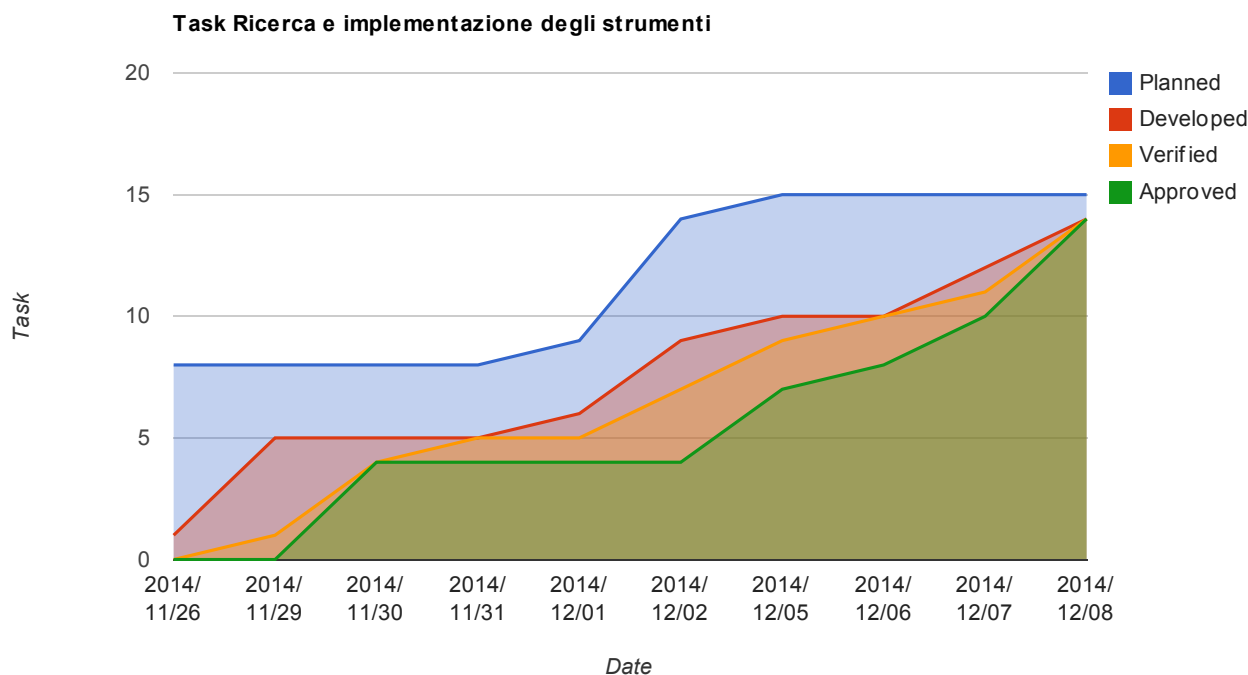


Figura 3: Grafico task Ricerca ed implementazione degli strumenti

D.1.2 Documenti

Essendo in questa fase non presente ancora alcun documento, non è stato possibile verificarli e quindi avere dei valori da esporre. Per correttezza è stato comunque inserito il nome della fase.

D.2 Analisi dei requisiti

D.2.1 Processi

La figura 4 rappresenta l'andamento dei task_G relativi a questa fase, si nota che dato l'inesperienza del gruppo e il tipo di fase in cui ci troviamo, inizialmente non ci sono molti task pianificati come ci si potrebbe aspettare in altre fasi. Il gruppo ha proseguito a pianificare pochi task alla volta fino al primo incontro con il proponente_G. Dal 2015/01/14 si osserva infatti un notevole picco inizialmente per i Planned ma poi anche per i Developed Verified e Approved.

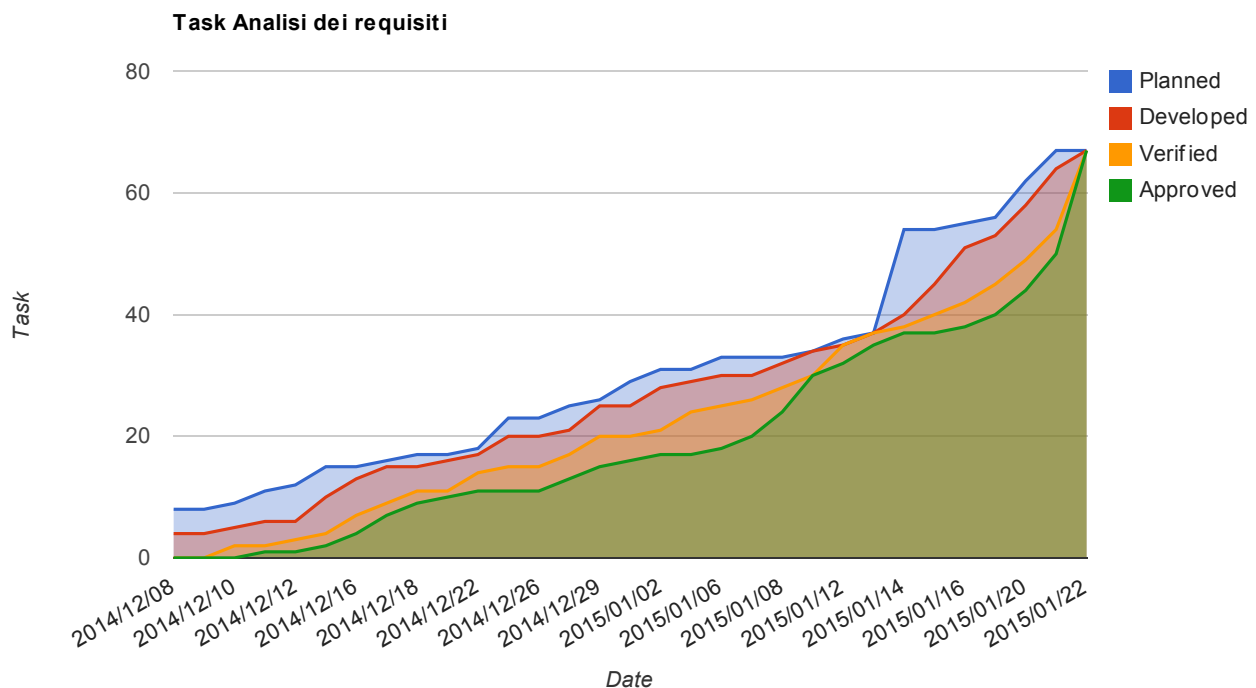


Figura 4: Grafico task Analisi dei requisiti

D.2.2 Documenti

Vengono riportati i valori dell'indice di Gulpease relativi alla fase Analisi dei requisiti. Un documento va considerato accettabile solamente se rientra nelle metriche definite nella sezione.

Nome documento	Valore indice	Esito
<i>Analisi dei Requisiti v1.0.0</i>	56	Superato
<i>Glossario v1.0.0</i>	50	Superato
<i>Norme di Progetto v1.0.0</i>	54	Superato
<i>Piano di Progetto v1.0.0</i>	55	Superato
<i>Piano di Qualifica v1.0.0</i>	55	Superato
<i>Studio di Fattibilità v2.0.0</i>	48	Superato

Tabella 6: Risultati indice Gulpease - Analisi dei requisiti

D.3 Analisi di dettaglio

D.3.1 Processi

Inizialmente sono presenti molti task_G già pianificati questo anche perché la riunione del team è stata fatta nel periodo iniziale della fase. Grazie a tale accorgimento è stato possibile pianificare con anticipo i task e quindi dare maggiore libertà in termini di tempistiche a coloro che dovevano completarli, approvarli e verificarli.

Nel grafico si può notare che c'è un notevole aumento dei task_G Verified e Approved tra il 2015/02/05 e il 2015/02/06, il motivo è che il *Responsabile di Progetto* ha fissato la data di consegna di tutte le parti della presentazione il 2015/02/06. Il gruppo di conseguenza, ha cercato di rispettare al meglio la consegna, questo ha fatto sì che ci fossero le giuste tempistiche per le prove della presentazione nonché l'integrazione con tutte le parti. Si osserva inoltre che non sono stati completati tutti i task pianificati, il motivo è che tutti i membri del gruppo negli ultimi giorni si sono impegnati a perfezionare la presentazione tralasciando in parte il lavoro da svolgere. Il non completamento di tutti i task andrà perciò ad appesantire la fase successiva nella quale bisognerà integrare anche questi task a quelli della fase.

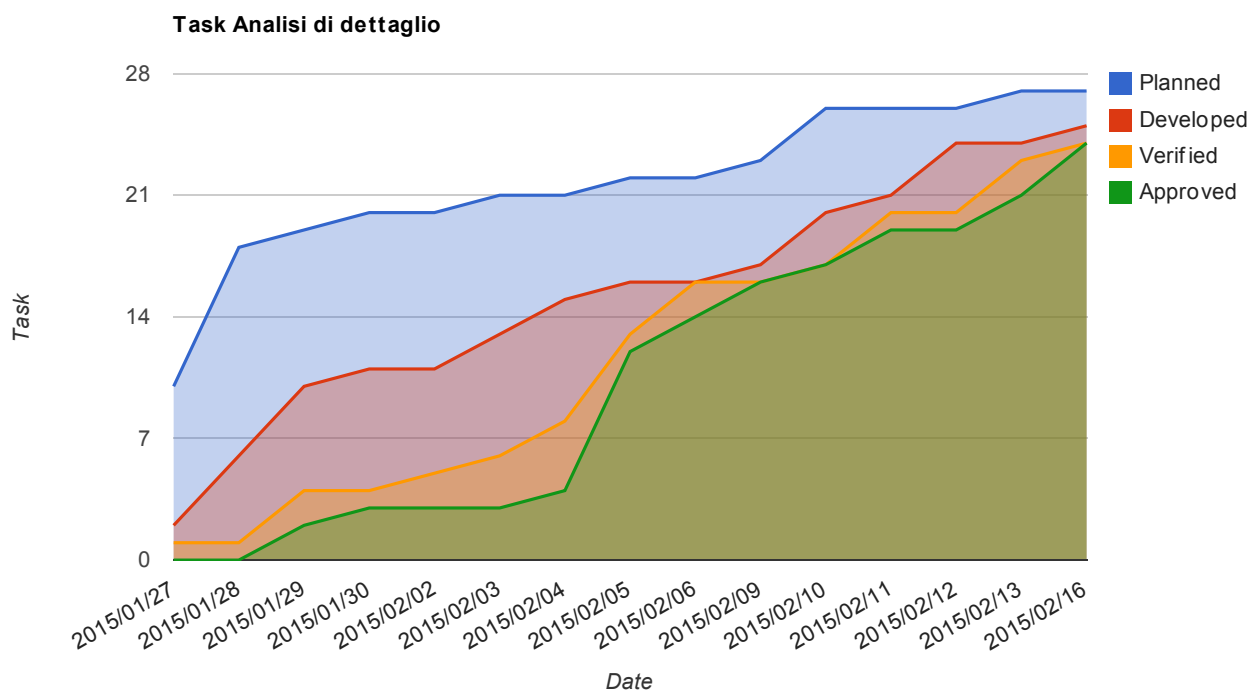


Figura 5: Grafico task Analisi di dettaglio

D.3.2 Documenti

Vengono riportati i valori dell'indice di Gulpease relativi alla fase Analisi di dettaglio. Un documento va considerato accettabile solamente se rientra nelle metriche definite nella sezione.

Nome documento	Valore indice	Esito
<i>Analisi dei Requisiti v2.0.0</i>	57	Superato
<i>Glossario v2.0.0</i>	50	Superato
<i>Norme di Progetto v2.0.0</i>	55	Superato
<i>Piano di Progetto v2.0.0</i>	57	Superato
<i>Piano di Qualifica v1.0.0</i>	59	Superato
<i>Studio di Fattibilità v2.0.0</i>	48	Superato

Tabella 7: Risultati indice Gulpease - Analisi di dettaglio

D.4 Progettazione architeturale

D.4.1 Processi

Visto il successo riscontrato nella fase precedente si è deciso di rendere una best practice la volontà di fissare una riunione nel primo periodo della fase per pianificare il più possibile e aver successivamente più tempo per il completamento, la verifica e l'approvazione. Nonostante quest'accorgimento il gruppo ha sforato la data pianificata di chiusura della fase che era il 2015/03/23, per finire ad approvare gli ultimi task_G il 2015/03/30. Il ritardo è stato dato in parte dal fatto che i membri del gruppo si sono scontrati con un nuovo linguaggio, e con nuovi software, questi hanno avuto bisogno di un apprendimento, attraverso studio e incontri col proponente_G, che è durato più del pianificato. Si è scelto di estendere il grafico fino alla data di effettiva approvazione dell'ultimo task per dar maggior visibilità al way of working.

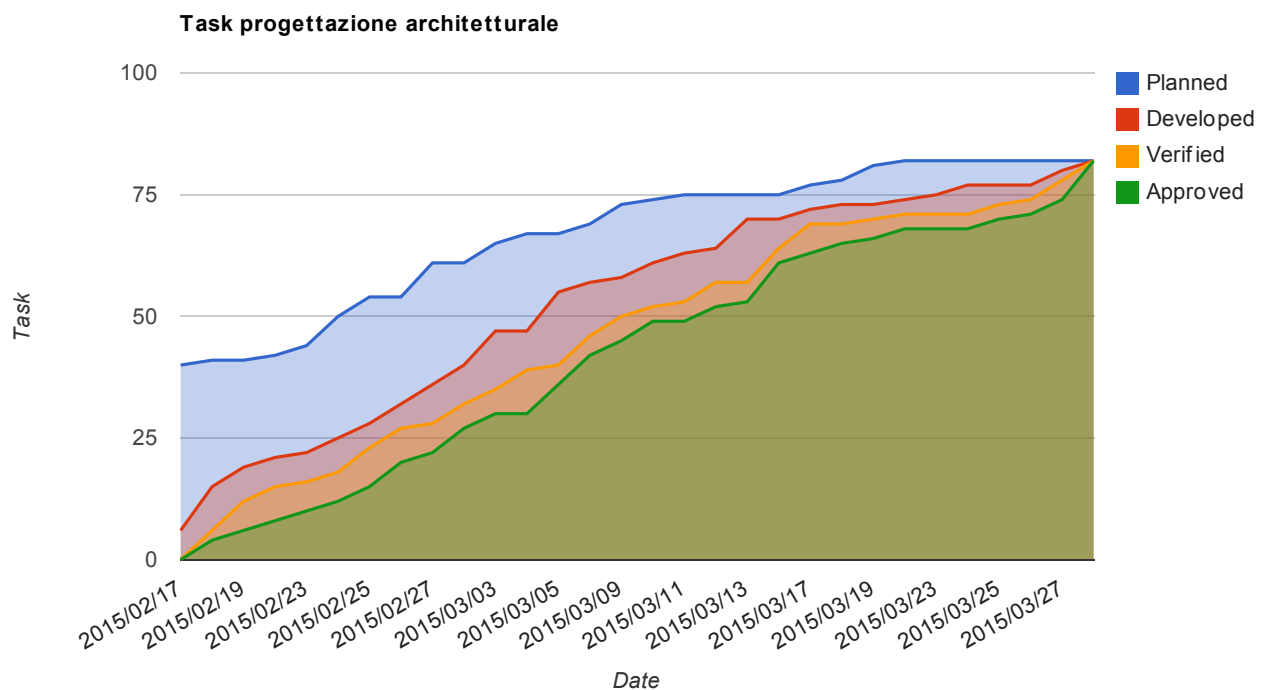


Figura 6: Grafico task Progettazione architeturale

D.4.2 Documenti

Vengono riportati i valori dell'indice di Gulpease relativi alla fase Progettazione architetturale. Un documento va considerato accettabile solamente se rientra nelle metriche definite nella sezione.

Nome documento	Valore indice	Esito
<i>Analisi dei Requisiti v3.0.0</i>	57	Superato
<i>Glossario v3.0.0</i>	51	Superato
<i>Norme di Progetto v3.0.0</i>	55	Superato
<i>Piano di Progetto v3.0.0</i>	57	Superato
<i>Piano di Qualifica v2.0.0</i>	58	Superato
<i>Studio di Fattibilità v2.0.0</i>	48	Superato
<i>Specifica Tecnica v1.0.0</i>	56	Superato

Tabella 8: Risultati indice Gulpease - Progettazione architetturale

D.5 Progettazione di dettaglio e codifica dei requisiti obbligatori

D.5.1 Processi

La best practice usata nelle fasi precedenti è stata applicata anche qui. Nonostante questo tutti i task_G pianificati non sono stati completati entro la data di fine fase, 2015/05/08. Il ritardo è stato causato dalla pianificazione in difetto per quanto riguarda le ore di programmazione ed alcune rivisitazioni dei diagrammi delle classi completati in precedenza. I task che non sono stati *Developed*, *Verified*, *Approved* andranno ad appesantire ulteriormente la fase successiva e chiederanno un impegno maggiore a tutto il gruppo per completarli entro la data fissata dal proponente_G.

D.5.2 Documenti

Vengono riportati i valori dell'indice di Gulpease relativi alla fase Progettazione di dettaglio e codifica dei requisiti obbligatori. Un documento va considerato accettabile solamente se rientra nelle metriche definite nella sezione.

Nome documento	Valore indice	Esito
<i>Analisi dei Requisiti v4.0.0</i>	57	Superato
<i>Definizione di Prodotto v1.0.0</i>	58	Superato
<i>Glossario v3.0.0</i>	51	Superato
<i>Norme di Progetto v4.0.0</i>	55	Superato
<i>Piano di Progetto v4.0.0</i>	59	Superato
<i>Piano di Qualifica v3.0.0</i>	60	Superato
<i>Studio di Fattibilità v2.0.0</i>	48	Superato
<i>Specifica Tecnica v2.0.0</i>	54	Superato
<i>Manuale Amministratore v1.0.0</i>	60	Superato
<i>Manuale Utente v1.0.0</i>	62	Superato

Tabella 9: Risultati indice Gulpease - Progettazione di dettaglio e codifica dei requisiti obbligatori

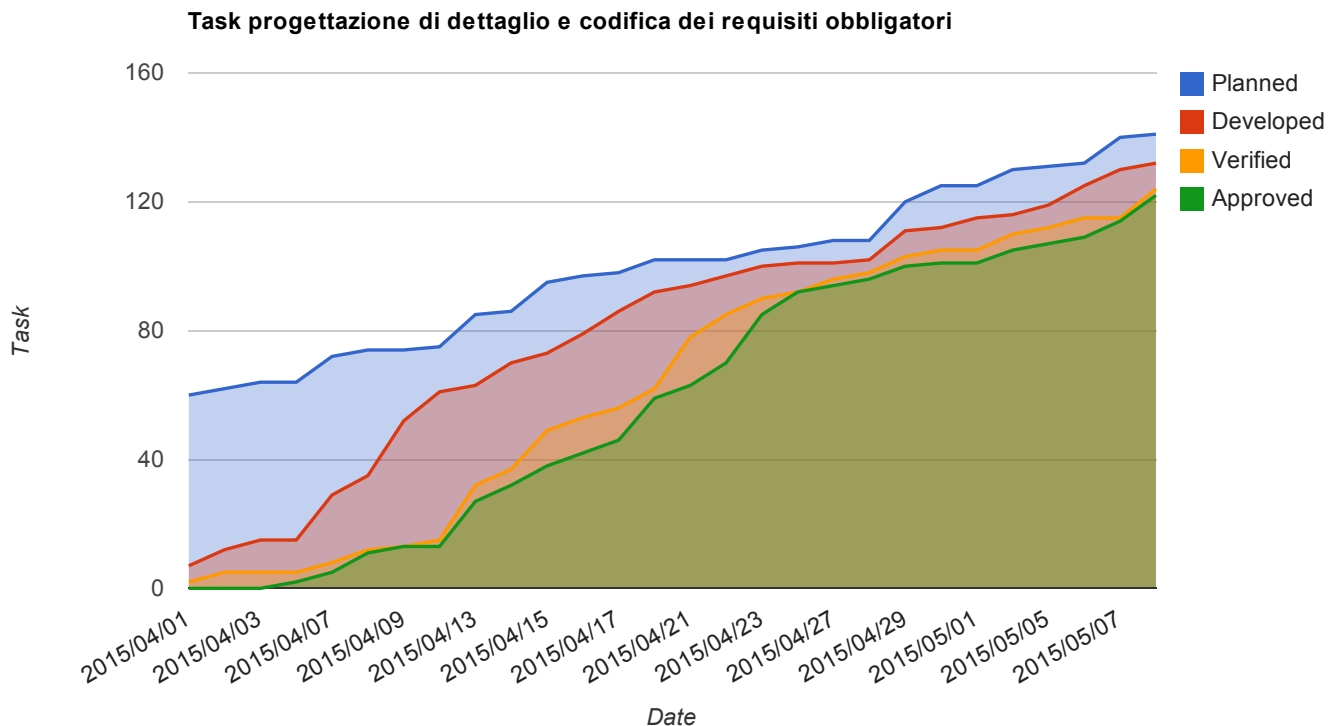


Figura 7: Grafico task Progettazione di dettaglio e codifica dei requisiti obbligatori

D.5.3 Progettazione

Viene riportata di seguito una tabella riassuntiva (tabella 10) che espone gli indici di instabilità per i componenti rilevati durante la progettazione. In questa tabella in oltre viene specificato come i componenti superano questo test, se in modo ottimale, scritta in **verde** o in modo accettabile **giallo**. Le componenti invece che non superano il test sono contrassegnate con la scritta *Non Superato* in **rosso**. Per quanto riguarda le componenti che non hanno superato il test non si pianificano degli obiettivi di miglioramento in quanto il non superamento è causato dalla natura e dalla funzionalità dei componenti stessi.

Componente	Indice instabilità	Esito
src::client::model	0	Superato
src::client::model::data	0.75	Superato
src::client::model::services	0.33	Superato
src::client::view _g	0.5	Superato
src::client::view _g ::public	0.5	Superato
src::client::view _g ::user	0.33	Superato
src::client::view _g ::admin	0.66	Superato
src::client::controller	0.75	Superato
src::client::controller::public	0.75	Superato

src::client::controller::user	0.75	Superato
src::client::controller::admin	0.75	Superato
src::server::db	0	Superato
src::server::db::raw_data	0	Superato
src::server::db::raw_data::fb	0.8	Superato
src::server::db::raw_data::tw	0.75	Superato
src::server::db::raw_data::ig	0.8	Superato
src::server::db::app_data	0	Superato
src::server::db::app_data::user	0.75	Superato
src::server::db::app_data::recipe _g	0	Superato
src::server::processor _g	0.8	Superato
src::server::processor _g ::commands	0.8	Superato
src::server::processor _g ::commands::user	1	Non Superato
src::server::processor _g ::commands::recipe _g	1	Non Superato
src::server::processor _g ::commands::requests	1	Non Superato
src::server::processor _g ::commands::social	1	Non Superato
src::server::miner _g	0.75	Superato
src::server::miner _g ::fb	1	Non Superato
src::server::miner _g ::tw	1	Non Superato
src::server::miner _g ::ig	1	Non Superato
src::server::endpoints	1	Non Superato
src::server::endpoints::api _g	1	Non Superato
src::server::endpoints::api _g ::public	1	Non Superato
src::server::endpoints::api _g ::private	1	Non Superato
src::server::endpoints::resp	0	Superato
src::server::endpoints::resp::public	0	Superato
src::server::endpoints::resp::public::fb	0	Superato
src::server::endpoints::resp::public::tw	0	Superato
src::server::endpoints::resp::public::ig	0	Superato
src::server::endpoints::resp::private	0	Superato

Tabella 10: Indice di instabilità

D.5.4 Metriche codice

Questa sezione traccia i risultati ricavati applicando le metriche relative al codice descritte nella sezione 2.8.3 di questo documento. Si è scelto di mantenere una separazione tra client e backend, in alcune metriche inoltre si specifica sia la media di tale indice che il valore massimo. Se i limiti della metrica_G sono superati in modo ottimale, scritta in **verde**, in modo accettabile **giallo**, le componenti invece che non superano il test sono contrassegnate con la scritta *Non Superato* in **rosso**. Segue ogni metrica, una breve descrizione dove si specifica se il test è stato superato o in caso negativo il motivo presunto per cui non lo è.

Indice di copertura:

- Backend:
 - Branch coverage: 70% **Non Superato**
 - Statements coverage: 75% **Non Superato**
- Client:
 - Branch coverage: 66% **Non Superato**
 - Statements coverage: 71% **Non Superato**

Il risultato ottenuto non supera i limiti di accettazione in quanto il gruppo non è riuscito a codificare tutti i test necessari per coprire meglio il software.

Complessità ciclomatica:

- Backend:
 - medio: 4.1 **Superato**
 - massimo: 8 **Superato**

Questo risultato così basso è stato raggiunto grazie alla divisione delle classi e alle numerose classi astratte che avranno complessità 1 e abbassano così la media.

- Client:
 - medio: 3.5 **Superato**
 - massimo: 7 **Superato**

Numero di parametri per metodo:

- Backend:
 - medio: 5.1 **Superato**
 - massimo: 11 **Non Superato**

In alcuni metodi non si è superato questo tipo di test in quanto la loro funzione era quella di istanziare dei nuovi oggetti nel database_G e quindi necessitavano di tutti i loro parametri per svolgere il compito.

- Client:
 - medio: 1.5 **Superato**
 - massimo: 4 **Superato**

Linee di codice per linee di commento:

- Backend:
 - medio: 0.68 **Superato**
- Client:
 - medio: 0.49 **Superato**

Questo test supera abbondantemente le aspettative, infatti il range ottimale era > 0.35 . Il risultato ottenuto è da implicare all'utilizzo da parte del gruppo delle Google style guide.

Numero di livelli di annidamento:

- Backend:
 - medio: 3.4 **Superato**
 - massimo: 5 **Superato**
- Client:
 - medio: 2.3 **Superato**
 - massimo: 4 **Superato**

Logical SLOC:

- Totale:
 - Backend: 2711
 - Client: 4015
- Parziale:
 - Backend:
 - * medio: 20.8 **Superato**
 - * massimo: 122 **Non Superato**
 - Client:
 - * medio: 35.5 **Superato**
 - * massimo: 115 **Non Superato**

In questo caso alcuni moduli, soprattutto del miner_G , e nella cancellazione delle recipe_G , non rientrano nel range di accettazione, questo è dovuto a causa dell'elevata quantità di attributi che questi devono istanziare o cancellare dal database_G .

Alcuni moduli non hanno superato questo test in quanto riguardano le funzioni principali dell'applicativo. Questo pur essendo oggetto di un futuro miglioramento, non preoccupa particolarmente in quanto la loro afferenza è molto elevata.

D.6 Progettazione di dettaglio e codifica dei requisiti desiderabili

D.6.1 Processi

Questa fase è stata appesantita dai ritardi accumulati nella fase precedente. Per questo motivo nel grafico non sono stati riportati i task_G pianificati nella fase antecedente, mentre i task che non erano stati *Developed*, *Verified*, *Approved*, sono stati completati qui, togliendo tempo ai task pianificati in questa fase. Il risultato è che i task per i requisiti obbligatori sono stati completati, mentre parte di quelli per i requisiti desiderabili non sono stati sviluppati. Questo andrà ad aumentare il lavoro della fase successiva, con la possibilità di non terminare i requisiti opzionali.

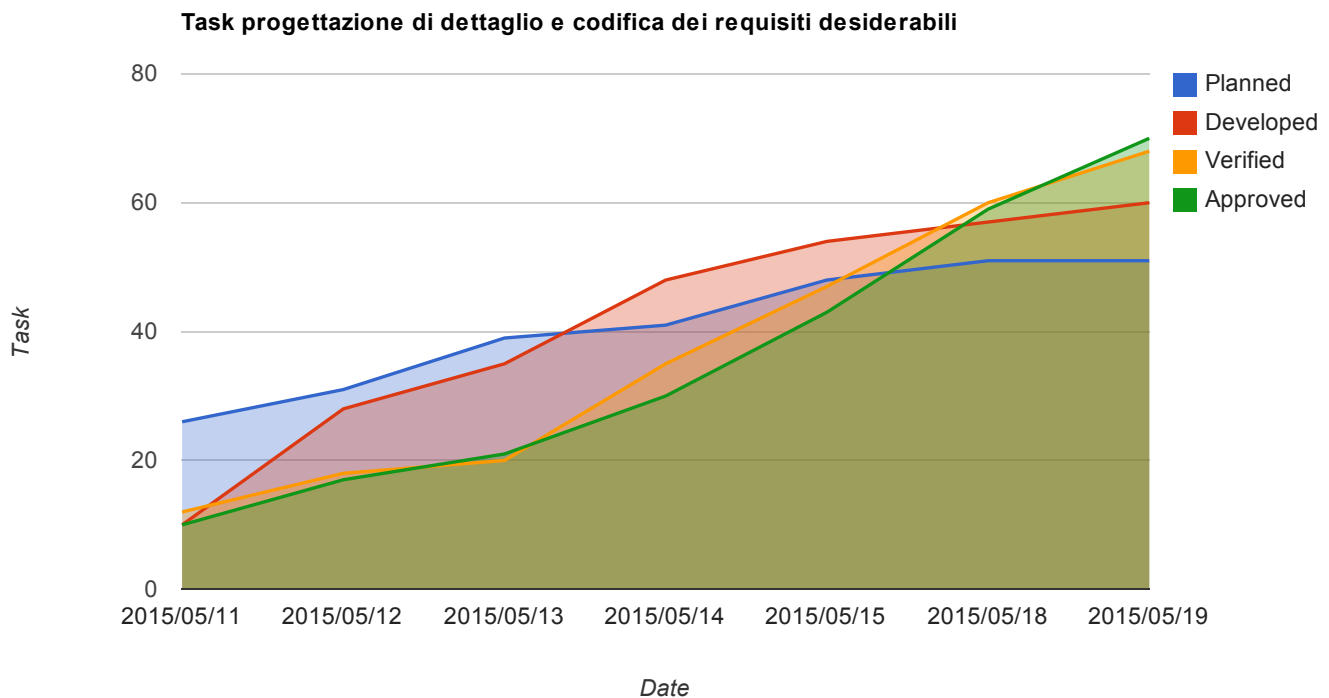


Figura 8: Grafico task Progettazione di dettaglio e codifica dei requisiti desiderabili

D.6.2 Documenti

Vengono riportati i valori dell'indice di Gulpease relativi alla fase Progettazione di dettaglio e codifica dei requisiti desiderabili. Un documento va considerato accettabile solamente se rientra nelle metriche definite nella sezione.

Nome documento	Valore indice	Esito
<i>Analisi dei Requisiti v4.0.0</i>	57	Superato
<i>Definizione di Prodotto v2.0.0</i>	58	Superato
<i>Glossario v3.0.0</i>	51	Superato
<i>Norme di Progetto v5.0.0</i>	55	Superato
<i>Piano di Progetto v5.0.0</i>	59	Superato
<i>Piano di Qualifica v4.0.0</i>	60	Superato
<i>Studio di Fattibilità v2.0.0</i>	48	Superato
<i>Specifica Tecnica v2.0.0</i>	54	Superato
<i>Manuale Amministratore v1.0.0</i>	60	Superato
<i>Manuale Utente v1.0.0</i>	62	Superato

Tabella 11: Risultati indice Gulpease - Progettazione di dettaglio e codifica dei requisiti desiderabili

D.7 Consolidamento della progettazione e codifica dei requisiti desiderabili

D.7.1 Processi

La fase analizzata include la consegna della revisione di progettazione e la relativa presentazione. Si può notare come fino al 27/05, data della consegna, i task_G sono

stati *Developed*, *Verified*, *Approved* con una certa rapidità e frequenza. Dopo la data questa invece si verifica un notevole calo di tutti gli stati, con la conseguenza che non tutti i task pianificati sono stati poi portati allo stato finale. Il ritardo è dovuto alla presenza del primo appello di ingegneria negli ultimi giorni della fase. Questo ha impegnato tutti i membri del gruppo e quindi sottratto tempo per le altre attività.

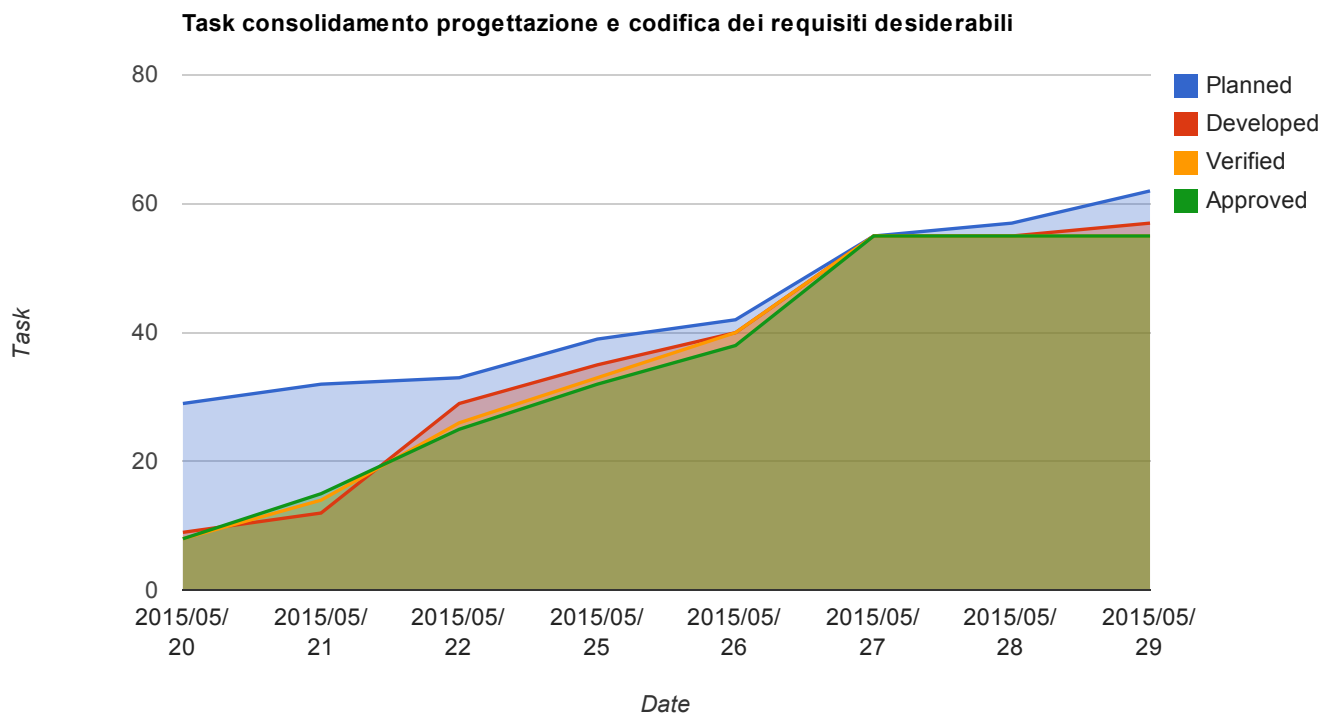


Figura 9: Grafico task Consolidamento della progettazione e codifica dei requisiti desiderabili

D.7.2 Documenti

Vengono riportati i valori dell'indice di Gulpease relativi alla fase Consolidamento della progettazione e codifica dei requisiti desiderabili. Un documento va considerato accettabile solamente se rientra nelle metriche definite nella sezione.

Nome documento	Valore indice	Esito
<i>Analisi dei Requisiti v4.0.0</i>	57	Superato
<i>Definizione di Prodotto v3.0.0</i>	59	Superato
<i>Glossario v3.0.0</i>	51	Superato
<i>Norme di Progetto v5.0.0</i>	55	Superato
<i>Piano di Progetto v6.0.0</i>	60	Superato
<i>Piano di Qualifica v5.0.0</i>	62	Superato
<i>Studio di Fattibilità v2.0.0</i>	48	Superato
<i>Specifica Tecnica v2.0.0</i>	54	Superato
<i>Manuale Amministratore v1.0.0</i>	60	Superato
<i>Manuale Utente v1.0.0</i>	62	Superato

Tabella 12: Risultati indice Gulpease - Consolidamento della progettazione e codifica dei requisiti desiderabili (TODO cambiare versione documenti)

D.8 Validazione

D.8.1 Processi

In questa ultima fase il gruppo ha cercato di focalizzarsi soprattutto nel primo periodo nel completare più velocemente possibile tutti i task_G pianificati, questo per non ricadere nel ritardo verificato nella fase precedente a causa di attività esterne al progetto. Inizialmente vediamo che si sono recuperati i task non sviluppati, poi si può notare un picco dei task pianificati al 08/06 che coincide con i risultati della revisione di qualifica. Grazie ai suggerimenti da parte del committente_G il gruppo ha pianificato le correzioni e le aggiunte da fare per poter partecipare alla revisione di accettazione.

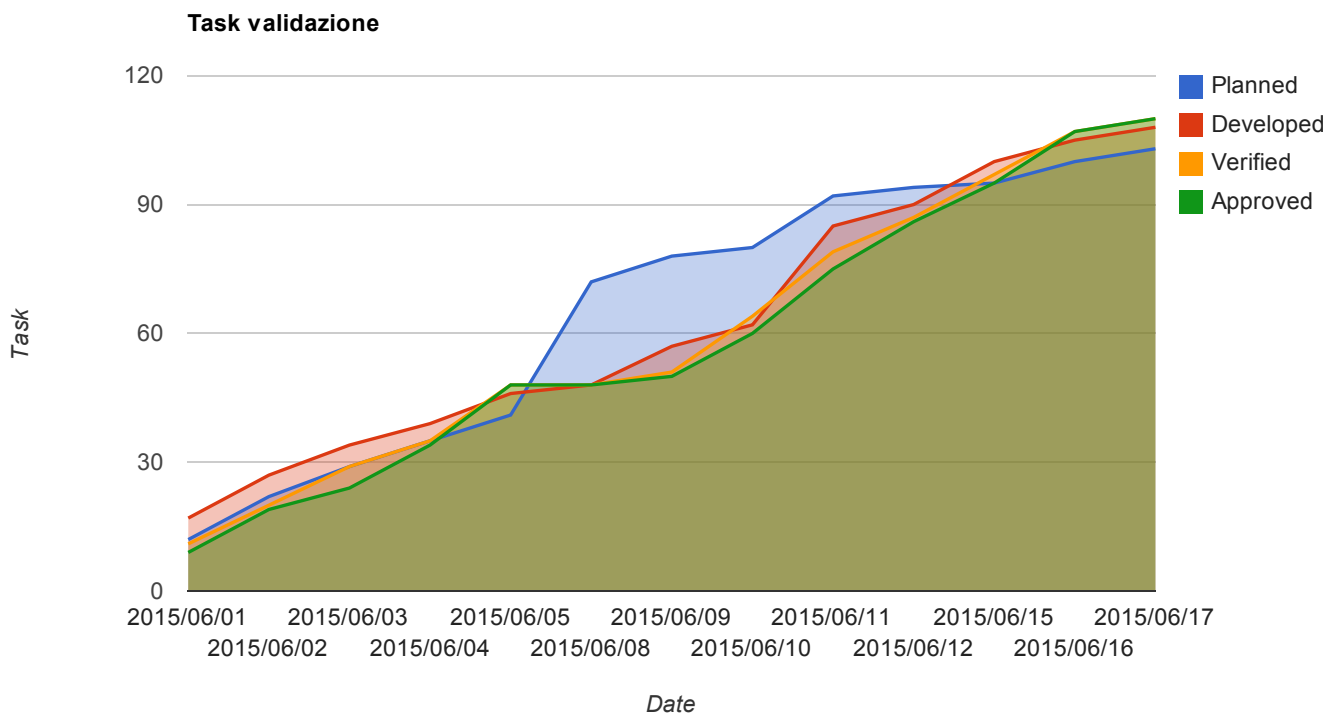


Figura 10: Grafico task Validazione

D.8.2 Documenti

Vengono riportati i valori dell'indice di Gulpease relativi alla fase Validazione_G. Un documento va considerato accettabile solamente se rientra nelle metriche definite nella sezione.

D.8.3 Progettazione

Viene riportata di seguito una tabella riassuntiva (tabella 10) che espone gli indici di instabilità per i componenti rilevati durante la progettazione. In questa tabella in oltre viene specificato come i componenti superano questo test, se in modo ottimale, scritta in verde o in modo accettabile giallo. Le componenti invece che non superano il test sono contrassegnate con la scritta *Non Superato* in rosso. Per quanto riguarda le componenti che non hanno superato il test non si pianificano degli obiettivi di miglioramento in quanto il non superamento è causato dalla natura e dalla funzionalità dei componenti stessi.

Nome documento	Valore indice	Esito
<i>Analisi dei Requisiti v6.0.0</i>	57	Superato
<i>Definizione di Prodotto v4.0.0</i>	52	Superato
<i>Glossario v4.0.0</i>	55	Superato
<i>Norme di Progetto v6.0.0</i>	59	Superato
<i>Piano di Progetto v7.0.0</i>	61	Superato
<i>Piano di Qualifica v6.0.0</i>	60	Superato
<i>Studio di Fattibilità v2.0.0</i>	48	Superato
<i>Specifica Tecnica v3.0.0</i>	54	Superato
<i>Manuale Amministratore v3.0.0</i>	61	Superato
<i>Manuale Utente v3.0.0</i>	60	Superato

Tabella 13: Risultati indice Gulpease - Validazione

Componente	Indice instabilità	Esito
<code>src::client::model</code>	0	Superato
<code>src::client::model::data</code>	0.75	Superato
<code>src::client::model::services</code>	0.33	Superato
<code>src::client::view_g</code>	0.5	Superato
<code>src::client::view_g::public</code>	0.5	Superato
<code>src::client::view_g::user</code>	0.33	Superato
<code>src::client::view_g::admin</code>	0.66	Superato
<code>src::client::controller</code>	0.75	Superato
<code>src::client::controller::public</code>	0.75	Superato
<code>src::client::controller::user</code>	0.75	Superato
<code>src::client::controller::admin</code>	0.75	Superato
<code>src::server::db</code>	0	Superato
<code>src::server::db::raw_data</code>	0	Superato
<code>src::server::db::raw_data::fb</code>	0.8	Superato
<code>src::server::db::raw_data::tw</code>	0.75	Superato
<code>src::server::db::raw_data::ig</code>	0.8	Superato
<code>src::server::db::app_data</code>	0	Superato
<code>src::server::db::app_data::user</code>	0.75	Superato
<code>src::server::db::app_data::recipe_g</code>	0	Superato
<code>src::server::processor_g</code>	0.8	Superato
<code>src::server::processor_g::commands</code>	0.8	Superato

src::server::processor _G ::commands::user	1	Non Superato
src::server::processor _G ::commands::recipe _G	1	Non Superato
src::server::processor _G ::commands::requests	1	Non Superato
src::server::processor _G ::commands::social	1	Non Superato
src::server::miner _G	0.75	Superato
src::server::miner _G ::fb	1	Non Superato
src::server::miner _G ::tw	1	Non Superato
src::server::miner _G ::ig	1	Non Superato
src::server::endpoints	1	Non Superato
src::server::endpoints::api _G	1	Non Superato
src::server::endpoints::api _G ::public	1	Non Superato
src::server::endpoints::api _G ::private	1	Non Superato
src::server::endpoints::resp	0	Superato
src::server::endpoints::resp::public	0	Superato
src::server::endpoints::resp::public::fb	0	Superato
src::server::endpoints::resp::public::tw	0	Superato
src::server::endpoints::resp::public::ig	0	Superato
src::server::endpoints::resp::private	0	Superato

Tabella 14: Indice di instabilità

D.8.4 Metriche codice

Questa sezione traccia i risultati ricavati applicando le metriche relative al codice descritte nella sezione 2.8.3 di questo documento. Si è scelto di mantenere una separazione tra client e backend, in alcune metriche inoltre si specifica sia la media di tale indice che il valore massimo. Segue ogni metrica_G, una breve descrizione dove si specifica gli eventuali problemi che causano il non superamento del test. Se i limiti della metrica sono superati in modo ottimale, scritta in **verde**, in modo accettabile **giallo**, le componenti invece che non superano il test sono contrassegnate con la scritta *Non Superato* in **rosso**.

Indice di copertura:

- Backend:
 - Branch coverage: 84% **Superato**
 - Statements coverage: 91% **Superato**
- Client:

- Branch coverage: 82% **Superato**
- Statements coverage: 89% **Superato**

Grazie alla riprogrammazione delle fasi e alla non codifica dei requisiti opzionali è stato possibile raggiungere un risultato che supera i limiti imposti.

Complessità ciclomatica:

- Backend:
 - medio: 4.8 **Superato**
 - massimo: 9 **Superato**

Questo risultato così basso è stato raggiunto grazie alla divisione delle classi e alle numerose classi astratte che avranno complessità 1 e abbassano così la media.

- Client:
 - medio: 3.7 **Superato**
 - massimo: 8 **Superato**

Numero di parametri per metodo:

- Backend:
 - medio: 5.6 **Superato**
 - massimo: 11 **Non Superato**

In alcuni metodi non si è superato questo tipo di test in quanto la loro funzione era quella di istanziare dei nuovi oggetti nel database e quindi necessitavano di tutti i loro parametri per svolgere il compito.

- Client:
 - medio: 1.7 **Superato**
 - massimo: 4 **Superato**

Linee di codice per linee di commento:

- Backend:
 - medio: 0.54 **Superato**
- Client:
 - medio: 0.38 **Superato**

Numero di livelli di annidamento:

- Backend:
 - medio: 3.4 **Superato**
 - massimo: 5 **Superato**
- Client:
 - medio: 2.3 **Superato**
 - massimo: 4 **Superato**

Logical SLOC:

- Totale:
 - Backend: 4823
 - Client: 8971
- Parziale:
 - Backend:
 - * medio: 32.5 **Superato**
 - * massimo: 122 **Non Superato**

In questo caso alcuni moduli, soprattutto del miner_G , e nella cancellazione delle recipe_G , non rientrano nel range di accettazione, questo è dovuto a causa dell'elevata quantità di attributi che questi devono istanziare o cancellare dal database_G .

- Client:
 - * medio: 35.5 **Superato**
 - * massimo: 300 **Superato**

Alcuni moduli non hanno superato questo test in quanto riguardano le funzioni principali dell'applicativo. Questi file non preoccupano particolarmente in quanto la loro afferenza è molto elevata. Quindi al fine di un eventuale processo di manutenzione, la manutenzione di queste funzionalità non comporterà il cambiamento di altri file.

E Esito delle revisioni

In questa sezione verranno riportate un elenco delle modifiche per ogni documento apportate dal gruppo a seguito delle valutazioni fatte dal committente_G alle revisioni. Viene qui sotto riportata una lista con tutte le modifiche effettuate divise per ogni revisione a cui il gruppo ha preso parte.

E.1 Revisione dei Requisiti

- **Analisi dei Requisiti:** a seguito delle segnalazioni, è stato rivisto completamente il documento. Sono stati rivisti completamente i casi d'uso e i requisiti. Il caso d'uso principale si è deciso di dividerlo in due grandi casi d'uso in modo da dividere i casi d'uso relativi ai servizi REST_G dal resto del sistema;
- **Norme di Progetto:** in base alle segnalazioni ricevute sono stati aggiunti diagrammi nella sezione relativi ai processi primari e sono state fornite le regole e le procedure di rotazione dei ruoli;
- **Piano di Progetto:** in base alle segnalazioni ricevute si sono rivisti completamente i consuntivi e preventivi, specialmente è stato rimossa la consuntivazione relativa alla prima parte. È stato corretto l'utilizzo del termine "fase" laddove utilizzato come sinonimo di "attività";
- **Piano di Qualifica:** a seguito delle segnalazioni è stato rivisto tutto il documento. Sono state rielaborate le strategie di perseguimento di qualità e migliorate le appendici relative agli standard di qualità. Infine, si è migliorata l'integrazione dei contenuti del documento con quelli delle *Norme di Progetto*.

E.2 Revisione di Progettazione

- **Analisi dei Requisiti:** a seguito delle segnalazioni, sono stati rivisti i diagrammi di attività presenti nel documento e eliminato un requisito già specificato precedentemente;
- **Specifica Tecnica:** in base alle segnalazioni ricevute sono stati rivisti alcuni design pattern_G, si è descritto il modo in cui i framework_G e le librerie utilizzate si integrano nell'architettura del prodotto e sono stati corretti alcuni errori lessicali e di significato.
Nella parte del backend sono state riviste le associazioni di composizione, è stato inoltre definito il tipo *Dictionary* usato nelle classi *Command*;
- **Piano di Progetto:** come da indicazione, sono state correlate l'analisi dei rischi e la pianificazione, è stata, inoltre, rivista in parte l'organizzazione del documento. Incrementato il contenuto del preventivo a finire e corretto il contenuto del consuntivo relativo alla revisione;
- **Piano di Qualifica:** in base alle indicazioni ricevute, è stata arricchita la sezione 3 e corretta la presentazione del PDCA_G.

E.3 Revisione di Qualifica

- **Analisi dei Requisiti:** a seguito delle segnalazioni sono stati corretti i diagrammi di attività che presentavano errori;

- **Specifica Tecnica:** in base alle segnalazioni è stato corretto l'utilizzo errato del termine design pattern_G per il MVW;
- **Definizione di Prodotto:** in base alle segnalazioni sono state corretti vari paragrafi che risultavano poco chiari; è stato quindi descritto in che modo viene generato il token d'accesso e come viene valorizzato il dizionario che associa le chiamate ai relativi comandi;
- **Manuali:** come da indicazione, sono stati sostituiti gli screenshot con altri di nuovi in cui vengono sottolineati meglio i passaggi fondamentali;
- **Piano di Progetto:** come da indicazione, è stata spostata la descrizione del ciclo di vita all'interno della pianificazione e sono stati corretti dei termini utilizzati erroneamente;
- **Piano di Qualifica:** in base alle indicazioni ricevute, è stata corretta la presentazione del ciclo PDCA_G.