



Piano di Qualifica

Jawa Druids

Versione	3.0.0
Data approvazione	2021-03-15
Responsabile	Andrea Cecchin
Redattori	Emma Roveroni Mattia Cocco Margherita Mitillo
Verificatori	Igli Mezini Andrea Cecchin Emma Roveroni
Stato	Approvato
Lista distribuzione	Jawa Druids Prof. Tullio Vardanega Prof. Riccardo Cardin Sync Lab
Uso	Esterno

Sommario

Il documento contiene le strategie di verifica e validazione seguite dal gruppo *Jawa Druids* durante lo svolgimento del progetto *Gathering Detection Platform*.



Registro delle modifiche

Versione	Data	Autore	Ruolo	Modifica	Verificatore
v2.0.7	2021-04-29	Andrea Dorigo	<i>Analista</i>	<i>Conclusa ste-sura § 5.3.2.2 e § 3.4</i>	Andrea Cecchin
v2.0.6	2021-04-23	Margherita Mitillo	<i>Analista</i>	<i>Conclusa ste-sura § 5.3.1 e § 4.2</i>	Andrea Cecchin
v2.0.5	2021-04-20	Emma Rove-roni	<i>Analista</i>	<i>Iniziata ste-sura § 5.3</i>	Andrea Cecchin
v2.0.4	2021-04-19	Emma Rove-roni	<i>Analista</i>	<i>Aggiunta § 4.2 e § 4.2.1</i>	Andrea Cecchin
v2.0.3	2021-04-19	Margherita Mitillo	<i>Analista</i>	<i>Aggiornato capitolo 6</i>	Igli Mezini
v2.0.2	2021-03-20	Emma Rove-roni	<i>Analista</i>	<i>Aggiunta § 4.3</i>	Andrea Cecchin
v2.0.1	2021-03-19	Margherita Mitillo	<i>Analista</i>	<i>Rielaborato capitolo 2 e 6</i>	Igli Mezini
v2.0.0	2021-03-15	Andrea Cecchin	<i>Responsabile</i>	<i>Approvazione del documen-to per RP</i>	-
v1.1.3	2021-03-10	Margherita Mitillo	<i>Analista</i>	<i>Aggiunto ca-pitolo 7</i>	Andrea Cecchin
v1.1.2	2021-03-09	Mattia Cocco	<i>Analista</i>	<i>Aggiunta § 4.4</i>	Andrea Cecchin
v1.1.1	2021-03-09	Emma Rove-roni	<i>Analista</i>	<i>Inizio ste-sura § 5.2 e § 3.6.1.2</i>	Igli Mezini



v1.1.0	2021-03-08	-	-	<i>Revisione complessiva del documento</i>	Emma Roveroni
v1.0.1	2021-02-25	Mattia Cocco	<i>Analista</i>	<i>Rielaborazione capitolo 2</i>	Igli Mezini
v1.0.0	2021-01-09	Emma Roveroni	<i>Responsabile di Progetto</i>	<i>Approvazione del documento per la RR</i>	-
v0.2.0	2021-01-08	-	-	<i>Verifica del documento</i>	Igli Mezini
v0.1.1	2021-01-08	Alfredo Graziano	<i>Analista</i>	<i>Aggiunta § 6.2</i>	-
v0.1.0	2021-01-07	-	-	<i>Verifica delle sezioni redatte finora</i>	Igli Mezini
v0.0.8	2021-01-04	Alfredo Graziano	<i>Analista</i>	<i>Aggiunta § 3.5.1</i>	-
v0.0.7	2021-01-04	Emma Roveroni	<i>Analista</i>	<i>Aggiunto capitolo 5, 6</i>	-
v0.0.6	2021-01-04	Mattia Cocco	<i>Analista</i>	<i>Aggiunta § 3.6.1, e capitolo 4.1</i>	-
v0.0.5	2021-01-03	Mattia Cocco	<i>Analista</i>	<i>Aggiunto capitolo 4</i>	-
v0.0.4	2021-01-03	Alfredo Graziano	<i>Analista</i>	<i>Aggiunto capitolo 2</i>	-
v0.0.3	2021-01-03	Emma Roveroni	<i>Analista</i>	<i>Aggiunto capitolo 3</i>	-
v0.0.2	2020-12-07	Alfredo Graziano	<i>Analista</i>	<i>Aggiunto capitolo 1</i>	-



v0.0.1	2020-11-26	Alfredo Graziano	<i>Analista</i>	<i>Prima stesura del documento</i>	-
--------	------------	------------------	-----------------	------------------------------------	---



Indice

1	Introduzione	9
1.1	Scopo del documento	9
1.2	Scopo del prodotto	9
1.3	Glossario	9
1.4	Riferimenti	10
1.4.1	Riferimenti normativi	10
1.4.2	Riferimenti informativi	10
2	Qualità di processo	11
2.1	Processi di supporto	11
2.1.1	Pianificazione	11
2.1.1.1	Metriche	11
2.2	Processi di sviluppo	13
2.2.1	Analisi dei requisiti	13
2.2.1.1	Metriche	13
3	Qualità del prodotto	14
3.1	Funzionalità	14
3.1.1	Metriche	14
3.1.1.1	MQPD01 Totalità dell'implementazione	14
3.2	Affidabilità	15
3.2.1	Metriche	15
3.2.1.1	MQPD03 Rilevamento Errori	15
3.3	Usabilità	15
3.3.1	Metriche	15
3.3.1.1	MQPD04 Validità dei dati in input	15
3.3.1.2	Indice di Gulpease	16
3.3.1.3	Errori Ortografici	16
3.4	Efficienza	16
3.4.1	Metriche	17
3.4.1.1	MQPD02 Efficienza raggiunta	17
3.5	Portabilità	17
3.5.1	Valutazione sulla Caratteristica	17
3.6	Manutenibilità	18
3.6.1	Metriche	18
3.6.1.1	MQPD05 Comprensione del codice	18
3.6.1.2	MQPD06 Structural fan-in	18



4	Specifica dei test	19
4.1	Tipi di test	20
4.2	Test di unità	21
4.2.1	Tracciamento dei test	23
4.3	Test di integrazione	26
4.4	Test di sistema	27
4.4.1	Tracciamento dei test	30
4.5	Test di accettazione	32
5	Resoconto attività di verifica	33
5.1	Revisione dei Requisiti	33
5.1.1	Strategia adoperata per l'analisi statica dei documenti	33
5.1.2	Esiti verifica	33
5.2	Revisione di Progettazione	36
5.2.1	Verifiche di processo	37
5.2.2	Verifiche di prodotto	37
5.2.2.1	Strategia adoperata per l'analisi statica dei documenti	37
5.2.2.1.1	Esiti Verifica	38
5.3	Revisione di qualifica	41
5.3.1	Verifiche di processo	41
5.3.1.1	MQPS05	41
5.3.1.2	PROI	41
5.3.2	Verifiche di prodotto	41
5.3.2.1	MQPD01	41
5.3.2.2	MQPD02	42
5.3.2.3	MQPD03	42
5.3.2.4	MQPD04	43
5.3.2.5	MQPD05	44
5.3.2.6	MQPD06	44
5.3.2.7	Strategia adoperata per l'analisi statica dei documenti	45
5.3.2.7.1	Esiti verifica	45
6	Valutazioni per il miglioramento	47
6.1	Valutazione su organizzazione	48
6.1.1	Revisione dei requisiti	48
6.1.2	Revisione di progettazione	48
6.1.3	Revisione di qualifica	49
6.2	Valutazione sui ruoli	49
6.2.1	Revisione dei requisiti	49
6.2.2	Revisione di progettazione	51
6.2.3	Revisione di qualifica	52



6.3	Valutazione su strumenti di lavoro	52
6.3.0.1	Revisione dei requisiti	52
6.3.0.2	Revisione dei progettazione	53
6.3.0.3	Revisione di qualifica	53
7	Esiti delle revisioni	56
7.1	Revisione dei requisiti	56
7.2	Revisione di progettazione	56



Elenco delle figure

5.1	Andamento Indice di Gulpease _g	Analisi dei Requisiti fino a RR	34
5.2	Andamento Indice di Gulpease _g	Norme di Progetto fino a RR	34
5.3	Andamento Indice di Gulpease _g	Piano di Qualifica fino a RR	35
5.4	Andamento Indice di Gulpease _g	Piano di Progetto fino a RR	35
5.5	Andamento Indice di Gulpease _g	Studio di Fattibilità fino a RR	35
5.6	MQPS05		37
5.7	Andamento Indice di Gulpease _g	Analisi dei Requisiti fino a RP	38
5.8	Andamento Indice di Gulpease _g	Norme di Progetto fino a RP	39
5.9	Andamento Indice di Gulpease _g	Piano di Qualifica fino a RP	39
5.10	Andamento Indice di Gulpease _g	Piano di Progetto fino a RP	40



Elenco delle tabelle

4.1	Elenco test di unità - Python	21
4.2	Elenco test di unità - Spring	22
4.3	Elenco test di unità - Vue.js	23
4.4	Tracciamento dei test di unità - Python	23
4.5	Tracciamento dei test di unità - Spring	24
4.6	Tracciamento dei test di unità - Vue.js	26
4.7	Elenco test di accettazione	26
4.8	Elenco test di sistema	30
4.9	Tracciamento dei test di sistema con i requisiti	32
5.1	Elenco Indici di Gulpease _g dei documenti fino alla RR	33
5.2	Elenco Indici di Gulpease _g dei verbali fino alla RR	36
5.3	Elenco Indici di Gulpease _g dei documenti fino alla RP	38
5.4	Elenco Indici di Gulpease _g dei verbali fino alla RP	40
5.6	PROI	41
5.8	MQPD01	42
5.10	MQPD02	42
5.12	Tabella dei test di unità effettuati	43
5.14	MQPD03	43
5.16	MQPD05	44
5.18	MQPD06	44
5.19	Elenco Indici di Gulpease _g dei documenti fino alla RQ	45
5.20	Elenco Indici di Gulpease _g dei verbali fino alla RQ	46
6.1	Tabella dei problemi relativi all'organizzazione - Revisione dei requisiti	48
6.2	Tabella dei problemi relativi all'organizzazione - Revisione di progettazione	48
6.3	Tabella dei problemi relativi all'organizzazione - Revisione di Qualifica	49
6.4	Tabella dei problemi relativi ai ruoli - Revisione dei requisiti	50
6.5	Tabella dei problemi relativi ai ruoli - Revisione di progettazione	51
6.6	Tabella dei problemi relativi ai ruoli - Revisione di qualifica	52
6.7	Tabella dei problemi relativi agli strumenti di lavoro - Revisione dei requisiti	53
6.8	Tabella dei problemi relativi agli strumenti di lavoro - Revisione di progettazione	53
6.10	Tabella dei problemi relativi agli strumenti di lavoro - Revisione di qualifica	55



1 Introduzione

1.1 Scopo del documento

Il Piano di Qualifica è un documento su cui si prevede di operare per l'intera durata del progetto e il cui scopo è presentare e descrivere le strategie di verifica e validazione adottate dal gruppo *Jawa Druids* al fine di garantire la qualità di prodotto e di processo. Per raggiungere questo obbiettivo viene applicato un sistema di verifica continua sui processi in corso e sulle attività_c svolte, in modo da rilevare e correggere subito eventuali anomalie, riducendo lo spreco di risorse ed il rischio di reiterare gli stessi errori.

1.2 Scopo del prodotto

In seguito alla pandemia del virus COVID-19 è nata l'esigenza di limitare il più possibile i contatti fra le persone, specialmente evitando la formazione di assembramenti. Il progetto *GDP: Gathering Detection Platform* di *Sync Lab* ha pertanto l'obiettivo di **creare una piattaforma in grado di rappresentare graficamente le zone potenzialmente a rischio di assembramento, al fine di prevenirlo**. Il prodotto finale è rivolto specificatamente agli organi amministrativi delle singole città, cosicché possano gestire al meglio i punti sensibili di affollamento, come piazze o siti turistici. Lo scopo che il software intende raggiungere non è solo quello della rappresentazione grafica real-time ma anche di poter riuscire a prevedere assembramenti in intervalli futuri di tempo.

A tal fine il gruppo *Jawa Druids* si prefigge di sviluppare un prototipo software in grado di acquisire, monitorare ed analizzare i molteplici dati provenienti dai diversi sistemi e dispositivi, a scopo di identificare i possibili eventi che concorrono all'insorgere di variazioni di flussi di utenti. Il gruppo prevede inoltre lo sviluppo di un'applicazione web da interporre fra i dati elaborati e l'utente, per favorirne la consultazione.

1.3 Glossario

All'interno della documentazione viene fornito un *Glossario*, con l'obiettivo di assistere il lettore specificando il significato e contesto d'utilizzo di alcuni termini strettamente tecnici o ambigui, segnalati con una *G* a pedice.



1.4 Riferimenti

1.4.1 Riferimenti normativi

- *Norme di Progetto 3.0.0.*

1.4.2 Riferimenti informativi

- *Qualità di processo:*
<https://www.math.unipd.it/~tullio/IS-1/2020/Dispense/L13.pdf>
- *Qualità di prodotto:*
<https://www.math.unipd.it/~tullio/IS-1/2020/Dispense/L12.pdf>
- *Verifica e validazione: introduzione:*
<https://www.math.unipd.it/~tullio/IS-1/2020/Dispense/L14.pdf>
- *Indice di Gulpease:*
https://it.wikipedia.org/wiki/Indice_Gulpease
- *IEEE Recommended Practice for Software Requirements Specifications:*
<https://ieeexplore.ieee.org/document/720574>
- *Validating the ISO/IEC 15504 measure of software requirements analysis process capability:*
<https://ieeexplore.ieee.org/document/852742>
- *Seminario per approfondimenti tecnici del capitolato C3:*
<https://www.math.unipd.it/~tullio/IS-1/2020/Progetto/ST1.pdf>
- *Standard ISO/IEC 9126:*
http://www.colonese.it/00-Manuali_Pubblicatii/07-ISO-IEC9126_v2.pdf
- *Metrica Comprensione del Codice:*
<https://www.aivosto.com/project/help/pm-loc.html>
- *Metriche per valutazione della Qualità di Prodotto:*
https://www.tricentis.com/blog/64-essential-testing-metrics-for-measuring-quality-assurance-success/?utm_source=qasymphony&utm_medium=redirect&utm_campaign=qas-redirects&utm_content=%2F%2Fwww.qasymphony.com%2Fblog%2F64-test-metrics%2F
- *Metriche per la valutazione della Qualità di Processo:*
https://it.wikipedia.org/wiki/Metriche_di_progetto



2 Qualità di processo

Per garantire un prodotto di qualità, che rispetti i costi ed i tempi stabiliti dal *Piano di Progetto 2.0.0*, il nostro gruppo ha deciso di aderire allo standard ISO/IEC 15504_G, anche noto come SPICE (*Software Process Improvement and Capability Determination*). Lo standard ISO/IEC 15504_G garantisce la qualità di tutti i processi che compongono il prodotto attraverso una definizione chiara degli obiettivi di ognuno di essi e delle soglie prestabilite. Per una descrizione più dettagliata dello standard ISO/IEC 15504_G riferirsi al capitolo 6 nel documento *Norme di Progetto 2.0.0*.

2.1 Processi di supporto

Di seguito si presentano le metriche relative alla qualità dei processi, come stabilito nel documento *Norme di Progetto 2.0.0*, indispensabili per ottenere gli obiettivi di qualità.

2.1.1 Pianificazione

2.1.1.1 Metriche

MQPS01 Budget at Completion

Quantità di budget totale allocato per il progetto. La misurazione viene effettuata tramite un numero intero.

- **valore preferibile:** corrispondente al preventivo;
- **valore accettabile:** il valore del preventivo con un errore massimo del 5%, ossia:

$$\text{preventivo} - 5\% \leq BAC \leq \text{preventivo} + 5\%$$

MQPS02 Planned value

Si tratta del valore del lavoro pianificato al momento del calcolo. La formula adottata è:

$$PV = BAC * \% \text{ di lavoro pianificato}$$

- **valore preferibile:** > 0 ;
- **valore accettabile:** ≥ 0 .

MQPS03 Actual cost

Il denaro speso fino al momento del calcolo per lo svolgimento del progetto. E' necessario un monitoraggio continuo per avere un Actual Cost al di sotto della soglia del Planned Value. Il valore è dato da un numero intero.



- **valore preferibile:** $0 \leq AC < PV$;
- **valore accettabile:** $0 \leq AC \leq \text{budget totale}$.

MQPS04 Earned value

Si tratta del valore del lavoro fatto fino al momento del calcolo. La formula corrispondente è:

$$\text{BAC} * \% \text{ di lavoro completato}$$

- **valore preferibile:** $\geq PV$;
- **valore accettabile:** ≥ 0 .

MQPS05 Schedule Variance

Indica lo stato di avanzamento nello svolgimento del progetto rispetto a quanto pianificato. La formula adottata è:

$$SV = EV - PV$$

- **valore preferibile:** > 0 ;
- **valore accettabile:** ≥ 0 .

In base al risultato ottenuto:

- $SV > 0$ indica che il gruppo è in anticipo rispetto alla pianificazione; in futuro, le previsioni dovranno essere eseguite con più precisione e tenendo conto di questo risultato;
- $SV = 0$ indica che il gruppo è in linea con la pianificazione; i criteri adottati per fare la pianificazione sono quindi efficaci, e dovranno essere usati anche per previsioni future;
- $SV < 0$ indica che il gruppo è in ritardo rispetto alla pianificazione: è necessaria una revisione delle pianificazioni da quel momento in poi, in modo da ridistribuire le risorse ed evitare di accumulare ulteriori ritardi.

MQPS06 Cost variance

Indica la differenza tra il costo di lavoro effettivamente completato ed il costo attualmente sostenuto. La formula adottata è:

$$CV = EV - AC$$

- **valore preferibile:** > 0 ;



- **valore accettabile:** ≥ 0 .

In base al risultato ottenuto:

- $CV > 0$ indica che lo svolgimento del progetto si mantiene al di sotto del budget;
- $CV = 0$ indica che il progetto è al pari con il budget;
- $CV < 0$ indica che il progetto è al di sopra del budget a disposizione, si devono correggere i metodi di lavoro.

2.2 Processi di sviluppo

La pianificazione è un'attività significativa della gestione di progetto. Consiste nel governare le risorse a disposizione, ovvero tempi, costi e ruoli, monitorarle nel tempo e reagire efficacemente ai cambiamenti.

2.2.1 Analisi dei requisiti

2.2.1.1 Metriche

PROI: Percentuale Requisiti Obbligatori Implementati Indicatore, mediante percentuale, dei requisiti_o obbligatori che sono stati implementati nel software.

$$\text{PROI} = \frac{ROI}{ROT} * 100$$

Dove:

- **ROI:** requisiti obbligatori implementati;
- **ROT:** requisiti obbligatori totali.

I range accettabili per il risultato di **PROI** sono così suddivisi:

- **valore preferibile:** 100%;
- **valore accettabile:** $>90\%$.



3 Qualità del prodotto

Per valutare la qualità del prodotto il gruppo *JawaDruids* ha stabilito di usare come riferimento lo standard ISO/IEC 9126_G, che definisce le caratteristiche, descritte attraverso dei parametri, da considerare affinché il prodotto finale sia di buona qualità. Per un approfondimento sullo standard si rimanda alla lettura del paragrafo §5 delle *Norme di Progetto 2.0.0*. Si riportano di seguito i parametri dello standard ritenuti più interessanti dal gruppo, nel contesto del progetto. Le metriche qui riportate si limitano a quelle individuate fino alla stesura di tale documento, dunque l'elenco di queste sarà opportunamente ampliato in futuro, se necessario per l'aumento della completezza della valutazione della qualità.

3.1 Funzionalità

Si tratta della capacità del prodotto software di fornire le funzioni appropriate e necessarie per soddisfare i bisogni emersi nell'*Analisi dei Requisiti 3.0.0* e per operare in un determinato contesto.

3.1.1 Metriche

3.1.1.1 MQPD01 Totalità dell'implementazione

Indice riportante l'interesse del prodotto software, rispetto ai requisiti_G funzionali posti, mediante un valore in percentuale:

$$T = (1 - \frac{RnI}{RI}) * 100$$

Dove:

- **T** sta per *Totalità*, riferito ai requisiti_G da implementare;
- **RnI** sta per *Requisito non Implementato*;
- **RI** sta per *Requisito Implementato*.

I range accettabili per il risultato di **T** sono così suddivisi:

- $90\% < T \leq 100\%$ indica che la copertura dei requisiti_G proposti è quasi totale;
- $80\% < T \leq 90\%$ indica che la copertura dei requisiti_G proposti è sufficiente, buona;
- $T \leq 80\%$ indica che la copertura dei requisiti_G proposti è insufficiente;
- **Valore Preferibile:** 100%;
- **Valore Accettabile:** $\geq 90\%$.



3.2 Affidabilità

Si tratta della capacità del prodotto software di mantenere il livello di prestazione elevato anche se usato in condizioni specifiche, che possono essere anomale o critiche.

3.2.1 Metriche

3.2.1.1 MQPD03 Rilevamento Errori

Indice che mostra qual'è la percentuale di errore basata sui test fatti. Come formula viene usata la seguente:

$$RE = \left(\frac{TE}{TT}\right) * 100$$

- **RE** sta per *Rilevamento Errori*;
- **TE** sta per *Test con Errori*;
- **TT** sta per *Test Totali*;
- **Valore preferibile**: 0%;
- **Valore accettabile**: $\leq 10\%$.

3.3 Usabilità

Si tratta della capacità del prodotto software di essere di facile comprensione e utilizzo da parte dell'utente, sotto determinate condizioni.

3.3.1 Metriche

3.3.1.1 MQPD04 Validità dei dati in input

Questo indice misura la veridicità dei dati che arrivano in input al software. Ovviamente più i dati si avvicinano alla realtà più elevato sarà il valore dell'indice. Viene usata la seguente formula:

$$VD = \frac{DIV}{DP} * 100$$

- **VD** sta per *Validità Dati*;
- **DIV** sta per *Dati Input Validati*;
- **DP** sta per *Dati Previsti*.



- **Valore Preferibile:** $\geq 100\%$;
- **Valore Accettabile:** $\geq 80\%$.

3.3.1.2 Indice di Gulpease

L'indice di Gulpease_G riporta il grado di leggibilità di un testo redatto in lingua italiana. La formula adottata è:

$$\text{GULP} = 89 + \frac{300 * (\text{numerofrasi}) - 10 * (\text{numeroparole})}{\text{numerolettere}}$$

L'indice così calcolato può pertanto assumere valori compresi tra 0 e 100, in cui:

- **GULP < 80:** indica una leggibilità difficile per un utente con licenza elementare;
- **GULP < 60:** indica una leggibilità difficile per un utente con licenza media;
- **GULP < 40:** indica una leggibilità difficile per un utente con licenza superiore;
- **Valore Preferibile:** > 80 ;
- **Valore Accettabile:** > 60 .

3.3.1.3 Errori Ortografici

La correttezza ortografica della lingua italiana è verificata attraverso l'apposito strumento integrato in di TexStudio, il quale sottolinea in tempo reale le parole ove ritiene sia presente un errore, consentendone la correzione.

- **Valore Preferibile:** 0 errori;
- **Valore Accettabile:** 0 errori.

3.4 Efficienza

Si tratta della capacità di un prodotto software di realizzare le funzioni richieste nel minor tempo possibile e sfruttando al meglio le risorse necessarie, quando opera in determinate condizioni.



3.4.1 Metriche

3.4.1.1 MQPD02 Efficienza raggiunta

Indice riportante l'efficienza del prodotto software, rispetto ai requisiti_G *prestazionali* posti, mediante un valore in percentuale:

$$T = (1 - \frac{RnI}{RI}) * 100$$

Dove:

- **T** sta per *Totalità*, riferito ai requisiti_G *prestazionali* da implementare;
- **RnI** sta per *Requisito prestazionale non Implementato*;
- **RI** sta per *Requisito prestazionale Implementato*.

I range accettabili per il risultato di **T** sono così suddivisi:

- $90\% < T \leq 100\%$ indica che la copertura dei requisiti_G proposti è quasi totale;
- $80\% < T \leq 80\%$ indica che la copertura dei requisiti_G proposti è sufficiente, buona;
- $T \leq 60\%$ indica che la copertura dei requisiti_G proposti è insufficiente;
- **Valore Preferibile:** 90%;
- **Valore Accettabile:** $\geq 60\%$.

3.5 Portabilità

La portabilità è definita come la capacità di un software nell'essere "trasportato" da un ambiente di lavoro, inteso sia come organizzativo che tecnologico, ad un altro.

3.5.1 Valutazione sulla Caratteristica

Il gruppo, dopo aver preso visione degli obbiettivi riguardanti questa caratteristica, si è soffermato sulla **Adattabilità**. Questo perché il prodotto software, essendo una web-app, dovrà essere capace di funzionare su qualsiasi piattaforma internet senza problemi. Inoltre si sta sviluppando e testando un Docker_G in modo da poter rendere portabile il software anche su altri sistemi operativi.



3.6 Mantenibilità

E' la capacità di un prodotto software di essere modificato. Le modifiche possono includere correzioni, adattamenti o miglioramenti del software.

3.6.1 Metriche

3.6.1.1 MQPD05 Comprensione del codice

Con questa metriche si intende calcolare un indice, in percentuale, riferito alla facilità della comprensione del codice da parte dell'utente. La formula utilizzata è la seguente:

$$F = \frac{N_{Lc}}{N_{Lcod}} * 100$$

Dove:

- **F**: è l'indice di facilità di comprensione;
- **N_Lc**: indica il numero di linee di commento presenti all'interno del codice;
- **N_Lcod**: indica il numero di linee di codice presente.
- **Valore Preferibile**: $\leq 10\%$;
- **Valore Accettabile**: $\leq 20\%$.

3.6.1.2 MQPD06 Structural fan-in

Indice che misura il numero di moduli che utilizzano un dato modulo. Un elevato valore di tale indice significa un alto uso del modulo in questione.

- **Valore Preferibile**: ≥ 1 ;
- **Valore Accettabile**: ≥ 0 .

4 Specifica dei test

Per assicurare un'ottima qualità del software prodotto, il gruppo *Jawa Druids*, dopo essersi confrontato, ha deciso di utilizzare come modello di sviluppo software il *V-Model*, o Modello a V, il quale è un'estensione del modello a cascata. Questo modello prevede un lavoro parallelo tra lo sviluppo dei test e le attività_e di analisi e progettazione. Grazie a ciò, i test permettono di verificare sia il corretto funzionamento delle parti di software programmate, sia la corretta implementazione di tutti i requisiti_e del progetto. Vengono utilizzate delle sigle, all'interno di tabelle, per fornire una comprensione più agevolata riguardo gli output prodotti tramite i test, specificando se il risultato è quello atteso, errato o non coerente con quanto aspettato. Le sigle per lo stato dei test sono:

- **NI**: non implementato;
- **I**: implementato.

Per quanto riguarda la qualità dei test si usa:

- **NS**: il test non ha soddisfatto la richiesta;
- **S**: il test ha soddisfatto la richiesta.

I test di *Sistema* e *Accettazione* hanno la seguente nomenclatura:

[TipoTest]RS[classificazione][tipo_di_requisito][codice_requisito]

dove:

- **TipoTest**: specifica il tipo di test applicato;
- **classificazione**:
 - **F**: indica se il requisito è funzionale;
 - **Q**: indica se il requisito è qualitativo;
 - **V**: indica se il requisito è vincolante;
 - **P**: indica se il requisito è prestazionale.
- **tipo_di_requisito**: assume i seguenti valori:
 - **O** per i requisiti_e obbligatori;
 - **D** per i requisiti_e desiderabili;
 - **F** per i requisiti_e facoltativi.



- **codice_requisito**: un numero incrementale per rendere univoco il requisito.

Invece i test di *Unità*, *Integrazione* e *Regressione* sono denominati nel seguente modo:

[TipoTest][Id]

dove:

- **Id** rappresenta un numero incrementale che inizia da 1.

4.1 Tipi di test

I test che verranno effettuati sul prodotto software sono così divisi:

- **Test di unità**: i test di unità servono per verificare le singole unità del software, ovvero le componenti con funzionamento autonomo. Il superamento di tali test non implica il corretto funzionamento del software. Viene contrassegnata da [TU].
- **Test di integrazione**: questa tipologia di test verifica i singoli moduli del software come fossero un gruppo unico. Vengono svolti successivamente ai TU e prima dei TS. Sono contrassegnati da [TI];
- **Test di sistema**: i test di sistema vengono eseguiti per verificare che i requisiti_c, scritti nel documento *Analisi dei Requisiti*, siano stati implementati e funzionanti. Viene rappresentato mediante la sigla [TS];
- **Test di accettazione**: i test di accettazione hanno come scopo la verifica che il software sviluppato soddisfi i requisiti_c presenti nel capitolato d'appalto_G e concordati col proponente_c. Questi saranno eseguiti durante il collaudo finale del prodotto software sotto l'osservazione sia dell'azienda proponente_c sia del gruppo di lavoro. Rappresentati mediante la sigla [TA];
- **Test di regressione**: Servono a garantire il corretto funzionamento del prodotto a seguito di modifiche del codice o di inserimento di nuove funzionalità. Vengono etichettati nel seguente modo [TR];



4.2 Test di unità

Sono stati individuati i seguenti test di unità per garantire il funzionamento di ogni minimo componente autonomo del sistema.

Test di unità Python			
Id Test	Descrizione	Esito	Qualità
<i>TU01</i>	Verifica che il metodo <code>get_onecall_api_response</code> in <code>weather_forecast.py</code> ottenga i dati delle 48 ore successive	<i>I</i>	<i>S</i>
<i>TU02</i>	Verifica che il metodo <code>fetch_read_m3u8</code> in <code>detect.py</code> scarichi e legga correttamente un file m3u8	<i>I</i>	<i>S</i>
<i>TU03</i>	Verifica che il metodo <code>extract_frame_from_video_url</code> in <code>get_frames.py</code> estragga e legga correttamente un frame da un video di cui viene fornito il link	<i>I</i>	<i>S</i>

Tabella 4.1: Elenco test di unità - Python

Test di unità Spring			
Id Test	Descrizione	Esito	Qualità
<i>TU04</i>	Verifica che il metodo in cui viene eseguita la query per il recupero della lista delle città presenti nel database ritorni effettivamente il tipo di ritorno corretto, ovvero <code>List<String></code>	<i>I</i>	<i>S</i>
<i>TU05</i>	Verifica che il metodo in cui viene eseguita la query per il recupero dell'ultimo dato presente nel database ritorni effettivamente il tipo di ritorno corretto, ovvero un <code>Detection</code>	<i>I</i>	<i>S</i>
<i>TU06</i>	Verifica che il metodo in cui viene eseguita la query per il recupero dei dati presente nel database in base alla città, alla data e all'orario, ritorni effettivamente il tipo di ritorno corretto, ovvero un <code>List<Detection></code>	<i>I</i>	<i>S</i>



<i>TU07</i>	Verifica che il metodo in cui viene eseguita la query per il recupero della latitudine e/o longitudine relativa a una città dal database ritorni effettivamente il tipo di ritorno corretto, ovvero List<String>.	<i>I</i>	<i>S</i>
-------------	---	----------	----------

Tabella 4.2: Elenco test di unità - Spring

Test di unità Vue.js			
Id Test	Descrizione	Esito	Qualità
<i>TU08</i>	Verifica che la componente "aboutUs.vue" esista	<i>I</i>	<i>S</i>
<i>TU09</i>	Verifica che il file "httpcommon.js" esista	<i>I</i>	<i>S</i>
<i>TU10</i>	Verifica che il metodo "getCities" presente nel file "httprequest.js" esista	<i>I</i>	<i>S</i>
<i>TU11</i>	Verifica che il metodo "getCoo" presente nel file "httprequest.js" esista	<i>I</i>	<i>S</i>
<i>TU12</i>	Verifica che il metodo "getDataRT" presente nel file "httprequest.js" esista	<i>I</i>	<i>S</i>
<i>TU13</i>	Verifica che il metodo "getLastValue" presente nel file "httprequest.js" esista	<i>I</i>	<i>S</i>
<i>TU14</i>	Verifica che la componente "autocompleteSearch.vue" esista	<i>I</i>	<i>S</i>
<i>TU15</i>	Verifica che il tag di input della componente "autocompleteSearch.vue" sia vuoto	<i>I</i>	<i>S</i>
<i>TU16</i>	Verifica che il tag di input della componente "autocompleteSearch.vue" non sia vuoto dopo che l'utente ha interagito scrivendo	<i>I</i>	<i>S</i>
<i>TU17</i>	Verifica che il metodo "getNameCity" presente nel file "autocompleteSearch.vue" esista	<i>I</i>	<i>S</i>
<i>TU18</i>	Verifica che il metodo "getNameCity" presente nel file "autocompleteSearch.vue" venga chiamato almeno una volta	<i>I</i>	<i>S</i>



<i>TU19</i>	Verifica che il metodo "itemSelected" presente nel file "autocompleteSearch.vue" esista	<i>I</i>	<i>S</i>
<i>TU20</i>	Verifica che la componente "datePicker.vue" esista	<i>I</i>	<i>NS</i>
<i>TU21</i>	Verifica che la componente "slider.vue" esista	<i>I</i>	<i>NS</i>
<i>TU22</i>	Verifica che il metodo "getTime" presente nel file "slider.vue" esista	<i>I</i>	<i>NS</i>
<i>TU23</i>	Verifica che la componente "app.vue" esista	<i>I</i>	<i>S</i>
<i>TU24</i>	Verifica che la componente "listCity.vue" esista	<i>I</i>	<i>NS</i>
<i>TU25</i>	Verifica che la componente "heatmap.vue" esista	<i>I</i>	<i>NS</i>
<i>TU26</i>	Verifica che la componente "vue2LeafletHeatmap.vue" esista	<i>I</i>	<i>NS</i>
<i>TU27</i>	Verifica che la componente "mainPage.vue" esista	<i>I</i>	<i>S</i>
<i>TU28</i>	Verifica che il metodo "getRetrieveCoordinate" presente nel file "mainPage.vue" venga chiamato dopo il click dell'utente sul bottone	<i>I</i>	<i>S</i>
<i>TU29</i>	Verifica che la componente "heatmap.vue" presente nel file "mainPage.vue" esista	<i>I</i>	<i>S</i>

Tabella 4.3: Elenco test di unità - Vue.js

4.2.1 Tracciamento dei test

Tracciamento test di unità Python		
Id Test	Percorso file	Metodo
TU01	acquisition/main/test/test_weather_forecast.py	test_response
TU02	acquisition/main/test/test_detect.py	test_fetch_read_m3u8
TU03	acquisition/main/test/test_detect.py	test_extract_frame_from_video_url

Tabella 4.4: Tracciamento dei test di unità - Python



Tracciamento test di unità Spring		
Id Test	Percorso file	Metodo
TU04	proof_of_concept/webapp/webapp/src/test/java/com/webapp/data/mongodb/DetectionCustomRepositoryImplTest/DetectionCustomRepositoryImplTest.java	getCitiesTest()
TU05	proof_of_concept/webapp/webapp/src/test/java/com/webapp/data/mongodb/DetectionCustomRepositoryImplTest/DetectionCustomRepositoryImplTest.java	getLastValueTest()
TU06	proof_of_concept/webapp/webapp/src/test/java/com/webapp/data/mongodb/DetectionCustomRepositoryImplTest/DetectionCustomRepositoryImplTest.java	getDataRTTest()
TU07	proof_of_concept/webapp/webapp/src/test/java/com/webapp/data/mongodb/DetectionCustomRepositoryImplTest/DetectionCustomRepositoryImplTest.java	getLatLngsTest()

Tabella 4.5: Tracciamento dei test di unità - Spring

Tracciamento test di unità Vue.js		
Id Test	Percorso file	Metodo
TU08	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	aboutUs:check if exist
TU09	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	httpcommon:check if exist
TU010	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	httpRequest:check if getCities work
TU011	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	httpRequest:check if getCoo work
TU012	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	httpRequest:check if getDataRT work
TU013	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	httpRequest:check if getLastValue work



TU014	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	autocompleteSearch: check if exist
TU015	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	autocompleteSearch: check if search bar is empty
TU016	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	autocompleteSearch: user should have written something
TU017	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	autocompleteSearch: check if the get work
TU018	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	autocompleteSearch: check if the get is called
TU019	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	autocompleteSearch: check if the item-Selected work
TU020	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	datePicker:check if exist
TU021	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	slider:check if exist
TU022	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	slider: the get should work
TU023	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	App:check if exist
TU024	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	listCity:check if exist
TU025	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	heatMap:check if exist
TU026	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	vue2LeafletHeatmap:check if exist



TU027	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	mainPage:check if exist
TU028	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	mainPage:check if getRetrieveCoordinate is called
TU029	proof_of_concept/webapp/vue-js-client-crud/src/tests/unit/appTest.spec.js	mainPage:check if component heatmap exist

Tabella 4.6: **Tracciamento dei test di unità - Vue.js**

4.3 Test di integrazione

Sono stati individuati i seguenti test di integrazione per garantire il funzionamento dei diversi moduli nel momento in cui vengono messi in relazione tra di loro.

Id Test	Descrizione	Esito	Qualità
<i>TI01</i>	Verifica che il sistema acquisisca i dati attraverso il modulo contapersone che utilizza l'algoritmo di Yolo v3	<i>I</i>	<i>S</i>
<i>TI02</i>	Verifica che il sistema invii i dati ottenuti dal modulo contapersone al database in modo corretto	<i>I</i>	<i>S</i>
<i>TI03</i>	Verifica che il sistema invii i dati ottenuti dal modulo di machine learning _c al database in modo corretto	<i>I</i>	<i>S</i>
<i>TI04</i>	Verifica che il sistema invii i dati presenti nel database alla web application in modo corretto	<i>I</i>	<i>S</i>

Tabella 4.7: **Elenco test di accettazione**



4.4 Test di sistema

Sono stati individuati i seguenti test di sistema per garantire il funzionamento del prodotto sviluppato. I test di sistema sono stati identificati attraverso i requisiti indicati nel documento *Analisi dei Requisiti 3.0.0*.

Id Test	Descrizione	Esito	Qualità
<i>TSRSFO1</i>	Verifica che il sistema utilizzi motori software 'contapersone'	<i>I</i>	<i>S</i>
<i>TSRSFF2</i>	Verifica che il sistema utilizzi simulatori di dati storici	<i>NI</i>	-
<i>TSRSFO3</i>	Verifica della visualizzazione di un messaggio d'errore in caso mancanza dati nella generazione della heat-map _G	<i>I</i>	<i>S</i>
<i>TSRSFO4</i>	Verifica che il sistema archivi tutti i dati nel database	<i>I</i>	<i>S</i>
<i>TSRSFO5</i>	Verifica che il sistema elabori i dati dalle sorgenti esterne in tempo reale	<i>I</i>	<i>S</i>
<i>TSRSFD6</i>	Verifica che il sistema effettui una previsione dell'insorgenza futura di variazioni significative di flussi di persone	<i>NI</i>	-
<i>TSRSFO7</i>	Verifica che l'utente possa poter visualizzare i dati in tempo reale tramite heat map _G	<i>I</i>	<i>S</i>
<i>TSRSFO8</i>	Verifica che l'utente possa poter visualizzare i dati storicizzati tramite heat map _G	<i>I</i>	<i>S</i>
<i>TSRSFO9</i>	Verifica che l'utente possa poter visualizzare una previsione tramite heat map _G	<i>I</i>	<i>S</i>
<i>TSRSFF10</i>	Verifica che l'utente possa poter distinguere tra i dati simulati e quelli reali	<i>NI</i>	-
<i>TSRSFD11</i>	Verifica che l'utente possa poter visualizzare un indice di affidabilità della previsione nella mappa	<i>NI</i>	-
<i>TSRSFD12</i>	Verifica che l'utente possa poter visualizzare un indice di affidabilità dei dati in tempo reale nella mappa	<i>NI</i>	-



<i>TSRSFF13</i>	Verifica che l'utente possa poter applicare dei filtri ai dati (reali, simulati)	<i>NI</i>	-
<i>TSRSFF14</i>	Verifica che l'utente abbia la possibilità di scegliere le sorgenti dati da cui prelevare dati tempo reale	<i>NI</i>	-
<i>TSRSFO15</i>	Verifica che il sistema aggiorni la mappa automaticamente ogni 10 minuti	<i>I</i>	<i>S</i>
<i>TSRSFO16</i>	Verifica che il modello di machine learning _g salvi i pesi e le predizioni in un file	<i>I</i>	<i>S</i>
<i>TSRSFO17</i>	Verifica che venga inviato un messaggio di errore al front end , dal backend, se non ci sono i dati richiesti	<i>I</i>	<i>S</i>
<i>TSRSFO18</i>	Verifica che l'utente possa selezionare una città tra quelle disponibili	<i>I</i>	<i>S</i>
<i>TSRSFO19</i>	Verifica che l'utente visualizzi le zone delle città rispettivamente alle zone utilizzate	<i>I</i>	<i>S</i>
<i>TSRSFO20</i>	Verifica che il sistema archivi i dati in tempo reale con la data e orario di riferimento associata	<i>I</i>	<i>S</i>
<i>TSRSFF21</i>	Verifica che il sistema utilizzi i dati delle predizioni in caso di mancanza di dati in tempo reale	<i>I</i>	<i>S</i>
<i>TSRSFO22</i>	Verifica che l'utente possa selezionare l'intervallo orario in fasce orarie	<i>I</i>	<i>S</i>
<i>TSRSFO23</i>	Verifica che il sistema utilizzi in modo prioritario i dati reali se presenti anche quelli determinati per le predizioni	<i>I</i>	<i>S</i>
<i>TSRSFO24</i>	Verifica che il sistema aggiorni automaticamente la mappa alla selezione di un diverso orario	<i>I</i>	<i>S</i>
<i>TSRSFO25</i>	Verifica che l'utente possa poter selezionare la data del giorno di cui vuole visualizzare i dati	<i>I</i>	<i>S</i>



<i>TSRSFO26</i>	Verifica che l'utente possa poter ripristinare la visione in tempo reale tramite un pulsante di ripristino	<i>I</i>	<i>S</i>
<i>TSRSFF27</i>	Verifica che l'utente possa poter reperire il manuale d'uso	<i>NI</i>	-
<i>TSRSFO28</i>	Verifica che l'utente possa poter variare il livello di zoom della heat map _c	<i>I</i>	<i>S</i>
<i>TSRSFO28.1</i>	Verifica che l'utente possa poter aumentare il livello di zoom della heat map _c	<i>I</i>	<i>S</i>
<i>TSRSFO28.1.1</i>	Verifica che l'utente possa poter attuare il drag _c della heat map _c	<i>I</i>	<i>S</i>
<i>TSRSFO28.1.2</i>	Verifica che l'utente possa poter visualizzare il pop-up _c legato ad un punto di interesse	<i>I</i>	<i>S</i>
<i>TSRSFO28.1.3</i>	Verifica che l'utente possa poter chiudere il pop-up _c legato ad un punto di interesse	<i>I</i>	<i>S</i>
<i>TSRSFO28.2</i>	Verifica che l'utente possa poter diminuire il livello di zoom della heat map _c	<i>I</i>	<i>S</i>
<i>TSRSFD29</i>	Verifica che l'utente possa poter ricercare tramite una barra di ricerca le città presenti nel database	<i>I</i>	<i>S</i>
<i>TSRSFD30</i>	Verifica che l'utente possa poter visualizzare il messaggio d'errore relativo alla mancanza dei dati ricercati attraverso la barra di ricerca nel database	<i>NI</i>	-
<i>TSRSFD31</i>	Verifica che l'utente possa poter visualizzare il confronto dei dati di due città selezionate dall'utente	<i>NI</i>	-
<i>TSRSFD32</i>	Verifica che l'utente possa poter salvare in un file locale i dati della città della mappa che sta visualizzando	<i>NI</i>	-
<i>TSRSFD33</i>	Verifica che l'utente possa poter inserire l'e-mail per il ricevimento delle informazioni delle informazioni della città selezionata	<i>NI</i>	-



<i>TSRSFD33.1</i>	Verifica che l'utente possa poter visualizzare un messaggio di errore nel caso l'email inserita sia scritta in modo errato	<i>NI</i>	-
<i>TSRSFD34</i>	Verifica che il sistema abbia inserito correttamente l'email e la città correlata nel database	<i>NI</i>	-
<i>TSRSFD35</i>	Verifica che il sistema invii correttamente l'email all'utente	<i>NI</i>	-
<i>TSRSFD36</i>	Verifica che l'utente possa poter visualizzare la lista delle città più ricercate	<i>NI</i>	-
<i>TSRSFD37</i>	Verifica che l'utente possa poter visualizzare la lista di tutte le città presenti nel database	<i>NI</i>	-

Tabella 4.8: **Elenco test di sistema**

4.4.1 Tracciamento dei test

Id Test	Id Requisito
TSRSFO1	RSFO1
TSRSFF2	RSFF2
TSRSFO3	RSFO3
TSRSFO4	RSFO4 RSFO4.1 RSFO4.2
TSRSFO5	RSFO5
TSRSFD6	RSFD6
TSRSFO7	RSFO9
TSRSFO8	RSFO10
TSRSFO9	RSFO7 RSFO11
TSRSFF10	RSFF12
TSRSFD11	RSFD13



TSRSFD12	RSFD14
TSRSFF13	RSFF15
TSRSFF14	RSFF16
TSRSFO15	RSFO17
TSRSFO16	RSFO18 RSFO18.1
TSRSFO17	RSFO19
TSRSFO18	RSFO20
TSRSFO19	RSFO21
TSRSFO20	RSFO22 RSFO22.1
TSRSFF21	RSFF23
TSRSFO22	RSFO24
TSRSFO23	RSFO25
TSRSFO24	RSFO26
TSRSFO25	RSFO27
TSRSFO26	RSFO28
TSRSFF27	RSFF31
TSRSFO28	RSFO32
TSRSFO28.1	RSFO32.1
TSRSFO28.1.1	RSFO32.1.1
TSRSFO28.1.2	RSFO32.1.2
TSRSFO28.1.3	RSFO32.1.3
TSRSFO28.2	RSFO32.2
TSRSFD29	RSFD33 RSFD33.1 RSFD33.2
TSRSFD30	RSFD34
TSRSFD31	RSFD35



TSRSFD32	RSFD36 RSFD36.1 RSFD36.2
TSRSFD33	RSFD37
TSRSFD33.1	RSFD37.1
TSRSFD34	RSFD38
TSRSFD35	RSFD39
TSRSFD36	RSFD40
TSRSFD37	RSFD41

Tabella 4.9: **Tracciamento dei test di sistema con i requisiti**

4.5 Test di accettazione

Essendo una tipologia di test che verrà eseguita in periodo di accettazione il gruppo si impegna a completare questa sezione in previsione della Revisione di Accettazione.

5 Resoconto attività di verifica

In questa sezione si riportano gli esiti, descritti ed analizzati, di tutte le attività_G di verifica svolte.

5.1 Revisione dei Requisiti

Tutta la documentazione sviluppata nella prima fase da consegnare per la Revisione dei Requisiti ha subito una meticolosa ed attenta revisione da parte dei Verificatori. Questi ultimi hanno seguito, in questa attività_G, per ogni documento, i metodi di *Walkthrough_G* ed *Inspection_G* relative all'analisi statica, stabilite nelle *Norme di Progetto 3.0.0*.

5.1.1 Strategia adoperata per l'analisi statica dei documenti

Il *Verificatore* si è occupato di valutare la correttezza del documento, concentrandosi nell'individuare gli errori presenti in questo. Una volta individuati gli errori la strategia adottata è la seguente:

- Correzione degli errori sia ortografici che sintattici, non fedeli alle norme tipografiche fissate nelle *Norme di Progetto 3.0.0*.

5.1.2 Esiti verifica

Per ciascun documento stilato si è calcolato l'indice di Gulpease_G. I risultati sono mostrati qui di seguito. Per evitare risultati errati nel calcolo di tale indice, non si sono tenuti in considerazione:

- il frontespizio di ogni documento;
- le eventuali tabelle presenti nei documenti;
- i diari delle modifiche di ogni documento.

Documento	Indice di Gulpease	Esito
<i>Analisi dei Requisiti 1.0.0</i>	96	<i>Superato</i>
<i>Norme di Progetto 1.0.0</i>	75	<i>Superato</i>
<i>Studio di Fattibilità 1.0.0</i>	70	<i>Superato</i>
<i>Piano di Progetto 1.0.0</i>	77	<i>Superato</i>
<i>Piano di Qualifica 1.0.0</i>	80	<i>Superato</i>

Tabella 5.1: Elenco Indici di Gulpease_g dei documenti fino alla RR

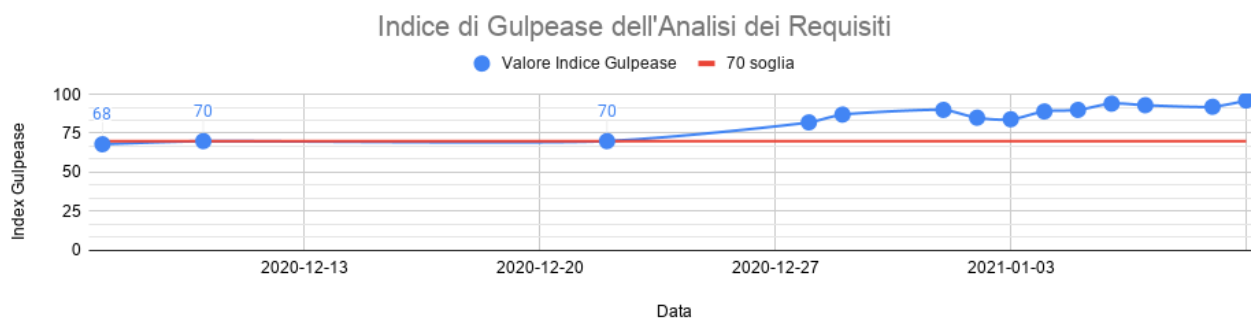


Figura 5.1: Andamento Indice di Gulpease_g Analisi dei Requisiti fino a RR

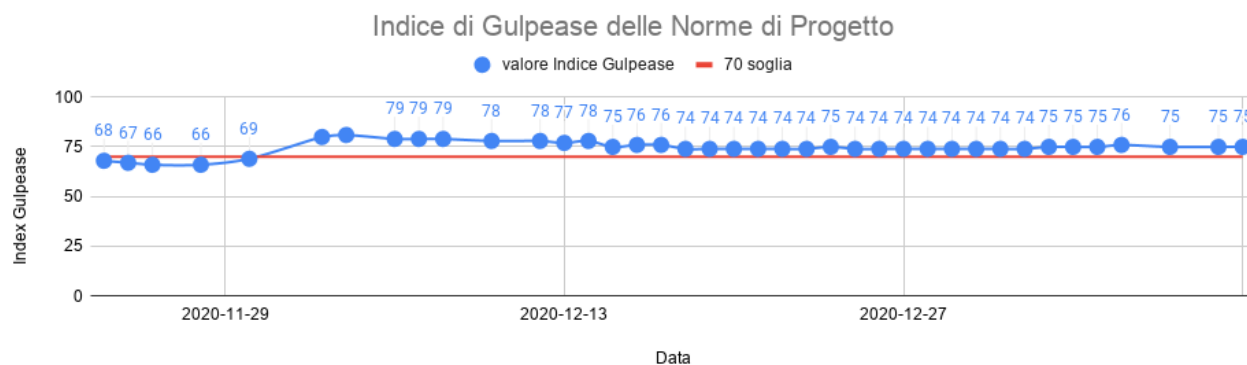
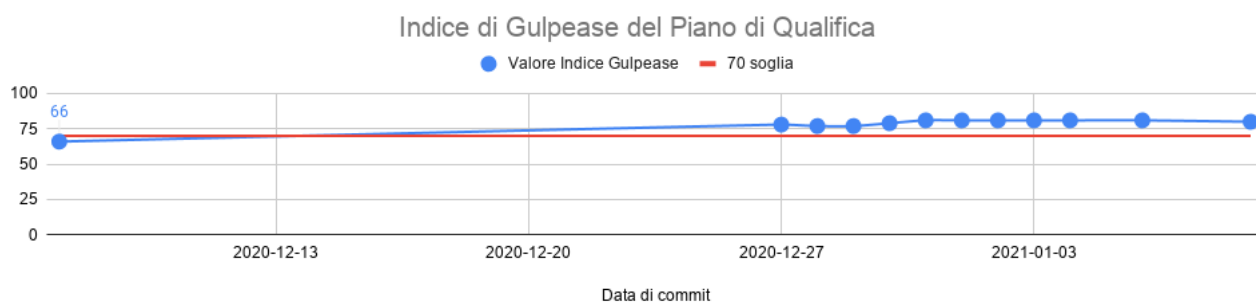
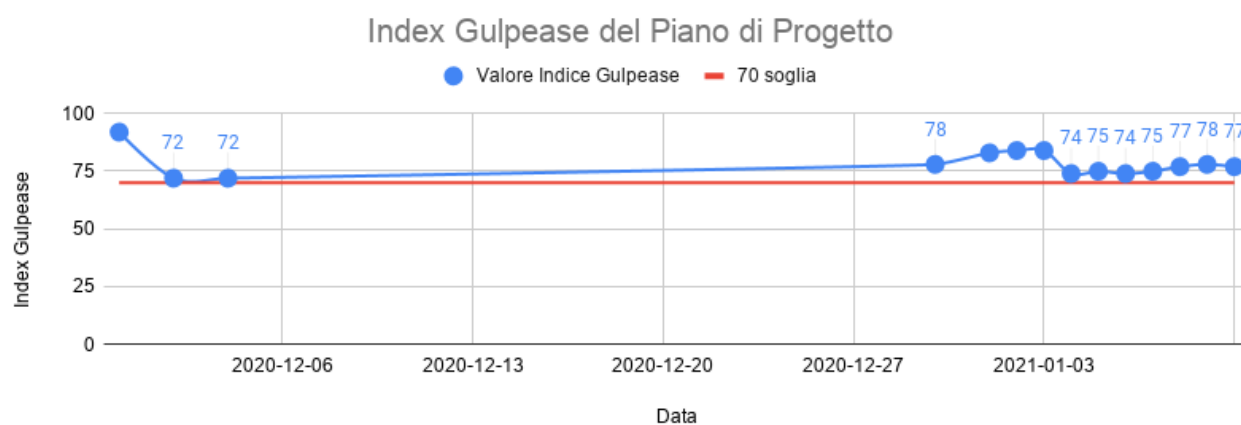
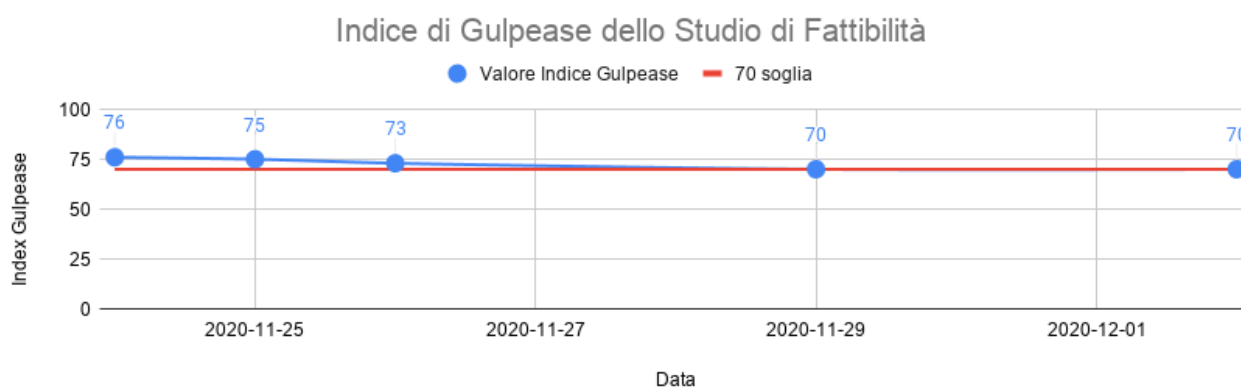


Figura 5.2: Andamento Indice di Gulpease_g Norme di Progetto fino a RR

Figura 5.3: Andamento Indice di Gulpease_G Piano di Qualifica fino a RRFigura 5.4: Andamento Indice di Gulpease_G Piano di Progetto fino a RRFigura 5.5: Andamento Indice di Gulpease_G Studio di Fattibilità fino a RR



Per quanto riguarda gli Indici di Gulpease_G dei verbali si è deciso di rappresentare i risultati in forma tabellare. Questo in quanto il verbale viene scritto tutta in una volta, quindi utilizzare un grafico temporale risulta non idoneo.

Documento	Indice di Gulpease	Esito
<i>verbale_interno_2020-10-28</i>	100	<i>Superato</i>
<i>verbale_interno_2020-11-19</i>	100	<i>Superato</i>
<i>verbale_interno_2020-11-24</i>	99	<i>Superato</i>
<i>verbale_interno_2020-12-04</i>	98	<i>Superato</i>
<i>verbale_esterno_2020-12-17</i>	99	<i>Superato</i>
<i>verbale_interno_2020-12-29</i>	100	<i>Superato</i>
<i>verbale_interno_2021-01-03</i>	100	<i>Superato</i>
<i>verbale_interno_2021-01-06</i>	100	<i>Superato</i>

Tabella 5.2: Elenco Indici di Gulpease_G dei verbali fino alla RR

5.2 Revisione di Progettazione

Tutta la documentazione da consegnare per la Revisione dei Progettazione ha subito una meticolosa ed attenta revisione da parte dei Vericatori. Questi ultimi hanno seguito, per ogni documento, i metodi di *Walkthrough_G* ed *Inspection_G* relativi all'analisi statica, stabiliti nelle *Norme di Progetto 2.0.0*.

5.2.1 Verifiche di processo

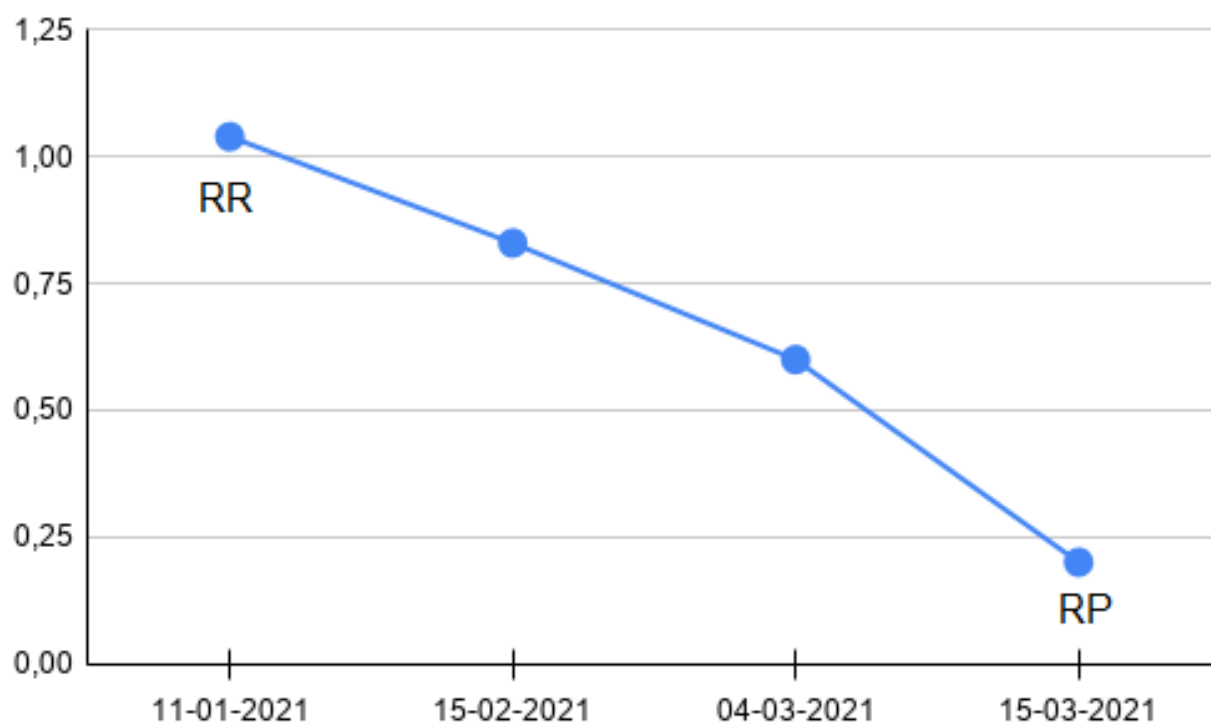


Figura 5.6: MQPS05

Il grafico indica la variazione di giorni di lavoro, dove l'asse dello zero individua il rispetto dei tempi di lavoro. Da questo si evidenzia come in entrambe le fasi ci sia una maggiorazione di giorni di lavoro rispetto a quelli preventivati. Anche se presente una maggiorazione si può notare come il gruppo sia riuscito ad ottimizzare il lavoro dalla *Revisione dei Requisiti* per diminuire il più possibile questo difetto.

5.2.2 Verifiche di prodotto

5.2.2.1 Strategia adoperata per l'analisi statica dei documenti

La strategia adoperata per l'analisi statica dei documenti per la *Revisione di Progettazione* è la medesima di quella descritta in § 5.1.1.

5.2.2.1.1 Esiti Verifica Per ciascun documento stilato si è calcolato l'indice di $Gulpease_g$. I risultati sono mostrati qui di seguito. Per evitare risultati errati nel calcolo di tale indice, non si sono tenuti in considerazione:

- il frontespizio di ogni documento;
- le eventuali tabelle presenti nei documenti;
- i diari delle modifiche di ogni documento.

Documento	Indice di Gulpease	Esito
<i>Analisi dei Requisiti 3.0.0</i>	92	<i>Superato</i>
<i>Norme di Progetto 2.0.0</i>	86	<i>Superato</i>
<i>Piano di Progetto 2.0.0</i>	79	<i>Superato</i>
<i>Piano di Qualifica 2.0.0</i>	84	<i>Superato</i>

Tabella 5.3: Elenco Indici di $Gulpease_g$ dei documenti fino alla RP

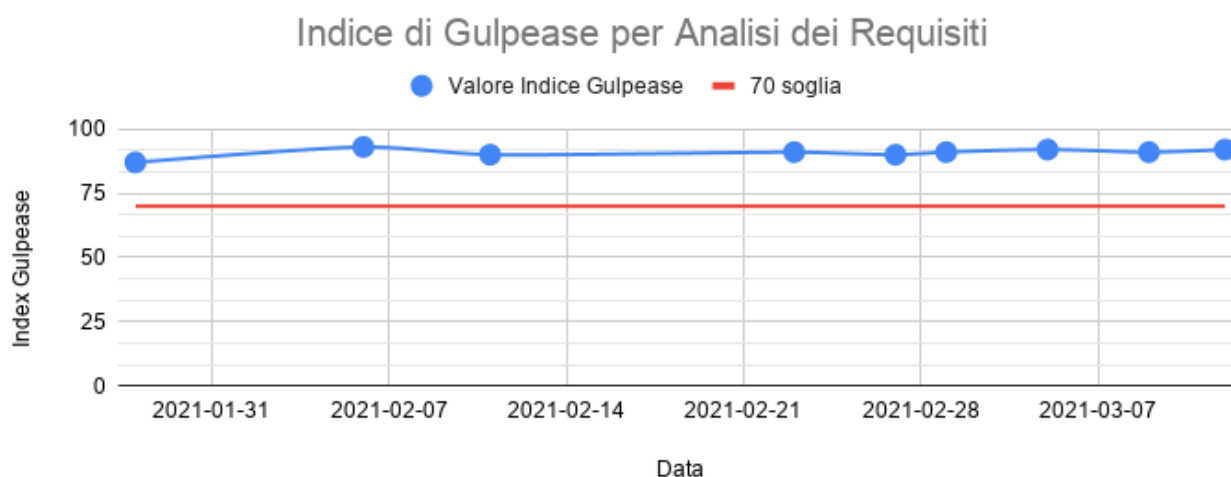


Figura 5.7: Andamento Indice di $Gulpease_g$ Analisi dei Requisiti fino a RP

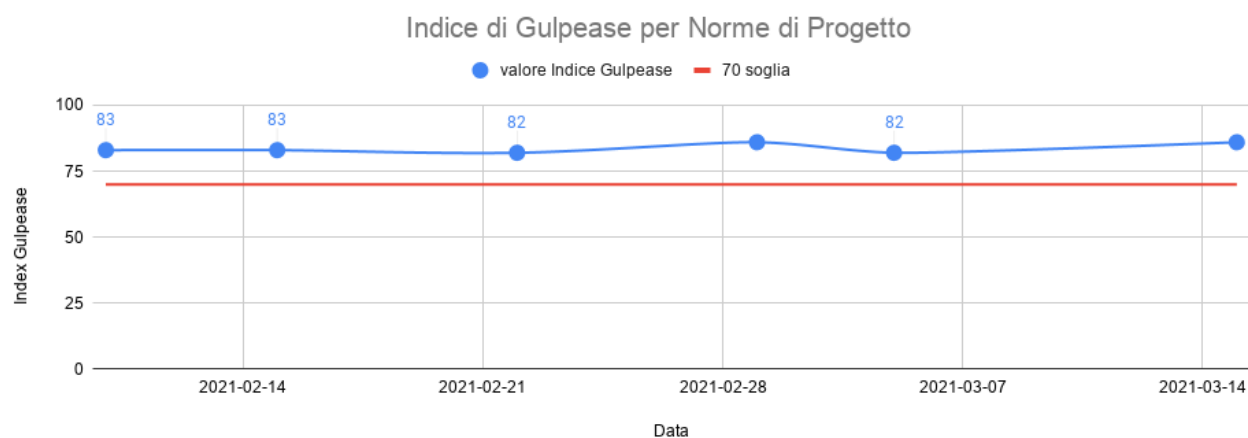


Figura 5.8: Andamento Indice di Gulpease_G Norme di Progetto fino a RP

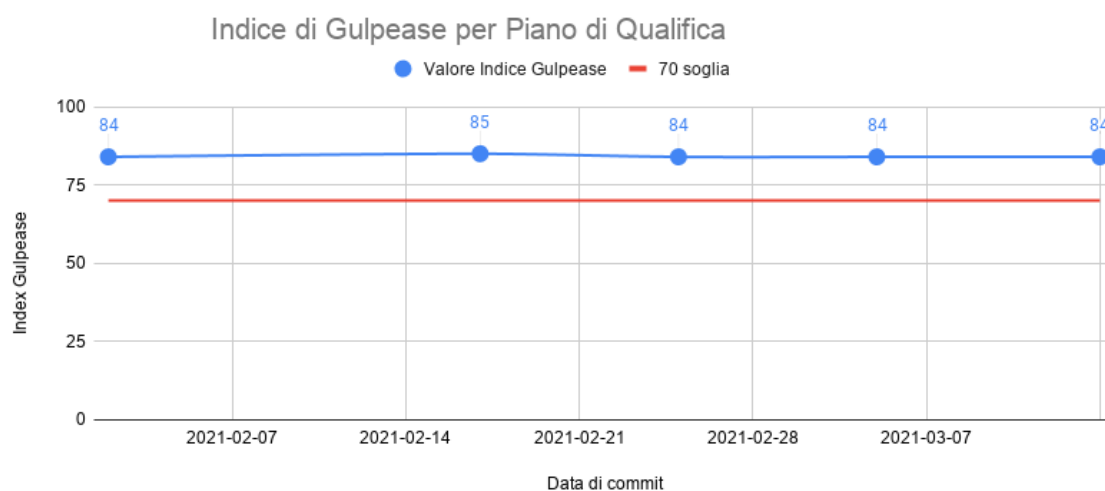


Figura 5.9: Andamento Indice di Gulpease_G Piano di Qualifica fino a RP

Index Gulpease per Piano di Progetto

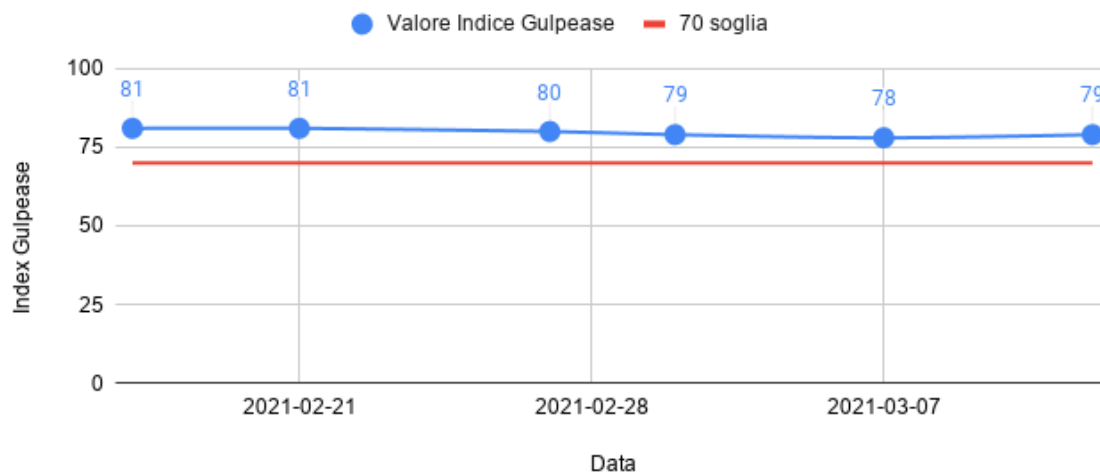


Figura 5.10: Andamento Indice di Gulpease_c Piano di Progetto fino a RP

Per quanto riguarda gli Indici di Gulpease_c dei verbali si è deciso di rappresentare i risultati in forma tabellare. Questo in quanto il verbale viene scritto tutta in una volta, quindi utilizzare un grafico temporale risulta non idoneo.

Documento	Indice di Gulpease	Esito
<i>v_e_2021-01-28</i>	99	<i>Superato</i>
<i>v_i_2021-01-29</i>	98	<i>Superato</i>
<i>v_e_2021-02-02</i>	100	<i>Superato</i>
<i>v_e_2021-02-08</i>	100	<i>Superato</i>
<i>v_i_2021-02-10</i>	100	<i>Superato</i>
<i>v_i_2021-02-17</i>	99	<i>Superato</i>
<i>v_e_2021-02-25</i>	98	<i>Superato</i>
<i>v_i_2021-02-26</i>	100	<i>Superato</i>
<i>v_i_2021-03-08</i>	100	<i>Superato</i>

Tabella 5.4: Elenco Indici di Gulpease_c dei verbali fino alla RP



5.3 Revisione di qualifica

Tutta la documentazione da consegnare per la Revisione di Qualifica ha subito una meticolosa ed attenta revisione da parte dei Verificatori. Questi ultimi, per attuare la verifica, hanno seguito per ogni documento i metodi di *Walkthrough_g* ed *Inspection_g*, relativi all'analisi statica stabiliti nelle *Norme di Progetto 3.0.0*.

5.3.1 Verifiche di processo

5.3.1.1 MQPS05

Il grafico sottostante indica la variazione di giorni di lavoro dove l'asse zero individua il rispetto dei tempi di lavoro.

5.3.1.2 PROI

Il numero di requisiti_g e la loro suddivisione in implementati e non implementati è visualizzabile nella seguente tabella.

Realizzazione	Quantità
<i>Implementato</i>	45
<i>Non Implementato</i>	1
<i>Totali</i>	46

Tabella 5.6: **PROI**

Effettuando il calcolo di questa metrica otteniamo una percentuale di requisiti obbligatori implementati pari ad un valore accettabile di 98%.

5.3.2 Verifiche di prodotto

5.3.2.1 MQPD01

Il numero di requisiti_g *funzionali* e la loro suddivisione in implementati e non implementati è presente nella seguente tabella.

Realizzazione	Quantità
<i>Implementato</i>	35
<i>Non Implementato</i>	22
<i>Totali</i>	57

Tabella 5.8: MQPD01

Effettuando il calcolo di questa metrica, otteniamo una percentuale di totalità dei requisiti *funzionali* pari al 61%, valore non accettabile, dovuto all'alto numero di requisiti funzionali non implementati in quanto non obbligatori. Per migliorare questo valore il gruppo si impegna nel cercare di implementare quanti più possibili requisiti non obbligatori nel periodo che intercorre tra la revisione di qualifica e la revisione di accettazione.

5.3.2.2 MQPD02

Il numero di requisiti *prestazionali* e la loro suddivisione in implementati e non implementati è presente nella seguente tabella.

Realizzazione	Quantità
<i>Implementato</i>	3
<i>Non Implementato</i>	2
<i>Totali</i>	5

Tabella 5.10: MQPD02

Effettuando il calcolo di questa metrica, otteniamo una percentuale di totalità dei requisiti *prestazionali* pari al 80%, valore accettabile e sufficiente.

5.3.2.3 MQPD03

Il gruppo ha iniziato a svolgere i primi test di unità dei differenti moduli sviluppati. Di seguito è presente una tabella che indica il numero dei test eseguiti, i test che passano e i test che generano errori.



Nel caso in cui il gruppo riesca a trovare un'implementazione dei test di unità del modulo legato al machine learning_G, la metrica verrà aggiornata nella fase successiva

Modulo	Test totali	Test passati	Test non passati
<i>Acquisizione</i>	3	3	0
<i>Back-end</i>	4	4	0
<i>Front-end</i>	21	18	3

Tabella 5.12: **Tabella dei test di unità effettuati**

Di seguito è presente una tabella che indica il valore di questa metrica per ogni modulo sviluppato ed infine il valore somma di ogni modulo. L'unico modulo ancora non testato è quello di predizione, legato al machine learning_G, in quanto varia ad ogni esecuzione.

Modulo	Valore metrica
<i>Acquisizione</i>	0%
<i>Back-end</i>	14%
<i>Front-end</i>	0%
<i>Totale</i>	11%

Tabella 5.14: **MQPD03**

Il valore totale di questa metrica risulta non accettabile in quanto superiore al 10%. Il gruppo si propone di raggiungere il valore accettabile riducendo i test con errore presenti.

5.3.2.4 MQPD04

Il calcolo di questa metrica non è possibile automatizzarlo in quanto è relativo al numero di persone contate dal software contapersone. Il controllo e il calcolo della metrica avvengono in intervalli di tempo periodici programmati. La media ottenuta è circa pari al 75%.

5.3.2.5 MQPD05

Poiché questa metrica è legata alla comprensione del codice, il gruppo ha deciso di evidenziarla attraverso la seguente tabella dove verranno mostrati i risultati per ogni modulo di sviluppo ed infine il risultato somma di ogni modulo.

Modulo	Valore metrica
<i>Acquisizione</i>	27%
<i>Predizione</i>	18%
<i>Back-end</i>	11%
<i>Front-end</i>	0,2%
<i>Totale</i>	8%

Tabella 5.16: MQPD05

Il modulo di acquisizione presenta un valore non accettabile perché superiore al 20%. I commenti presenti all'interno dei file di tale modulo sono necessari alla comprensione del codice riguardante l'esecuzione delle librerie esterne. L'incidenza di questo valore non rende il totale superiore alla soglia preferibile prevista.

5.3.2.6 MQPD06

Di seguito è presente la tabella che indica il numero di volte che ogni modulo venga utilizzato da un dato modulo.

Modulo	Numero di chiamate	Modulo chiamante
<i>Acquisizione</i>	0	-
<i>Predizione</i>	0	-
<i>Back-end</i>	1	Front end
<i>Front-end</i>	0	-
<i>Totale</i>	1	Front end

Tabella 5.18: MQPD06

Il valore di questa metrica, come si può evincere dalla tabella, è pari al valore preferibile ovvero 1. Se si dovesse considerare il database come modulo, questo valore sarebbe nettamente maggiore ma il gruppo ha deciso di non includerlo nel conteggio perchè si otterrebbe un valore non veritiero dello structural fan-in.

5.3.2.7 Strategia adoperata per l'analisi statica dei documenti

La strategia adoperata per l'analisi statica dei documenti per la *Revisione di Progettazione* è la medesima di quella descritta in § 5.1.1.

5.3.2.7.1 Esiti verifica Per ciascun documento stilato si è calcolato l'indice di Gulpease_g. I risultati sono mostrati qui di seguito. Per evitare risultati errati nel calcolo di tale indice, non si sono tenuti in considerazione:

- il frontespizio di ogni documento;
- le eventuali tabelle presenti nei documenti;
- i diari delle modifiche di ogni documento.

Documento	Indice di Gulpease	Esito
<i>Analisi dei Requisiti 4.0.0</i>		<i>Superato</i>
<i>Norme di Progetto 3.0.0</i>		<i>Superato</i>
<i>Piano di Progetto 3.0.0</i>		<i>Superato</i>
<i>Piano di Qualifica 3.0.0</i>		<i>Superato</i>
<i>Manuale Utente 1.0.0</i>		<i>Superato</i>
<i>Manuale Sviluppatore 1.0.0</i>		<i>Superato</i>
<i>Product Baseline 1.0.0</i>		<i>Superato</i>

Tabella 5.19: Elenco Indici di Gulpease_g dei documenti fino alla RQ

Per quanto riguarda gli Indici di Gulpease_g dei verbali si è deciso di rappresentare i risultati in forma tabellare. Questo in quanto il verbale viene scritto tutta in una volta, quindi utilizzare un grafico temporale risulta non idoneo.



Documento	Indice di Gulpease	Esito
<i>v_e_2021-03-19</i>	98	<i>Superato</i>
<i>v_e_2021-03-25</i>	100	<i>Superato</i>
<i>v_i_2021-03-26</i>	99	<i>Superato</i>
<i>v_e_2021-03-29</i>	99	<i>Superato</i>
<i>v_e_2021-03-31</i>	100	<i>Superato</i>
<i>v_i_2021-04-09</i>	100	<i>Superato</i>
<i>v_i_2021-04-12</i>	100	<i>Superato</i>
<i>v_e_2021-04-15</i>	98	<i>Superato</i>
<i>v_e_2021-04-16</i>	99	<i>Superato</i>
<i>v_i_2021-04-19</i>	100	<i>Superato</i>

Tabella 5.20: **Elenco Indici di Gulpease_g dei verbali fino alla RQ**



6 Valutazioni per il miglioramento

Questa sezione riporta una valutazione complessiva sul lavoro svolto fino ad ora, con l'obiettivo di far emergere e, quindi, risolvere in maniera efficace tutte le problematiche portate all'attenzione durante agli incontri con il gruppo e segnalate di conseguenza nei verbali interni. In questo modo il gruppo si impegna ad evitare che queste si ripresentino in futuro.

A causa dell'assenza di una figura esterna che possa effettivamente fornire una valutazione oggettiva del lavoro svolto, questa si basa su un'autovalutazione di ciascun membro del gruppo. Nel caso in cui si presentassero nuove problematiche con l'avanzamento del lavoro, il gruppo provvederà ad integrare opportunamente la seguente sezione. Qui di seguito si trovano in forma tabellare le difficoltà incontrate durante ciascuna revisione per ogni tipologia di problema:

- la prima è legata all'organizzazione (§ 6.1);
- la seconda è legata ai ruoli (§ 6.2);
- la terza è legata agli strumenti (§ 6.3).

Nella tabella di ogni problematiche è presentata la descrizione del problema con la rispettiva soluzione, ed inoltre, ad ognuna è attribuito un livello di gravità che varia da 1, che indica il livello minimo di difficoltà nella corretta risoluzione, a 3, che indica il livello massimo. La strutturazione di queste tabelle è stata decisa ed uniformata all'interno delle *Norme di Progetto 3.0.0*

Questa sezione verrà continuamente aggiornata allo scadere delle revisioni in modo da tenere traccia dei problemi riscontrati durante lo sviluppo del progetto e delle relative soluzioni.



6.1 Valutazione su organizzazione

6.1.1 Revisione dei requisiti

Problema	Descrizione	Gravità	Soluzione
Incontro con il gruppo	Si è riscontrata una difficoltà nel riuscire ad organizzare tutti gli incontri in modo che ogni membro del gruppo fosse presente.	2	Si è fatto un Poll sul canale Discord del gruppo, in cui ciascun membro ha votato la propria preferenza. Alla fine si è raggiunti ad una decisione unanime.

Tabella 6.1: Tabella dei problemi relativi all'organizzazione - Revisione dei requisiti

6.1.2 Revisione di progettazione

Problema	Descrizione	Gravità	Soluzione
Comunicazione interna del gruppo	Si è riscontrata una difficoltà nel riuscire a comunicare internamente al sorgere di problemi con il rischio di accumularli e doverli risolvere troppo sommariamente	2	Si è deciso di aumentare l'impegno da parte di ogni componente di comunicare più spesso in modo da poter risolvere i possibili problemi che possono sorgere durante lo sviluppo
Organizzazione interna del gruppo	Si è riscontrata una difficoltà nello scandire il tempo ed i compiti portando ad uno sviluppo poco organizzato sia dei documenti che del prodotto software	2	Si è deciso di suddividere a monte ogni obiettivo da raggiungere per poter avere una migliore organizzazione e tracciabilità di cosa manca per ottenere un prodotto di qualità
Utilizzo di Trello	Si è riscontrata una difficoltà nell'utilizzo di Trello	2	Per risolvere il problema il gruppo ha deciso di impegnarsi nell'utilizzo della piattaforma



Tabella 6.2: Tabella dei problemi relativi all'organizzazione - Revisione di progettazione

6.1.3 Revisione di qualifica

Problema	Descrizione	Gravità	Soluzione
Lavoro interno al gruppo	Alcuni componenti del gruppo non hanno lavorato in maniera efficiente in modo da pesare sul lavoro generale in prossimità delle scadenze	2	Per arginare questa difficoltà abbiamo deciso di organizzare ulteriormente il lavoro e aumentare i resoconti del lavoro svolto in modo da poter tenere traccia degli obiettivi da svolgere

Tabella 6.3: Tabella dei problemi relativi all'organizzazione - Revisione di Qualifica

6.2 Valutazione sui ruoli

6.2.1 Revisione dei requisiti



Problema	Descrizione	Gravità	Soluzione
Rivestimento del ruolo di <i>Responsabile</i>	A causa dell'inesperienza, la maggiore difficoltà riscontrata nel rivestire il ruolo di <i>Responsabile</i> è stata la stima delle risorse necessarie ed un'assegnazione adeguata delle stesse	2	Per arginare tale difficoltà, in questa fase iniziale del progetto, il gruppo si aggiorna con maggior frequenza per avere un riscontro sulle stime e per poterle correggere
Rivestimento del ruolo di <i>Analista</i>	Nessuno del gruppo ha redatto tale documentazione prima, per questo motivo è risultato difficile comprendere la struttura e le "competenze" di ogni documento	2	Abbiamo cercato di capire più a fondo le indicazioni del committente _G e ci siamo confrontati tra di noi per cercare di trovare la soluzione migliore.
Rivestimento del ruolo di <i>Amministratore</i>	Il ruolo di <i>Amministratore</i> inizialmente ha creato delle problematiche relative all'approfondimento degli standard ISO per capire come adattarli al nostro progetto, mantenendo la qualità.	2	Tutti i membri del gruppo hanno contribuito alla ricerca di materiale informativo e condiviso le informazioni con gli altri membri, per velocizzare l'apprendimento iniziale.

Tabella 6.4: Tabella dei problemi relativi ai ruoli - Revisione dei requisiti



6.2.2 Revisione di progettazione

Problema	Descrizione	Gravità	Soluzione
Rivestimento del ruolo di <i>Progettista</i>	A causa dell'inesperienza tecnologica il gruppo ha ritrovato difficoltà ad organizzare lo sviluppo dell'architettura del prodotto software	3	Come soluzione il gruppo ha deciso di impiegare più ore nello studio personale per conoscere meglio le tecnologie con cui dovrà lavorare
Rivestimento del ruolo di <i>Analista</i>	Il gruppo, dato lo studio poco approfondito del progetto e dei suoi requisiti, ha riscontrato difficoltà nello svolgere un'analisi corretta che ha portato ad un'Analisi dei Requisiti non sufficiente	2	Il gruppo ha organizzato incontri con l'azienda proponente ed il professor Riccardo Cardin per poter sanare le lacune e produrre un'Analisi dei Requisiti soddisfacente

Tabella 6.5: Tabella dei problemi relativi ai ruoli - Revisione di progettazione



6.2.3 Revisione di qualifica

Problema	Descrizione	Gravità	Soluzione
Rivestimento del ruolo di <i>Programmatore</i>	A causa dell'inesperienza tecnologica, il gruppo ha riscontrato difficoltà a padroneggiare correttamente i linguaggi e framework da usare. Ciò ha portato anche a produrre del codice poco chiaro ed efficiente	3	Per arginare il problema i componenti del gruppo hanno deciso di dedicare molto tempo allo studio individuale per poter colmare le lacune e produrre del codice soddisfacente
Rivestimento del ruolo di <i>Analista</i>	Il gruppo, in seguito alle correzioni ricevute alla revisione di progettazione, ha dovuto impiegare più ore di quelle preventivate per questo ruolo per produrre una versione migliore del documento <i>Analisi dei Requisiti</i>	3	Per risolvere il problema il gruppo ha organizzato incontri con l'azienda proponente e ha seguito le indicazioni del professor Riccardo Cardin tramite corrispondenza via email

Tabella 6.6: Tabella dei problemi relativi ai ruoli - Revisione di qualifica

6.3 Valutazione su strumenti di lavoro

6.3.0.1 Revisione dei requisiti



Problema	Descrizione	Gravità	Soluzione
<i>GitHub_G</i>	Alcuni membri del gruppo avevano meno esperienza con l'uso di questo strumento, quindi ci sono state alcune difficoltà iniziali.	2	Per risolvere tale problema, i membri meno pratici si sono impegnati nel sanare le loro lacune e quelli più ferrati, invece, si resi disponibili nell'aiutare chi in difficoltà.
L ^A T _E X	Per via dell'inesperienza della maggior parte dei membri del gruppo riguardo l'uso di tale strumento, si sono riscontrate diverse difficoltà, specie con la costruzione di tabelle ed il frontespizio.	2	Per cercare di risolvere in breve tempo il problema, si è dedicato del tempo nelle prime settimane all'apprendimento di questo strumento.

Tabella 6.7: Tabella dei problemi relativi agli strumenti di lavoro - Revisione dei requisiti

6.3.0.2 Revisione dei progettazione

Problema	Descrizione	Gravità	Soluzione
Software di riconoscimento oggetti	Il gruppo ha riscontrato difficoltà nel trovare ed implementare un software di riconoscimento oggetti	3	Per risolvere questa problematica il gruppo ha effettuato più ricerche contemporaneamente testando ogni risultato ottenuto per poi decidere quale software potesse essere il migliore per lo sviluppo del nostro progetto.

Tabella 6.8: Tabella dei problemi relativi agli strumenti di lavoro - Revisione di progettazione

6.3.0.3 Revisione di qualifica



Problema	Descrizione	Gravità	Soluzione
Difficoltà nell'implementare il framework leaflet	Per generare correttamente la mappa il gruppo ha riscontrato difficoltà nell'implementare il framework leaflet come componente di Vue.js	3	Più membri del gruppo hanno dedicato diverse ore a documentarsi per risolvere questa problematica il prima possibile, così da riuscire ad avanzare con lo sviluppo della web application
Difficoltà nel gestire le dipendenze in Vue.js	Il gruppo ha riscontrato difficoltà nel gestire le dipendenze tra le differenti componenti	2	I componenti del gruppo che dovevano sviluppare questa parte hanno impiegato molto tempo a leggere la documentazione e a fare esercizio pratico per poter comprendere al meglio il funzionamento del framework.
Difficoltà nella scrittura delle query lato backend	Il gruppo ha riscontrato una difficoltà iniziale a comprendere come strutturare il codice per eseguire correttamente le query al database	2	Per risolvere questa problematica si è effettuato uno studio della documentazione trovando la soluzione più adatta al nostro problema



Sviluppo del modulo legato al machine learning _c	Il gruppo ha riscontrato diverse difficoltà nello sviluppare un modello funzionante di machine learning	3	Per risolvere questa problematica Dario Stagnitto, dipendente dell'azienda proponente, durante un incontro, ha proposto di non utilizzare più keras, libreria troppo complessa per le conoscenze pregresse del gruppo legate al machine learning, e di utilizzare invece scikit learn che ha una curva di apprendimento molto più bassa. Inoltre il gruppo ha mantenuto una corrispondenza con Dario Stagnitto il quale ha aiutato allo sviluppo dell'ambiente in modo da fornire delle buone basi da cui partire per procedere con lo sviluppo.
---	---	---	--

Tabella 6.10: **Tabella dei problemi relativi agli strumenti di lavoro - Revisione di qualifica**

7 Esiti delle revisioni

7.1 Revisione dei requisiti

Successivamente alla prima revisione il gruppo, basandosi sulla prima valutazione, ha apportato diverse modifiche. Di seguito vengono elencate le modifiche effettuate:

- aggiunta il numero del capitolo in ogni documento;
- modifica della denominazione dei verbali in modo da poter ordinarli tramite una codifica alfanumerica;
- in tutti i documenti il gruppo ha rielaborato la tabella del registro delle modifiche in modo tale che sia coerente con lo scatto di versione legato al modello incrementale;
- ristrutturazione dell'*Analisi dei requisiti* attraverso l'aggiunta:
 - dei casi d'uso concordati con l'azienda e i requisiti collegati ad essi;
 - della tabella riassuntiva rappresentate il numero dei requisiti ed il loro tipo;
- ristrutturazione del *Piano di Progetto* attraverso la modifica del capitolo 3, relativo al modello di sviluppo;
- ristrutturazione delle *Norme di Progetto* attraverso la modifica dei capitoli 2 e 3;
- ristrutturazione del *Piano di Qualifica* attraverso:
 - l'aggiunta del capitolo 7 e della § 5.2;
 - riorganizzazione dei capitoli 2,4,5.

7.2 Revisione di progettazione

In seguito alla Revisione di progettazione, basandosi sulla valutazione data, il gruppo ha apportato diverse modifiche sia ai documenti che al codice prodotto. Di seguito vengono elencate le modifiche effettuate:

- riscrittura del codice già presente per renderlo più leggibile ed ordinato;
- proseguimento della stesura del codice per soddisfare tutti i requisiti obbligatori.
- stesura dei verbali interni ed esterni di tutti gli incontri interni e delle comunicazioni esterne;



- ristrutturazione della tabella del registro delle modifiche in tutti i documenti in modo da rendere più chiaro chi effettua la modifica e chi la verifica;
- modifica del formato della data in tutti i documenti in modo tale da seguire il formato AAAA-MM-GG;
- stesura del documento *Manuale Utente*;
- stesura del documento *Manuale Sviluppatore*;
- stesura dell'allegato tecnico *Product Baseline*;
- ristrutturazione del *Piano di Progetto* attraverso:
 - rivisitazione dei capitoli 4, 5 e 6;
 - l'aggiunta del capitolo 7;
- ristrutturazione dell'*Analisi dei requisiti* attraverso:
 - studio ulteriore e riorganizzazione del capitolo 3 dove si sono modificati i casi d'uso;
 - l'aggiornamento del capitolo 4, ovvero dei requisiti, in funzione della nuova organizzazione dei casi d'uso;
- ristrutturazione delle *Norme di Progetto* attraverso:
 - l'aggiunta della sezione legata ai processi di miglioramento del capitolo 4;
 - aggiunti strumenti di codifica ??
- ristrutturazione del *Piano di Qualifica* attraverso:
 - l'aggiornamento dei capitoli 4, 6 e 7;
 - l'aggiunta di grafici legati alle metriche del capitolo 5.