

Tweet Analysis

Relazione finale

Cheikh Ibrahim · Zaid

PO Operativo

Matricola: 0000974909

Xia · Tian Cheng

Scrum master

Matricola: 0000975129

Lee · Qun Hao Henry

Developer

Matricola: 0000990259

Paris · Manuel

Developer

Matricola: 0000997526

Anno accademico

2022 — 2023

Corso di Ingegneria del Software

Alma Mater Studiorum · Università di Bologna

Indice

1	Descrizione del prodotto	2
1.1	Scope	2
1.2	Casi d'uso	2
1.3	Diagramma delle classi	2
2	Descrizione degli sprint	3
2.1	Sprint 1	3
2.1.1	Sprint goal	3
2.1.2	Backlog	3
2.1.3	Burndown	4
2.1.4	Retrospettiva	5
2.2	Sprint 2	6
2.2.1	Sprint goal	6
2.2.2	Backlog	7
2.2.3	Burndown	8
2.2.4	Retrospettiva	9
2.3	Sprint 3	9
2.3.1	Sprint goal	9
2.3.2	Backlog	9
2.3.3	Burndown	11
2.3.4	Retrospettiva	12
2.4	Sprint 4	12
2.4.1	Sprint goal	12
2.4.2	Backlog	12
2.4.3	Burndown	12
2.4.4	Retrospettiva	13
3	Descrizione del processo	14
3.1	Team building	14
3.1.1	Scrumble	14
3.1.2	Escape the Boom	14
3.2	Gitinspector	14
3.3	Retrospettiva finale	14
3.4	Deployment	14
4	Artefatti	15

1 Descrizione del prodotto

1.1 Scope

1.2 Casi d'uso

1.3 Diagramma delle classi

2 Descrizione degli sprint

Sono stati svolti quattro sprint della durata di 14 giorni ciascuno.

La stima dei punti delle user stories è stata effettuata con una scala da 0 a 10, valutando separatamente il frontend dal backend. Il punteggio complessivo è quindi ottenuto dalla somma di quest'ultimi.

2.1 Sprint 1

2.1.1 Sprint goal

Lo sprint è stato principalmente dedicato a studiare le API di Twitter e produrre le prime funzionalità per la visualizzazione e l'analisi dei tweet.

In particolare le feature pianificate per lo sprint sono state:

- Ricerca di tweet per username
- Ricerca di tweet per hashtag
- Analisi dei tweet tramite componenti grafiche (grafico a torta per il sentiment analysis, grafico a barre per la frequenza dei tweet e word cloud)

2.1.2 Backlog

US: Come utente interessato ai tweet, voglio poter cercare dei tweet per hashtag per leggerli.	Punti: 8 (3 frontend + 5 backend)
---	---

DOD: L'utente, cercando un hashtag in un apposito textbox, è in grado di leggere tutti i tweet correlati visualizzando: nome account Twitter, username, immagine profilo, contenuto Tweet (testo + foto e video), data e ora, luogo (se applicabile), numero like, numero commenti, numero retweet

Test: Richiamare l'API implementata, verificare che il formato sia corretto e che il contenuti dei tweet contenga l'hashtag ricercato

US: Come utente interessato ai tweet, voglio poter cercare dei tweet per nome utente per leggerli.	Punti: 8 (3 frontend + 5 backend)
---	---

DOD: L'utente, cercando un nome utente in un apposito textbox, è in grado di leggere tutti i tweet correlati visualizzando: nome account Twitter, username, immagine profilo, contenuto Tweet (testo + foto e video), data e ora, luogo (se applicabile), numero like, numero commenti, numero retweet

Test: Richiamare l'API implementata, verificare che il formato sia corretto e che l'autore dei tweet sia quello ricercato

<p>US: Come analista, voglio poter analizzare il sentimento per stabilire se un tweet è positivo o meno.</p> <p>DOD: L'utente, dato un tweet, vede se è positivo, negativo o neutro tramite immagine o testo.</p> <p>Test: Analizzare frasi di cui è noto il sentimento</p>	<p>Punti: 9 (2 frontend + 7 backend)</p>
--	---

<p>US: Come analista, voglio vedere un grafico a barre per vedere il numero di tweet nell'unità di tempo.</p> <p>DOD: L'utente apre una pagina web contenente il grafico a barre con il numero di tweet nell'unità di tempo.</p> <p>Test: Manualmente verificare che il grafico sia corretto</p>	<p>Punti: 2 (2 frontend)</p>
---	---

<p>US: Come analista, voglio vedere un grafico a torta per vedere il rapporto di sentiment positivi, negativi e neutri.</p> <p>DOD: L'utente apre una pagina web contenente il grafico a torta con sentiment positivi, negativi e neutri.</p> <p>Test: Manualmente verificare che il grafico sia corretto</p>	<p>Punti: 2 (2 frontend)</p>
--	---

<p>US: Come analista, voglio vedere una term cloud per vedere le parole più utilizzate nei tweet.</p> <p>DOD: L'utente apre una pagina web contenente una term cloud con le parole più utilizzate.</p> <p>Test: Manualmente verificare che il grafico sia corretto</p>	<p>Punti: 4 (2 frontend + 2 backend)</p>
---	---

2.1.3 Burndown

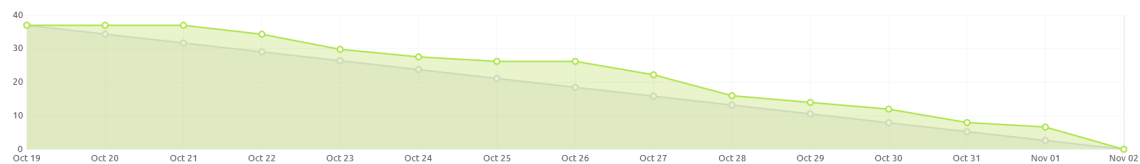


Figura 1: Burndown

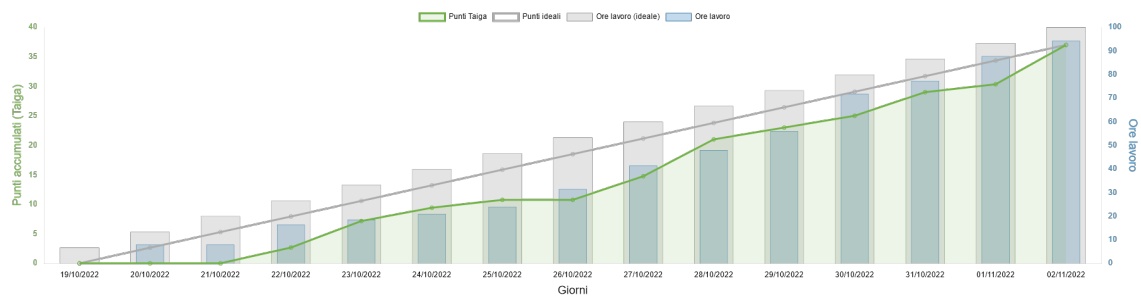


Figura 2: Progresso dei punti (asse a sinistra) e ore di lavoro (asse a destra)

2.1.4 Retrospettiva

	<div><div></div><div>Scrum Team</div><div>The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master. Scrum Teams are:<ul style="list-style-type: none">• Self-organizing• Cross-functional• Flexible• Creative• ProductiveScrum Teams deliver products iteratively and incrementally, maximizing opportunities for feedback.</div><div><div></div><div>Applies to: <input checked="" type="checkbox"/> Team <input type="checkbox"/> Way of Working</div><div></div><div>2.04</div></div></div>	<div><div></div><div>Definition of Done</div><div>The quality criteria used to assess when work is complete on the product Increment. Any one product or system should have a definition of done that is standard for any work done on it.</div><div><div>Completion Conditions Listed</div><div>Quality Criteria and Evidence Described</div></div><div><div></div><div>Describes: <input type="checkbox"/> Team <input checked="" type="checkbox"/> Way of Working</div><div></div><div>2.04</div></div></div> <div>Poco dettagliata</div>	
		<div><div></div><div>Scrum Pillars</div><div>Scrum is founded on empirical process control theory, or empiricism. Three pillars uphold every implementation of empirical process control:<ul style="list-style-type: none">• Transparency• Inspection• AdaptationIn turn they are the foundation of every successful Scrum implementation.</div><div><div></div><div>Applies to: <input type="checkbox"/> Team and <input checked="" type="checkbox"/> Way of Working</div><div></div><div>2.04</div></div></div> <div>Comunicazione presente, ma ridotta</div>	

Figura 3: Pre-retrospettiva del 27/10/2022

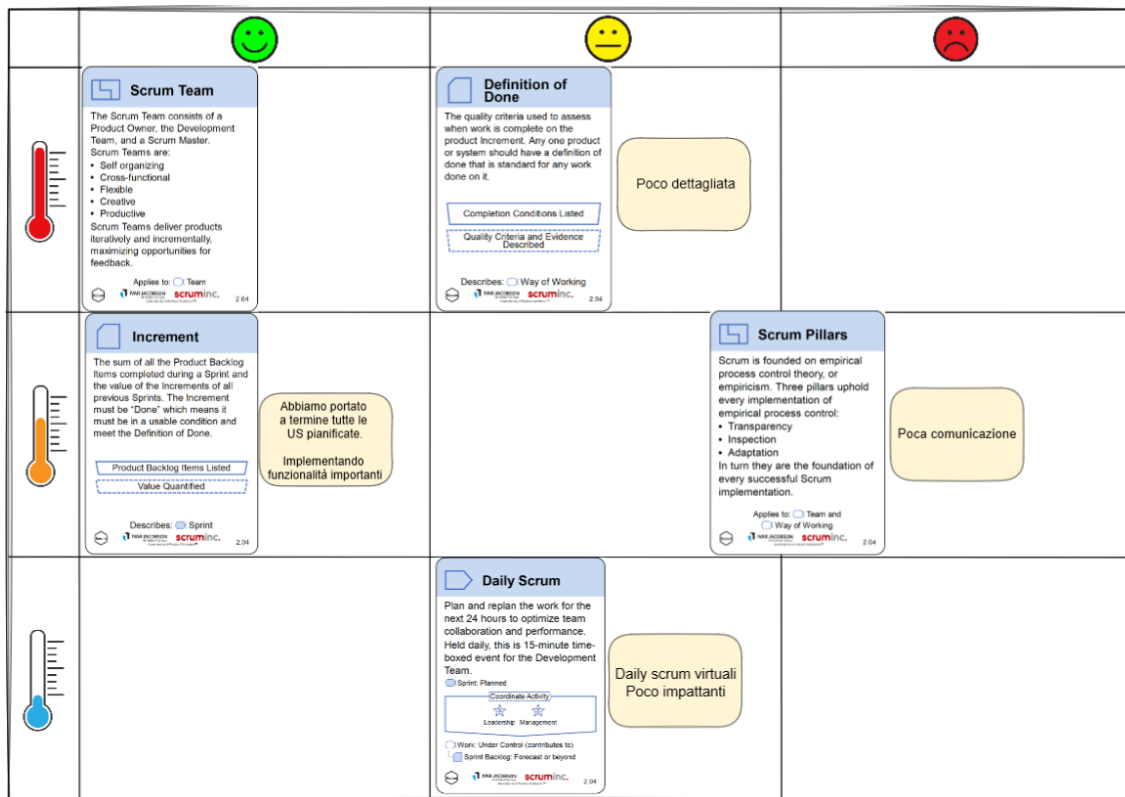


Figura 4: Retrospettiva del 01/11/2022

2.2 Sprint 2

2.2.1 Sprint goal

L'obiettivo dello sprint è stato quello di concludere l'epica riguardante la visualizzazione e l'analisi dei tweet.

Nello specifico, sono state implementate le seguenti funzionalità:

- Ricerca di tweet per intervallo temporale (richiesto dal cliente allo sprint review precedente)
- Ricerca di un determinato numero di tweet con una singola ricerca (richiesto dal cliente allo sprint review precedente)
- Ricerca di tweet per parola chiave
- Mappa per visualizzare la posizione dei tweet con geolocalizzazione
- Raccolta di tweet in tempo reale

2.2.2 Backlog

US: Come utente,
voglio scegliere un numero di tweet da poter caricare in
una volta
per poterli analizzare in modo aggregato.

Punti: 5
(3 frontend + 2 backend)

DOD: Possibilità di ricercare un largo numero di tweet scrivendone la quantità in un textbox. Tale quantità serve per la ricerca iniziale e anche per mostrare pagine successive. È possibile inoltre ricercare un numero di tweet inizialmente per poi cambiare quantità e cercare una pagina successiva (esempio: ricerca di 150 tweet iniziali, poi viene modificata la quantità (dallo stesso textbox) in 20 e si preme su “Pagina successiva”. Il numero di tweet così mostrato diventa 170).

US: Come utente,
voglio scegliere un intervallo di tempo e raccogliere tweet
per analizzarne le tendenze storiche.

Punti: 5
(3 frontend + 2 backend)

DOD: Possibilità di ricercare dei tweet dato un intervallo temporale. Non deve essere possibile cercare tweet nel futuro. Non deve essere possibile cliccare su “Prossima pagina” quando non ci sono più tweet da visualizzare.

US: Come utente interessato a vedere tweet,
voglio poter cercare dei tweet per parola chiave
per vedere cosa ne pensa la gente a riguardo.

Punti: 4
(2 frontend + 2 backend)

DOD: Possibilità di cercare tweet per parola o frase chiave. I grafici già presenti devono funzionare anche con questa ricerca.

US: Come utente interessato ai tweet,
voglio poter visualizzare su una mappa la posizione dei
tweet cercati
per avere un’idea della località dalla quale sono stati
pubblicati.

Punti: 7
(3 frontend + 4 backend)

DOD: Possibilità di visualizzare una mappa con le posizioni dei tweet ricercati. Se sono presenti più tweet nella stessa zona è possibile aggregarli (in base alla distanza) e mostrare un unico valore, ovvero il numero di tweet in tale zona. La mappa deve essere sempre visibile anche durante lo scorrimento della pagina (come i grafici).

US: Come utente,
voglio vedere la posizione di tutti i tweet di una data
persona
per conoscere i suoi spostamenti.

Punti: 5
(5 frontend + 0 backend)

DOD: Possibilità di inserire il nome utente di una persona e visualizzare su una mappa le posizioni dei suoi tweet e i suoi spostamenti. Per gli spostamenti si mostrano sulla mappa delle frecce basandosi sulla posizione e sulla data del tweet.

US: Come lettore di tweet,
voglio poter vedere i tweet che ricerco in tempo reale
per sapere cosa la gente posta.

Punti: 10
(3 frontend + 7 backend)

DOD:

2.2.3 Burndown

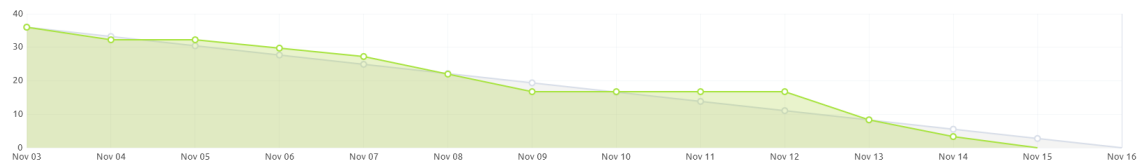


Figura 5: Burndown

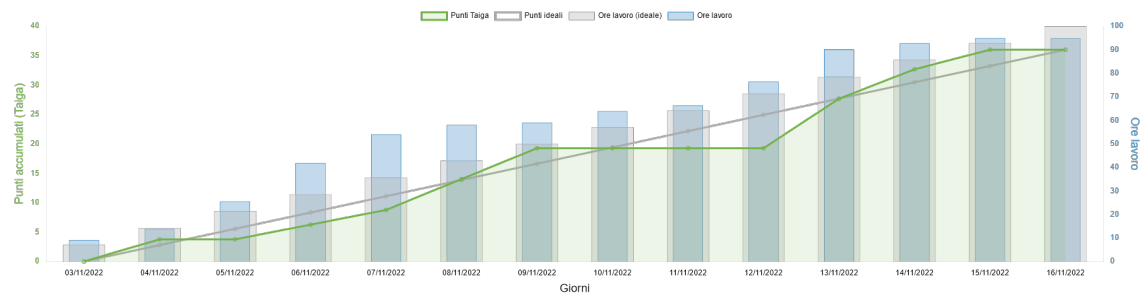


Figura 6: Progresso dei punti (asse a sinistra) e ore di lavoro (asse a destra)

2.2.4 Retrospettiva

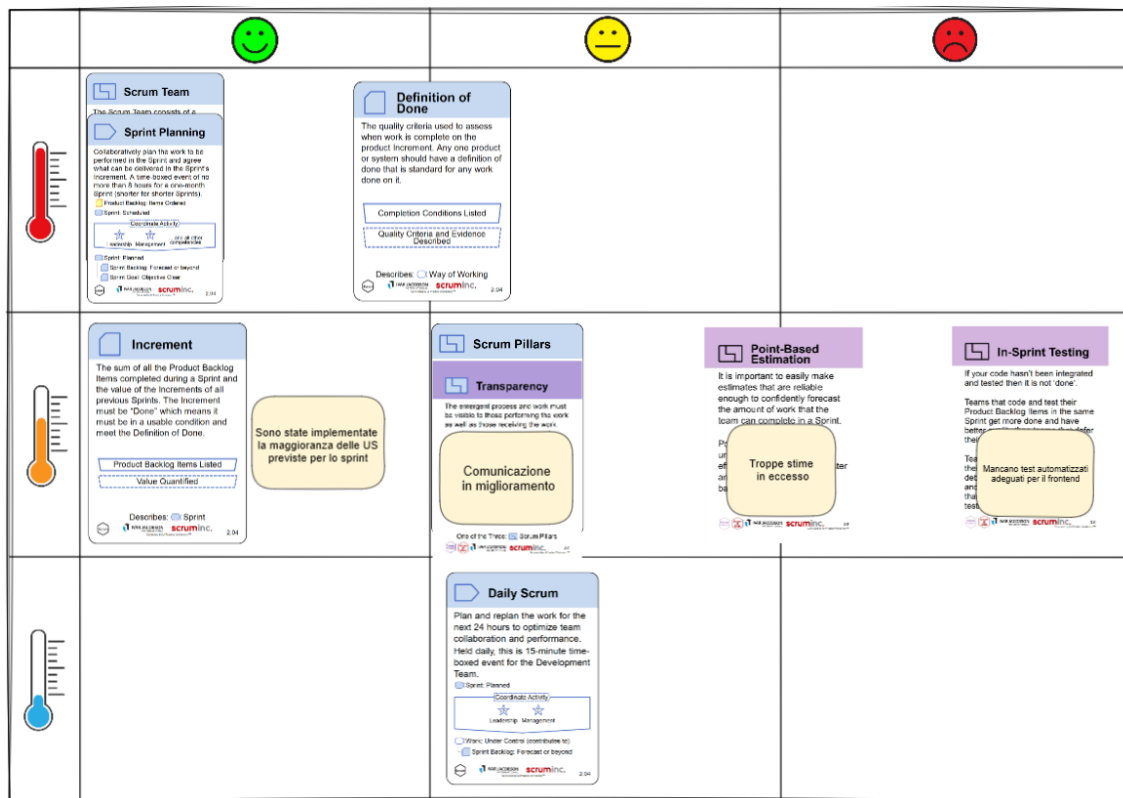


Figura 7: Pre-retrospettiva del 10/11/2022

2.3 Sprint 3

2.3.1 Sprint goal

2.3.2 Backlog

US: Come spettatore de #leredita
voglio raccogliere i tweet di chi prova a indovinare la
ghigliottina,
per visualizzare, in ordine temporale, tutti coloro che
provano ad indovinare.

Punti: 5
(2 frontend + 3 backend)

DOD: Possibilità di visualizzare una pagina con tutti i tweet di tutte le persone che
usano l' #leredita e provano ad indovinare la ghigliottina.

Test: Cercando per data, verificare che vengano visualizzati i tentativi del giorno

US: Come spettatore de #leredita
voglio visualizzare, su una mappa, la posizione di tutti
coloro che provano ad indovinare
per conoscere la posizione di giocatori.

Punti: 4
(4 frontend)

DOD: Possibilità di visualizzare una pagina con tutte le posizioni su una mappa di
tweet di tutte le persone che usano l’#leredita e provano ad indovinare la ghigliottina.

Test: Manualmente verificare che nella mappa siano presenti i marker dei tweet con la
geolocalizzazione

US: Come spettatore de #leredita
voglio visualizzare tutti coloro che indovinano la
ghigliottina
per sapere chi ha indovinato.

Punti: 7
(4 frontend + 3 backend)

DOD: Possibilità di visualizzare la parola del giorno assieme a tutti i vincitori che
hanno indovinato.

Test: Cercando per data, verificare che venga trovata la parola del giorno

US: Come spettatore di #reazioneacatena
voglio raccogliere i tweet di chi prova a indovinare
l’ultima parola,
per visualizzare, in ordine temporale, tutti coloro che
provano ad indovinare.

Punti: 2
(2 backend)

DOD: Possibilità di visualizzare una pagina con tutti i tweet di tutte le persone che
usano l’#reazioneacatena e provano ad indovinare la parola finale.

Test: Cercando per data, verificare che vengano visualizzati i tentativi del giorno

US: Come spettatore di #reazioneacatena
voglio visualizzare, su una mappa, la posizione di tutti
coloro che provano ad indovinare
per conoscere la posizione di giocatori.

Punti: 0

DOD: Possibilità di visualizzare una pagina con tutte le posizioni su una mappa di
tweet di tutte le persone che usano l’#reazioneacatena e provano ad indovinare la parola
finale.

Test: Manualmente verificare che nella mappa siano presenti i marker dei tweet con la
geolocalizzazione

US: Come spettatore de #reazioneacatena
voglio visualizzare tutti coloro che indovinano l’ultima
parola
per sapere chi ha indovinato.

Punti: 2
(0 frontend + 2 backend)

DOD: Possibilità di visualizzare la parola del giorno assieme a tutti i vincitori che
hanno indovinato.

Test: Cercando per data, verificare che venga trovata la parola del giorno

US: Come giocatore di scacchi Voglio poter muovere una pedina Per fare la mia mossa	Punti: 12 (6 frontend + 6 backend)
DOD: Possibilità di fare una mossa a scacchi e visualizzarla	
Test: Provare a effettuare mosse valide e invalide	

2.3.3 Burndown

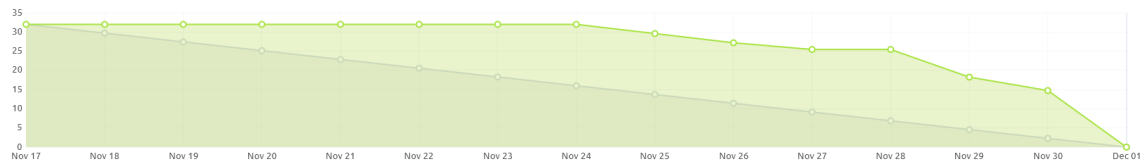


Figura 8: Burndown

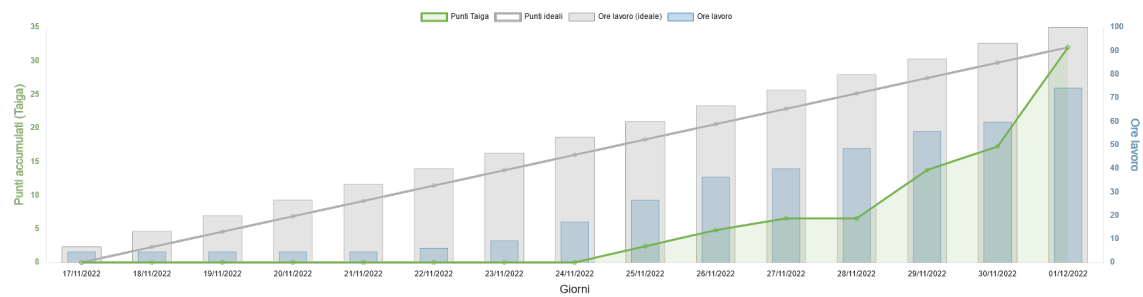


Figura 9: Progresso dei punti (asse a sinistra) e ore di lavoro (asse a destra)

2.3.4 Retrospettiva

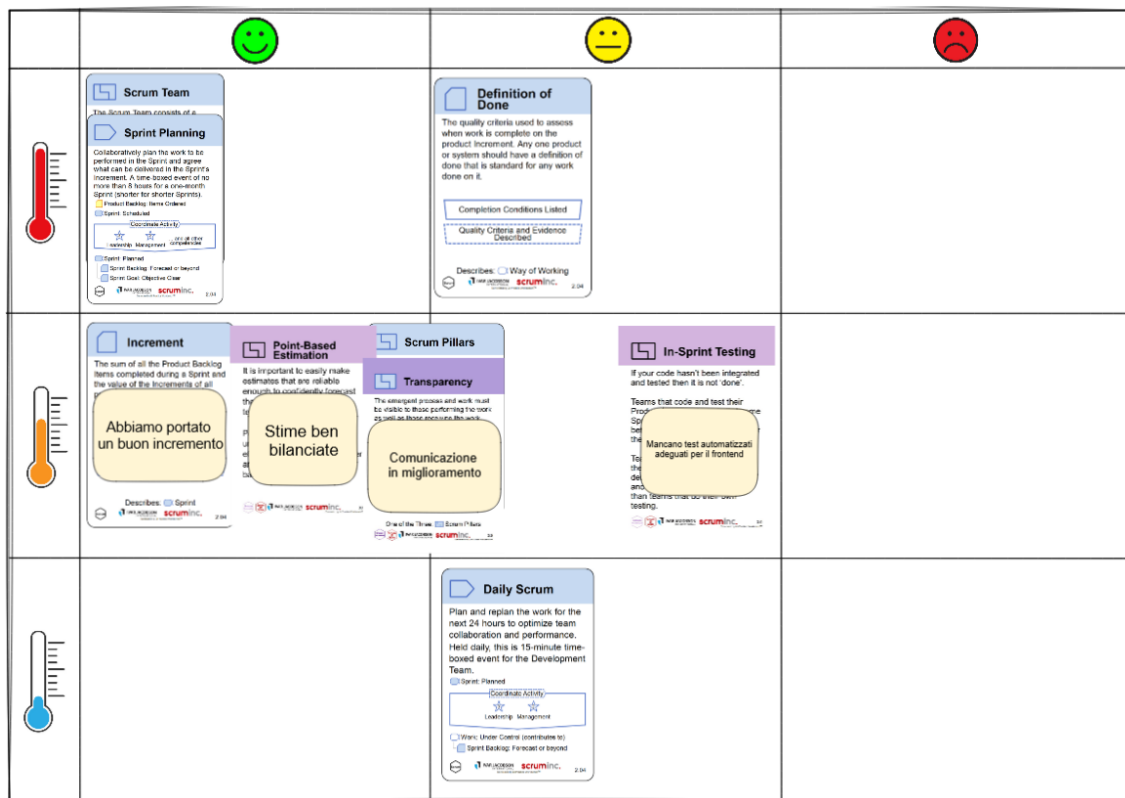


Figura 10: Pre-retrospettiva del 02/12/2022

2.4 Sprint 4

2.4.1 Sprint goal

2.4.2 Backlog

2.4.3 Burndown

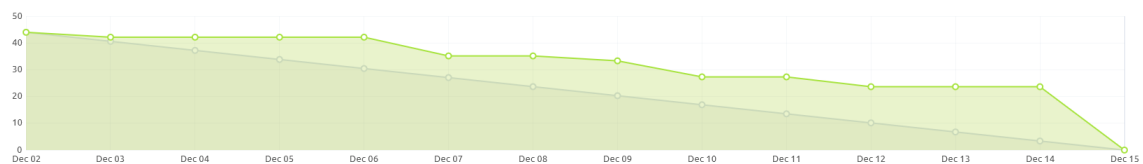


Figura 11: Burndown

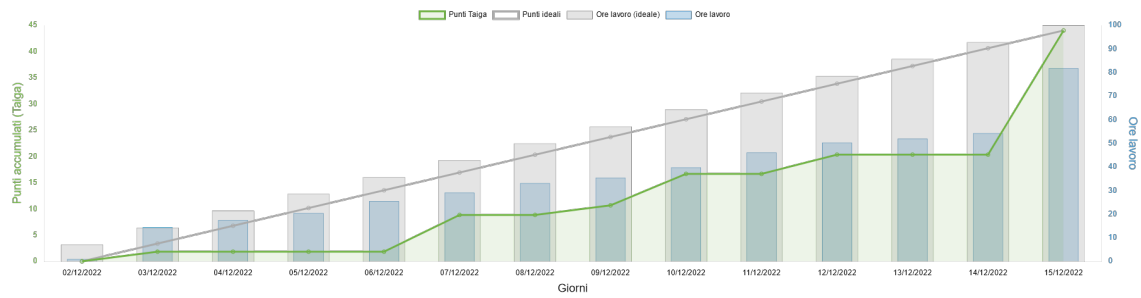


Figura 12: Progresso dei punti (asse a sinistra) e ore di lavoro (asse a destra)

2.4.4 Retrospettiva

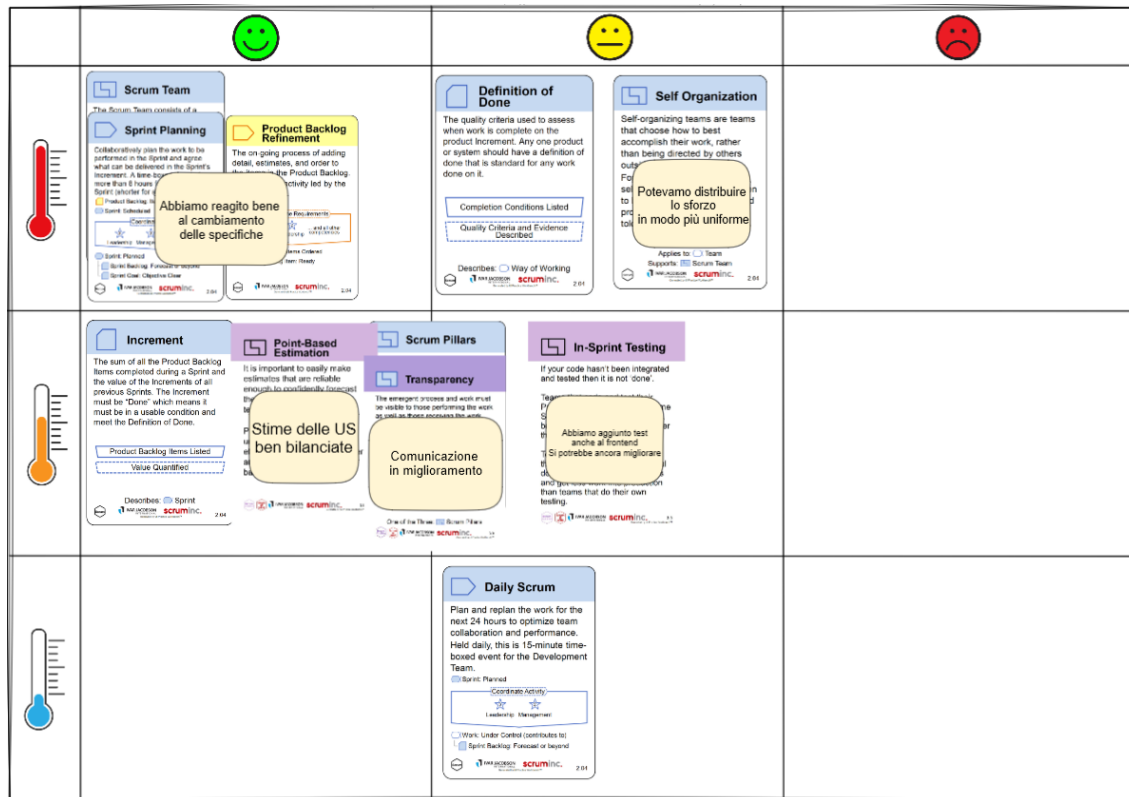


Figura 13: Pre-retrospettiva del 02/12/2022

3 Descrizione del processo

3.1 Team building

3.1.1 Scrumble

3.1.2 Escape the Boom

3.2 Gitinspector

3.3 Retrospettiva finale

3.4 Deployment

4 Artefatti