



**UNIVERSITÀ POLITECNICA DELLE MARCHE
FACOLTÀ DI INGEGNERIA**

**Corso di Laurea magistrale in Ingegneria Informatica e
dell'Automazione**

**Progettazione e sviluppo di un sistema di gestione
di una stazione osservativa mediante un approccio
Scrum e LeanUX.**

**Design and development of an observation station
management system using a Scrum and LeanUX
approach**

**Relatore:
Prof. LUCA SPALAZZI**

**Tesi di Laurea di:
ANDREA LANGONE**

**Correlatore:
Ing. MATTEO CANZARI**

SOMMARIO

SOMMARIO	2
INDICE FIGURE	4
INTRODUZIONE	6
DOMINIO APPLICATIVO	8
METODOLOGIE PROGETTUALI	10
a. AGILE	11
Le origini della metodologia AGILE	12
i. Scrum	13
Come funziona Scrum?	14
Ruoli	14
Artefatti	15
Eventi	17
ii. Agile vs V-Model	19
Che cos'è il V-Model?	19
Quale è la differenza tra modello Agile e V-Model?	21
b. LeanUX	22
LeanUX - Cos'è?	22
La necessità di Ipotesi in LeanUX	22
Creare un'Ipotesi in LeanUX	24
Il Minimum Viable Product (MVP) e LeanUX	24
Ricerca e test in LeanUX	25
I tre pilastri fondamentali del LeanUX	26
i. LeanUX Canvas	28
c. Vantaggi integrazione LeanUX con Scrum	32
TECNOLOGIE	35
OpenProject	35
Features OpenProject	37
PROGETTAZIONE e SVILUPPO	40
a. Implementazione Agile	41
Ruoli	41
Artefatti	42
Eventi	45

b. Implementazione LeanUx	47
c. Struttura sistema e resoconto Sprint	50
Sprint 1	50
Sprint Review 1	52
Menù, Wiki e Struttura Progetti	54
Sprint 2	58
Sprint Review 2	66
Logistica	67
Sprint 3	71
Sprint Review 3	72
Strumentazione Scientifica	73
Sprint 4	79
Sprint Review 4	80
Attività Scientifica	81
USABILITA' e ACCESSIBILITA'	86
Questionario	86
Analisi risultati	89
Risultati USABILITÀ	89
Risultati ACCESSIBILITÀ	93
Risultati finali	97
Considerazioni risultati domande generiche	101
CONCLUSIONI	103
Sviluppi futuri	105
APPENDICE	106
Osservatorio d'Abruzzo - INAF	106
TASK	109
INSTALLAZIONE OPENPROJECT e AMBIENTE DI SVILUPPO	112
Installazione OpenProject	112
Ambiente di Sviluppo	114
WEBGRAFIA	115
RINGRAZIAMENTI	117

INDICE FIGURE

Figura 1: <i>Manifesto AGILE</i>	12
Figura 2: <i>Ciclo sviluppo / progettazione modello Waterfall.</i>	19
Figura 3: <i>Ciclo sviluppo / progettazione V-Model</i>	20
Figura 4: <i>Minimum Viable Product</i>	25
Figura 5: <i>Design Thinking</i>	26
Figura 6: <i>LeanUX Canvas V2</i>	29
Figura 7: <i>Logo OpenProject</i>	35
Figura 8: <i>Sottoscrizioni offerte da OpenProject</i>	36
Figura 9: <i>Ciclo progettazione OpenProject</i>	36
Figura 10: <i>Board Agile di Trello durante lo Sprint 3</i>	44
Figura 11: <i>Parte di tabella descrittiva della struttura di una sezione del sistema di gestione</i>	49
Figura 12: <i>Struttura provvisoria dei progetti</i>	54
Figura 13: <i>Esempio di una pagina Wiki</i>	55
Figura 14: <i>menù di default e menù ridotto</i>	56
Figura 15: <i>Esempio di “Panoramica” di un progetto</i>	57
Figura 16: <i>Esempio di “Macro-attività” di un progetto.</i>	57
Figura 17: <i>Esempio di “Membri” di un progetto</i>	57
Figura 18: <i>Sottoprogetti definiti per il macro-progetto LOGISTICA</i>	67
Figura 19: <i>Esempio di un TASK</i>	70
Figura 20: <i>Sottoprogetti definiti per il macro-progetto STRUMENTAZIONE SCIENTIFICA</i>	74
Figura 21: <i>Esempio di un TASK in STRUMENTAZIONE SCIENTIFICA</i>	78
Figura 22: <i>Sottoprogetti definiti per il macro-progetto ATTIVITA’ SCIENTIFICA</i>	81
Figura 23: <i>Esempio di un TASK in ATTIVITA’ SCIENTIFICA</i>	84
Figura 24: <i>Esempio elenco di TASK relazionati</i>	84
Figura 25: <i>Grafico risultati domanda 1 USABILITÀ</i>	89
Figura 26: <i>Grafico risultati domanda 2 USABILITÀ</i>	90
Figura 27: <i>Grafico risultati domanda 3 USABILITÀ</i>	90
Figura 28: <i>Grafico risultati domanda 4 USABILITÀ</i>	91

Figura 29: <i>Grafico risultati domanda 5 USABILITÀ</i>	91
Figura 30: <i>Grafico risultati domanda 6 USABILITÀ</i>	92
Figura 31: <i>Grafico risultati domanda 7 USABILITÀ</i>	92
Figura 32: <i>Grafico risultati domanda 8 USABILITÀ</i>	93
Figura 33: <i>Grafico risultati domanda 1 ACCESSIBILITÀ</i>	93
Figura 34: <i>Grafico risultati domanda 2 ACCESSIBILITÀ</i>	94
Figura 35: <i>Grafico risultati domanda 3 ACCESSIBILITÀ</i>	94
Figura 36: <i>Grafico risultati domanda 4 ACCESSIBILITÀ</i>	95
Figura 37: <i>Grafico risultati domanda 5 ACCESSIBILITÀ</i>	95
Figura 38: <i>Grafico risultati domanda 6 ACCESSIBILITÀ</i>	96
Figura 39: <i>Grafico risultati domanda 7 ACCESSIBILITÀ</i>	96
Figura 40: <i>Grafico risultati domanda generica 1</i>	101
Figura 41: <i>Grafico risultati domanda generica 2</i>	102
Figura 42: <i>Osservatorio Astronomico di Teramo (Abruzzo)</i>	106
Figura 43: <i>Stazione osservativa di Campo Imperatore</i>	108
Figura 44: <i>Esempio di un task completo</i>	109

INTRODUZIONE

La stesura di questa tesi è il risultato delle attività svolte, durante il periodo di tirocinio, presso l'*Osservatorio Astronomico d'Abruzzo - INAF*, nell'ambito del progetto “**Attività di analisi, modellazione e sviluppo del sistema di gestione della Stazione Osservativa di Campo Imperatore dell'INAF**”.

L'obiettivo del progetto si focalizza sull'analisi, modellazione e sviluppo del sistema di gestione utilizzato dagli astronomi e tecnici che operano nella Stazione Osservativa di Campo Imperatore. In particolare, si sono effettuate delle analisi attraverso l'approccio LeanUX e si è eseguita l'implementazione utilizzando il software OpenProject attraverso una metodologia Scrum.

Nei capitoli seguenti verrà descritta la fase di progettazione e di sviluppo, e illustrate le metodologie e le tecnologie utilizzate.

In particolare:

- Nel [Capitolo 2](#) viene fatta una panoramica generale sull'obiettivo e le motivazioni che hanno spinto alla realizzazione del sistema di gestione.
- Nel [Capitolo 3](#) si parlerà e si motiverà la scelta delle metodologie AGILE, Scrum e LeanUX.
- Nel [Capitolo 4](#) si descriverà OpenProject, ovvero la tecnologia utilizzata per lo sviluppo.
- Nel [Capitolo 5](#) si entrerà nel dettaglio della progettazione e dello sviluppo del sistema utilizzando le tecnologie e metodologie scelte.

- Nel [Capitolo 6](#) si faranno delle valutazioni finali attraverso un questionario su USABILITÀ e ACCESSIBILITÀ, compilato dagli Stakeholders.

In aggiunta, l'obiettivo personale, oltre a quello pratico della progettazione di tale sistema, è stato l'approfondimento di metodologie AGILE, ritenuto da me, un requisito fondamentale per la carriera lavorativa futura.

DOMINIO APPLICATIVO

Al giorno d'oggi, con il progredire delle tecnologie informatiche, è quasi impensabile gestire un'organizzazione, grande o piccola che sia, senza il minimo supporto di un software. I vantaggi che ne derivano sono palesi e di assoluta necessità.

La **Stazione Osservativa di Campo Imperatore**, con la scoperta delle onde gravitazionali, ha visto crescere nuovamente la sua importanza e utilità, dopo un periodo limitato a visite turistiche e didattiche. Essendo una stazione di vecchia generazione, nata nel 1965, mancava di un sistema informatico che gestisse ogni aspetto organizzativo della stazione, necessità che è diventata fondamentale negli ultimi anni.

Dal 2017 è gestita dall'**Osservatorio Astronomico d'Abruzzo - INAF** (vedere [*APPENDICE - Osservatorio D'Abruzzo - INAF*](#)), con sede a Teramo, che fa uso dei telescopi di Campo Imperatore per le proprie ricerche. Tale stazione si trova a 2200 m s.l.m sul Gran Sasso D'Italia e questo comporta ad una serie di problematiche.

Raggiungere questo luogo risulta difficoltoso, in quanto accessibile in auto tramite una strada il cui il transito dipende dalle condizioni metereologiche ed è vietato nei mesi invernali o tramite funivia. Questo comporta ad una mancata presenza in loco da parte degli astronomi, e una presenza limitata a tecnici specializzati che si occupano di tutti gli aspetti legati alla stazione (controllare e preparare i telescopi, controllare gli impianti termici, eseguire varie manutenzioni ecc...). Da un punto di vista organizzativo, la struttura osservativa non ha un responsabile unico, ma vari responsabili e tecnici che vi lavorano con disponibilità ed orari diversi, in quanto impegnati su differenti progetti. Ultimo vincolo è che la struttura non deve essere disponibile a delle osservazioni programmate, ma deve essere attiva e gestibile in

remoto ogni volta che arrivi un alert di gravitazionale. La gestione remota della stazione comporta stringenti vincoli di sicurezza poiché quest'ultima è posta in una posizione esposta a avverse condizioni atmosferiche che, se mal interpretate, potrebbero causare danni alla strumentazione e mettere a rischio la sicurezza di chi opera all'interno della struttura o nelle zone limitrofe.

Un sistema di gestione informatico aiuta a coordinare e sincronizzare ogni operazione o intervento, a migliorare la comunicazione e la condivisione delle informazioni e ridurre la probabilità di errore o incomprensione. Inoltre fornisce uno storico circa lo stato di manutenzione della strumentazione.

Da qui nasce principalmente la necessità di avere un sistema condiviso con tutto lo staff dell'INAF, che permetta di sincronizzare e condividere, attraverso dei TASK (vedere [APPENDICE - TASK](#)), ogni attività svolta tra i vari utenti del sistema.

Ma le motivazioni sono più di una:

- Comunicazione condivisa tra tecnici e astronomi;
- Sistema unico per l'intera organizzazione, che gestisca e condivida ogni attività della stazione, anche diverse tra di loro;
- Monitoraggio più semplice delle attività;
- Rinnovare le tecnologie della stazione osservativa.

L'attività di tirocinio, svolta presso l'INAF, aveva come obiettivo la progettazione e lo sviluppo di tale sistema, oltre all'apprendimento delle metodologie *AGILE* e *LeanUX*. Sistema che doveva soddisfare le esigenze di 3 Stakeholders, addetti alle 3 macro-attività principali dell'osservatorio, esigenze da soddisfare in poco tempo, ovvero 375 ore.

METODOLOGIE PROGETTUALI

Gli approcci e le metodologie scelte, per lo sviluppo del sistema di gestione della stazione osservativa, sono **Agile** e **LeanUX**. Le motivazioni di tali scelte sono diverse. Le valutazioni sulla scelta dell'Agile sono partite sicuramente dalla problematica della tempistica. Il tempo necessario per implementare tale sistema è stato di circa 2 mesi, tempo abbastanza breve per poter pensare ad uno sviluppo da zero di un sistema, quindi l'Agile, in particolare il framework *Scrum*, era la scelta logica per poterlo realizzare con queste tempistiche.

Altra motivazione è stata la relativa urgenza nell'avere il prima possibile un sistema funzionante e utilizzabile (anche non completo), cosa che l'Agile attraverso gli *Sprint* permette di realizzare, infatti, alla fine di ogni Sprint, almeno una parte del sistema era pronto per essere usato e testato, per poi eventualmente correggere dopo il feedback dello stakeholder. Tale motivo ha fatto ricadere la scelta sull'Agile a dispetto del V-Model.

La disponibilità degli stakeholders, inoltre, è stata anch'essa un fattore determinante sia per la scelta dell'Agile e sia per *LeanUX*. Il primo non è possibile utilizzarlo se non si hanno a disposizione continuamente i feedback degli *stakeholders*, e il secondo invece, oltre ad essere esso stesso un approccio orientato all'utente, ha permesso di fornire una guida semi-strutturata delle interviste agli *stakeholders*, permettendo di ridurre lo svantaggio dovuto all'inesperienza del *Team di Sviluppo* e degli *stakeholders* stessi.

Non sapendo come orientare lo stakeholder nell'illustrazione delle proprie esigenze durante l'intervista, una guida era necessaria, e *LeanUX Canvas* l'ha fornita

permettendo così di poter dare un filo logico e pratico alle interviste e permettere una progettazione precisa e rapida, incentrata sull'utente.

Di seguito verranno spiegati in linee generali gli approcci *Agile* con il framework *Scrum* e *LeanUX* e il suo Canvas, infine la loro applicazione nell'ambito di tale progetto.

a. AGILE^[1]

La "metodologia agile" è un approccio allo sviluppo del software basato sulla distribuzione continua di software efficienti creati in modo rapido e iterativo.

Benché l'espressione faccia riferimento specifico a un determinato sviluppo del software, questa metodologia non prevede regole da rispettare tassativamente nell'ambito dello sviluppo del software. Si tratta piuttosto di un tipo di approccio alla collaborazione e ai flussi di lavoro fondato su una serie di valori in grado di guidare il nostro modo di procedere.

In pratica, le metodologie di sviluppo software agile consistono nel rilasciare rapidamente modifiche al software in piccole porzioni con l'obiettivo di migliorare la soddisfazione dei clienti. Le metodologie utilizzano approcci flessibili e lavoro di gruppo per concentrarsi sul miglioramento continuo. Con l'adozione dello sviluppo agile i vari team costituiti da pochi sviluppatori ciascuno, si organizzano in autonomia e collaborano direttamente con i rappresentanti aziendali tramite incontri periodici durante l'intero ciclo di vita dello sviluppo del software. La metodologia agile consente di adottare un approccio più leggero alla stesura della documentazione software e di integrare le modifiche in qualsiasi fase del ciclo di vita, anziché ostacolarle.

Le origini della metodologia AGILE

L'approccio agile al lavoro è stato concepito per far fronte ai limiti percepiti del metodo a cascata, che trae le sue sorgenti dal metodo manifatturiero basato sulla catena di montaggio del 1913 di Henry Ford, applicato più tardi allo sviluppo di software. Dalla sua nascita nel 2001, lo sviluppo agile si è fatto strada nel settore dei software e del project management subendo molti cambiamenti.

La “metodologia Agile” fa riferimento e viene riassunta attraverso un manifesto:

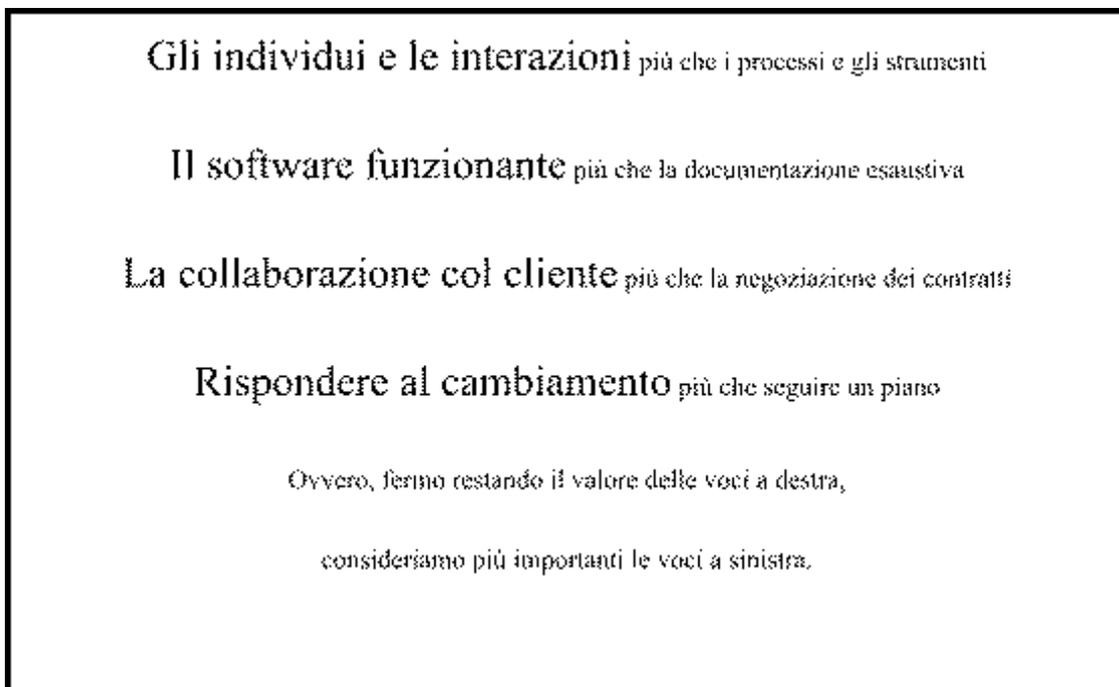


FIGURA 1. *Manifesto AGILE.*

L'impulso che ha spinto alcuni sviluppatori a introdurre una metodologia agile era legato alla mancata capacità dei cicli di produzione e dei metodi di collaborazione a cascata di portare ai risultati attesi. Questo problema ha iniziato a diffondersi nei primi anni '90, quando potevano trascorrere anni tra l'identificazione di un'esigenza aziendale e l'erogazione di un'applicazione efficiente volta a risolverla. Quegli anni erano caratterizzati da esigenze di business in costante mutamento, e molti dei progetti

software venivano addirittura annullati prima che fossero erogati. Questa perdita di tempo e di risorse ha portato molti sviluppatori software a cercare un'alternativa.

Messe in difficoltà dalle continue innovazioni, le organizzazioni hanno iniziato ad adottare sempre più strategie di trasformazione digitale per tenere il passo con un settore in continua evoluzione. E qui entra in gioco lo sviluppo agile del software.

La metodologia agile è alla base di molti flussi di lavoro digitale odierni. Il cloud computing, con la sua infrastruttura IT flessibile e scalabile, è cresciuto di pari passo con le richieste dello sviluppo agile di software. Lo sviluppo cloud native integra la nozione di software "agile" per ottenere una serie di servizi interconnessi e scalabili volti a soddisfare le esigenze aziendali.

Ne deduciamo quindi che il concetto di "metodologia agile" incorpora un'idea agile in grado di rispondere alle esigenze dei suoi clienti (ovvero, gli sviluppatori software) nel tempo. È importante tenere conto di questo aspetto quando si analizzano alcuni framework agili, che hanno nomi diversi e spesso variano da un'implementazione all'altra.

i. Scrum^[2]

Uno dei tanti framework Agile, forse il più utilizzato, è *Scrum*.

Scrum è un approccio basato sulla teoria del controllo empirico dei processi: le decisioni vengono prese sulla base dell'esperienza (empirismo). Tutti gli aspetti del lavoro devono essere visibili ai responsabili del risultato finale (**trasparenza**). Per rendere trasparenti questi elementi, il Team Scrum ispeziona di frequente il prodotto mentre lo sta sviluppando (**ispezione**). Così il processo e il prodotto possono essere adattati immediatamente nel caso di nuove esigenze o di condizioni mutate del mercato (**adattamento**).

Il senso di Scrum è quello di far lavorare il team insieme, in modo coordinato e organizzato.

Come funziona Scrum?

Come tutte le metodologie Agile , si basa sulla divisione del progetto in più fasi , chiamate *Sprint*.

Ad ogni Sprint, il gruppo di lavoro presenta nuove funzionalità, operative e implementabili, che vengono fatte valutare al cliente. Si configura così un sistema iterativo che consente di incrementare poco alla volta, ma frequentemente, le funzionalità del progetto, e quindi allo stesso tempo, si può verificare costantemente l'andamento complessivo e la soddisfazione del cliente su quanto già prodotto.

La metodologia Scrum viene definita da *Ruoli*, *Artefatti* ed *Eventi*.

Ruoli

Sono tre i ruoli che vengono individuati nella metodologia Scrum: *Product Owner*, *Scrum Master* e *Team di Sviluppo*. Insieme costituiscono lo **Scrum Team**.

Il **Product Owner** definisce il lavoro da svolgere e l'ordine con cui viene completato. Raccoglie la voce degli stakeholder (clienti, management e chiunque abbia un interesse nel prodotto), le necessità dell'utente finale, i requisiti del mercato e sulla base di questi elementi stabilisce le priorità di sviluppo per il Team Scrum ed è il responsabile del backlog, ossia delle attività che il team di sviluppo deve fare.

Lo **Scrum Master** è il responsabile del processo, e un leader a servizio dello Scrum Team. Conoscitore esperto della metodologia Scrum, sa come applicarla e si assicura che il Team comprenda e segua le regole che la caratterizzano, perché il progetto abbia successo. Inoltre favorisce il lavoro del Team di Sviluppo, rimuovendo ostacoli, organizzando meeting di confronto e soprattutto proteggendolo da ogni possibile distrazione: ogni membro del gruppo deve poter lavorare al 100% sullo sviluppo, e lo Scrum Master si assicura che questo avvenga.

Il **Team di Sviluppo** è la squadra di sviluppo composta da solitamente dalle 3 alle 9 persone. Anche lo Scrum Master può far parte del Team di Sviluppo. Chi concretamente porta a termine gli Sprint e fornisce le funzionalità da implementare è questo insieme coordinato di persone, autogestito e cross-funzionale.

Autogestito perché decide in autonomia come sviluppare le funzionalità individuate da Product Owner. *Cross-funzionale* perché contiene al proprio interno tutte le competenze e le funzioni necessarie per portare a termine i task. Inoltre ogni membro del team è responsabile dell'intero sviluppo: condividere la responsabilità aumenta il senso di appartenenza e incentiva la collaborazione tra i componenti del Team di Sviluppo.

Artefatti

Gli Artefatti sono tre: *Product Backlog*, *Sprint Backlog* e *Incremento*.

Sono pensati per aumentare la trasparenza delle informazioni principali, così che tutto il team ne sia al corrente.

Il **Product Backlog** è la lista ordinata di tutti gli elementi che servono nel prodotto. Elenca caratteristiche, funzioni, requisiti, miglioramenti e correzioni che costituiscono le modifiche da apportare alle versioni future del prodotto. Viene creato dal Product Owner, che è il responsabile dell'individuazione degli elementi (*Product Backlog Items*) e del loro ordine: stabilisce infatti la priorità con cui il Team di Sviluppo dovrà lavorarci.

Il Product Backlog non è fisso, ma evolve: diventa più complesso man mano che il prodotto viene costruito, e asseconda anche in feedback degli utenti che hanno testato le funzionalità già rilasciate.

Le singole voci del Product Backlog vengono descritte come **User Story** brevi e semplici descrizioni di una funzionalità raccontata dal punto di vista dell'utente o del cliente del sistema.

La struttura di una User Story è questa:

Come <tipo di utente>, voglio <obiettivo> in modo da <ragione>

Un esempio pratico di User Story potrebbe essere:

“Come venditore, mi piacerebbe impostare la mia password, così posso accedere al sistema”.

In questo caso la funzionalità descritta è particolarmente complessa e generica: questo tipo di User Story viene chiamata **Epic**. Le Epic vengono poi divise in User Story più specifiche, per permettere una corretta valutazione delle funzionalità da creare.

L'esempio precedente potrebbe, per esempio, essere diviso così:

“Come venditore, mi piacerebbe modificare il mio profilo, così posso impostare la mia password”

“Come amministratore, vorrei inviare un'email a un nuovo venditore con un link di accesso tokenizzato, in modo che possa accedere temporaneamente al sistema per impostare la propria password”.

Lo **Sprint Backlog**, invece, è l'insieme degli elementi del Product Backlog che sono stati selezionati per essere completati in una iterazione. E' anche una previsione del Team di Sviluppo su quanto verrà prodotto alla fine dello Sprint. Rende evidente a tutti gli sviluppatori il lavoro che deve essere fatto per raggiungere l'obiettivo dell'interazione (lo **Sprint Goal**). Lo Sprint Backlog è in continua evoluzione: tiene traccia del lavoro svolto giornalmente, delle attività completate e di quanto rimane da fare. Solo il Team di Sviluppo può intervenire sullo Sprint Backlog, anche aggiungendo o eliminando task .

L'**Incremento** è la somma di tutte le funzionalità completate in uno Sprint, insieme a quelle già completate negli Sprint precedenti. L'incremento prodotto deve soddisfare la "**Definizione di Fatto**" (in inglese, "*Definition of Done*"): ogni Team Scrum crea una propria definizione, ovvero un documento condiviso su quali siano i criteri per stabilire che cosa significa "Fatto".

Eventi

Gli Eventi che caratterizzano la strategia Scrum sono quattro, pensati per strutturare il processo, aumentare la trasparenza del lavoro, fornire un momento formale per adattare il progetto, e ridurre la necessità di incontri non programmati.

Lo **Sprint** è il cuore del framework Scrum ed è il contenitore di questi quattro eventi. Ha una durata compresa tra 2 e 4 settimane, ogni Sprint è nella pratica un mini progetto, il cui scopo è presentare un Incremento che risponde ai requisiti di "Fatto", utilizzabile e potenzialmente rilasciabile. Durante lo Sprint, gli obiettivi individuati nella fase di pianificazione restano fissi, così come la durata: scaduto il termine, lo sprint si conclude anche se non tutti i risultati sono stati raggiunti. Se l'obiettivo di uno Sprint diventa obsoleto e palesemente inutile, un intero Sprint può essere cancellato: ma vista la breve durata dello Sprint, è un'eventualità molto rara.

Lo Sprint è composto da 4 eventi: *Sprint Planning*, *Daily Scrum*, *Sprint Review* e *Sprint Retrospective*. Sono tutti eventi con una durata prestabilita.

Lo **Sprint Planning** è l'inizio di ogni Sprint: una riunione in cui il Team Scrum concorda l'obiettivo dello Sprint (lo *Sprint Goal*) e il Team di Sviluppo seleziona i Product Backlog Items da realizzare e pianifica come portarli a termine (viene impostato lo Sprint Backlog). Dura massimo 8 ore per gli Sprint di 4 settimane.

Il **Daily Scrum** è un evento giornaliero di 15 minuti in cui il Team di Sviluppo programma la giornata di lavoro. I punti affrontati in questa riunione sono tre: cosa è

stato fatto il giorno prima, cosa si farà nelle successive 24 ore e gli eventuali ostacoli incontrati. Inoltre, incentiva la collaborazione e la sincronizzazione del lavoro tra gli sviluppatori, rende chiara a tutti la progressione verso lo Sprint Goal e permette allo Scrum Master di intervenire tempestivamente sui problemi che rallentano il Team di Sviluppo.

Lo **Sprint Review** è una riunione prevista alla fine di ogni Sprint in cui lo Scrum Team discute insieme agli stakeholder l'incremento realizzato. La riunione serve per dimostrare quanto è stato svolto, e per iniziare a raccogliere gli input per le successive riunioni di Sprint Planning. Ha una durata proporzionata alla lunghezza dello *Sprint*, e non supera mai le 4 ore di riunione per uno Sprint di 4 settimane.

La **Sprint Retrospective**, che si apre dopo la *Sprint Review*, è un momento formale in cui il Team Scrum analizza lo Sprint concluso, identifica le cose che sono andate bene, individua i miglioramenti che possono essere fatti e pianifica il modo in cui implementarli nello *Sprint* successivo. Dura al massimo 3 ore.

Dopo la *Sprint Retrospective*, si ricomincia da capo, con un nuovo *Sprint Planning*, finché il progetto non è concluso.

ii. Agile vs V-Model^[3]

Esistono diverse metodologie di sviluppo del software utilizzate oggi nell'industria del software, una delle più importanti e più usate è sicuramente il **V-Model**.

L'obiettivo principale del *V-Model* è dare uguale peso alla codifica e ai test, e il modello *Agile* è un modello più recente, introdotto per risolvere le carenze riscontrate nei modelli esistenti. L'obiettivo principale di *Agile* è quello di integrare i test il prima possibile e di rilasciare versioni del prodotto in maniera incrementale suddividendo il sistema in parti secondarie molto piccole e gestibili.

Che cos'è il V-Model?

Il *V-Model*, come detto, è un modello di sviluppo software. E' considerato un'estensione del tipico modello di sviluppo del software Waterfall (*Figura 2*).

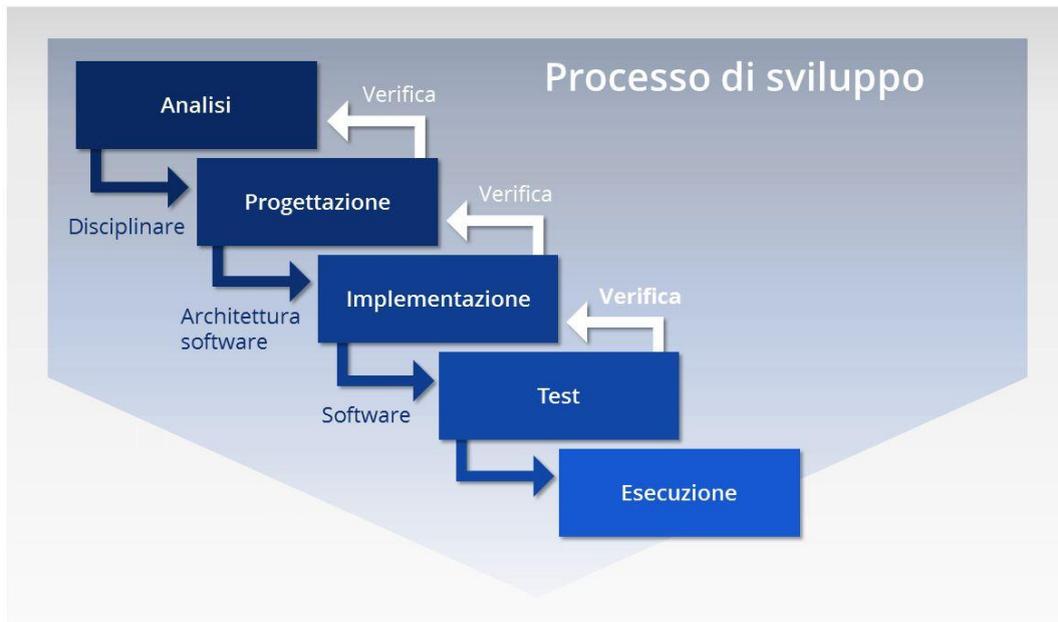


FIGURA 2. Ciclo sviluppo / progettazione modello Waterfall. ^[2]

V-Model usa le stesse relazioni tra le fasi definite nel modello Waterfall. Ma invece di scendere linearmente (come il modello Waterfall), *V-model* si sposta in diagonale e poi si sposta indietro (dopo la fase di codifica), formando la forma della lettera V (*Figura 3*), da cui prende il nome. Questa forma a V è formata per mostrare la relazione tra ciascuna fase di sviluppo / progettazione e la fase di test corrispondente.

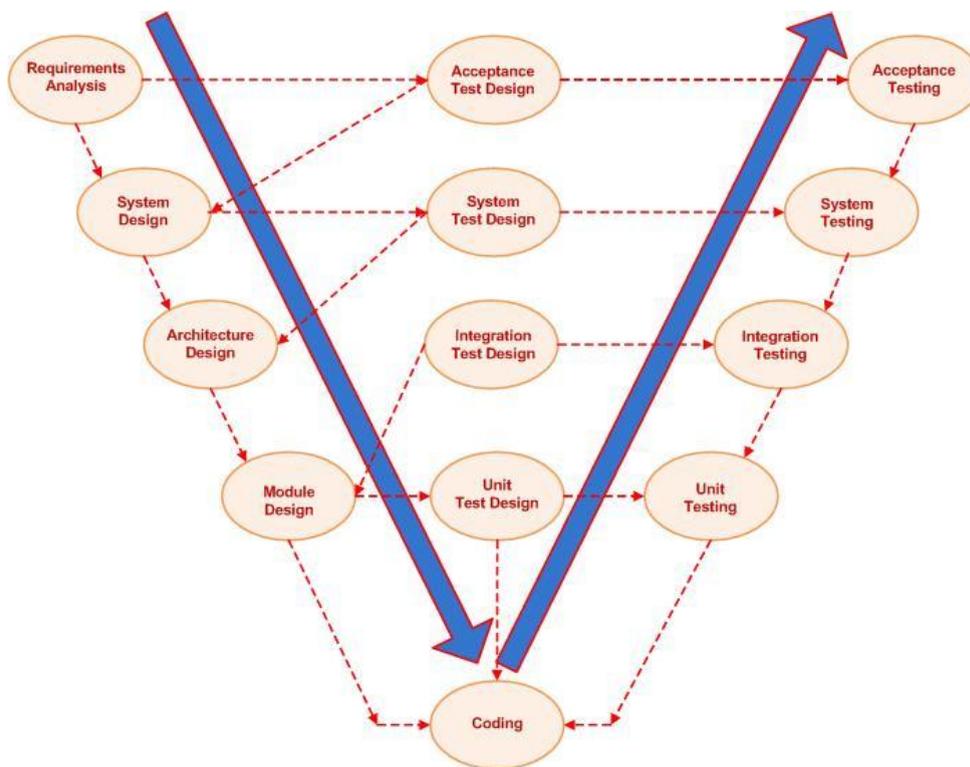


FIGURA 3. Ciclo sviluppo / progettazione V-Model. [\[8\]](#)

Il test (il percorso ascendente, lato destro della V) viene eseguito per la verifica, mentre le fasi di progettazione corrispondenti (il percorso discendente, lato sinistro della V) vengono utilizzate per la convalida. Nel modello V, lo stesso peso viene assegnato alla codifica e al test.

V-Model consiglia di creare un documento di test accanto al documento / codice di progetto. Ad esempio, i documenti di test di integrazione dovrebbero essere documentati mentre viene elaborato il piano di progettazione dettagliato. Ciò significa che un piano di implementazione per ogni test deve essere creato in anticipo, non aspettando che lo sviluppo sia completato in modo che possa essere trasferito al team di test.

Quale è la differenza tra modello Agile e V-Model?

Il modello *Agile* offre una versione funzionante del prodotto molto presto rispetto al *V-Model*, uno dei motivi per cui nella scelta delle metodologie progettuali è stato preferito *Agile* rispetto al *V-Model*. Dato che più funzioni vengono fornite in modo incrementale, il cliente può utilizzarle appena sviluppate e fornire un feedback immediato. Il tempo di ciclo di prova di *Agile* è relativamente breve rispetto a *V-Model*, poichè il test viene eseguito parallelamente allo sviluppo.

Agile è un modello proattivo (grazie ai suoi cicli molto brevi) rispetto al modello *V* molto più reattivo ai cambiamenti richiesti dagli utenti. *V-Model* è molto rigido e relativamente meno flessibile rispetto al modello *Agile*.

Attualmente, a causa di tutti questi vantaggi, nel campo industriale, soprattutto nelle startup, *Agile* è preferito rispetto al modello *V*.

Viceversa, il *V-Model* è ancora molto utilizzato in alcuni settori della ricerca, come il settore spaziale, perchè sono molto più vincolati, soprattutto nelle tempistiche (es. le finestre di lancio).

b. LeanUX^[4]

LeanUX è una tecnica incredibilmente utile quando si lavora su progetti in cui viene utilizzato il metodo di sviluppo Agile. Le tecniche di User Experience (UX) tradizionali spesso non funzionano quando lo sviluppo viene condotto in burst rapidi: non c'è abbastanza tempo per fornire UX allo stesso modo.

Fondamentalmente la LeanUX e altre forme di UX hanno tutte in mente lo stesso obiettivo; offrendo un'ottima esperienza utente, ma il modo in cui si lavora su un progetto è leggermente differente.

LeanUX - Cos'è?

La LeanUX è focalizzata sull'esperienza in fase di progettazione ed è meno focalizzata sui risultati finali rispetto al tradizionale UX. Richiede maggiore livello di collaborazione con l'intero team. L'obiettivo principale è quello di concentrarsi sull'ottenimento di un feedback il prima possibile in modo che possa essere utilizzato per prendere decisioni rapide. La natura dello sviluppo Agile è di lavorare in cicli rapidi e iterativi, e LeanUX imita questi cicli per garantire che i dati generati possano essere utilizzati in ogni iterazione.

La necessità di Ipotesi in LeanUX

Nella UX tradizionale il progetto si basa sull'acquisizione dei requisiti e sui risultati finali. L'obiettivo è garantire che i risultati finali siano il più dettagliati possibile e rispondano adeguatamente ai requisiti stabiliti all'inizio del progetto.

La LeanUX è leggermente diversa. Non sei concentrato sui risultati dettagliati. Stai cercando di produrre cambiamenti che migliorino il prodotto nell'immediatezza, essenzialmente per modellare il risultato in meglio.

Questo funziona in pratica abbandonando i “requisiti” e utilizzando una “affermazione del problema” che dovrebbe portare a una serie di ipotesi che possono essere utilizzate per creare ipotesi.

Che cos'è un'Ipotesi? Un'Ipotesi è fondamentalmente un'affermazione di qualcosa che pensiamo sia vero. Sono progettati per generare una comprensione comune intorno a un'idea che consenta a tutti di iniziare. E' pienamente compreso che le Ipotesi potrebbero non essere corrette e potrebbero essere modificate durante il progetto man mano che si sviluppa una migliore comprensione all'interno del team.

Le Ipotesi sono normalmente generate su base workshop. Riunisci il team e esponi il problema e poi permetti al team di fare un brainstorming delle loro idee per risolvere il problema. Nel processo generi risposte a determinate domande che formano le tue Ipotesi.

Domande tipiche potrebbero essere:

- Chi sono i nostri utenti?
- A cosa serve il prodotto?
- Quando viene utilizzato?
- In quali situazioni viene utilizzato?
- Quale sarà la funzionalità più importante?
- Qual è il rischio maggiore per la consegna del prodotto?

Ci può essere più di una risposta per ogni domanda. Questo ci lascia con un numero di ipotesi maggiore di quello che potrebbe essere pratico gestire. Se è questo il caso, il team può dare priorità alle proprie Ipotesi rapidamente dopo la loro generazione. In generale, daresti la priorità alle tue Ipotesi in base al rischio che rappresentano (quali sono le conseguenze di un grave errore? Più grave è la conseguenza, maggiore è la priorità) e il livello di comprensione del problema in questione (meno sai, maggiore è la priorità).

Creare un'Ipotesi in LeanUX

Le Ipotesi create in LeanUX sono progettate per testare le nostre Ipotesi. C'è un formato semplice che si può usare per creare le Ipotesi in modo rapido e semplice.

Un esempio:

*“**Riteniamo che** consentire alle persone di salvare i propri progressi in qualsiasi momento sia essenziale per gli utenti degli smartphone. **Ciò consentirà di raggiungere un livello più elevato di completamenti di iscrizione. Lo avremo dimostrato quando potremo misurare un miglioramento dell'attuale tasso di completamento del 20%.**”*

Dichiariamo l'Ipotesi, il perché è importante e per chi è importante. Infine, determiniamo di quali prove avremmo bisogno per dimostrare che l'Ipotesi era vera.

Se scopriamo che non c'è modo di dimostrare l'Ipotesi, si potrebbe andare nella direzione sbagliata perché i risultati non sono chiaramente definiti.

Uno dei grandi vantaggi di lavorare in questo modo è che rimuove gran parte del “non penso che sia una buona idea” e delle lotte politiche interne dal processo di progettazione della UX. Ogni idea sarà testata e i criteri di prova chiaramente determinati. Nessuna prova? Quindi è il momento di abbandonare l'idea e provare qualcos'altro.

Se tutti possono capire un'Ipotesi e le aspettative da essa, tendono ad essere felici di aspettare per vedere se è vero piuttosto che discutere appassionatamente il proprio punto di vista soggettivo.

Il Minimum Viable Product (MVP) e LeanUX

Il MVP è un concetto fondamentale in LeanUX. L'idea è quella di costruire la versione più elementare possibile del concetto, testarlo e se non ci sono risultati preziosi abbandonarlo. Gli MVP che sono promettenti possono essere quindi incorporati in ulteriori round di progettazione e sviluppo.

Questo è un ottimo modo per massimizzare le risorse e uno dei motivi per cui funziona così bene con lo sviluppo Agile.

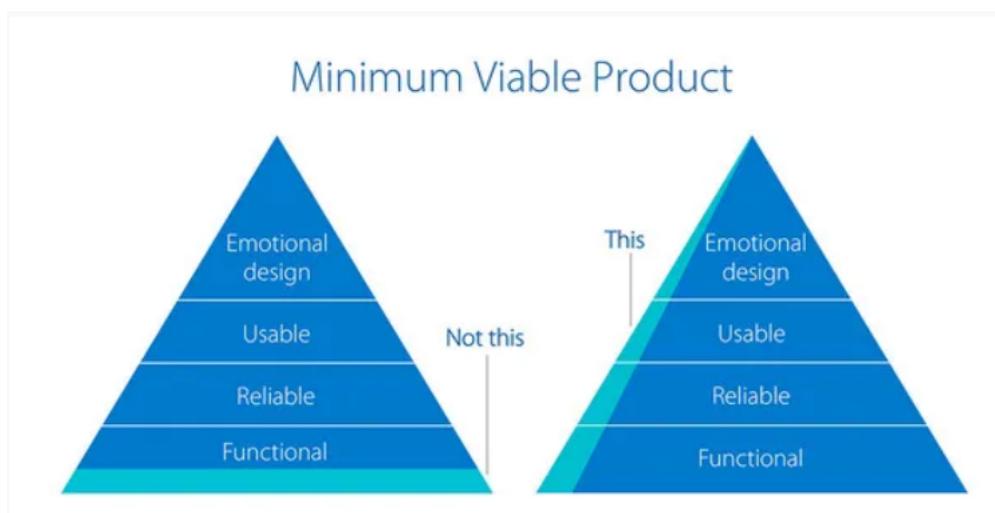


FIGURA 4. *Minimum Viable Product*

Ricerca e test in LeanUX

La ricerca e il test, per la natura stessa di LeanUX, si basano sugli stessi principi utilizzati negli ambienti LeanUX tradizionali. Tuttavia, l'approccio tende ad essere rapido: i risultati devono essere consegnati prima dell'inizio del prossimo Agile Sprint; quindi c'è molta meno attenzione su output di documenti pesanti e maggiore attenzione sui dati grezzi.

Le responsabilità per la ricerca tendono anche a essere distribuite più ampiamente all'intero team in modo che non si crei un "collo di bottiglia" se una singola risorsa di progettazione UX cerca di portare a termine l'intero lavoro in tempi ristretti da sola.

I tre pilastri fondamentali del LeanUX^[5]

La LeanUX è composta da tre principi fondamentali:

- Il primo principio fondamentale della LeanUX è il **Design Thinking**. Il *Design Thinking* è importante per la LeanUX perché richiede che ogni aspetto di un'azienda possa essere affrontato con gli strumenti che questo propone. Dà ai progettisti il permesso di lavorare oltre i loro confini tipici e incoraggia inoltre i non progettisti ad utilizzare metodi di progettazione per risolvere i problemi che si trovano ad affrontare nei loro ruoli. Il *Design Thinking* incoraggia i team a collaborare in maniera trasversale rispetto ai loro ruoli e a prendere in considerazione il product design da una prospettiva olistica.

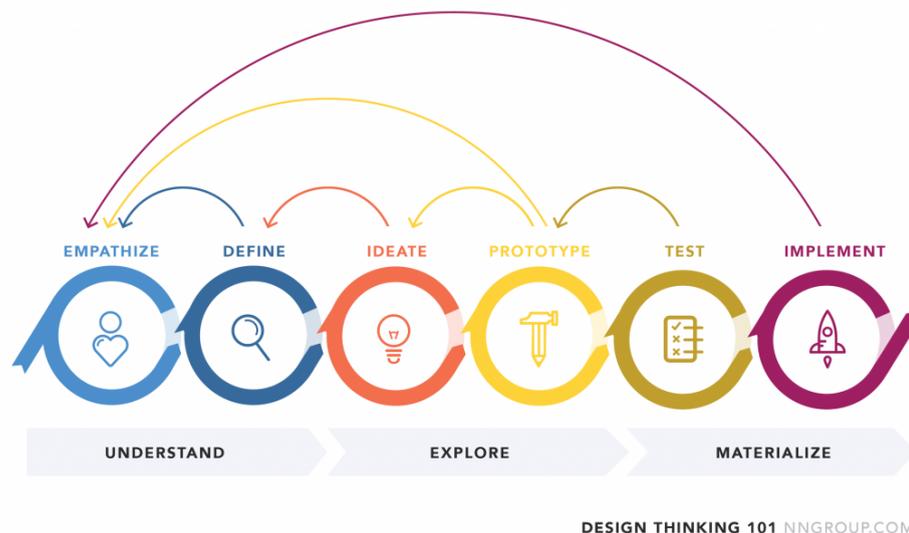


FIGURA 5. *Design Thinking*

- Il secondo principio fondamentale della LeanUX è la metodologia **Agile**. Gli sviluppatori utilizzano da anni i metodi *Agile* per ridurre il tempo necessario a completare un progetto e per fornire soluzioni al cliente in modo continuo. Anche se la metodologia *Agile* può creare degli ostacoli durante il processo per designer, i valori fondamentali di questa sono il cuore della LeanUX.
- Il terzo principio fondamentale della LeanUX è il metodo “**Lean Startup**” creato da Eric Ries. Il metodo *Lean Startup* usa un ciclo di feedback chiamato “build, measure, learn” ovvero “costruisci, misura, impara”. Questo ciclo è necessario affinché sia possibile ridurre al minimo il rischio relativo alla creazione del progetto e allo stesso tempo permettere al team di creare il prodotto rapidamente e di apprendere dagli errori. I team devono costruire dei “**Minimum Viable Products**” (MVP) e consegnarli al cliente in modo da apprendere dagli errori il prima possibile.

i. LeanUX Canvas

I canvas, ma anche template, kanban e grid sono griglie, più o meno grandi (dal formato A₄ a poster da 3 metri) che offrono, a chi lo utilizza, domande o temi da analizzare.

Oggi assistiamo ad un proliferare di canvas o template, come per esempio il **Business Model Canvas** e tanti altri, che aziende e accorti professionisti offrono gratuitamente per facilitare il lavoro di UX.

Anche LeanUX ne ha, tra cui quello utilizzato nell'ambito di questo progetto, ovvero **LeanUX Canvas V2** di Jeff Gothelf [\[6\]](#), che va a risolvere le problematiche relative alla versione precedente (*LeanUX Canvas VI*).

Gli scopi e l'idea di utilizzo, secondo l'autore del canvas, sono:

- E' uno strumento di facilitazione per team interfunzionali, progettato per creare una conversazione incentrata sul cliente sul lavoro che il team sta svolgendo;
- L'obiettivo è quello di aiutare il team a concentrarsi sul “perché” stanno facendo il lavoro attuale;
- Fornisce una guida iniziale da seguire per i team nelle prime fasi dell'adozione della metodologia Agile;
- E' una sorta di “polizza assicurativa” per garantire che l'apprendimento avvenga in ogni sprint.
- Espone le lacune del team nella comprensione del problema che stanno risolvendo, per chi lo stanno risolvendo e perché credono che le loro soluzioni funzioneranno;
- E' un primo passo nel passaggio della conversione dagli output ai risultati.

Il *LeanUx Canvas V2* utilizzato è questo (*Figura 6*):

The image shows the Lean UX Canvas (v2) form, which is a structured template for defining and testing a business idea. It consists of eight numbered boxes arranged in a 3x3 grid (with the bottom-right cell empty). Each box contains a specific question or instruction related to business strategy and user experience. The boxes are: 1. Business Problem, 2. Business Outcomes, 3. Users, 4. User Outcomes & Benefits, 5. Solutions, 6. Hypotheses, 7. What's the most important thing we need to learn first?, and 8. What's the least amount of work we need to do to learn the next most important thing?. The form also includes fields for 'Title of initiative', 'Date', and 'Iteration' at the top.

FIGURA 6. *LeanUX Canvas V2*

Questo, attraverso 8 Box, permette di avere una guida per il team, incentrata sull'utente, durante il processo di progettazione. Lo scopo di ogni Box:

- **BOX 1 - Business problem statement:** questo Box è progettato per aiutare il team a comprendere il problema che si sta cercando di risolvere.

Esempio:

“Perdiamo l'80% dei nuovi utenti perché non riescono a capire come effettuare una prenotazione”

“Ciò rende l'assistenza e il marketing dei nostri clienti molto più costosi.”

- **BOX 2 - Business outcomes:** in questo Box si indica quali sono i “*business impact*” che si vogliono ottenere.

Esempio:

“Aumento delle prenotazioni.”

“Aumento dei referrals.”

- **BOX 3 - Users**: In questo Box vengono indicati la tipologia di utenti che vengono coinvolti.

Esempio:

“Nuovi clienti che non hanno familiarità con la nostra piattaforma di prenotazione.”

- **BOX 4 - User outcomes and benefits**: in questo Box vengono descritti quali sono i benefici attesi per gli utenti e come vengono misurati / valutati.

Esempio:

“Un modo più semplice di prenotare per le persone.”

“Misureremo i miglioramenti attraverso il feedback dei clienti.”

- **BOX 5 - Solutions**: in questo Box si indicano quali sono le soluzioni adottate per risolvere i problemi.

Esempio:

“Progettare un nuovo flusso di onboarding.”

“Creare un video che spieghi come funziona il nostro sistema di prenotazioni.”

“Semplificare il nostro sistema di prenotazioni.”

“Aprire un forum per i clienti dove trovare aiuto dalla community.”

- **BOX 6 - Hypotheses**: in questo Box vengono definite le ipotesi seguendo lo schema *“Riteniamo che [business outcomes]. Ciò sarà raggiunto quando [customer] vedono [benefit] dalle [solutions]”*.

Esempio:

“Noi crediamo che possiamo incrementare le prenotazioni e i referral dalla semplificazione del nostro sistema di prenotazioni.”

- **BOX 7 - What's the most important thing we need to learn first?**: in questo Box si indica quale è la parte più rischiosa delle nostre ipotesi.

Esempio:

“Ci vorrà troppo tempo per realizzarlo.”

- **BOX 8 - What's the least amount of work to learn the next most important thing?**: in questo Box si descrive quale è il modo più veloce per dimostrare le ipotesi.

Esempio:

“Realizzare un prototipo grezzo.”

“Testarlo con i nuovi utenti.”

c. Vantaggi integrazione LeanUX con Scrum

User Experience o UX (in italiano "esperienza di utente") è un termine utilizzato per definire la relazione tra una persona e un prodotto, un servizio, un sistema. L'esperienza utente coinvolge tutti gli aspetti esperienziali, affettivi, l'attribuzione di senso e di valore collegati ad un prodotto o servizio, all'interazione con esso e quanto ad esso correlato, ma include anche le percezioni personali su aspetti quali l'utilità, la semplicità d'utilizzo e l'efficienza del sistema.

UX copre ogni aspetto dell'interazione di una persona con un'organizzazione, i suoi prodotti e servizi. Ispezionare e comprendere bene questi aspetti aiuterà gli sviluppatori di prodotti ad aumentare il valore di quell'esperienza.

A partire dall'organizzazione del team, LEAN UX guida i team a essere interfunzionali, co-localizzati, autosufficienti e responsabilizzati. Inoltre, è tutto progettato per offrire risultati di alta qualità al massimo valore. Questa è davvero roba che non è affatto facile da ottenere.

Le attività UX tradizionali generalmente ruotano attorno alla creazione di sottoprodotti come wireframe esaurienti, prototipi interattivi, personaggi di marketing, flussi di utenti e percorsi dei clienti.

LEAN UX si concentra sulla massimizzazione del valore e sulla riduzione degli sprechi. Si perde meno tempo nella pre-produzione dei sottoprodotti. Più valore si ottiene attraverso la collaborazione tra specialisti e il coinvolgimento di clienti e utenti finali. Esso ruota intorno all'iniezione, in fase di sviluppo, di decisioni basate sulle informazioni fornite dal cliente. Considera l'apprendimento e lo sviluppo come attività in corso che corrono insieme come un percorso integrato all'interno di un singolo team.

Quale valore aggiungerebbero gli UX Designer o gli UX Engineer quando performeranno all'interno del team di sviluppo interfunzionale in Scrum, applicando LEAN UX come strategia?

Faccio affidamento a un articolo di Sjoerd Nijland su Medium^[15] che elenca quali sono, secondo lui, i vantaggi che spingono all'integrazione di LeanUX con Scrum:

- 1. Connettere maggiormente e “avvicinare” lo Scrum Team con gli utenti finali e i clienti;*
- 2. Facilitare le attività che migliorano la collaborazione all'interno dei team interfunzionali;*
- 3. Iniettare, in fase di sviluppo, decisioni basate su informazioni fornite dai clienti;*
- 4. Valorizzare l'apprendimento più della consegna del prodotto;*
- 5. Supportare il Product Owner nel prendere decisioni migliori;*
- 6. Allineare le priorità del team sulla risoluzione dei problemi e non sulla creazione di funzionalità;*
- 7. Incoraggiare a pensare ai risultati da raggiungere;*
- 8. Incoraggiare la creatività e l'esplorazione che crea una visione più ampia del lavoro;*
- 9. Creare un ambiente in cui è sicuro e prezioso fallire;*
- 10. Reagire rapidamente alle situazioni.*

La verità è che non esiste un framework o una metodologia adatta al 100% ad ogni sviluppo o progettazione, per cui i team, le aziende ecc... sono spesso costrette ad

adottare soluzioni di integrazioni di metodologie o parti di framework, al fine di modellare i propri metodi in base allo sviluppo/progetto che stanno affrontando.

Questa integrazione di LeanUX con Scrum, cerca appunto di colmare alcune mancanze di quest'ultimo come ad esempio la poca interazione tra team di sviluppo e utente finale, quando il team ha esigenze di avere un feedback rapido e preciso, che arrivi direttamente dagli utilizzatori del software e non non modo indiretto tramite degli "intermediari". Ma anche Scrum aggiunge vantaggi a LeanUX, come per esempio l'organizzazione tramite Ruoli, Eventi e Artefatti, e un maggiore focus sul prodotto finale, ed è esattamente ciò che abbiamo fatto noi.

TECNOLOGIE

L'INAF che ha pensato alla realizzazione di questo sistema, al momento dell'illustrazione del progetto, come unico vincolo, ha imposto l'utilizzo del software OpenProject, che oltre a semplificare il lavoro, ha permesso di soddisfare quasi tutte le richieste degli Stakeholders nel rispetto delle regole della metodologia AGILE.

OpenProject^[9]



FIGURA 7. Logo OpenProject.

OpenProject è un sistema di gestione progettuale via web per la collaborazione in squadra indipendentemente dal luogo, che permette appunto di gestire progetti classici o in metodo AGILE al fine di supportare il team durante l'intero ciclo di vita del progetto. OpenProject è disponibile in più di 30 lingue.

OpenProject è sotto licenza GNU GPL V3 ed il codice sorgente è pubblicato liberamente su Github [\[10\]](#).

Vengono offerte delle sottoscrizioni a pagamento, annuali e in base al numero di utenti, che offrono diverse funzionalità:



FIGURA 8. Sottoscrizioni offerte da OpenProject.

Per la realizzazione di questo progetto è stata utilizzata la versione “*COMMUNITY*”. OpenProject offre un set completo di funzionalità per supportare i team di progetto lungo l’intero processo di gestione del progetto:

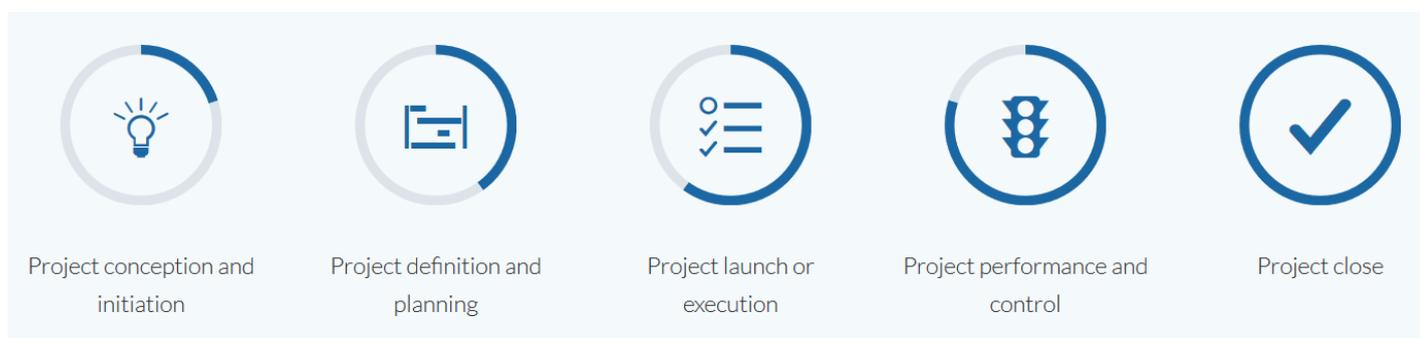


FIGURA 9. Ciclo progettazione OpenProject.

Consente la collaborazione e la comunicazione senza interruzioni del sistema dall'idea iniziale fino alla chiusura e alla documentazione del progetto.

Features OpenProject

Come descritto in precedenza le features di OpenProject permettono di gestire la progettazione da parte del team durante tutto il ciclo di vita del progetto.

- *Pianificazione e programmazione del progetto*

In Openproject è possibile definire gli obiettivi e il lavoro da svolgere in relazione ad essi, analizzare le attività e creare un piano dettagliato che mostri come e quando il progetto fornirà dei risultati.

Tools: *Diagrammi di Gantt, Work Packages e Calendario.*

- *Roadmap del prodotto e pianificazione del rilascio*

OpenProject è il modo più semplice per pianificare, visualizzare e comunicare la roadmap del prodotto. Permette di condividere la roadmap del prodotto con le parti interessate, permettendo di ottenere feedback sulle idee e scomporle in un piano di rilascio dettagliato.

Tools: *Roadmap e Timeline del Prodotto.*

- *Gestione dei Task e collaborazione in team*

OpenProject permette ai team di monitorare il proprio lavoro, facendo sì che tutti conoscano gli obiettivi e lavorino insieme per raggiungerli, avendo tutte le attività e le comunicazioni in un unico posto, e permette di organizzare e assegnare facilmente i compiti ai singoli.

Tools: *Lista Task, Commenti e Storia, Osservatori.*

- **Kanban, Agile e Scrum**

OpenProject è uno strumento che facilita i team Agile che desiderano utilizzare metodologie Agile come Scrum. I team Agile possono creare e stimare storie, dare priorità a uno sprint backlog e tenere traccia delle attività. È strettamente integrato in altri moduli come la pianificazione della roadmap, il monitoraggio dei bug e la gestione delle attività.

Tools: *Agile Boards, Backlogs e Task Board.*

- **Tracciamento Bug**

OpenProject permette il tracciamento dei bug, offrendo ai responsabili della Quality Assurance e ai tester una piattaforma per acquisire, classificare e dare priorità ai bug. Grazie al supporto della metodologia Agile, OpenProject riduce al minimo il ciclo di feedback tra la garanzia della qualità e lo sviluppo del software.

Tools: *Tracking, Filtri, Workflows e Campi Personalizzati.*

- **Tracciamento del tempo, reporting dei costi e dei Budget**

OpenProject semplifica il monitoraggio del tempo. Permette di tenere sotto controllo i budget del progetto e i costi spesi, nonché il tempo speso. Permette di creare report

personalizzati per informazioni accurate e aggiornate sulle prestazioni del progetto e sulle risorse allocate. Inoltre permette di pianificare il costo per ogni fase del progetto e di guardare quanto del budget assegnato è stato speso.

Tools: *Time Tracking, Report dei Costi, Budgeting e Log del Tempo.*

- **Wiki del progetto**

OpenProject permette di creare e gestire in modo collaborativo una wiki di progetto. Quindi, gestire la documentazione del progetto, i riferimenti, le linee guida, i manuali per l'utente e molto altro in un'unica posizione centrale e di creare una base di conoscenza per i progetti e condividerla con il team e altre parti interessate. Si può facilmente inserire, modificare e formattare testi (ad es. copiare e incollare da Word), includere tabelle, immagini, collegamenti e altri riferimenti a OpenProject (ad es. tabelle del pacchetto di lavoro, diagrammi di Gantt o ticket). Si hanno tutte le modifiche in una vista o esportare i testi (ad esempio in Markdown). È possibile creare una struttura di navigazione dettagliata con menu laterale, sommario e gerarchie di siti.

Tools: *Project Wiki, Contenuti, Tabelle Incorporate e Storia.*

PROGETTAZIONE e SVILUPPO

In questo capitolo verrà spiegato come sono state usate le metodologie e le tecnologie scelte nell'ambito dello sviluppo del sistema di gestione della stazione osservativa. In primis si illustrerà come sono stati applicati i metodi, e con quali tools, di **Agile** e di **LeanUx**.

Inoltre, si illustrerà come **OpenProject** è stato utilizzato per sviluppare il sistema richiesto, spiegando le fasi più importanti dell'implementazione e le funzionalità del sistema.

a. Implementazione Agile

In questo paragrafo verrà spiegato come è stato utilizzato l'approccio **Agile** nell'ambito di questo progetto. Come già accennato nell'introduzione del Capitolo 3, le motivazioni che hanno spinto sulla scelta della metodologia *Agile* sono sicuramente le tempistiche, 375 ore ovvero circa 2 mesi a disposizione per fornire un sistema completo o quasi, e la possibilità di fornire, in tempi rapidi, piccoli parti funzionanti dell'intero sistema. Inoltre la disponibilità continua da parte degli stakeholders ne ha facilitato l'approccio.

Il framework *Agile* utilizzato è stato **Scrum**.

Ruoli

Partendo dal presupposto che si trattava di una attività di tirocinio e che quindi non si aveva a disposizione un vero e proprio team, si è cercato di simulare nel modo più reale possibile la composizione dei ruoli di Scrum, ma di piccole dimensioni.

Infatti il ruolo dello **Product Owner** e dello **Scrum Master** sono stati assegnati alla stessa persona, ovvero l'ing. *Matteo Canzari*, essendo egli un esperto della metodologia Agile, ha potuto gestire tranquillamente entrambe i ruoli, per cui non c'è stato bisogno di una ulteriore persona che prendesse uno dei due ruoli.

Invece per quanto riguarda il **Team di Sviluppo**, era composto da una singola persona, ovvero me stesso, che come da definizione Scrum, era autogestito e cross-funzionale.

Infine, gli **stakeholders** interessati al sistema e che quindi fornivano continuamente feedback, erano astronomi o persone dello staff dell'INAF, in particolare:

- *Enzo Broncato* - Stakeholder & Direttore Osservatorio Astronomico d'Abruzzo;
- *Mauro Dolci* - Stakeholder come responsabile strumentazione scientifica;
- *Gaetano Valentini* - Stakeholder come responsabile logistica;

- *Fiore De Luise* - Stakeholder & come responsabile delle osservazioni astronomiche;

La dimensione del *Team di Sviluppo* e l'unica figura di *Product Owner / Scrum Master*, rimarcano ancor di più l'importanza della scelta di *Agile*, perché con poche persone e nel poco tempo a disposizione, si è riuscito a sviluppare un sistema di gestione per la stazione osservativa funzionante e che soddisfasse le esigenze degli *stakeholders*.

Artefatti

Per quanto riguarda gli Artefatti, la classica struttura fornita dal framework *Scrum* (*Product Backlog, Sprint Backlog, Incremento, User Story*), è stata leggermente modificata o per lo meno interpretata diversamente.

Per poter gestire gli artefatti, quindi creare un board *Agile* condivisa tra il *Team di Sviluppo* e il *Product owner / Scrum Master*, si è utilizzato il tool [Trello.com](https://trello.com) ^[11].

Questo sito consente agli utenti di creare le loro schede attività con più colonne e scambiare le attività tra di loro. Il software è utilizzabile per uso personale e aziendale. Ha una varietà di possibilità di impiego, come la gestione immobiliare, la gestione di progetti software, i bollettini scolastici, la pianificazione delle lezioni, la contabilità, il web design, i giochi e la gestione di casi legali.... .

Nella board creata, le colonne definite sono:

- **LEGEND**: descrive le tipologie di schede presenti nella board. Ovvero:
 - **User Story**: scheda che indica ciò che deve essere o è stato implementato. Anche il classico schema per le *User Story*, ovvero “*Come <tipo di utente>, voglio <obiettivo> in modo da <ragione>*”, non è stato utilizzato, ma semplicemente sono state descritte liberamente dal creatore

della scheda. Lo schema classico è stato utilizzato di più durante le interviste agli stakeholders attraverso il *LeanUX Canvas v2*.

Esempio: “Definizione Macro-Attività”

→ **Spike**: scheda che descrive un’attività che necessita di ulteriori approfondimenti.

Esempio: “Studio Documentazione Openproject”

→ **Enabler**: scheda che descrive un’attività di supporto alle User Story, ma non di immediato valore per l’utente.

Esempio: “Intervista a Fiore e chiarimenti con Mauro”

→ **Blocked**: scheda che descrive un’attività attualmente bloccata per qualche motivazione. E’ uno stato dell’attività spesso transitorio.

- **BACKLOG**: funge da **Product Backlog**, per cui vengono inserite tutte le schede / attività in ordine di priorità che devono essere sviluppate dal *Team di Sviluppo*.
- **TO-DO**: funge da **Sprint Backlog**, contenendo tutte le schede / attività da sviluppare durante il singolo *Sprint*. In questa colonna, solo il *Team di Sviluppo* può creare o cancellare schede, ma per la maggior parte le schede qui contenute sono prese e spostate dal *Backlog*.
- **DOING**: contiene le schede / attività in corso di sviluppo, prese e spostate dalla colonna *TO-DO*.

- **CODE REVIEW**: contiene tutte le schede / attività sviluppate che sono in fase di revisione, prima di essere approvate. Le schede presenti in questa colonna sono prese e spostate dalla colonna *DOING*.
- **DONE**: contiene tutte le schede che sono state approvate o che hanno superato la fase di revisione, e quindi presi e spostati dalla colonna *CODE REVIEW*.
- **SPRINT n°**: dopo lo **Sprint Review**, le schede contenute in *DONE* vengono spostate in questa colonna. Quindi contiene tutte le schede completate relative ad uno *Sprint*.

Da considerare che per ogni *Sprint* è stata definita una capacità (**Capacity**), solitamente minimo 9.5 giorni, e un carico (**Load**), di almeno 6 giorni. Inoltre ad ogni scheda è stata assegnata una stima del carico, ovvero la stima dei giorni necessari al suo sviluppo, indicati nelle schede con un numero tra parentesi (*n*) nel titolo. La somma di tutte le stime delle schede / attività relative ad uno *Sprint*, formano il totale del carico avuto dal *Team di Sviluppo* durante un singolo *Sprint*.

Il carico e la capacità sono molto utili a comprendere l'efficienza e l'efficacia del *Team di Sviluppo*, e hanno permesso al *Product Owner / Scrum Master* di capire il lavoro da assegnare al *Team di Sviluppo*, *Sprint* dopo *Sprint*.

Di seguito, nella figura, è mostrato come si presenta la UI della board di *Trello*:

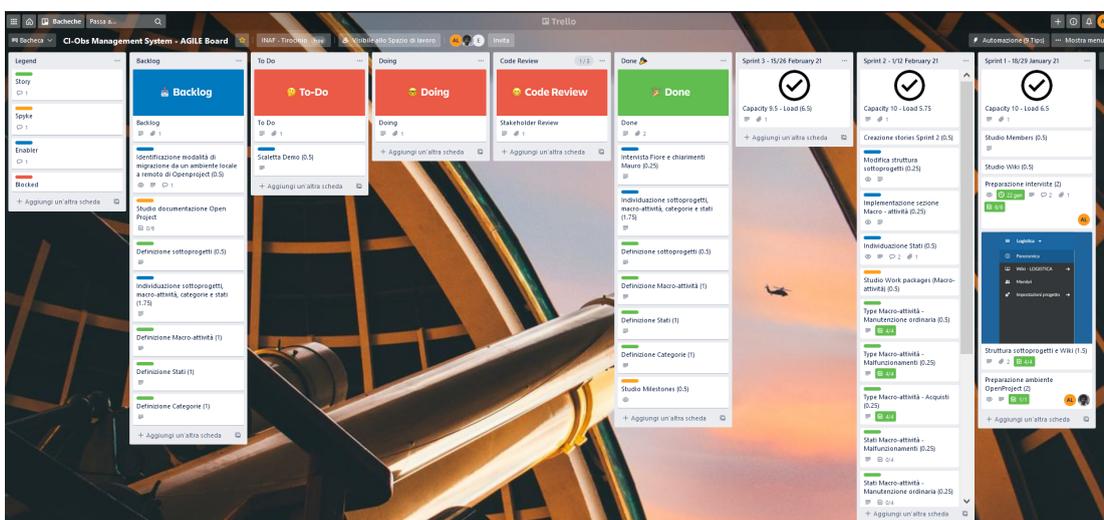


FIGURA 10. Board Agile di Trello durante lo Sprint 3.

Eventi

Gli Eventi definiti nel framework *Scrum* sono stati, nell'ambito di questo progetto, adattati in base al tempo a disposizione. Come detto nei paragrafi precedenti, il tempo a disposizione è stato di 375 ore ovvero circa 2 mesi, per cui gli eventi sono stati così organizzati:

- **Sprint**: il numero totale degli *Sprint* è stato 4, ognuno della durata di 2 settimane così da poter riempire interamente i due mesi. Per ognuno degli *Sprint* è stato definito una *Capacity* di almeno 9.5 giorni, coprendo così interamente le giornate lavorative della settimana, ma escludendo Sabato e Domenica. La durata degli *Sprint* è stata ampiamente soddisfacente, soprattutto per l'implementazione di *User Story*, per cui il tempo avanzato è stato usato per approfondimento di *Spike* o studio di *Enabler*.
- **Sprint Planning**: tale evento, a cui partecipava il *Team di Sviluppo* e il *Product Owner / Scrum Master*, avveniva il primo giorno lavorativo dello *Sprint*, ovvero il primo Lunedì delle 2 settimane. In questa riunione, della durata di circa 1 ora, venivano riordinate le idee nate dalla prima intervista agli Stakeholders, o nate da uno *Sprint Review*, e definito *Sprint Backlog* e *Sprint Goal*.
- **Daily Scrum**: tale evento, a cui partecipava il *Team di Sviluppo* e il *Product Owner / Scrum Master*, avveniva il Martedì e Giovedì alle 9.00 di mattino. La durata del *Daily Scrum* era di circa 15 minuti, ma in base a ciò che si andava a discutere, la durata poteva essere estesa. In questa “piccola” riunione veniva mostrato velocemente ciò che era stato fatto il giorno prima, esposti i problemi riscontrati, eventuali blockers e pianificato il lavoro fino al *Daily Scrum* successivo.

- **Sprint Review**: tale evento, a cui partecipava non solo il *Team di Sviluppo* e il *Product Owner / Scrum Master* ma anche gli *Stakeholders*, avveniva alla fine di ogni *Sprint* ovvero il secondo Venerdì delle 2 settimane, dalla durata di circa 2 ore. In primis, veniva mostrato una demo di ciò che era stato implementato durante lo *Sprint*, al fine di far comprendere le funzionalità agli *Stakeholders* e in modo da verificare se le esigenze fossero soddisfatte.
Dopo di ciò venivano raccolti gli input dagli *Stakeholders* per il prossimo *Sprint Planning*, sempre basandosi sull'esperienza delle interviste degli *Sprint* precedenti e sulla guida fornita dal *LeanUX Canvas v2*.
- **Sprint Retrospective**: tale evento, a cui partecipava il *Team di Sviluppo* e il *Product Owner / Scrum Master*, che solitamente si tiene dopo lo *Sprint Review*, avveniva prima dell'inizio dello *Sprint Planning*. La durata dello *Sprint Retrospective* era di circa 15 minuti, in cui si discuteva dell'andamento dello *Sprint Review* precedente e delle problematiche venute fuori.

b. Implementazione LeanUx

In questo paragrafo verrà spiegato come è stato utilizzato l'approccio **LeanUx**, nello specifico il *LeanUX Canvas v2* in fase di progettazione. Come già accennato nell'introduzione del Capitolo 3, le motivazioni che hanno spinto sulla scelta della metodologia *LeanUX* sono dovute, sicuramente alla necessità di una guida per le interviste agli *Stakeholders*, ma anche alla volontà di voler basare lo sviluppo attorno all'utente attraverso continue interazioni tra il *Team di Sviluppo* e gli *Stakeholders*, esigenze che vengono soddisfatte dalla natura stessa di *LeanUX*. La disponibilità continua da parte degli *Stakeholders* è fondamentale, in quanto, senza di essa, non è possibile applicare tale metodologia.

La progettazione del sistema, è partita attraverso una prima intervista agli *Stakeholders*. Qui, solo ai fini del tirocinio, il lavoro di guidare l'intervista (e anche quelle a seguire) è stata assegnata al *Team di Sviluppo* invece che al *Product Owner*, contrariamente alla definizione di *Scrum*.

Come già detto, c'è stata inizialmente inesperienza da entrambe le parti, quindi nè chi poneva le domande e nè lo *Stakeholder* interessato sapevano precisamente come comportarsi e come esporre o raccogliere informazioni. Quindi, per facilitare l'intervista, si è utilizzato il *LeanUx Canvas v2*, ovvero le domande, poste dal *Team di Sviluppo*, si basavano su ciò che richiedevano i BOX del Canvas, fino ad ottenere delle *Ipotesi*.

Il totale delle interviste è stato 3, ovvero una per ogni *Stakeholder*, rispettivamente alla fine dello *Sprint* 1, 2 e 3. Anche se l'intervista era principalmente rivolta ad un singolo *Stakeholder* per volta, erano comunque presenti tutti. Questo ha fatto sì che, intervista dopo intervista, l'uso del Canvas è stato sempre minore, in quanto lo *Stakeholder* successivo sapeva già come illustrare le proprie esigenze e il materiale da fornire al *Team di Sviluppo*, e quest'ultimo sapeva già le domande da porre, come raccogliere le informazioni, e come proporre soluzioni. Si è passato quindi a delle interviste con la

metodologia di interviste semi-strutturate, alla discussione di un modello di dati che è stato concordato nel corso degli incontri.

Il motivo per cui le interviste sono state fatte ad un *Stakeholder* singolo per volta, è che ognuno di essi era interessato in particolare ad una sezione del sistema. Infatti il sistema di gestione sviluppato, è suddiviso sostanzialmente in 3 parti (vedere paragrafo *c. Struttura sistema e resoconto Sprint*), quindi per ogni intervista sono state illustrate le esigenze e raccolte informazioni riguardanti una parte.

Questo ha permesso di generare ogni volta *MVP* su una delle parti, fornendo così alla fine di uno *Sprint* una parte di sistema funzionante e testabile (come vuole per definizione Agile e LeanUX).

Inoltre, durante l'intervista (successiva alla prima) di uno *Stakeholder*, è stato raccolto il feedback dello *Stakeholder* precedente sulla parte a lui interessata, e durante gli *Sprint* si interagiva continuamente tra le parti per avere chiarimenti su alcuni aspetti.

Nei giorni seguenti all'intervista, lo *Stakeholder* era tenuto a fornire al *Team di Sviluppo* un file Excel con una tabella che descrivesse la struttura della parte del sistema di gestione a egli interessato, secondo le proprie esigenze e il proprio punto di vista, la quale veniva poi valutata dal Team, eventualmente modificata e riproposta allo *Stakeholder*.

In questo schema venivano indicate:

- INFRASTRUTTURA: indica l'infrastruttura a cui si fa riferimento;
- ATTIVITÀ: indica la tipologia di attività riferita ad una infrastruttura;
- STATUS: indica gli stati che un'attività può assumere;
- CATEGORIA: indica il nome di possibili parole chiave da utilizzare, riferite all'insieme *INFRASTRUTTURA - ATTIVITÀ - STATUS*, ovvero ad un *TASK* (vedi [APPENDICE - TASK](#)).

Un esempio di tabella:

	A	B	C	D	E	F	G	H	
1									
2		INFRASTRUTTURA							
3			ATTIVITA'		STATUS		Categoria		
4		SISTEMA ANTINCENDIO	Controllo Semestrale		Contattare Ditta		Controllo Antincendio		
5					Definizione appuntamento				
6					Sopralluogo effettuato				
7					Acquisizione rapporto ditta				
8				Malfunzionamento		Acquisizione rapporto ditta		Malfunzionamento Antincendio	
9					Definizione intervento				
10					Organizzazione intervento				
11						Intervento effettuato			
12				Acquisti		Definizione lista acquisti		Acquisti Antincendio	
13					Consegna PDA Acquisto				
14					Materiale Consegnato				
15					Materiale Testato / Via al pagamento				
16			GRUPPO ELETTROGENO	Controllo Mensile		Test da eseguire		Controllo Gruppo Elettrogeno	
17						Test eseguito			
18					Compilazione rapporto test				
19					Consegna rapporto				
20				Malfunzionamento		Acquisizione rapporto		Malfunzionamento Gruppo Elettrogeno	
21					Definizione intervento				
22					Organizzazione intervento				
23						Intervento effettuato			
24				Acquisti		Definizione lista acquisti		Acquisti Gruppo Elettrogeno	
25					Consegna PDA Acquisto				
26			Materiale Consegnato						
27			Materiale Testato / Via al pagamento						
28					Controllo da eseguire				
29					Controllo lista materiale				

FIGURA 11. Parte di tabella descrittiva della struttura di una sezione del sistema di gestione.

Da notare che ci sono state più interazioni con lo *Stakeholder* interessato alla prima intervista, dovuto al grado di incertezza generato dalla maggiore inesperienza, riducendosi per gli altri *Stakeholder* grazie ai feedback empirici ottenuti nelle interviste precedenti.

c. Struttura sistema e resoconto Sprint

Il sistema di gestione della stazione osservativa, come detto nel *Dominio Applicativo*, è stato progettato per gestire vari aspetti e problematiche della stazione osservativa, aspetti anche molto diversi tra di loro, ma che necessitavano di un sistema comune di gestione e condivisione per tutto lo staff dell'*Osservatorio D'Abruzzo*.

I settori principali, chiamati macro-progetti, su cui è stata basata la progettazione e lo sviluppo sono 3:

- 1. LOGISTICA;**
- 2. STRUMENTAZIONE SCIENTIFICA;**
- 3. ATTIVITÀ SCIENTIFICA;**

Ognuno dei quali era di interesse specifico per uno *Stakeholder* e si suddividono in ulteriori sotto-progetti, che verranno analizzati nei paragrafi successivi.

L'implementazione e la progettazione di un macro-progetto prendeva un intero *Sprint*, al termine del quale rappresentava il *MVP* da presentare allo *Stakeholder*. Quindi 3 dei 4 *Sprint* sono stati utilizzati per lo sviluppo dei tre macro-progetti.

Ora entriamo più nel dettaglio di ogni *Sprint*, gli *Stakeholders* interessati e le attività svolte.

Sprint 1

Questo *Sprint* è stato svolto dal 18 Gennaio 2021 al 31 Gennaio 2021.

E' il primo dei 4, quindi è stato caratterizzato nella prima settimana da una fase di studio degli strumenti e da un colloquio di presentazione ed esplorativo con tutti gli *Stakeholders*, oltre a frequenti colloqui con il *Product Owner*.

Il colloquio con tutti è servito a capire il progetto e le esigenze in linee generali, da cui è venuta fuori l'esigenza di una suddivisione in 3 parti del sistema di gestione e la

necessità immediata di una **Wiki**. Invece i frequenti colloqui con il *Product Owner*, oltre a pianificare il lavoro, hanno espresso il vincolo di utilizzo di **OpenProject**.

Invece la seconda fase è stata caratterizzata dalla preparazione dell'ambiente di sviluppo (vedere *APPENDICE - INSTALLAZIONE OPENPROJECT*), dalla preparazione delle domande per la prima intervista ad uno *Stakeholder* in particolare e dalle prime attività di sviluppo riguardanti la struttura generale dei menù del sistema di gestione e della *Wiki* per ogni macro-progetto.

Questo *Sprint* non interessava in particolare nessuno dei tre *Stakeholders*, in quanto non riguardava attività strettamente legate a parti di loro interesse.

Le attività svolte quindi sono state:

- *Studio Members* (0.5) - Studio documentazione di *OpenProject* relativa ai "Members".
- *Studio Wiki* (0.5) - Studio documentazione di *OpenProject* relativa alla "Wiki".
- *Preparazione ambiente Openproject* (2) - Installazione di *OpenProject*.
- *Preparazione interviste* (2) - Programmare le domande agli stakeholders identificando:
 - * due argomenti da approfondire per ogni stakeholder;
 - * le priorità maggiori.
- *Struttura Progetti e Wiki* (1.5) - Creazione della struttura dei vari progetti / sottoprogetti e rispettive Wiki.

**nelle parentesi () è indicata la stima del carico in giorni*

Da notare come la stima del carico della “*Preparazione interviste*” è stata di 2 giorni, tempo abbastanza elevato, questo dovuto all'inesperienza iniziale.

Inoltre, la stima del carico indicata non è stata quella iniziale, ma è stata modificata empiricamente alla fine dello *Sprint* perchè inizialmente era stata sovrastimata, dettato dal fatto che il *Product Owner* non conoscesse l'efficienza e l'efficacia del *Team di Sviluppo*. Questo solo per il primo *Sprint*, ai successivi invece non è stata modificata la stima iniziale.

Invece, la struttura dei progetti e i menù definiti sono frutto dell'intervista esplorativa con gli *Stakeholders* e sono stati modificati negli *Sprint* successivi, soprattutto i sottoprogetti, ma lasciando inalterata la divisione in 3 macro-progetti.

Per quanto riguarda la **Wiki**, la sua implementazione, oltre ad essere una delle esigenze nate durante il primo colloquio, è anche una delle funzionalità che ha permesso di fornire il *MVP* per il primo *Sprint*. La presenza di una Wiki al termine dello *Sprint* ha fornito un sistema non completo ma funzionante, testabile ed utilizzabile.

Sprint Review 1

Al termine del primo *Sprint*, per la precisione Venerdì 29 Gennaio 2021, si è tenuto il primo *Sprint Review*.

Durante questo evento è stata mostrata una demo di ciò che è stato implementato durante lo *Sprint*, ovvero la struttura provvisoria del sistema di gestione e la *Wiki*, in modo da far comprendere come utilizzare le varie funzionalità agli *Stakeholders*.

Inoltre, è stata effettuata la prima intervista ad uno *Stakeholder*, ovvero **Gaetano Valentini**, interessato alla parte **LOGISTICA** del sistema di gestione, utilizzando il *LeanUX Canvas v2*.

Le esigenze nate durante l'intervista, oltre a creare un sottoprogetto per ogni impianto o settore da lui gestito, anche quella di poter aprire al loro interno dei **TASK** che permettessero di condividere in tempo reale, con tutto lo staff dell'INAF, un'attività in esecuzione nella stazione osservativa, attraverso una descrizione dettagliata del *TASK*, la possibilità di definire uno stato e di avere un modo di comunicare in maniera istantanea.

Quindi le *IPOTESI* generate con il Canvas sono state:

- *Noi crediamo che L'APERTURA DI UN TASK DELLA MANUTENZIONE ORDINARIA sarà raggiunta se GAETANO riesce a INSERIRE LA TIPOLOGIA "MANUTENZIONE ORDINARIA" dalla LISTA DEI TYPE .*
- *Noi crediamo che L'APERTURA DI UN TASK DEI MALFUNZIONAMENTI sarà raggiunta se GAETANO riesce a INSERIRE LA TIPOLOGIA "MALFUNZIONAMENTO" dalla LISTA DEI TYPE .*
- *Noi crediamo che L'APERTURA DI UN TASK DEGLI ACQUISTI sarà raggiunta se GAETANO riesce a INSERIRE LA TIPOLOGIA "ACQUISTO" dalla LISTA DEI TYPE .*
- *Noi crediamo che la MONITORAZIONE DELLO STATO ATTUALE DI UN TASK DEI MALFUNZIONAMENTI sarà raggiunta se GAETANO riesce a INSERIRE UNO STATO dalla LISTA DEGLI STATI .*
- *Noi crediamo che la MONITORAZIONE DELLO STATO ATTUALE DI UN TASK DELLA MANUTENZIONE ORDINARIA sarà raggiunta se GAETANO riesce a INSERIRE UNO STATO dalla LISTA DEGLI STATI .*
- *Noi crediamo che la MONITORAZIONE DELLO STATO ATTUALE DI UN TASK DEGLI ACQUISTI sarà raggiunta se GAETANO riesce a INSERIRE UNO STATO dalla LISTA DEGLI STATI .*
- *Noi crediamo che la COMUNICAZIONE REAL-TIME sarà raggiunta se GAETANO riesce a INSERIRE COMMENTI all'interno di ogni TASK.*

Al termine dell'intervista e raccolte tutte le informazioni, è stato chiesto a **G. Valentini** di fornire, nei giorni seguenti, il file Excel con una tabella che descrivesse tutti gli impianti e settori, i tipi di attività, gli stati assunti e le categorie (vedere *Figura 11*), il quale è stato fornito tempestivamente il giorno seguente.

Infine è stato chiesto a tutti gli *Stakeholders* di testare e usare le funzionalità presenti per poter fornire un feedback durante lo *Sprint Review* successivo.

Menù, Wiki e Struttura Progetti

Come già detto, lo sviluppo in questo Sprint si è basato sull'implementazione della struttura dei progetti e sottoprogetti, della struttura base dei menù e sulla Wiki.

E' stata data una struttura provvisoria ai progetti, soprattutto per quanto riguarda i sottoprogetti. Infatti troviamo un macro-progetto chiamato "Campo Imperatore" che è il progetto generico che comprende i tre macro-progetti:

1. **Logistica**;
2. **Strumentazione Scientifica**;
3. **Osservazioni** (diventato poi **Attività Scientifica**);

Ognuno di questi tre ha dei sottoprogetti (es. *Osservazione 1*, *Telescopio 1* ecc...) che sono cambiati nel corso degli Sprint, in quanto non si avevano ancora informazioni sulle esigenze specifiche degli Stakeholders. Di seguito l'immagine di come si presentava la struttura dei progetti dopo tale implementazione.

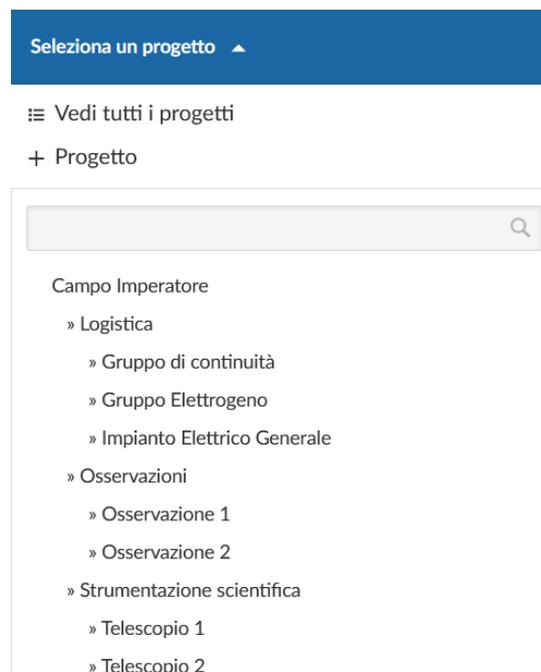


FIGURA 12. *Struttura provvisoria dei progetti del sistema di gestione dello Sprint 1.*

Per ogni macro-progetto e sottoprogetto è stata implementata la *Wiki*, per cui gli *Stakeholders* hanno potuto da subito testarla e fornire feedback nello *Sprint* successivo. Di seguito un'immagine di una pagina Wiki.

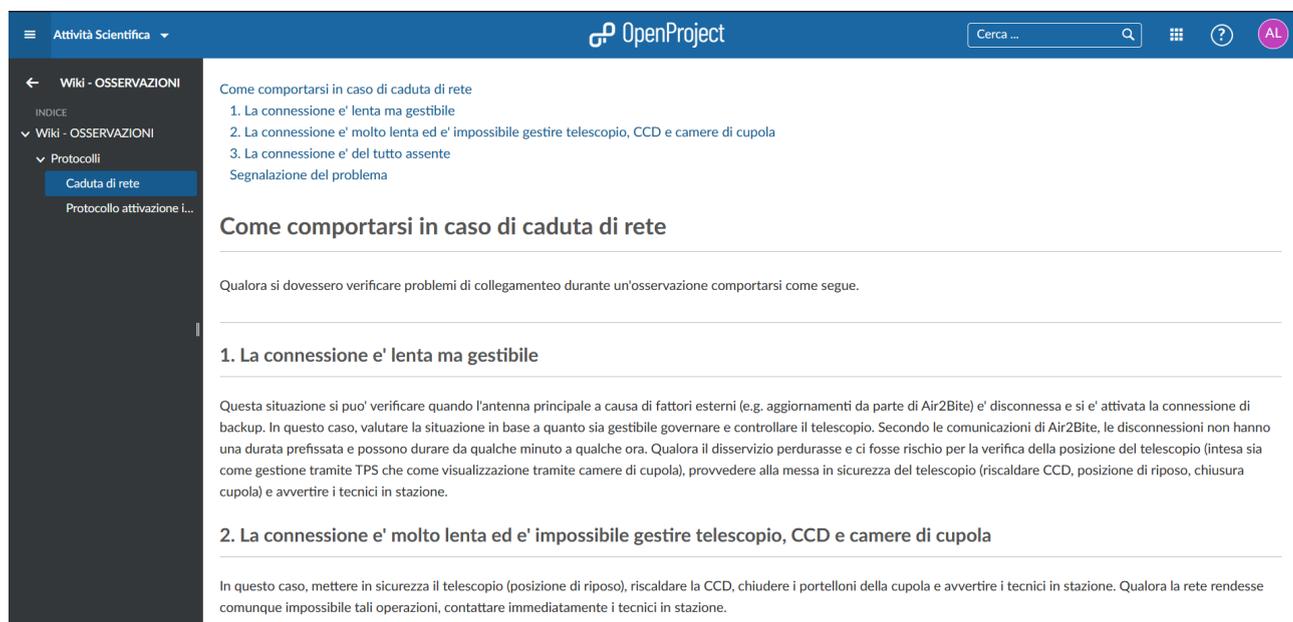


FIGURA 13. Esempio di una pagina Wiki, in questo caso del macro-progetto Attività Scientifica (ex Osservazioni).

Da notare la possibilità di creare una struttura di pagine (Genitore: Wiki - OSSERVAZIONI, Figlio: Protocolli ecc.....), dare una formattazione alle pagine, inserire immagini ed allegati.

Inoltre ai vari progetti sono stati ordinati i menù lasciando solo le voci utilizzate, perché alla creazione di un nuovo progetto, *OpenProject* lascia di default tutte le voci del menù.

L'immagine mostra a sinistra il menù di default e a destra come è stato ridotto.

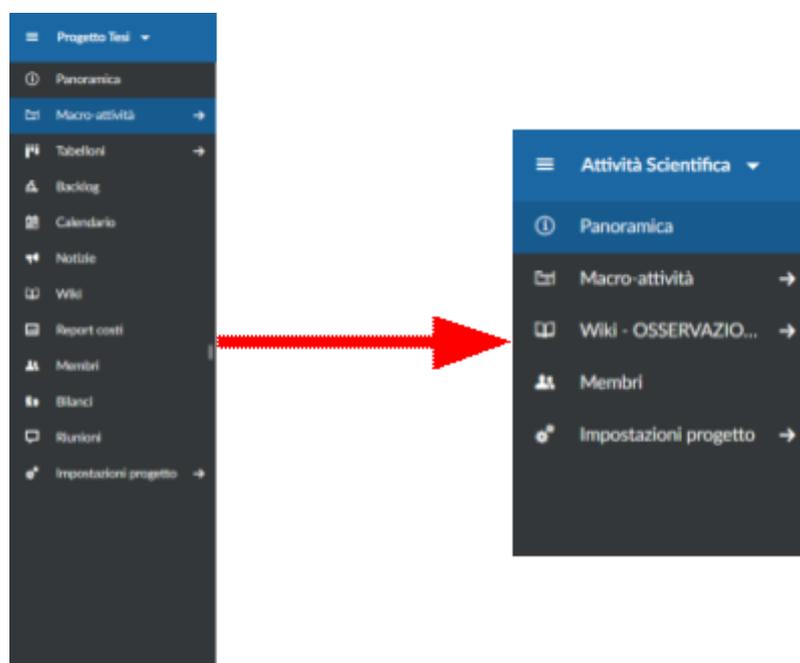


FIGURA 14. A sinistra il menù di default alla creazione di un nuovo progetto, a destra come è stato ridotto il menù.

La voce “*Panoramica*” presenta, appunto, una panoramica del progetto / sottoprogetto in questione e si presenta come in *Figura 15*. Invece, “*Macro-attività*” permette di visualizzare tutti TASK aperti e chiusi nel rispettivo progetto / sottoprogetto, e permette di gestire i TASK, si mostra come in *Figura 16*.

“*Wiki*” è la voce del menù che permette di accedere alle pagine della wiki, come mostrato nella *Figura 13* precedente.

Infine “*Membri*” e “*Impostazioni Progetto*”, il primo mostra una pagina con tutti gli utenti che possono accedere al progetto / sottoprogetto e i rispettivi poteri / ruoli (*Figura 17*) e la possibilità di gestire gli accessi e i poteri ad ogni utente, il secondo invece permette all’admin di sistema di gestire delle impostazioni del progetto in questione.

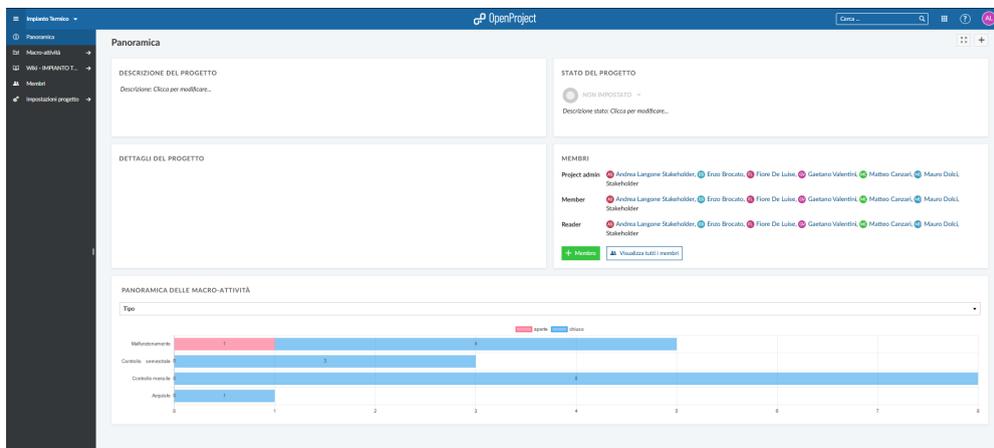


FIGURA 15. Esempio di “Panoramica” di un progetto.

ID	PROGETTO	SOGGETTO	TIPO	STATO	PRIORITÀ	ASSEGNATARIO
162	Impianto Termico	Cappello tubo di scarico fumi	MALFUNZIONAMENTO	Organizzazione intervento	Normal	Giuseppe Valentini
165	Sensistica	Protocollo Sensistica	ACQUISTO	Definizione lista acquisti	Normal	Matteo Carcari
199	Diagnostica e Diagnostica	Telesempio diagnostico	ACQUISTO	Definizione lista acquisti	Normal	Giuseppe Valentini
223	Software di rete	Togliere velocità di rete dalla linea di rete backup	TEST	Nuova attività	Normal	Matteo Carcari
224	Software di rete	Definire una procedura quando si attiva la linea di backup	SVILUPPO SW	Nuova attività	Normal	Matteo Carcari
225	Software di rete	Sviluppo software Valada	SVILUPPO SW	Nuova attività	Normal	Matteo Carcari
226	Software di rete	Problemi con il server di rete posta	MALFUNZIONAMENTO	Intervento effettuato	Normal	Matteo Carcari
238	Servizi	Controllo finestre esterne	CONTROLLO ANNUALE	Correggere rapporto	Normal	Giuseppe Valentini

FIGURA 16. Esempio di “Macro-attività” di un progetto.

COGNOME	NOME	EMAIL	RUOLI	GRUPPI	STATO
Valentini	Giuseppe	giuseppe.valentini@unipi.it	Member, Reader, Project admin	Stakeholder	attivo
Orion	Francesca	francesca.orion@unipi.it	Reader	Astronomi	attivo
Langone Stakeholder	Andrea	15081180@studenti.unipi.it	Member, Reader, Project admin	Stakeholder	attivo
Langone	Andrea	andrea.langone7@unipi.it	Project admin	(Tutti i ruoli a causa dello stato di admin)	attivo
Daki	Mauro	mauro.daki@unipi.it	Member, Reader, Project admin	Stakeholder	attivo
Di Cianno	Antonio	antonio@unipi.it	Member, Reader		attivo
De Lallo	Flore	flore.de.lallo@unipi.it	Member, Reader, Project admin	Stakeholder	attivo
Carli	Roberta	roberta.carli@unipi.it	Reader	Astronomi	attivo
Carcari	Matteo	matteo.carcari@unipi.it	Member, Reader, Project admin	Astronomi, Stakeholder	attivo
Brocato	Enzo	enzo.brocato@unipi.it	Member, Reader, Project admin	Stakeholder	attivo
			Reader	Astronomi	attivo

FIGURA 17. Esempio di “Membri” di un progetto.

Sprint 2

Questo *Sprint* è stato svolto dal 1 Febbraio 2021 al 14 Febbraio 2021.

Durante lo *Sprint Planning* è stata analizzata la tabella fornita da **Gaetano Valentini** e cercata una possibile soluzione. La tabella ha fornito queste informazioni:

LEGGENDA:

- In verde le **INFRASTRUTTURE**;
- In blu le **TIPOLOGIE** di TASK;
- In arancione gli **STATI** dei TASK.

Sistema antincendio		
Controllo semestrale	Malfunzionamento	Acquisti
Contattare ditta	Acquisizione rapporto ditta	Definizione lista acquisti
Definizione appuntamento	Definizione intervento	Consegna PDA, Acquisto
Sopralluogo effettuato	Organizzazione intervento	Materiale Consegnato
Acquisizione rapporto ditta	Intervento effettuato	Materiale Testato

Gruppo Elettrogeno		
Controllo mensile	Malfunzionamento	Acquisti
Test da eseguire	Acquisizione rapporto	Definizione lista acquisti
Test eseguito	Definizione intervento	Consegna PDA, Acquisto
Compilazione rapporto test	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Cassette Primo Soccorso	
Controllo semestrale	Acquisti
Contattare ditta	Definizione lista acquisti
Controllo lista materiale	Consegna PDA, Acquisto
Compilazione rapporto	Materiale Consegnato
Consegna rapporto	Materiale Testato

Illuminazione d'emergenza		
Controllo semestrale	Malfunzionamento	Acquisti
Test da eseguire	Acquisizione rapporto	Definizione lista acquisti
Test eseguito	Definizione intervento	Consegna PDA, Acquisto
Compilazione rapporto test	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Cisterne Gasolio Riscaldamento		
Controllo semestrale	Malfunzionamento	Acquisti
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti
Controllo eseguito	Definizione intervento	Consegna PDA Acquisto
Compilazione rapporto controllo	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Cupole Telescopi		
Manutenzione annuale	Malfunzionamento	Acquisti
Manutenzione da eseguire	Acquisizione rapporto	Definizione lista acquisti
Manutenzione eseguita	Definizione intervento	Consegna PDA Acquisto
Compilazione rapporto manut.	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Impianto Termico 1		
Manutenzione semestrale	Malfunzionamento	Acquisti
Manutenzione da eseguire	Acquisizione rapporto ditta	Definizione lista acquisti
Appuntamento terzo responsabile	Definizione intervento	Consegna PDA Acquisto
Manutenzione effettuata	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Impianto Termico 2		
Controllo mensile	Malfunzionamento	Acquisti
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti
Controllo eseguito	Definizione intervento	Consegna PDA Acquisto
Compilazione rapporto controllo	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Impianto Refrigerante CCD Schmidt		
Controllo semestrale	Malfunzionamento	Acquisti
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti
Controllo eseguito	Definizione intervento	Consegna PDA Acquisto
Compilazione rapporto controllo	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Impianto Liquefattore Azoto		
Controllo semestrale	Malfunzionamento	Acquisti
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti
Controllo eseguito	Definizione intervento	Consegna PDA Acquisto
Compilazione rapporto controllo	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Materiale Tecnico e Sicurezza del personale	
Controllo semestrale	Acquisti
Controllo da eseguire	Definizione lista acquisti
Controllo lista materiale	Consegna PDA Acquisto
Compilazione rapporto	Materiale Consegnato
Consegna rapporto	Materiale Testato

Materiale di Consumo	
Controllo semestrale	Acquisti
Controllo da eseguire	Definizione lista acquisti
Controllo lista materiale	Consegna PDA Acquisto
Compilazione rapporto	Materiale Consegnato
Consegna rapporto	Materiale Testato

Integrità Terrazzo		
Controllo Bimestrale	Malfunzionamento	Acquisti
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti
Controllo eseguito	Definizione intervento	Consegna PDA Acquisto
Compilazione rapporto controllo	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Servizio Pulizie
Controllo mensile
Appuntamento con la ditta
Controllo date e firme servizio
Compilazione rapporto
Consegna rapporto

Accesso Funivia del Personale
Controllo annuale
Contatti con Centro Turistico G.S.
Rinnovo Convenzione
Convenzione Rinnovata
Accesso Consentito

Funzionalità Servizi Igienici		
Controllo Annuale	Malfunzionamento	Acquisti
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti
Controllo eseguito	Definizione intervento	Consegna PDA Acquisto
Compilazione rapporto controllo	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Recinzione Osservatorio		
Montaggio / Smontaggio	Malfunzionamento	Acquisti
Definizione montaggio/montaggio	Acquisizione rapporto	Definizione lista acquisti
Controllo integrità delle parti	Definizione intervento	Consegna PDA Acquisto
Manutenzione effettuata	Organizzazione intervento	Materiale Consegnato
Compilazione e consegna rapporto	Intervento effettuato	Materiale Testato

Controllo Sala Conferenze		
Controllo Annuale	Malfunzionamento	Acquisti
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti
Controllo eseguito	Definizione intervento	Consegna PDA Acquisto
Compilazione rapporto controllo	Organizzazione intervento	Materiale Consegnato
Consegna rapporto	Intervento effettuato	Materiale Testato

Visite Guidate del Centro Turistico	
Organizzazione	Acquisti
Convenzione da Rinnovare	Definizione lista acquisti
Definizione del calendario visite	Consegna PDA Acquisto
Svolgimento delle visite con report	Materiale Consegnato
Chiusura delle visite e analisi	Materiale Testato

Didattica	
Organizzazione	Acquisti
Contatti con scuole	Definizione lista acquisti
Definizione del calendario didattico	Consegna PDA, Acquisto
Svolgimento lezioni con report	Materiale Consegnato
Chiusura della attività di didattica	Materiale Testato

Manifestazioni	
Organizzazione	Acquisti
Definizione della manifestazione	Definizione lista acquisti
Organizzazione del Personale e della Strumentazione	Consegna PDA, Acquisto
Svolgimento della manifestazione	Materiale Consegnato
Report della manifestazione	Materiale Testato

Analizzate le informazioni, la soluzione da noi proposta, ha cercato di raggruppare le infrastrutture che trattassero quasi le stesse cose, di utilizzare gli stessi stati per più tipologie di TASK in modo da ridurre il numero, inoltre si è cercato di utilizzare gli stessi stati per le stesse tipologie di TASK per non creare confusione in fase di creazione di nuove features da parte dell'Admin.

Quindi le infrastrutture sono state sostanzialmente rappresentate dai sotto-progetti, ognuno di essi avente una sezione MACRO-ATTIVITÀ' dove poter gestire e visualizzare i TASK a cui poter modificare i rispettivi stati e informazioni, inserendo al proprio interno, ove necessario, dei CAMPI PERSONALIZZATI per poter soddisfare alcune esigenze.

La soluzione da noi proposta è questa:

LEGGENDA:

- In verde le **INFRASTRUTTURE**;
- In blu le **TIPOLOGIE** di TASK;
- In arancione gli **STATI** dei TASK;
- In rosso i **CAMPI PERSONALIZZATI** dei TASK;
- In rosa le **SCELTE** dei CAMPI PERSONALIZZATI.

Didattica e Divulgazione			
Controllo annuale	Malfunzionamento	Acquisto	Gestione visite
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti	Visita fissata
Compilazione rapporto	Definizione intervento	Consegna PDA acquisto	Visita effettuata
Consegna rapporto	Organizzazione intervento	Materiale Consegnato	Report della visita
Controllo eseguito	Intervento effettuato	Pagamento da effettuare	
		Pagamento inviato	

Gruppo Elettrogeno		
Controllo mensile	Malfunzionamento	Acquisto
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti
Compilazione rapporto	Definizione intervento	Consegna PDA acquisto
Consegna rapporto	Organizzazione intervento	Materiale consegnato
Controllo eseguito	Intervento effettuato	Pagamento da effettuare
		Pagamento inviato

Illuminazione d'Emergenza		
Controllo semestrale	Malfunzionamento	Acquisto
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti
Compilazione rapporto	Definizione intervento	Consegna PDA acquisto
Consegna rapporto	Organizzazione intervento	Materiale consegnato
Controllo eseguito	Intervento effettuato	Pagamento da effettuare
		Pagamento inviato

Sistema Anti-Incendio		
Controllo semestrale	Malfunzionamento	Acquisto
Contattare ditta	Acquisizione rapporto	Definizione lista acquisti
Definizione appuntamento	Definizione intervento	Consegna PDA acquisto
Sopralluogo effettuato	Organizzazione intervento	Materiale consegnato
Acquisizione rapporto	Intervento effettuato	Pagamento da effettuare
		Pagamento inviato

Materiale di Sicurezza e Consumo	
Controllo semestrale	Acquisto
Controllo da eseguire	Definizione lista acquisti
Compilazione rapporto	Consegna PDA Acquisto
Consegna rapporto	Materiale consegnato
Controllo eseguito	Pagamento da effettuare
	Pagamento inviato

Categoria sicurezza
Cassette primo soccorso
Materiale sicurezza e tecnico del personale
Materiale di consumo

Impianti Cupole e Telescopi			
Controllo semestrale	Malfunzionamento	Acquisto	Manutenzione annuale
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti	Compilazione rapporto
Compilazione rapporto	Definizione intervento	Consegna PDA acquisto	Manutenzione da eseguire
Consegna rapporto	Organizzazione intervento	Materiale Consegnato	Consegna rapporto
Controllo eseguito	Intervento effettuato	Pagamento da effettuare	Manutenzione eseguita
		Pagamento inviato	

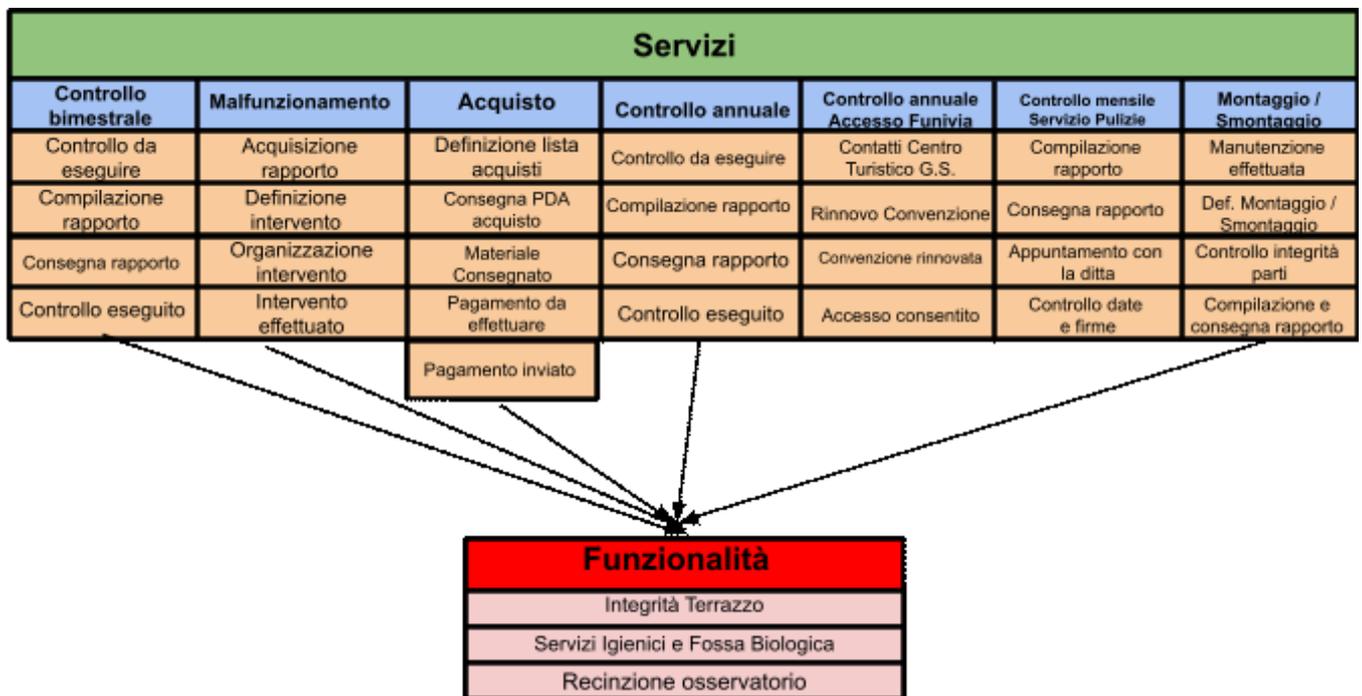
Telescopio
AZT-24
Schmidt

Impianto
Cupole telescopi
Impianto refrigerante CCD
Impianto liquefattore azoto

Cupola
AZT-24
Schmidt

Impianto Termico				
Controllo semestrale	Malfunzionamento	Acquisto	Controllo mensile	Manutenzione semestrale
Controllo da eseguire	Acquisizione rapporto	Definizione lista acquisti	Controllo da eseguire	Manutenzione da eseguire
Compilazione rapporto	Definizione intervento	Consegna PDA acquisto	Compilazione rapporto	Appuntamento con responsabile
Consegna rapporto	Organizzazione intervento	Materiale Consegnato	Consegna rapporto	Manutenzione effettuata
Controllo eseguito	Intervento effettuato	Pagamento da effettuare	Controllo eseguito	Consegna rapporto
		Pagamento inviato		

Impianto
Impianto Termico 1
Impianto Termico 2
Cisterne gasolio riscaldamento



Quindi lo Sprint è stato caratterizzato soprattutto da attività di sviluppo della soluzione della sezione *LOGISTICA*.

Le attività svolte sono state:

- Creazione Stories Sprint 2 (0.25) - Creazione delle attività su *Trello* per lo *Sprint 2*;
- Studio Work packages (Macro-attività) (0.5) - Studio documentazione *OpenProject* riguardo le Macro-Attività;
- Individuazione Stati (0.5) - Individuazione degli stati da assegnare ai vari TASK.
- Modifica struttura sottoprogetti (0.25) - Modifica alla struttura dei sottoprogetti d'esempio implementata nel primo sprint, aggiungendo sottoprogetti sulla base delle informazioni date da Gaetano.
- Implementazione sezione Macro-Attività (0.25) - Implementazione della sezione "Macro-attività" per ogni sottoprogetto.
- Type Macro-attività - Manutenzione ordinaria (0.5) - Implementazione del TASK di tipo MANUTENZIONE ORDINARIA.

- Type Macro-attività - Malfunzionamenti (0.25) - Implementazione del TASK di tipo MALFUNZIONAMENTO.
- Type Macro-attività - Acquisti (0.25) - Implementazione del TASK di tipo ACQUISTI.
- Stati Macro-attività - Malfunzionamenti (0.25) - Implementazione degli stati di per il TASK di tipo MALFUNZIONAMENTO.
- Stati Macro-attività - Manutenzione ordinaria (0.25) - Implementazione degli stati di per il TASK di tipo MANUTENZIONE ORDINARIA.
- Stati Macro-attività - Acquisti (0.25) - Implementazione degli stati di per il TASK di tipo ACQUISTO.
- Individuazione campi personalizzati (0.5) - Individuazione campi personalizzati da utilizzare per descrivere al meglio il task.
- Campi personalizzati Macro-attività - Malfunzionamenti (0.5) - Implementazione campi personalizzati per il task MALFUNZIONAMENTO.
- Campi personalizzati Macro-attività - Manutenzione ordinaria (0.5) - Implementazione campi personalizzati per il task MANUTENZIONE ORDINARIA.
- Campi personalizzati Macro-attività - Acquisti (0.5) - Implementazione campi personalizzati per il task ACQUISTO.
- Preparazione intervista Mauro (0.5) - Preparazione domande per l'intervista a Mauro Dolci.

**nelle parentesi () è indicata la stima del carico in giorni*

Per quanto riguarda l'esigenza di voler avere un sistema di messaggistica condivisa altri membri dello staff, non c'è stata nessuna reale implementazione in quanto all'interno di un TASK è possibile inserire commenti, taggare membri dello staff e visualizzare le modifiche apportate al TASK stesso, per cui viene utilizzato tale sistema per poter comunicare.

Sprint Review 2

Al termine del secondo *Sprint*, per la precisione Venerdì 12 Febbraio 2021, si è tenuto il secondo *Sprint Review*.

Durante questo evento è stata mostrata una demo di ciò che è stato implementato durante lo *Sprint*, ovvero la sezione *LOGISTICA* interessata in particolare a G. Valentini, in modo da fargli comprendere come utilizzare le varie funzionalità e illustrargli la soluzione proposta.

Inoltre, dopo la demo, si è passati all'intervista a Mauro Dolci, interessato alla sezione del progetto **STRUMENTAZIONE SCIENTIFICA**.

Come detto nei paragrafi precedenti, il *LeanUX Canvas v2* è stato utilizzato molto meno rispetto alla prima intervista, in quanto gli stakeholders già sapessero come esporre le proprie esigenze e il materiale da fornire al Team di Sviluppo. Infatti le esigenze di **M. Dolci** erano pressoché le stesse di **G. Valentini** ma applicate al suo settore.

Le tipologie di TASK sono molto simili rispetto al settore *LOGISTICA*, in quanto si tratta di strumenti scientifici/astronomici o componenti di telescopi, per cui *MALFUNZIONAMENTO*, *CONTROLLO* (simile a *MANUTENZIONE ORDINARIA*) e *ACQUISTO* sono le tipologie usate.

L'unica esigenza differente è stata quella di poter indicare tramite un campo personalizzato il telescopio a cui fa riferimento il TASK, in modo da poterli filtrare in base al nome del telescopio. Quindi l'ipotesi generata dal Canvas è questa:

- Noi crediamo che LA POSSIBILITÀ' DI ASSOCIARE UN TELESCOPIO SPECIFICO AD UN TASK sarà raggiunta se MAURO riesce a SCEGLIERE IL TELESCOPIO da UN CAMPO PERSONALIZZATO .

Al termine dell'intervista e raccolte tutte le informazioni, è stato chiesto a **M. Dolci** di fornire, nei giorni seguenti, il file Excel con tutte le informazioni su infrastrutture, attività e stati.

Infine è stato chiesto a **G. Valentini** di testare e usare le funzionalità sviluppate per poter fornire un feedback durante lo *Sprint Review* successivo e sono state raccolti i feedback da tutti gli *Stakeholders* riguardo il MVP dello *Sprint 1* (Wiki e Menù), dove non è stato evidenziato alcun problema.

Logistica

Lo sviluppo in questo Sprint si è basato per la maggior parte sull'implementazione del progetto LOGISTICA.

A partire dalla soluzione elaborata e dalle informazioni fornite da G. Valentini, sono stati definiti in primis i nuovi sottoprogetti sulla base delle infrastrutture proposte nella soluzione. L'immagine mostra i sottoprogetti di LOGISTICA definiti:

- » Logistica
 - » Didattica e Divulgazione
 - » Gruppo Elettrogeno
 - » Illuminazione d'emergenza
 - » Impianti Cupole e Telescopi
 - » Impianto Termico
 - » Materiale di sicurezza e consumo
 - » Sensoristica
 - » Servizi
 - » Sistema anti-incendio
 - » Software & rete

FIGURA 18. *Sottoprogetti definiti per il macro-progetto LOGISTICA.*

I sottoprogetti “Sensoristica” e “Software & Rete”, sono stati definiti dopo il lavoro di tirocinio da un altro sviluppatore. Per ognuno di questi, sono state definite le corrispondenti tipologie di TASK e rispettivi stati. Inoltre, ove richiesto, sono stati aggiunti dei campi personalizzati all'interno dei TASK. Nella tabella sono elencate tutte le tipologie di TASK, gli stati e i campi personalizzati associati:

N.B. : Nelle parentesi dei campi personalizzati sono indicati nei TASK di quali sottoprogetti è possibile trovarli.

TIPOLOGIA	STATI	CAMPI PERSONALIZZATI
Malfunzionamento	Acquisizione rapporto Definizione intervento Organizzazione intervento Intervento effettuato	Telescopio (Impianti Cupole e Telescopi) Impianto (Impianti Cupole e Telescopi) (Impianto Termico) Funzionalità (Servizi)
Acquisto	Definizione lista acquisti Consegna PDA acquisto Materiale consegnato Pagamento da effettuare Pagamento inviato	Telescopio (Impianti Cupole e Telescopi) Impianto (Impianti Cupole e Telescopi) (Impianto Termico) Funzionalità (Servizi) Categoria Sicurezza (Materiale di Sicurezza e Consumo)
Controllo Annuale	Controllo da eseguire Compilazione rapporto Consegna rapporto Intervento effettuato	Funzionalità (Servizi)
Gestione Visite	Visita fissata Visita effettuata Report della visita	
Controllo Mensile	Controllo da eseguire Compilazione rapporto Consegna rapporto Intervento effettuato	Impianto (Impianto Termico)
Controllo Semestrale	Controllo da eseguire Compilazione rapporto Consegna rapporto Intervento effettuato	Telescopio (Impianti Cupole e Telescopi) Impianto (Impianti Cupole e Telescopi) (Impianto Termico) Categoria Sicurezza (Materiale di Sicurezza e Consumo)
Manutenzione annuale	Compilazione rapporto Manutenzione da eseguire Consegna rapporto Manutenzione eseguita	Impianto (Impianti Cupole e Telescopi) Cupola (Impianti Cupole e Telescopi)
Manutenzione semestrale	Manutenzione da eseguire Appuntamento con responsabile Manutenzione effettuata Consegna rapporto	Impianto (Impianto Termico)
Controllo bimestrale	Controllo da eseguire Compilazione rapporto Consegna rapporto Intervento effettuato	Funzionalità (Servizi)
Controllo Annuale Accesso Funivia	Contatti Centro Turistico G.S. Rinnovo Convenzione Convenzione Rinnovata Accesso Consentito	Funzionalità (Servizi)
Controllo Mensile Servizio Pulizie	Compilazione rapporto Consegna rapporto Appuntamento con la ditta Controllo date e firme	Funzionalità (Servizi)

Montaggio / Smontaggio	Manutenzione effettuata Def. Montaggio / Smontaggio Controllo integrità parti Compilazione e consegna rapporto	Funzionalità (<i>Servizi</i>)
------------------------	---	------------------------------------

Tramite campi personalizzati è possibile scegliere tra le voci:

- CATEGORIA SICUREZZA (*Materiale di Sicurezza e Consumo*):
 - Cassette primo soccorso;
 - Materiale sicurezza e tecnico del personale;
 - Materiale di consumo.

- TELESCOPIO (*Impianti Cupole e Telescopi*):
 - AZT-24;
 - Schmidt.

- IMPIANTO (*Impianti Cupole e Telescopi*):
 - Cupole telescopi;
 - Impianto refrigerante CCD;
 - Impianto liquefattore azoto.

- CUPOLA (*Impianti Cupole e Telescopi*):
 - AZT-24;
 - Schmidt.

- IMPIANTO (*Impianto Termico*):
 - Impianto Termico 1;
 - Impianto Termico 2;
 - Cisterne gasolio riscaldamento.

- FUNZIONALITÀ (*Servizi*):
 - Integrità terrazzo;

- Servizi igienici e fossa biologica;
- Recinzione osservatorio.

Nell'immagine seguente è rappresentato un TASK del sottoprogetto "Servizi" con tipologia CONTROLLO ANNUALE e stato "CONSEGNA RAPPORTO", inoltre sono mostrate le possibili scelte del campo personalizzato "FUNZIONALITÀ":

The screenshot displays the OpenProject interface for a task. The task title is "CONTROLLO ANNUALE Controllo finestre esterne". The status is "Consegna rapporto". The description contains two paragraphs: "Sono state controllate le finestre esterne. Alcune sono state trovate non bene in asse." and "Sono in corso i lavori di ripristino piena funzionalità meccanica delle finestre. Effettuato interventi al lato sud. Controllore Tedesco: "Ho finito di sistemare tutte le finestre del lato sud, in alcune, quelle che apriamo più frequentemente ho effettuato una modica per non far scivolare la regolazione della maniglia."". The "PERSONE" section lists "Assegnatario" as "-" and "Responsabile" as "Gaetano Valentini". The "DETTAGLI" section shows "Data" as "Recinzione Osservatorio" and "Avanzamento (%)" as 40%. The "Funzionalità" dropdown menu is open, showing options: "-", "Integrità Terrazzo", "Servizi Igenici e Fossa Bio...", and "Recinzione Osservatorio". The "Priorità" is set to "Normal".

FIGURA 19. Esempio di un TASK del progetto LOGISTICA e sottoprogetto SERVIZI.

Per avere un quadro dettagliato di un TASK, vedere l'[APPENDICE - TASK](#)

Sprint 3

Questo *Sprint* è stato svolto dal 15 Febbraio 2021 al 28 Febbraio 2021.

Durante lo *Sprint Planning* non è stata analizzata la tabella del file Excel, in quanto **M. Dolci** ha ritardato la sua consegna, per cui sono state definite nel *Backlog* alcune attività di studio complementari per poter sopperire alla mancanza delle informazioni e raggiungere un Carico di almeno 6 giorni. Tabella che abbiamo ottenuto, simile a quella descritta nello *Sprint 2*, subito dopo lo *Sprint Planning*, e per pianificare le attività di sviluppo è stato prolungata la durata di un *Daily Scrum*.

Questo aspetto fa emergere un difetto, delle metodologie *Agile* e *LeanUX*, ovvero la mancanza di continua interazione con gli Stakeholder può portare, come in questo caso, al blocco di alcune attività, ma fa emergere anche un vantaggio , ovvero la possibilità di sopperire a tale blocco cercando comunque di fornire alla fine dello *Sprint* un *MVP*.

Detto ciò, dopo l'analisi della tabella Excel ricevuta, la soluzione proposta è molto simile a quella della parte *LOGISTICA*, quindi ha cercato di raggruppare le infrastrutture che trattassero quasi le stesse cose, di utilizzare gli stessi stati per più tipologie di TASK in modo da ridurre il numero, inoltre si è cercato di utilizzare gli stessi stati per le stesse tipologie di TASK. Invece per esigenza di poter scegliere il telescopio di riferimento all'interno di un TASK è stata soddisfatta inserendo un campo personalizzato dove poter selezionare da un elenco il telescopio d'interesse (*Figura 21*).

Quindi lo *Sprint* è stato caratterizzato soprattutto da attività di sviluppo della soluzione della sezione *STRUMENTAZIONE SCIENTIFICA*.

Le attività svolte sono state:

- Creazione stories Sprint 3 (0.25) - Creazione delle attività su *Trello* per lo *Sprint 3*;
- Studio Milestones (0.5) - Studio documentazione *OpenProject* riguardo le *Milestones*;

- Individuazione sottoprogetti, macro-attività, campi personalizzati e stati (1.75) - Individuare, dalle informazioni fornite da Mauro, i vari sottoprogetti da implementare, le macro-attività, le campi personalizzati e gli stati da includere.
- Definizione sottoprogetti (0.5) - Implementazione della struttura dei sottoprogetti.
- Definizione Macro-attività (1) - Implementazione delle Macro-attività in ogni sottoprogetto.
- Definizione Stati (1) - Implementazione degli stati per ogni macro attività.
- Definizione Campi personalizzati (1) - Implementazione dei campi personalizzati nei TASK di interesse.
- Scaletta Demo (0.5) - Definizione della scaletta per la demo da presentare agli stakeholders.
- Preparazione Intervista Fiore (0.25) - preparazione domande per l'intervista allo stakeholder Fiore De Luise.

**nelle parentesi () è indicata la stima del carico in giorni*

L'unica criticità riscontrata in questo *Sprint* è stata la mancata ricezione rapida del file Excel che ne ha bloccato per un giorno la progettazione e lo sviluppo.

Sprint Review 3

Al termine del terzo *Sprint*, per la precisione Venerdì 26 Febbraio 2021, si è tenuto il terzo *Sprint Review*.

Durante questo evento è stata mostrata una demo di ciò che è stato implementato durante lo *Sprint*, ovvero la sezione *STRUMENTAZIONE SCIENTIFICA* interessata in particolare a M. Dolci, in modo da fargli comprendere come utilizzare le varie funzionalità e illustrargli la soluzione proposta.

Inoltre, dopo la demo, si è passati all'intervista a **Fiore De Luise**, interessato alla sezione del progetto **ATTIVITÀ' SCIENTIFICA**.

In questo caso, il *LeanUX Canvas v2* non è stato utilizzato, in quanto lo stakeholder, essendo presente alle interviste precedenti, già sapeva come esporre le proprie esigenze e il materiale da fornire al Team di Sviluppo.

Le esigenze illustrate da F. De Luise, essendo addetto alle osservazioni astronomiche, sono quelle di poter rappresentare un'osservazione e il backup dei dati di quest'ultima, segnalare problemi vari, eventuali analisi di dati e pubblicazioni, con un TASK.

Per il TASK relativo al backup, si è richiesto di poterlo creare automaticamente ogni qual volta viene creato un TASK di osservazione e collegarlo ad esso.

Inoltre è stato richiesto di poter caricare allegati all'interno dei TASK e di indicare i progetti scientifici a cui appartiene l'osservazione.

Al termine dell'intervista e raccolte tutte le informazioni, è stato chiesto a **F. De Luise** di fornire il file Excel con tutte le informazioni su infrastrutture, attività e stati. Tale file ci è stato fornito immediatamente, in quanto lo Stakeholder, ascoltando le esperienze degli altri due, lo stava già preparando prima che "arrivasse il suo turno".

Infine è stato chiesto a **M. Dolci** di testare e usare le funzionalità sviluppate per poter fornire un feedback durante lo *Sprint Review* successivo e sono stati raccolti i feedback da **G. Valentini** riguardo il MVP dello *Sprint 2* (progetto LOGISTICA), dove è stata riferita una sua mancanza nel file Excel fornito al Team di Sviluppo.

Tale mancanza riguardava il sottoprogetto di DIDATTICA E DIVULGAZIONE, e la tabella comprensiva di tali dati è stata prontamente fornita durante questo Sprint Review.

Strumentazione Scientifica

Lo sviluppo in questo *Sprint* si è basato per la maggior parte sull'implementazione del progetto STRUMENTAZIONE SCIENTIFICA.

A partire dalla soluzione elaborata e dalle informazioni fornite da M. Dolci, sono stati definiti in primis i nuovi sottoprogetti sulla base delle infrastrutture proposte nella

soluzione. L'immagine mostra i sottoprogetti di STRUMENTAZIONE SCIENTIFICA definiti:

- » Strumentazione scientifica
 - » Accessori
 - » All-Sky System
 - » Altri Dispositivi
 - » Weather Station
 - » Sviluppo
 - » Telescopi
 - » CCD
 - » Cupola
 - » Liquefattore
 - » Settori
 - » Strumentazione
 - » Swircam

FIGURA 20. *Sottoprogetti definiti per il macro-progetto STRUMENTAZIONE SCIENTIFICA.*

Da notare come, in questo caso, è stato definito un ulteriore sottolivello di progetti, in quanto c'era una maggiore necessità di suddividere le varie infrastrutture a livello di TASK ma comunque dargli un raggruppamento generico.

Ad esempio, dal progetto “*Accessori*” è possibile visualizzare e gestire i TASK dei suoi sottoprogetti, ovvero “*All-Sky System*”, “*Weather Station*” e “*Altri Dispositivi*”, ma questi ultimi non possono né visualizzare e né gestire i TASK di “*Accessori*”, tanto meno quelli dei progetti allo stesso livello.

Per ognuno dei progetti di livello “più basso” e “*Sviluppo*”, quindi esclusi “*Accessori*” e “*Telescopi*”, sono state definite le corrispondenti tipologie di TASK e rispettivi stati. Inoltre, ove richiesto, sono stati aggiunti dei campi personalizzati all'interno dei TASK. Nella tabella sono elencate tutte le tipologie di TASK, gli stati e i campi personalizzati associati:

N.B. : Nelle parentesi dei campi personalizzati sono indicati nei TASK di quali sottoprogetti è possibile trovarli.

TIPOLOGIA	STATI	CAMPI PERSONALIZZATI
Malfunzionamento Strumentazione	Failure identification Restoring in progress Restored	All-Sky (<i>All-Sky system</i>) Strumento (<i>Altri Dispositivi</i>) Weather (<i>Weather Station</i>) CCD (<i>CCD</i>) Telescopio (<i>CCD, Cupola, Liquefattore, Settori, Strumentazione, Swircam</i>) Cupola (<i>Cupola</i>) Liquefattore (<i>Liquefattore</i>) Settore (<i>Settori</i>) Strumentazione (<i>Strumentazione</i>) Swircam (<i>Swircam</i>)
Acquisto Strumentazione	Acquisizione dati Ricezione offerte Ordine emesso Ispezione Collaudo Ordine liquidato	All-Sky (<i>All-Sky system</i>) Strumento (<i>Altri Dispositivi</i>) Weather (<i>Weather Station</i>) CCD (<i>CCD</i>) Telescopio (<i>CCD, Cupola, Liquefattore, Settori, Strumentazione, Swircam</i>) Cupola (<i>Cupola</i>) Liquefattore (<i>Liquefattore</i>) Settore (<i>Settori</i>) Strumentazione (<i>Strumentazione</i>) Swircam (<i>Swircam</i>) Categoria Strumentazione (<i>Sviluppo</i>)
Controllo	Scheduled Warning Updated	All-Sky (<i>All-Sky system</i>) Strumento (<i>Altri Dispositivi</i>) Weather (<i>Weather Station</i>) CCD (<i>CCD</i>) Telescopio (<i>CCD, Cupola, Liquefattore, Settori, Strumentazione, Swircam</i>) Cupola (<i>Cupola</i>) Liquefattore (<i>Liquefattore</i>) Settore (<i>Settori</i>) Strumentazione (<i>Strumentazione</i>) Swircam (<i>Swircam</i>)
Update	Scheduled Update completed Update unsuccessful	All-Sky (<i>All-Sky system</i>) Telescopio (<i>Strumentazione</i>) Strumentazione (<i>Strumentazione</i>)
Progettazione	Concept study Design Progettazione completata	Categoria Strumentazione (<i>Sviluppo</i>)

Costruzione	Procurement Integration Costruzione completata	Categoria Strumentazione (Sviluppo)
Installazione	Installation Commissioning Installazione Completata	Categoria Strumentazione (Sviluppo)

Tramite campi personalizzati è possibile scegliere tra le voci:

- ALL-SKY (*All-Sky system*):
 - Camera;
 - Software.

- STRUMENTO (*Altri dispositivi*):
 - Support Equipment;
 - Handling Tools;
 - Seeing Monitor.

- WEATHER (*Weather Station*):
 - Station;
 - Software.

- CCD (*CCD*):
 - Camera;
 - Ruota Portafiltri;
 - Cooling.

- TELESCOPIO (*CCD, Cupola, Liquefattore, Settori, Strumentazione, Swircam*):

- AZT-24;
- Schmidt;
- CUPOLA (*Cupola*):
 - AZT-24;
 - Schmidt;
- LIQUEFATTORE (*Liquefattore*):
 - Compressore N2;
 - Compressore He;
 - Coldhead & Vessel;
 - Recupero vapori;
 - Elettronica di controllo;
- SETTORE (*Settori*):
 - Ottica;
 - Meccanica;
 - Movimentazione.
- STRUMENTAZIONE (*Strumentazione*):
 - Spettrografo Ottico;
 - Instrument Selector;
 - Illuminatori Flat Field;
 - Lampade Spettrali;
 - Elettronica di Controllo;
 - Software di Controllo;
 - Sensoristica;
 - Fuoceggiatore;
 - Camera di Guida;
- SWIRCAM (*Swircam*):

- Elettronica di Controllo;
- Camera Infrarossa.
- CATEGORIA STRUMENTAZIONE (Sviluppo):
 - Nuova Strumentazione.

Come detto in precedenza, l'esigenza esplicita dello *Stakeholder M. Dolci* di poter scegliere a quale Telescopio fare riferimento in TASK, è stata soddisfatta utilizzando il campo personalizzato TELESCOPIO, e di seguito vediamo un'immagine che rappresenta un TASK con il campo personalizzato e le scelte effettuabili.

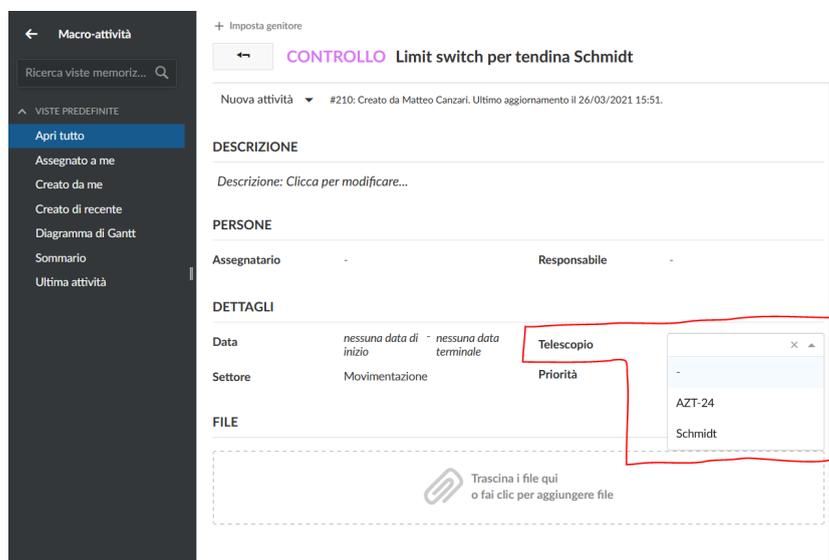


FIGURA 21. Esempio di un TASK in STRUMENTAZIONE SCIENTIFICA, scelta telescopio tramite campo personalizzato.

Invece per un TASK qualsiasi di STRUMENTAZIONE SCIENTIFICA, è simile all'implementazione di LOGISTICA mostrata in Figura 19.

Sprint 4

Questo *Sprint* è stato svolto dal 1 Marzo 2021 al 14 Marzo 2021.

Durante lo *Sprint Planning* è stata analizzata la tabella fornita da **Fiore De Luise**, simile a quella illustrata nello *Sprint 2*, e cercata una possibile soluzione.

Analizzate le informazioni, la soluzione da noi proposta definiva sottoprogetti diversi che riguardassero le osservazioni, la segnalazione di problemi e l'analisi dei dati.

L'esigenza di poter caricare file allegati ad un TASK, è stata soddisfatta senza alcuna implementazione in quanto *OpenProject* permette di caricare file allegati all'interno dei TASK aperti.

Invece, non è stato possibile soddisfare pienamente la richiesta di creare automaticamente un TASK di backup per ogni TASK di osservazione creato e collegarlo a quest'ultimo, perché le funzionalità della versione free di OpenProject non permettevano di farlo automaticamente, e c'era bisogno di sviluppare qualche plug-in, ma sarà il creatore del TASK di osservazione a creare manualmente anche quello di backup e collegarlo all'osservazione corrispondente, funzione permessa nella versione free.

Per quanto riguarda la possibilità di assegnare più progetti ad un TASK di osservazione, è stato risolto attraverso più campi personalizzati con il nome del progetto e una scelta Sì/No. Non si è potuto utilizzare una scelta multipla differente, in quanto la versione Free di *OpenProject* non metteva a disposizione tale feature, altrimenti disponibile con l'Enterprise Edition.

Lo *Sprint* è stato caratterizzato da una prima attività di sviluppo per la correzione della sezione *LOGISTICA* attraverso le informazioni fornite da **G. Valentini** durante lo *Sprint Review 3* e da una seconda parte di sviluppo della soluzione della sezione *ATTIVITÀ SCIENTIFICA*.

Le attività svolte sono state:

- Creazione stories Sprint 4 (0.25) - Creazione delle attività su *Trello* per lo *Sprint 4*;
- Correzione LOGISTICA (0.5) - Correzione parte relativa allo stakeholder G. Valentini.

- Individuazione sottoprogetti, macro-attività, campi personalizzati e stati (1.75) - Individuare, dalle informazioni fornite da Fiore, i vari sottoprogetti da implementare, le macro-attività, i campi personalizzati e gli stati da includere.
- Definizione sottoprogetti (0.5) - Implementazione della struttura dei sottoprogetti.
- Definizione Macro-attività (1) - Implementazione delle Macro-attività in ogni sottoprogetto.
- Definizione Stati (1) - Implementazione degli stati per ogni macro attività.
- Definizione Campi personalizzati (1) - Implementazione dei campi personalizzati nei TASK di interesse.
- Scaletta Demo (0.5) - Definizione della scaletta per la demo da presentare agli stakeholders.

Non è stata riscontrata alcuna criticità durante lo *Sprint 4*.

Sprint Review 4

Al termine del quarto *Sprint*, per la precisione Venerdì 12 Marzo 2021, si è tenuto il quarto *Sprint Review*.

Durante questo evento è stata mostrata una demo di ciò che è stato implementato durante lo *Sprint*, ovvero la sezione *ATTIVITÀ SCIENTIFICA* interessata in particolare a **F. De Luise**, in modo da fargli comprendere come utilizzare le varie funzionalità e illustrargli la soluzione proposta, e mostrate a **G. Valentini** le correzioni fatte sulla parte *LOGISTICA*.

In questo caso, non è stata effettuata nessuna intervista con il *LeanUX Canvas v2*, in quanto è l'ultimo *Sprint*.

Invece, sono stati raccolti i feedback di **M. Dolci** sulla sezione *STRUMENTAZIONE SCIENTIFICA*, che non ha evidenziato alcuna problematica.

Attività Scientifica

Lo sviluppo in questo *Sprint* si è basato per la maggior parte sull'implementazione del progetto *ATTIVITÀ SCIENTIFICA*.

A partire dalla soluzione elaborata e dalle informazioni fornite da **F. De Luise**, sono stati definiti in primis i nuovi sottoprogetti sulla base delle infrastrutture proposte nella soluzione. L'immagine mostra i sottoprogetti di *ATTIVITÀ SCIENTIFICA* definiti:

- » Attività Scientifica
 - » Analisi Dati
 - » Osservazioni
 - » Segnalazioni

FIGURA 22. *Sottoprogetti definiti per il macro-progetto ATTIVITA' SCIENTIFICA.*

Per ognuno di questi, sono state definite le corrispondenti tipologie di TASK e rispettivi stati. Inoltre, ove richiesto, sono stati aggiunti dei campi personalizzati all'interno dei TASK. Nella tabella sono elencate tutte le tipologie di TASK, gli stati e i campi personalizzati associati:

N.B.: *Nelle parentesi dei campi personalizzati sono indicati nei TASK di quali sottoprogetti è possibile trovarli.*

TIPOLOGIA	STATI	CAMPI PERSONALIZZATI
Analisi Dati	Preriduction Completed Calibration completed Validation OK Data Available for Download	Calibration (<i>Analisi Dati</i>) Education (<i>Analisi Dati</i>) Grawita (<i>Analisi Dati</i>) Space Debris (<i>Analisi Dati</i>) Service (<i>Analisi Dati</i>) Test (<i>Analisi Dati</i>) Transients (<i>Analisi Dati</i>) Other (<i>Analisi Dati</i>)
Pubblicazione	In progress Completed	Calibration (<i>Analisi Dati</i>) Education (<i>Analisi Dati</i>) Grawita (<i>Analisi Dati</i>) Space Debris (<i>Analisi Dati</i>) Service (<i>Analisi Dati</i>) Test (<i>Analisi Dati</i>) Transients (<i>Analisi Dati</i>) Other (<i>Analisi Dati</i>)
Osservazione	Preparazione Esecuzione Chiusa	Calibration (<i>Osservazioni</i>) Education (<i>Osservazioni</i>) Grawita (<i>Osservazioni</i>) Space Debris (<i>Osservazioni</i>) Service (<i>Osservazioni</i>) Test (<i>Osservazioni</i>) Transients (<i>Osservazioni</i>) Other (<i>Osservazioni</i>)
Backup Dati	Scheduled Backup Completato Backup Failed	
Segnalazione problema	Segnalato Preso in visione In lavorazione Risolto	Apparato (<i>Segnalazioni</i>)
Verifica Strumentazione	Scheduled In progress Verifica completata	

Tramite campi personalizzati è possibile scegliere tra le voci:

- APPARATO (*Segnalazioni*):
 - Telescopio;
 - CCD;
 - Cupola;
 - Software.

- CALIBRATION (*Analisi Dati, Osservazioni*):
 - Sì;
 - No;

- EDUCATION (*Analisi Dati, Osservazioni*):
 - Sì;
 - No;

- GRAWITA (*Analisi Dati, Osservazioni*):
 - Sì;
 - No;

- SPACE DEBRIS (*Analisi Dati, Osservazioni*):
 - Sì;
 - No;

- SERVICE (*Analisi Dati, Osservazioni*):
 - Sì;
 - No;

- TEST (*Analisi Dati, Osservazioni*):
 - Sì;
 - No;

- OTHER (*Analisi Dati, Osservazioni*): campo per testo libero.

Come detto in precedenza, la richiesta di poter allegare file all'interno di un TASK è stata soddisfatta senza alcuna implementazione ma semplicemente esiste già l'apposita sezione per gli allegati. Invece, l'esigenza di poter selezionare più progetti scientifici a cui un TASK fa riferimento, è stata soddisfatta utilizzando più campi personalizzati con i nomi di tali progetti da cui scegliere Sì / No. Nell'immagine seguente è mostrato un esempio di TASK in cui sono stati caricati degli allegati e in cui sono presenti i campi personalizzati dei progetti scientifici.

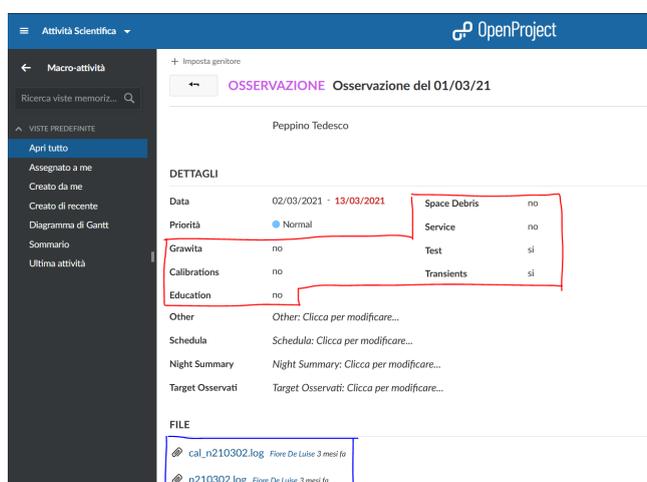


FIGURA 23. Esempio di un TASK in ATTIVITA' SCIENTIFICA, in **rosso** scelta progetti tramite campo personalizzato, in **blu** sezione allegati.

Non è stato possibile creare automaticamente i TASK di *BACKUP DATI* relazionati con i rispettivi TASK di *OSSERVAZIONE*, ma lo è in modo manuale. Nell'immagine è mostrato un esempio di come si presenta un TASK di *OSSERVAZIONE* nell'elenco quando ne viene creato uno relionato di *BACKUP DATI*.

ID ↑	PROGETTO	↳ SOGGETTO	TIPO	STATO	PRIORITÀ	ASSEGNATARIO	*
124	Osservazioni	↳ Osservazione del 01/03/21	OSSERVAZIONE	Esecuzione	Normal	-	...
125	Osservazioni	↳ Backup notte 01-02/03/21	BACKUP DATI	Scheduled	Normal	FL Fiore De Luise	

FIGURA 24. Esempio elenco di TASK relazionati (da notare l'indentazione dei SOGGETTO).

Invece per un TASK qualsiasi di *ATTIVITÀ SCIENTIFICA*, è simile all'implementazione di *LOGISTICA* mostrata in *Figura 19*.

CAPITOLO 6

USABILITA' e ACCESSIBILITA'

In questo capitolo si descriverà il questionario per valutare l'USABILITÀ e l'ACCESSIBILITÀ sul sistema di gestione, compilato da parte degli utenti, e si analizzeranno i risultati ottenuti.

Questionario

Il questionario è stato fornito agli utenti dopo un mese dalla consegna del software, in modo che potessero acquisire un pò di esperienza e dare una valutazione migliore.

Questi utenti sono i tre Stakeholders, più **ing. Matteo Canzari**, ovvero il *Product Owner / Scrum Master*, che essendo anch'egli un dipendente dell'INAF, è un utilizzatore del sistema.

La compilazione è completamente anonima.

E' composto da 17 domande a risposta chiusa, di cui 8 sull'*USABILITÀ*, 7 sull'*ACCESSIBILITÀ* e 2 generiche:

- DOMANDE GENERICHE:

1. Stia attualmente utilizzando il software?

- a. Si;
- b. No.

2. Se dovessi dare un voto da 1 a 10 al software, quale voto daresti?

- Voto da 1 (Pessimo) a 10 (Ottimo).

- **DOMANDE USABILITÀ:**

1. Quanto è facile creare e gestire i task?

- Voto da 1 (Molto difficile) a 10 (Molto facile).

2. Quanto è facile navigare nei menù?

- Voto da 1 (Molto difficile) a 10 (Molto facile).

3. Quanto è facile personalizzare i task?

- Voto da 1 (Molto difficile) a 10 (Molto facile).

4. Quanto è facile filtrare i task per una ricerca?

- Voto da 1 (Molto difficile) a 10 (Molto facile).

5. Quanto è facile monitorare nel complesso i vari task?

- Voto da 1 (Molto difficile) a 10 (Molto facile).

6. Quanto è facile creare e gestire una pagina Wiki?

- Voto da 1 (Molto difficile) a 10 (Molto facile).

7. Quanto è facile comunicare con gli altri utenti?

- Voto da 1 (Molto difficile) a 10 (Molto facile).

8. Quanto è facile allegare dei file?

- Voto da 1 (Molto difficile) a 10 (Molto facile).

- **DOMANDE ACCESSIBILITÀ:**

1. Quanto trovi gradevole lo stile grafico di OpenProject?
 - Voto da 1 (Molto sgradevole) a 10 (Molto gradevole).

2. Quanto trovi gradevole la combinazione di colori scelti per le tipologie di task (es. Malfunzionamento, Controllo, Osservazione ecc...)?
 - Voto da 1 (Molto sgradevole) a 10 (Molto gradevole).

3. Quanto trovi gradevole la combinazione di colori scelti per gli stati dei task?
 - Voto da 1 (Molto sgradevole) a 10 (Molto gradevole).

4. Quanto trovi gradevole la struttura data ai progetti e sottoprogetti?
 - Voto da 1 (Molto sgradevole) a 10 (Molto gradevole).

5. Quanto trovi gradevole i nomi dati ai campi personalizzati all'interno dei task (es. Impianto, Telescopio ecc...)?
 - Voto da 1 (Molto sgradevole) a 10 (Molto gradevole).

6. Quanto trovi gradevole il layout delle pagine "Panoramica"?
 - Voto da 1 (Molto sgradevole) a 10 (Molto gradevole).

7. Quanto trovi gradevole il layout interno dei task?
 - Voto da 1 (Molto sgradevole) a 10 (Molto gradevole).

Analisi risultati

Dopo la compilazione del questionario, vengono analizzati i risultati al fine di comprendere il grado di *USABILITÀ* e *ACCESSIBILITÀ*.

In tale analisi, nel calcolo dei punteggi, non vengono prese in considerazione le risposte alle *DOMANDE GENERICHE*, ma vengono utilizzate per alcune considerazioni.

Dai risultati ci si aspetta una buona *USABILITÀ*, dato l'utilizzo di *LeanUX* e *AGILE* che prevedono continui feedback e interazioni con l'utente, permettendo di modellare il software in base alle richieste dell'utente per cui si presuppone tale risultato.

Diversamente per l'*ACCESSIBILITÀ* in quanto, avendo usato *OpenProject* per lo sviluppo, la personalizzazione della UI è limitata alla scelta di alcuni colori, per cui si è cercato di soddisfare l'utente per quanto possibile.

Detto ciò, di seguito vengono analizzati singolarmente i risultati di ogni domanda, facendo, infine, una valutazione complessiva sia per l'*USABILITÀ* che per l'*ACCESSIBILITÀ*.

Risultati USABILITÀ

- RISULTATI domanda 1:

Quanto è facile creare e gestire i task?
4 risposte

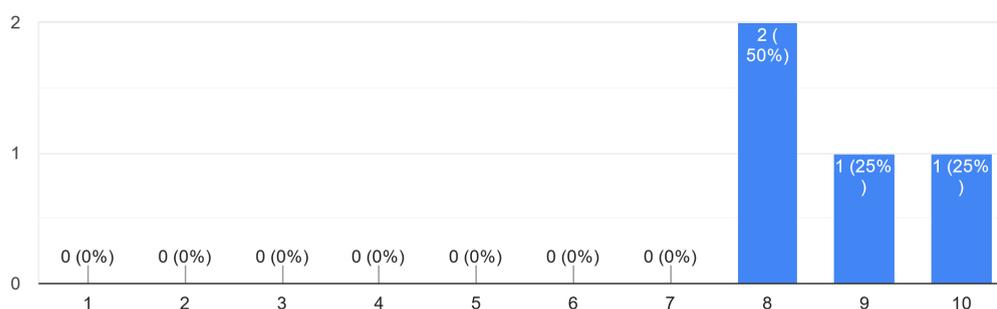


FIGURA 25. Grafico risultati domanda 1 *USABILITÀ*.

Sicuramente è una delle domande più importanti, in quanto i TASK sono il cuore del progetto. Vediamo come i punteggi sono alti, per cui risulta essere facile poter lavorare con i TASK.

- RISULTATI domanda 2:

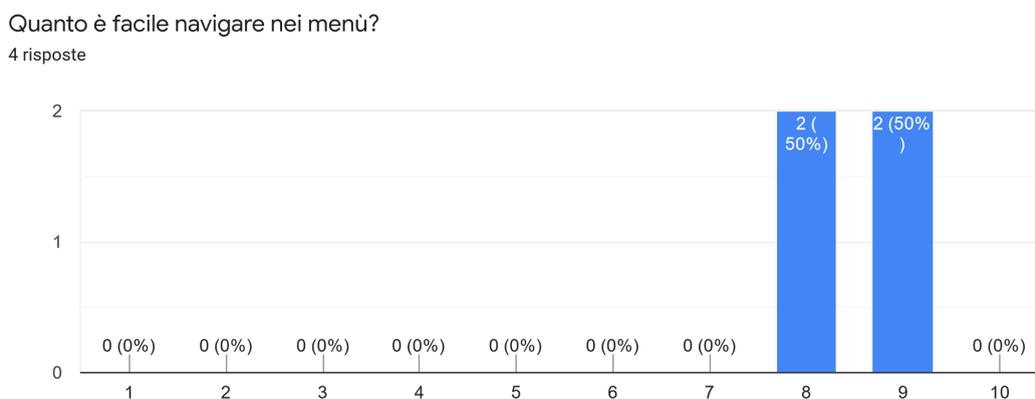


FIGURA 26. Grafico risultati domanda 2 USABILITÀ.

Dai risultati si nota come sia facile navigare attraverso i tanti menù di *OpenProject*, anche se nessuno ha dato il punteggio massimo. Un risultato non scontato, in quanto la personalizzazione del layout dei menù si limita a scegliere le voci da visualizzare o a creare un elenco di progetti / sottoprogetti.

- RISULTATI domanda 3:

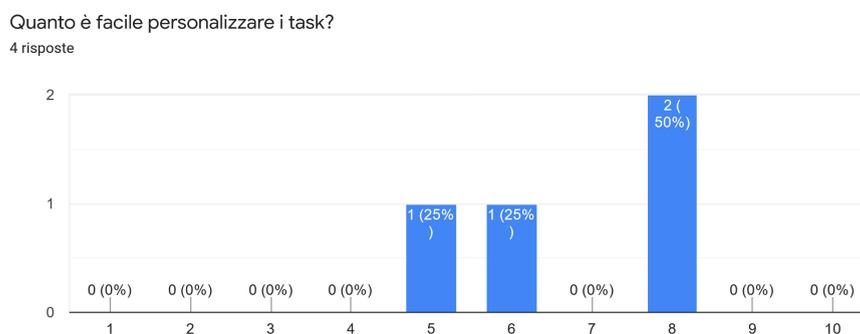


FIGURA 27. Grafico risultati domanda 3 USABILITÀ.

I risultati mostrano un pò di incertezza per alcuni utenti, vediamo due risposte con un punteggio mediocre e due abbastanza buono. Probabilmente questo risultato è dovuto al fatto che tale operazione richiede tanti passaggi, per cui non risulta essere immediata.

- **RISULTATI domanda 4:**

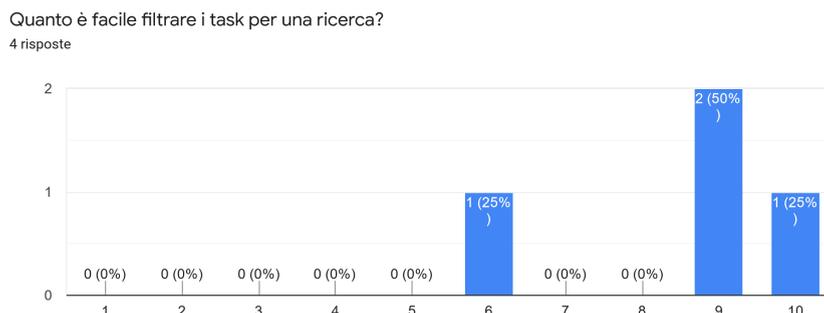


FIGURA 28. Grafico risultati domanda 4 USABILITÀ.

A questa domanda c'è solo una risposta incerta con un voto mediocre, dovuto probabilmente alla poca immediatezza o alla poca esperienza, mentre per gli altri tre la ricerca dei TASK attraverso filtri sembra essere di facile utilizzo.

- **RISULTATI domanda 5:**

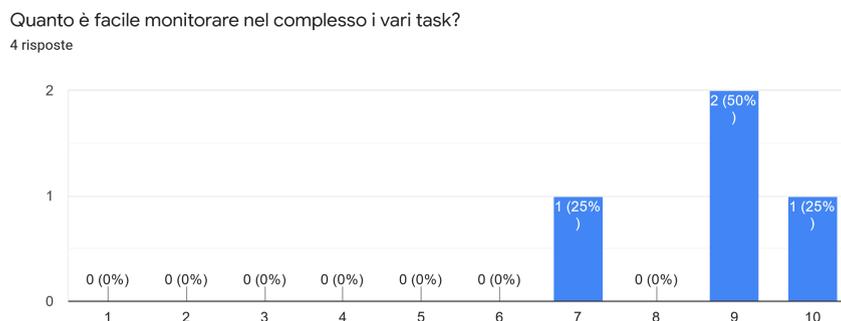


FIGURA 29. Grafico risultati domanda 5 USABILITÀ.

I risultati a questa domanda dimostrano una buona facilità nel monitorare i task attraverso le pagine “Panoramica”, leggermente incerto un utente ma comunque ha fornito un voto più che sufficiente.

- RISULTATI domanda 6:

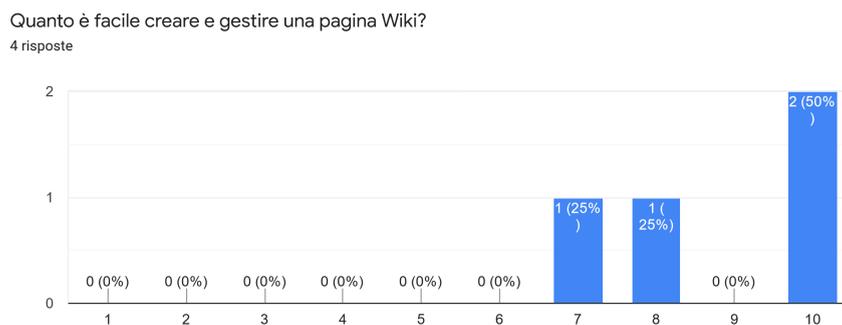


FIGURA 30. Grafico risultati domanda 6 USABILITA'.

La Wiki per due utenti è molto facile utilizzarla, fornendo il voto massimo, per altri due è facile con voti altamente sufficienti.

- RISULTATI domanda 7:

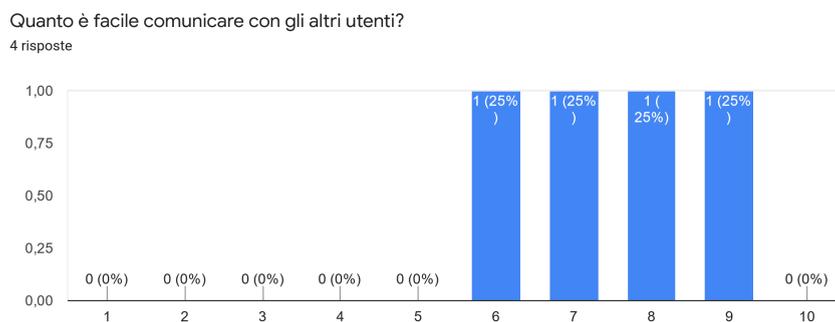


FIGURA 31. Grafico risultati domanda 7 USABILITÀ.

A questa domanda, gli utenti hanno tutti un parere diverso, in quanto forniscono tutti un voto diverso ma comunque sufficienti. Evidentemente il metodo di comunicazione tramite commenti all'interno dei TASK non è immediato, soprattutto nel raggiungere la conversazione, in quanto bisogna trovare il TASK d'interesse ed aprirlo.

- RISULTATI domanda 8:

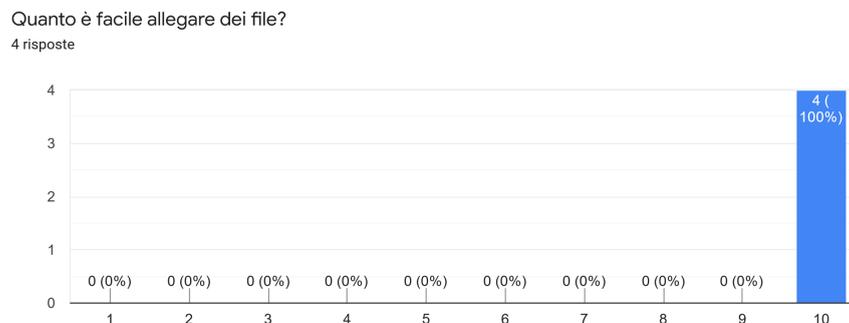


FIGURA 32. Grafico risultati domanda 8 *USABILITÀ*.

Qui, tutti d'accordo. Allegare file in un TASK è molto facile.

Risultati ACCESSIBILITÀ

- RISULTATI domanda 1:

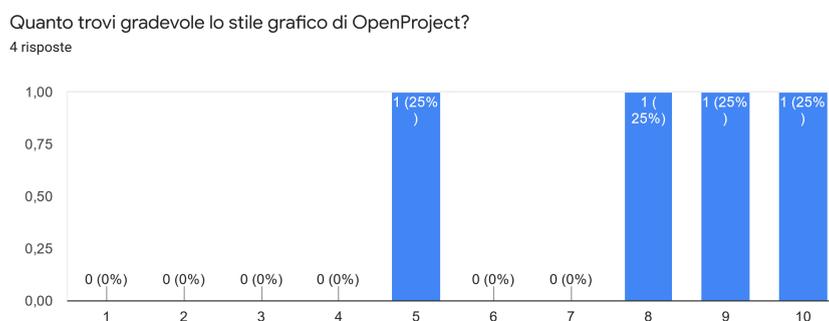


FIGURA 33. Grafico risultati domanda 1 *ACCESSIBILITÀ*.

Lo stile grafico di *OpenProject* è considerato gradevole da tre utenti mentre meno gradevole da un utente con una votazione di 5.

- RISULTATI domanda 2:

Quanto trovi gradevole la combinazione di colori scelti per le tipologie di task (es. Malfunzionamento, Controllo, Osservazione ecc...)?
4 risposte

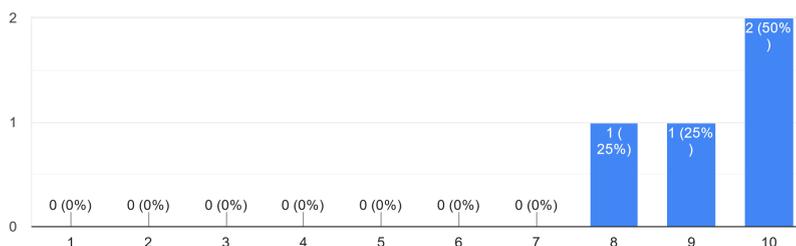


FIGURA 34. Grafico risultati domanda 2 ACCESSIBILITÀ.

Sulla combinazione di colori scelta, delle tipologie, sono tutti molto soddisfatti. E' una risposta abbastanza attesa, in quanto anche la scelta dei colori veniva mostrata agli *Stakeholders* durante gli *Sprint*.

- RISULTATI domanda 3:

Quanto trovi gradevole la combinazione di colori scelti per gli stati dei task?
4 risposte

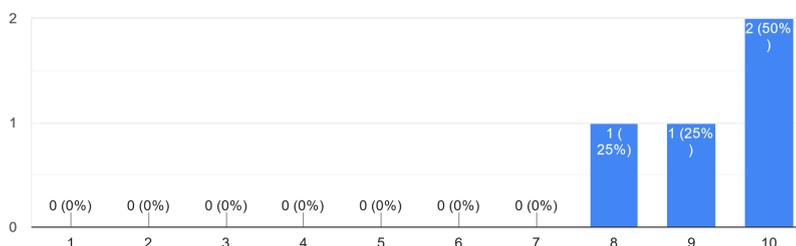


FIGURA 35. Grafico risultati domanda 3 ACCESSIBILITÀ.

Sulla combinazione di colori scelta, degli stati di un task, sono tutti molto soddisfatti. E' una risposta abbastanza attesa, in quanto anche la scelta dei colori veniva mostrata agli Stakeholders durante gli *Sprint*.

- RISULTATI domanda 4:

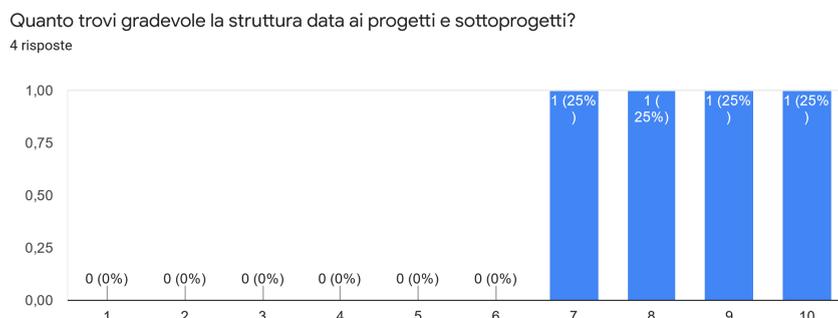


FIGURA 36. Grafico risultati domanda 4 ACCESSIBILITÀ.

Sulla struttura data ai progetti e sottoprogetti sono tutti soddisfatti, anche se ognuno ha un parere differente, ma i voti dati sono più che sufficienti.

- RISULTATI domanda 5:

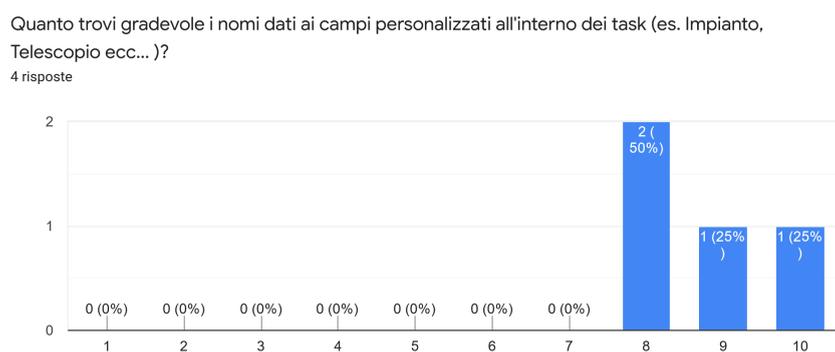


FIGURA 37. Grafico risultati domanda 5 ACCESSIBILITÀ.

Trovano tutti molto gradevole la scelta dei nomi dei campi personalizzati. E' una risposta abbastanza attesa, in quanto anche la scelta dei campi personalizzati veniva mostrata agli *Stakeholders* durante gli *Sprint*.

- RISULTATI domanda 6:

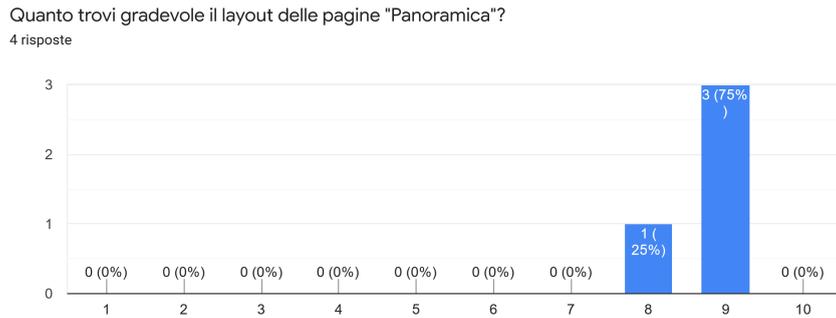


FIGURA 38. Grafico risultati domanda 6 ACCESSIBILITÀ.

Trovano tutti molto gradevole il layout delle pagine “*Panoramica*”.

- RISULTATI domanda 7:

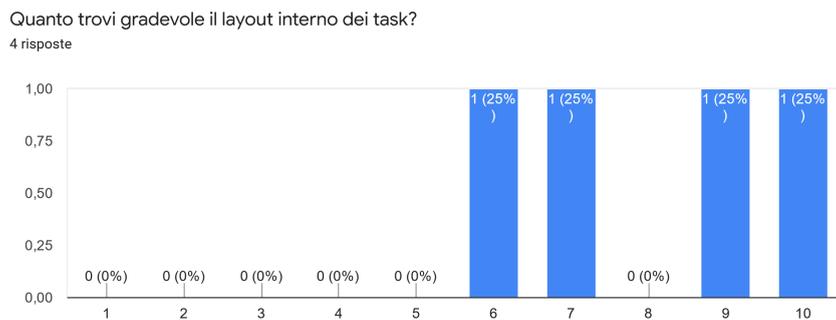


FIGURA 39. Grafico risultati domanda 7 ACCESSIBILITÀ.

A questa domanda il grado di incertezza è maggiore, due risposte presentano voti eccellenti e altre due intorno alla sufficienza. Sicuramente il layout dei task è pieno di di features quindi può sembrare confusionario. E' fornito di base da *OpenProject* quindi non ci sono state modifiche e personalizzazioni in fase di sviluppo.

Risultati finali

Di seguito vengono riportate delle tabelle, una per l'USABILITÀ e una per l'ACCESSIBILITÀ, che mostrano il punteggio medio e la deviazione standard per ogni domanda.

Dopo ogni tabella vengono fatte alcune considerazioni sui punteggi ottenuti e le eventuali soluzioni future per migliorare il punteggio.

Per i punteggi finali viene fatta la media di tutti i punteggi medi e riportati in un'altra tabella finale, insieme alla deviazione standard.

Infine, vengono fatte alcune considerazioni riguardo i risultati delle *DOMANDE GENERICHE*.

- **USABILITÀ:**

<u>DOMANDA</u>	<u>PUNTEGGIO MEDIO</u>	<u>DEVIAZIONE STANDARD</u>
Quanto è facile creare e gestire i task?	8.75	0.83
Quanto è facile navigare nei menù?	8.5	0.5
Quanto è facile personalizzare i task?	6.75	1.3
Quanto è facile filtrare i task per una ricerca?	8.5	1.5

Quanto è facile monitorare nel complesso i task?	8.75	1.09
Quanto è facile creare e gestire una pagina Wiki?	8.75	1.3
Quanto è facile comunicare con gli altri utenti?	7.5	1.12
Quanto è facile allegare dei file?	10	0

Dai punteggi medi ottenuti, si nota che ad ogni domanda il punteggio medio di usabilità non è mai sotto la sufficienza, questo dimostra come sia abbastanza facile utilizzare le features del sistema di gestione.

Inoltre anche la deviazione standard, ad ogni domanda, è al massimo qualche decimo sopra l'unità, quindi il grado di incertezza degli utenti è molto basso se consideriamo che i voti sono unitari (1, 2, ..., 10).

Detto ciò, visti i punteggi, si potrebbero fare alcuni miglioramenti o modifiche dove il voto non raggiunge l'8. I miglioramenti si potrebbero apportare attraverso lo sviluppo di plugin, che permetterebbero di aggiungere feature al sistema rispetto a quelle predefinite di Openproject.

Si potrebbe pensare di creare una sezione apposita, esterna ai TASK, dove poter comunicare con gli altri utenti del sistema, e sicuramente provare a migliorare la semplicità di personalizzazione dei TASK intervistando gli Stakeholders per comprendere nel dettaglio come poter facilitare tale operazione.

Dove il punteggio di usabilità supera l'8, non ci sono miglioramenti da fare, perché il voto non perfetto probabilmente riguarda il "gusto personale", ma è comunque un voto più molto buono.

- **ACCESSIBILITÀ:**

<u>DOMANDA</u>	<u>PUNTEGGIO MEDIO</u>	<u>DEVIAZIONE STANDARD</u>
Quanto trovi gradevole lo stile grafico di OpenProject?	8	1.87
Quanto trovi gradevole la combinazione di colori scelti per le tipologie di task?	9.25	0.83
Quanto trovi gradevole la combinazione di colori scelta per gli stati dei task?	9.25	0.83
Quanto trovi gradevole la struttura data ai progetti e sottoprogetti?	8.5	1.12
Quanto trovi gradevole i nomi dati ai campi personalizzati all'interno dei task?	8.75	0.83
Quanto trovi gradevole il layout delle pagine "Panoramica"?	8.75	0.43
Quanto trovi gradevole il layout interno dei task?	8	1.58

I punteggi medi di accessibilità sono tutti superiori all'8, dimostrando un alto grado di gradevolezza, da parte degli utenti, nei layout e nei dettagli grafici utilizzati.

Unica nota, sono alcuni valori di deviazione standard leggermente alti, come sulla gradevolezza del layout interno dei task o lo stile grafico di OpenProject.

Miglioramenti non andrebbero fatti, visti i punteggi medi elevati, ma si potrebbe provare a cambiare qualcosa (cambiando alcuni colori o sbloccando nuove personalizzazioni con l'Enterprise Edition di OpenProject) cercando di soddisfare anche gli utenti incerti, senza però ridurre il grado di accessibilità degli altri utenti.

- **RISULTATI FINALI:**

	<u>PUNTEGGIO FINALE</u>	<u>DEVIAZIONE STANDARD MEDIA</u>
<u>USABILITÀ</u>	8.44	0.995
<u>ACCESSIBILITÀ</u>	8.64	1.07

Considerando i voti da 1 a 10, il punteggio finale ottenuto, esprime un grado di *USABILITÀ* del sistema di gestione, molto alto, così come per l'*ACCESSIBILITÀ*.

Il primo è un valore atteso, che afferma ancor di più le potenzialità di *AGILE* e *LeanUX*, in quanto tali metodologie garantiscono quasi il pieno soddisfacimento degli *Stakeholders*, grazie ai continui feedback e interazioni con essi.

Inoltre, tale punteggio esprime la buona riuscita nello sviluppo e progettazione da parte dell'intero Team, e quindi un lavoro ben fatto.

Invece, la "*DEVIAZIONE STANDARD MEDIA*" ottenuta, considerando valori unitari nei punteggi (es. 1, 2, 3 ecc...), esprime un grado di incertezza molto basso dell'*USABILITÀ*, da parte degli utenti, in quanto il valore è inferiore all'unità.

Il punteggio medio ottenuto per l'ACCESSIBILITÀ è risultato non scontato, in quanto, come detto in precedenza, la personalizzazione del layout è molto limitata con *OpenProject*.

Invece, la “*DEVIAZIONE STANDARD MEDIA*” ottenuta, considerando valori unitari nei punteggi (es. 1, 2, 3 ecc...), esprime, anche se più alto di quello dell'*USABILITÀ*, un grado di incertezza comunque molto basso da parte degli utenti, in quanto il valore supera di poco l'unità.

Considerazioni risultati domande generiche

I risultati sulle *DOMANDE GENERICHE* del questionario non hanno contribuito ai punteggi dell'*USABILITÀ* e dell'*ACCESSIBILITÀ*, ma servono semplicemente a fare alcune considerazioni su di esse.

- RISULTATI domanda 1:

Se dovessi dare un voto da 1 a 10 al software, quale voto daresti?

4 risposte

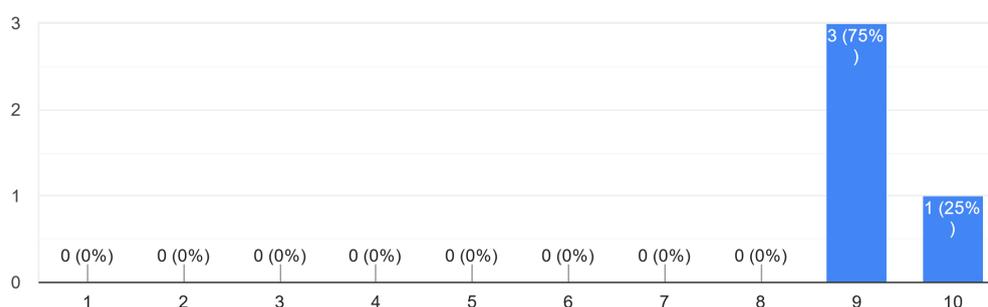


FIGURA 40. Grafico risultati domanda generica 1.

Questa domanda è stata posta per capire come gli utenti valutano il software complessivamente, considerando tutti gli aspetti, non solo quelli di *ACCESSIBILITÀ* e *USABILITÀ*.

DOMANDA	VOTO COMPLESSIVO	DEVIAZIONE STANDARD
Se dovessi dare un voto da 1 a 10 al software, quale voto daresti?	9.25	0.43

Il voto complessivo e la deviazione standard esprimono un soddisfacimento degli Stakeholders quasi perfetto e un grado di incertezza quasi nullo. Il voto complessivo supera sia quello di *ACCESSIBILITÀ* (8.64) che quello di *USABILITÀ* (8.44), a dimostrazione del fatto che, aldilà dei fattori legati a questi ultimi, anche le altre funzionalità fornite dal sistema di gestione implementato sono alquanto soddisfacenti.

- RISULTATI domanda 2:

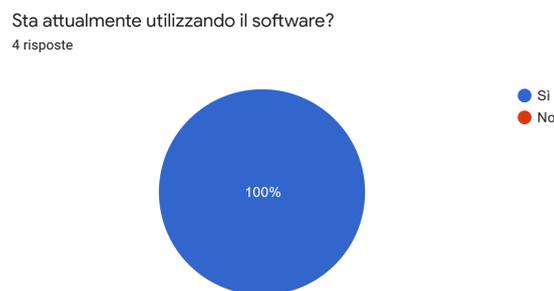


FIGURA 41. Grafico risultati domanda generica 2.

Domanda posta per capire se gli utenti utilizzano il software. I risultati del grafico a torta sono chiari. Questo è molto importante perché fa comprendere come l'*AGILE* e *LeanUX* hanno permesso di realizzare, in così poco tempo e con un Team composto da così poche persone, un software utilizzabile da subito, e che quindi la scelta di queste metodologie e tecnologie (*OpenProject*) è stata vincente, visto anche il grado di soddisfazione degli utenti.

DOMANDA	SI	NO
Sta attualmente utilizzando il software?	4	0

CONCLUSIONI

Giunti alla conclusione di questo lavoro di tesi, possiamo trarre delle conclusioni e fare alcune considerazioni sui temi affrontati nei vari capitoli.

In primis si è descritto, in modo generico, di cosa tratta questo progetto e le motivazioni che hanno portato al suo sviluppo. Con la scoperta delle onde gravitazionali, la stazione osservativa di Campo Imperatore è tornata in auge, per cui non si poteva pensare di non avere un sistema che gestisse ogni aspetto, considerando i tempi moderni e i motivi discussi nel Capitolo 2.

In secondo luogo sono state illustrate nel dettaglio le metodologie, Agile, in particolare il framework Scrum, e LeanUX, oltre alla tecnologia Openproject, motivandone la scelta.

Considerando il poco tempo per fornire un software funzionante e il poco personale per lo sviluppo del sistema, Scrum con l'integrazione di LeanUX, ha fornito una metodologia "vincente" che ha permesso di raggiungere l'obiettivo preposto.

Scrum ha permesso di strutturare lo sviluppo e la progettazione attraverso gli Eventi, i Ruoli e gli Artefatti, ma mancava di una maggiore interazione con gli utenti finali, necessaria per sviluppare in poco tempo un software che soddisfacesse la maggior parte delle esigenze degli stakeholders, lacuna compensata dall'integrazione di LeanUX.

Infatti, LeanUX attraverso il suo "credo", ha permesso di migliorare l'interazione tra Team di Sviluppo e utenti finali, o di basare lo sviluppo del software sulle decisioni degli stakeholder "iniettate" durante l'implementazione, e di fornire Sprint dopo Sprint una parte di sistema funzionante ed utilizzabile (MVP). Inoltre ha fornito, grazie al LeanUX Canvas v2, una guida per le interviste semi-strutturate.

Viceversa, con LeanUX si perderebbe un pò il focus sul prodotto finale, problema che viene risolto dalla natura stessa di Scrum, risultando così metodologie complementari. Anche la scelta di OpenProject, grazie alla sua facilità di approccio allo sviluppo, e delle features che facevano al caso delle esigenze, ha contribuito a raggiungere velocemente il traguardo.

Poi si è descritto come le metodologie e tecnologie scelte sono state utilizzate durante la fase di sviluppo del sistema di gestione, spiegando nel dettaglio ogni Sprint, quindi cosa si è implementato, come si sono svolte le interviste, le interazioni con gli utenti, i vari eventi ecc... .

Non si è evidenziata nessuna grande criticità durante lo sviluppo, se non un piccolo ritardo nella consegna del file Excel (Figura 11), tranquillamente superata. Questo grazie alle tante interazioni con gli stakeholders, che hanno fornito continui feedback su ciò che si implementava, alla loro chiarezza nell'espone le proprie esigenze, e all'utilizzo delle giuste metodologie.

Infine, sono state fatte valutazione sull'accessibilità e l'usabilità, attraverso un questionario fornito agli utenti del software, dopo un mese dal termine dello sviluppo. Ci si aspettava un grado di accessibilità e usabilità elevato, in quanto con le metodologie usate, attraverso le continue interazioni con gli utenti, ha permesso di progettare un sistema sulla base delle loro esigenze e che lo sviluppo sarebbe andato avanti fino a pieno soddisfacimento, per cui ottenere scarse valutazioni sarebbe stato un controsenso. Infatti, i risultati attesi sono stati rispettati.

Considerando che i punteggi da 1 a 10, i risultati ottenuti sono un 8.44 per l'usabilità e un 8.64 per l'accessibilità, valutazioni che superano ampiamente la sufficienza.

Inoltre, dal questionario è emerso che tutti gli utenti stanno attualmente utilizzando il software, cosa assolutamente non scontata considerando l'inesperienza ed il tempo a disposizione. Questo non fa altro che dimostrare l'importanza che hanno avuto le metodologie usate, e che la scelta di integrare LeanUX con Scrum è stata vincente.

Sviluppi futuri

Al termine delle ore di tirocinio e dei 4 Sprint, è stato fornito agli stakeholders un software che soddisfasse tutti i requisiti, ma sicuramente migliorabile.

Il risultato ottenuto, dal questionario di usabilità e accessibilità, è molto buono e che lascia spazio ad alcuni miglioramenti e non a stravolgimenti, per provare ad aumentare il punteggio.

Basti considerare che tali risultati sono stati raggiunti utilizzando la versione free di OpenProject, quindi la versione Enterprise potrebbe aggiungere features e dettagli che vadano a migliorare l'esperienza dell'utente, ad esempio aggiungere la scelta multipla in alcuni campi personalizzati, migliorare il layout interno dei task con l'aggiunta di nuovi contenitori o sbloccare nuovi temi per OpenProject.

Inoltre, durante la progettazione e l'implementazione, non è stata approfondita la possibilità di sviluppare dei Plugin, possibilità che andrebbe presa in considerazione in futuro per poter aggiungere features e migliorare la personalizzazione del sistema.

APPENDICE

Osservatorio d'Abruzzo - INAF^{[12][13]}

L'**Osservatorio Astronomico d'Abruzzo** con direttore **Enzo Broncato** è una delle strutture di ricerca dell'Istituto Nazionale di Astrofisica (INAF), il principale ente di ricerca italiano per lo studio dell'Universo. Come da statuto, l'INAF ha “il compito di svolgere, promuovere e valorizzare la ricerca scientifica e tecnologica nei campi dell'astronomia e dell'astrofisica e di diffonderne e divulgarne i relativi risultati, di promuovere e favorire il trasferimento tecnologico verso l'industria, perseguendo obiettivi di eccellenza a livello internazionale”.



FIGURA 42. *Osservatorio Astronomico di Teramo (Abruzzo).*

L'**Osservatorio astronomico d'Abruzzo**, sito in località Collurania a Teramo, è uno degli osservatori astronomici pubblici italiani afferente all'INAF. Di recente

costituzione, riunisce lo storico Osservatorio Astronomico di Teramo "*Vincenzo Cerulli*" con la Stazione Osservativa di Campo Imperatore.

L'attività scientifica svolta presso l'Osservatorio Astronomico d'Abruzzo, riguarda diversi campi dell'astrofisica stellare teorica e osservativa:

- Evoluzione Stellare
- Progenitori di Supernovae
- Astrofisica Nucleare e Nucleositesi
- Popolazioni Multiple negli Ammassi Globulari Galattici
- Sintesi di Popolazioni Stellari

Tra i vari progetti scientifici, l'Istituto è particolarmente coinvolto nel supporto alla missione GAIA e nella gestione del progetto AMICA.

L'Osservatorio propone anche alcune attività per il pubblico, come visite guidate, iniziative didattiche, eventi e attività di divulgazione.

Osservatorio astronomico di Campo Imperatore^[14], oggi denominato **Stazione osservativa di Campo Imperatore**, è un centro di osservazione astronomica fondato nel 1965 come parte dell'Osservatorio astronomico di Roma a 2200 m s.l.m. in provincia dell'Aquila, in Abruzzo, nel cuore del massiccio del Gran Sasso d'Italia, e usato ancora oggi per attività di ricerca scientifica e divulgative. Dal 2017 è gestito dall'**Osservatorio astronomico d'Abruzzo**. Lo strumento principale dell'osservatorio è l'AZT-24, un riflettore Ritchey-Chrétien del diametro di 108 cm costruito per l'Osservatorio di Pulkovo ma trasferito in Italia nel 1995 e posto sul Gran Sasso nel 1997. Il telescopio è equipaggiato con la camera ad infrarossi (1,1-2,5 μm) SWIRCAM, fornita dall'osservatorio di Teramo, e coinvolta nel progetto SWIRT sulla ricerca di supernove extragalattiche, in cui collabora anche la Russia. Questo telescopio è l'unico telescopio a infrarossi presente in Italia.



FIGURA 43. *Stazione osservativa di Campo Imperatore.*

TASK

Un **TASK** in OpenProject è sostanzialmente una “Attività” eseguita o da eseguire, di cui si vuol tener traccia attraverso stati e varie informazioni.

OpenProject permette di implementare un Task Manager dove appunto è possibile gestire, personalizzare e condividere con altri utenti i vari task.

Vediamo nel dettaglio come si presenta un task in OpenProject e quali sono le parti più significative. Nell’immagine seguente è rappresentato un task d’esempio:

The screenshot shows a task page in OpenProject. At the top, there's a breadcrumb trail: '+ Imposta genitore' (1), 'Malfunzionamento' (2), and 'Cappello tubo di scarico fumi'. Below this is a dropdown menu for 'Organizzazione intervento' (3) and a status bar indicating '#162: Creato da Gaetano Valentini. Ultimo aggiornamento il 09/04/2021 13:19.'. The main content area is divided into sections: 'DESCRIZIONE' (4) with text about a gas outlet cap being removed and replaced; 'PERSONE' (8) showing 'Assegnatario' and 'Responsabile' as Gaetano Valentini; 'DETTAGLI' (5) with fields for 'Data' (17/03/2021), 'Impianto' (Impianto Termico II), 'Avanzamento (%)' (50%), and 'Priorità' (Normal); and 'FILE' (6) with an attached file 'cappello_divetto.jpeg'. On the right, the 'ATTIVITÀ' (7) section shows a log of updates: creation on 17/03/2021, a modification on 19/03/2021, and another modification on 09/04/2021.

FIGURA 44. Esempio di un task completo.

1. **TIPOLOGIA**: indica di che tipo di task si tratta. E’ possibile personalizzare le tipologie e quindi sceglierle durante la creazione del task.
2. **NOME**: indica il nome / titolo che viene dato al task durante la sua creazione. Solitamente è una brevissima descrizione dell’attività o di ciò che definisce il task.

3. **STATO**: indica lo stato in cui si trova il task, permettendo di comprendere al lettore a che punto o in che condizioni è l'attività. E' possibile definire in modo personalizzato più stati per lo stesso task, definire un ordine, un colore e un nome.
4. **DESCRIZIONE**: contiene una descrizione, solitamente dettagliata, del task. E' possibile inserire un numero illimitato di caratteri e dargli una formattazione.
5. **PERSONE** e **DETTAGLI**: sono contenitori predefiniti di informazioni varie sul task, solitamente campi predefiniti o personalizzati. Nella versione Enterprise di OpenProject è possibile definire altri contenitori di questo tipo.
6. **ALLEGATI**: è il contenitore degli allegati, in quanto nei task è possibile allegare dei file.
7. **LOG** e **COMMENTI**: sezione che funge da log delle modifiche sul task e da "simil chat" tra gli utenti, infatti è sia possibile visualizzare chi ha apportato la modifica e cosa ha modificato, e sia è possibile aggiungere e visualizzare commenti relativi al task in questione.
8. **ASSEGNATARIO** e **RESPONSABILE**: sono due campi predefiniti che si presentano alla creazione del task e permettono di selezionare, appunto, l'assegnatario del task e chi ne è responsabile.
9. **DATA**: è un campo predefinito che si presenta alla creazione del task. Permetti di selezionare la data di inizio e di fine di un task.
10. **AVANZAMENTO**: è un campo predefinito rappresentato da una barra con cui è possibile indicare la percentuale di avanzamento del task.

11. **CAMPO PERSONALIZZATO**: è un campo personalizzabile dallo sviluppatore da inserire nei contenitori stile “Persone” o “Dettagli”. E’ possibile dargli un nome e selezionare varie tipologie di valore, dalla scelta in un elenco alla scrittura libera ecc... .

12. **PRIORITÀ**: è un campo predefinito che si presenta alla creazione del task ed indica la priorità. Permette di scegliere il livello di priorità da dedicare al task.

INSTALLAZIONE OPENPROJECT e AMBIENTE DI SVILUPPO

Installazione OpenProject

L'installazione di OpenProject è stata eseguita su un server dell'Osservatorio d'Abruzzo, dove risiede il sistema di gestione dell'osservatorio.

L'installazione è stata effettuata attraverso Docker su un sistema Linux / Ubuntu.

Docker è un progetto open-source che automatizza il processo di deployment di applicazioni all'interno di contenitori software, fornendo un'astrazione aggiuntiva grazie alla virtualizzazione a livello di sistema operativo di Linux.

Per poter usufruire di Docker c'è bisogno che sulla macchina sia installato il Docker Engine, quindi se non è presente, bisogna procedere prima all'installazione di quest'ultimo.

Per una guida all'installazione di Docker Engine:

- <https://docs.docker.com/engine/install/ubuntu/>

Fatto ciò si può procedere all'installazione di OpenProject.

Installazione OpenProject

Primo, bisogna clonare la repository openproject-deploy:

```
git clone https://github.com/opf/openproject-deploy --depth=1 --branch=stable/11
openproject
```

Quindi, vai nella cartella *compose*:

```
openproject/compose
```

Assicurati di utilizzare l'ultima versione delle immagini Docker:

```
docker-compose pull
```

Lancia il container:

```
docker-compose up -d
```

Dopo un po', OpenProject dovrebbe essere attivo e funzionante su <http://localhost:8080>. Il nome utente e la password predefiniti sono login: `admin` e password: `admin`.

Puoi interrompere lo stack Compose eseguendo:

```
docker-compose down
```

Per ulteriori informazioni e per metodi diversi di installazione, visitare il sito al seguente link:

<https://www.openproject.org/docs/installation-and-operations/installation/docker/>

Ambiente di Sviluppo

Il server è stato protetto da VPN quindi per poter accedere da remoto bisognava connettersi attraverso una VPN all'indirizzo <http://193.204.1.146>. E' stato utilizzato il software SoftEther VPN per poter connettersi tramite VPN.

Quindi il sistema di gestione è stato sviluppato su server ma con accesso da remoto.

WEBGRAFIA

[1] Definizioni e informazioni sulla metodologia AGILE.

- <https://www.redhat.com/it/devops/what-is-agile-methodology>

[2] Definizioni e informazioni sul framework Scrum.

- <https://medium.com/geekandjob-blog/scrum-cos%C3%A8-e-come-funziona-metodologia-agile-7c8988feec01>

[3] Confronto tra V-Model e Agile.

- <https://it.sawakinome.com/articles/programming/difference-between-agile-and-v-methodologies-model.html>

[4] Definizioni e informazioni sulla metodologia LeanUX.

- <https://www.interaction-design.org/literature/article/a-simple-introduction-to-lean-ux>

[5] I 3 pilastri fondamentali della LeanUX.

- <https://www.uxblog.it/i-3-pilastri-fondamentali-della-lean-ux/>

[6] LeanUX Canvas v2.

- <https://jeffgothelf.com/blog/leanuxcanvas-v2/>

[7] Immagine metodo a cascata.

- <https://www.ionos.it/digitalguide/siti-web/programmazione-del-sito-web/modello-a-cascata/>

[8] Immagine V-Model

- <https://it.wikipedia.org/wiki/V-Model>

[9] Definizioni e informazioni sul software OpenProject.

- <https://docs.openproject.org/getting-started/openproject-introduction/>

[10] Github repository OpenProject.

- <https://github.com/opf/openproject>

[11] Sito Trello.

- <https://trello.com/>

[12] Informazioni Osservatorio Astronomico D'Abruzzo - INAF.

- <http://www.ia-teramo.inaf.it/>

[13] Informazioni Osservatorio Astronomico D'Abruzzo - INAF.

- <https://edu.inaf.it/edu-inaf/osservatorio-astronomico-dabruzzo/>

[14] Informazioni Stazione Osservativa di Campo Imperatore.

- https://it.wikipedia.org/wiki/Osservatorio_astronomico_di_Campo_Imperatore

[15] Articolo Medium su integrazione di LeanUX con Scrum.

- <https://medium.com/serious-scrum/10-ways-lean-ux-boosts-value-in-scrum-ebb6a2ea1d13>

RINGRAZIAMENTI

Ringrazio il professore **Luca Spalazzi** per la sua solita grande disponibilità e umanità, e per avermi offerto questo interessantissimo lavoro.

Ringrazio tutto lo staff dell'INAF, dal direttore **Enzo Broncato**, a **Gaetano Valentini**, **Mauro Dolci** e **Fiore De Luise** per la loro disponibilità, serietà e simpatia, inoltre vanno i miei complimenti per il grande lavoro che svolgono.

Ringrazio, in modo speciale, l'ing. **Matteo Canzari**, tutor, guida e mentore di questo percorso, professionale e competente. Il trattamento ricevuto va ben oltre il tirocinio, per cui spero di essere stato all'altezza dei suoi consigli e insegnamenti, e che tutto ciò possa essere stato ripagato con il mio impegno. GRAZIE INFINITE.

Ringrazio i miei genitori e mia sorella. Senza i loro sacrifici, la loro vicinanza e caparbieta, nulla sarebbe stato realizzabile. Spero di aver soddisfatto le loro aspettative, e che questo risultato sia solo l'inizio di un percorso che possa portare a vedermi realizzato.

Questo titolo non è solo mio, ma è anche vostro! Vi ringrazio con tutto il cuore, spero che siate fieri di me.

Ringrazio, i miei amici che mi sono stati vicini in questo lungo percorso, il vostro supporto è stato fondamentale! L'amicizia si vede soprattutto nei momenti di difficoltà, e voi ci siete stati! GRAZIE.

Infine, questa volta ringrazio soprattutto me stesso, per la costanza, la testardaggine e la determinazione. Un percorso lunghissimo fatto di gioie e di momenti difficili, a cui ho cercato sempre di reagire e questo titolo è il risultato di tutti i miei sforzi. Ho capito tante cose, ho perso e guadagnato persone, ho vissuto periodi bui, ma la risposta è stata sempre forte e coraggiosa, ed il merito è mio e di chi ha creduto in me.

Ma questo è solo l'inizio, ora viene il bello....

