



Laurea Magistrale in informatica-Università di Salerno
Corso di Gestione dei Progetti Software- Prof.ssa F. Ferrucci



Scrum Product Backlog UniRentHub

Riferimento	C11_SPB
Versione	1.0
Data	22/01/2024
Destinatario	Prof.ssa Filomena Ferrucci
Presentato da	Rocco Iuliano Simone Della Porta
Approvato da	



Laurea Magistrale in informatica-Università di Salerno
Corso di Gestione dei Progetti Software- Prof.ssa F. Ferrucci

Revision History

Data	Versione	Descrizione	Autori
18/12/2023	0.1	Prima stesura	R.Iuliano S.Della Porta
19/12/2023	0.2	Aggiunta story points	R. Iuliano S. Della Porta
22/01/2024	1.0	Revisione finale	R. Iuliano S. Della Porta



Sommario

Revision History	2
1. Introduction.....	4
2. Approach	4
2.1. Team figures	4
2.2. Sprint	4
2.3. Meeting	4
3. Product Backlog.....	5



1. Introduction

In questo documento saranno riportati i dettagli su come verrà applicato il framework SCRUM il cui impiego è stato pianificato per la fase di implementazione della piattaforma UniRentHub. Saranno indicati i requisiti funzionali che verranno implementati, oltre alle informazioni pertinenti agli sprint.

2. Approach

Siccome si è deciso di utilizzare il framework SCRUM solo durante la fase di implementazione della piattaforma UniRentHub e che la durata di tale fase è di circa un mese, è stato necessario rivisitare alcuni concetti di SCRUM per adattarli alle necessità del progetto, in particolare:

- La durata degli sprint;
- Le figure del team;
- I meeting da svolgere.

2.1. Team figures

Nel contesto del progetto si è deciso di mantenere solo le seguenti figure:

- **Team**, composto dai team member i cui nomi sono riportati nei precedenti documenti. Essi si occuperanno dello sviluppo della piattaforma;
- **Scrum master**, interpretato dal Project Manager. Sotto questa veste il Project Manager non si occuperà di tutte le responsabilità legate a tale ruolo ma solo ed esclusivamente della gestione del lavoro, dei meeting e della supervisione.
- **Product owner**, interpretato sempre dal PM.

2.2. Sprint

Come riportato in precedenza la durata della fase di implementazione è di circa un mese; quindi, è stato necessario rivisitare la durata degli sprint, ovvero, passando dalle 2 settimane ad 1 settimana. Per ogni sprint è previsto un meeting iniziale nel quale verranno selezionate le funzionalità da implementare in quello sprint.

2.3. Meeting

Dato che per lo sviluppo si utilizzerà il framework SCRUM, non è più previsto lo svolgimento dei meeting come in precedenza, ma bensì verranno effettuati **due daily scrum meeting** per sprint di una durata massima di 15 minuti. Si è deciso di non effettuare i daily scrum meeting con cadenza giornaliera per non sovraccaricare i team member.



3. Product Backlog

Siccome i requisiti funzionali sono stati definiti utilizzando la notazione IEEE nella fase di *requirements analysis*, si è deciso di utilizzarli al posto delle user stories previste da SCRUM. Ad ogni requisito da implementare è associato un numero di **story point** definiti in un meeting iniziale che ha coinvolto tutto il team. La tecnica utilizzata per la loro definizione è la **t-shirt size** e i valori associati sono:

- S equivale a 3 story points;
- M equivale a 4 story points;
- L equivale a 5 story points;
- XL equivale a 6 story points.

ID	Nome	Descrizione	Story Points
RF_1.0	Recensire una casa	Il sistema deve consentire ad un utente registrato studente di poter recensire la casa in cui risiede	M
RF_1.1	Segnalare locatore	Il sistema deve consentire ad un utente registrato come studente di segnalare il locatore che gli ha affittato l'immobile	S
RF_1.2	Prenotare una visita	Il sistema deve permettere all'utente registrato come studente di prenotare una visita in una data specifica ad un appartamento.	L
RF_1.3	Affitto alloggio	Il sistema deve permettere ad un utente registrato come studente di poter affittare un'immobile	M
RF_1.4 – RF_6.0	Ricerca alloggi	Il sistema deve consentire ad un utente registrato come studente di poter cercare alloggi filtrando la ricerca secondo vari criteri	L
RF_1.5	Pagamento	Il sistema deve permettere allo studente di finalizzare il pagamento dell'affitto	S
RF_1.6	Visualizza alloggio	Il sistema deve permettere ad un utente registrato come studente di poter visualizzare un alloggio	S



RF_1.7	Modificare recensione	L'utente registrato come studente deve avere la possibilità di modificare la recensione, pubblicata in precedenza, in caso di problematiche.	S
RF_1.9	Post studente	Il sistema deve permettere all'utente studente di creare un annuncio per la ricerca di un coinquilino.	S
RF_1.10	Aggiungere un annuncio alla lista preferiti	Il sistema deve permettere ad un utente registrato come studente di poter aggiungere un annuncio selezionato alla propria lista dei preferiti.	M
RF_1.11	Visualizzare lista preferiti	Il sistema deve permettere ad un utente registrato come studente di poter visionare gli annunci inseriti nella propria lista preferiti.	S
RF_1.12	Rimuovere annuncio dalla lista preferiti	Il sistema deve permettere ad un utente registrato come studente di poter rimuovere l'annuncio precedentemente inserito nella lista dei preferiti.	S
RF_2.0	Log- in	Il sistema deve consentire ad un utente registrato di autenticarsi nel sito	S
RF_2.1	Log-out	Il sistema deve permettere all'utente registrato di scollegarsi dal sito	S
RF_2.3	Modificare i propri dati	Il sistema deve permettere all'utente registrato di poter modificare i dati inseriti in fase di registrazione	M
RF_2.4	Verifica account	Il sistema deve permettere ad un utente appena registrato di verificare il proprio account tramite un link di verifica che l'utente riceve nella sua mail.	M
RF_2.5	Informativa cookie	Il sistema deve obbligatoriamente informare gli utenti registrati della presenza di eventuali cookie, dei loro diritti a riguardo e di chiederne il consenso all'installazione (requisito obbligatorio dalla direttiva ePrivacy e dell'UE ai sensi del GDPR)	S
RF_3.1	Creazione annuncio	Il sistema deve consentire ad un utente proprietario di poter creare un annuncio di affitto di un proprio immobile, inserendo tramite un checkbox i vari servizi messi a disposizione	XL



RF_3.2	Modifica annuncio	L'utente proprietario deve avere la possibilità di modificare l'annuncio pubblicato	L
RF_3.4	Data visita proprietario	Il sistema deve permettere ad un utente proprietario di inserire delle date per la visita	M
RF_4.0	Visualizza case da approvare	Il sistema deve permettere all'homechecker di visualizzare le case da approvare	S
RF_4.1	Annullare una casa	Il sistema deve permettere all'homechecker di poter annullare una casa dopo averla visionata nel caso non rispetti le normative vigenti	S
RF_4.2	Approvare una casa	Il sistema deve permettere all'homechecker di poter approvare una casa dopo averla visionata nel caso rispetti le normative vigenti	S
RF_5.0	Visualizzare homechecker	Il sistema deve permettere all'utente admin di visualizzare i vari homechecker	S
RF_5.1	Aggiungere homechecker	Il sistema deve permettere all'utente admin di aggiungere homechecker	S
RF_5.2	Eliminare homechecker	Il sistema deve permettere all'utente admin di eliminare l'homechecker	S
RF_5.4	Bloccare utenti	Il sistema deve permettere all'utente admin di poter bloccare gli utenti segnalati	S
RF_6.1	Registrazione	Il sistema deve permettere ad un utente guest di potersi registrare al sito	S