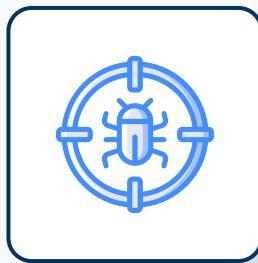




Università degli Studi di Napoli Federico II

Dipartimento di Ingegneria Elettrica e
delle Tecnologie dell'Informazione
Corso di Laurea in Informatica



BUG BOARD 26 Sistema di Gestione Issue Collaborativo

Insegnamento

Ingegneria del Software

Docenti

Prof. Sergio Di Martino

Prof. Luigi Libero Lucio Starace

Candidati

GRUPPO INGSW2526_005

Virginia Antonia
Esposito

Matricola: N86004987

Giuseppe Paolo
Esposito

Matricola: N86005174

Anno Accademico
2025/2026

Indice

1	Documento di Specifica dei Requisiti Software	2
	a. Glossario.....	2
	b. Modellazione dei Casi d'Uso	3
	c. Personas.....	4
	d. Requisiti non-funzionali e di dominio	8
	(i) Requisiti non-funzionali.....	8
	(ii) Vincoli di dominio.....	9
	e. Formalizzazione Caso d'Uso: CreaIssue	10
	(i) Descrizione testuale strutturata	10
	(ii) Prototipazione visuale via Mock-up.....	12

Documento di Specifica dei Requisiti Software

a. Glossario

- **Issue:** Segnalazione creata da un developer per documentare un problema, porre un quesito o richiedere l'inserimento di una funzionalità.
- **Bug:** Tipo di issue che indica un errore o un malfunzionamento del software.
- **Feature:** Tipo di issue che rappresenta una proposta di aggiunta di una nuova funzionalità.
- **Documentation:** Tipo di issue che riguarda errori o mancanze nella documentazione del progetto.
- **Question:** Tipo di issue utilizzata per porre quesiti o chiedere chiarimenti sul progetto.
- **Priorità:** Livello di urgenza associato a una issue.
- **Stato:** Indica la fase corrente della issue nel suo ciclo di esistenza (todo, risolta, ecc.).
- **Admin:** Utente con privilegi elevati, può creare progetti, assegnare issue, gestire utenti e archiviare bug.
- **GenericUser:** Utente non ancora loggato, può solo effettuare il login.
- **LoggedUser:** Utente con privilegi base, visualizza, crea e risolve issues assegnate dall'admin.
- **Team:** Gruppo di utenti loggati responsabili di un progetto, gestito da un admin.
- **Filtro:** Funzione che consente di individuare issue in base a tipologia, stato, priorità o altri parametri.
- **Archivio:** Sezione dove le issue archiviate non sono più visibili nella lista principale; solo l'admin può archiviare.
- **Pull Request:** Permette di notificare la risoluzione locale di una issue, che potrebbe essere accettata o meno a discrezione dell'admin del progetto.

b. Modellazione dei Casi d'Uso

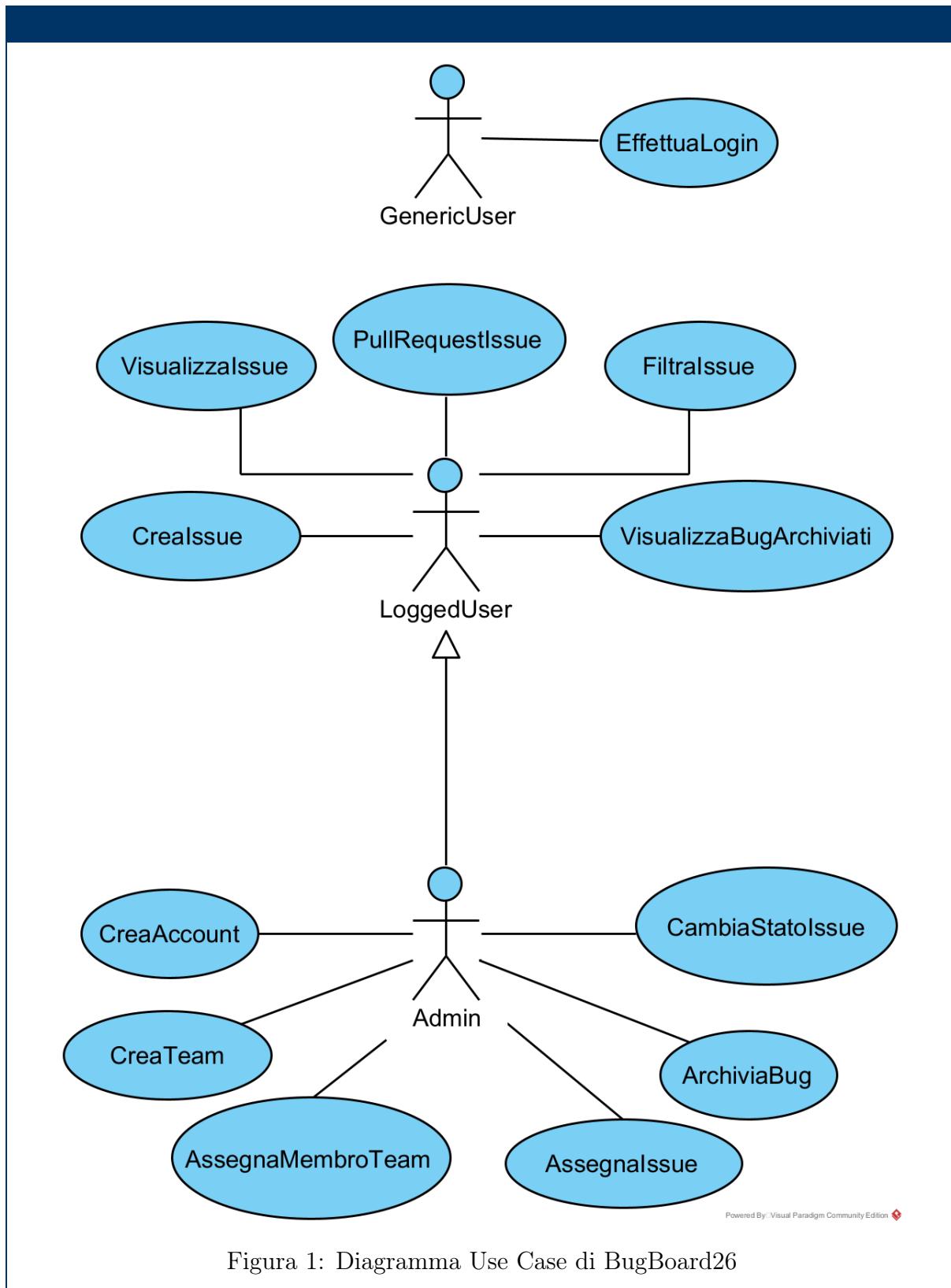


Figura 1: Diagramma Use Case di BugBoard26

c. Personas

1. Sheldon Cooper



Età: 35

Luogo di nascita: Texas, USA

Stato civile: Sposato

Titolo lavorativo: Senior Developer

Tratti caratteriali: Perfezionista, Narcisista, Competitivo, Poco incline al lavoro di squadra

GOALS:

- Dimostrare la propria superiorità nella risoluzione delle issue complesse
- Ottenere il ruolo di amministratore dell'azienda
- Ottenere riconoscimenti per il contributo apportato al team

INTERESSI:

- Fumetti: spende gran parte del suo tempo libero collezionandoli
- Fisica: la sua prima passione prima dell'informatica
- Videogiochi: utilizza i giochi strategici per tenere la mente allenata

BIO: Attualmente Sheldon lavora presso la SoftEngUniNA come Senior Developer e si occupa della analisi e risoluzione di bug complessi, oltre che alla manutenzione del codice.

Le sue eccezionali capacità lo rendono uno sviluppatore molto competente nella risoluzione di bug complessi. Tuttavia la sua natura perfezionista e la difficoltà nel lavorare con gli altri creano spesso tensioni nel team di sviluppo.

Sheldon preferisce interfacce chiare e ben strutturate, che gli consentano di concentrarsi sulla risoluzione dei bug piuttosto che sulla navigazione di interfacce complesse.

2. Armando Caputo



Età: 19

Luogo di nascita: Campobasso, IT

Stato civile: Single

Titolo lavorativo: Junior Developer

Tratti caratteriali: Insicuro, Taciturno, Determinato

GOALS:

- Dimostrare competenza agli amministratori e crescere professionalmente
- Imparare rapidamente tecniche e metodologie per la gestione e risoluzione dei bug
- Completare correttamente le issues assegnate senza dover chiedere aiuto

INTERESSI:

- Computer: Appassionato di hardware, sperimenta nuove configurazioni nel tempo libero
- Serie Tv: Utilizza le pause per distrarsi e staccare la mente dal lavoro
- Musica: Crea playlist personali per concentrarsi mentre lavora

BIO: Armando è al suo primo lavoro come sviluppatore presso SoftEngUniNA. La sua inesperienza e timidezza rendono complicato il lavoro in team, soprattutto quando deve interagire con colleghi più esperti. Per evitare di perdgersi o confondersi tra le molte attività in corso, preferisce strumenti che gli permettano di concentrarsi sulle issue più urgenti e rilevanti, con informazioni complete e facilmente accessibili. Ha bisogno di un'interfaccia semplice, chiara e lineare, con indicazioni passo-passo e notifiche che lo guidino senza sovraccaricarlo di informazioni inutili.

3. Tony Stark



Età: 38

Luogo di nascita: New York, USA

Stato civile: Celibe

Titolo lavorativo: Amministratore di sistema

Tratti caratteriali: Carismatico, impulsivo, geniale, ironico, ambizioso, visionario

GOALS:

- Creare un ambiente di lavoro stimolante e competitivo dove i membri del team possano mostrare le proprie capacità.
- Coordinare il team di developer per garantire la risoluzione tempestiva dei bug e il rispetto delle scadenze.
- Vincere il premio di migliore amministratore dell'azienda ogni mese

INTERESSI:

- Tecnologia e innovazione: Creare e sperimentare nuovi dispositivi e software avanzato
- Comunicazione: Eccelle nella gestione dei rapporti con gli stakeholder, trovando sempre soluzioni creative
- Vita sociale: Party di lusso e feste esclusive

BIO: Tony è il visionario dietro la Stark Industries ed è conosciuto per il suo genio ingegneristico. Come amministratore in SoftEngUniNA porta la sua esperienza di gestione di progetti e team ad alto impatto, assicurandosi che gli sviluppatori abbiano sempre gli strumenti migliori per un lavoro efficace. La sua natura carismatica e ironica lo rendono un leader stimolante, anche se spesso impulsivo. Si irrita facilmente se le informazioni non sono immediatamente visibili e richiedono troppi click per essere raggiunte, odia non poter tenere tutto sotto controllo del suo team e non poter monitorare lo svolgimento delle Issues.

4. Amy Farrah Fowler



Età: 33

Luogo di nascita: California, USA

Stato civile: Sposata

Titolo lavorativo: Amministratrice di sistema

Tratti caratteriali: Introversa, metodica, analitica, paziente

GOALS:

- Mantenere l'efficienza del team e garantire qualità nello sviluppo del codice
- Coordinare i team di sviluppatori con rigidità e chiarezza
- Ridurre il tempo di risoluzione dei bug fornendo strumenti adatti e formazione mirata

INTERESSI:

- Neuroscienze: E' il suo campo scientifico preferito
- Attività sociali: Le piace partecipare a feste a tema
- Lettura scientifica: E' la sua attività preferita nel tempo libero

BIO: Amy, dopo la sua carriera nell'ambito della ricerca nelle Neuroscienze, ha deciso di dedicarsi all'amministrazione di progetti software presso SoftEngUniNA. Coordina i team di sviluppo e supervisiona la gestione delle issue assicurandosi un approccio metodico e garantendo un codice di alta qualità. Tuttavia, il suo perfezionismo e la sua introversione a volte rallentano il flusso decisionale del team e la comunicazione. Vuole gestire personalmente ogni dettaglio e fa fatica a delegare sotto pressione, ha bisogno di strumenti di comunicazione chiari e un sistema di monitoraggio delle Issues sintetico e impeccabile.

d. Requisiti non-funzionali e di dominio

(i) Requisiti non-funzionali

RNF-1 Prestazioni

- **RNF-P1 (Carico Utenti):** Il sistema deve supportare fino a 20 utenti registrati senza degradazione significativa delle prestazioni.

RNF-2 Sicurezza

- **RNF-S1 (Autenticazione):** Il sistema deve impedire l'accesso a utenti non autenticati a tutte le funzionalità e dati, eccetto la funzionalità di login.
- **RNF-S2 (Protezione Dati):** Tutte le password devono essere memorizzate in forma hash (con algoritmo bcrypt) e mai in chiaro nel database.

RNF-3 Affidabilità

- **RNF-R1 (Uptime):** Il back-end deve garantire un uptime maggiore del 95% in un periodo di monitoraggio di 30 giorni.

RNF-4 Usabilità

- **RNF-U1 (Efficienza):** Un utente autenticato deve poter creare una nuova issue in meno di 10 clic/interazioni dall'homepage.
- **RNF-U2 (Apprendibilità):** Un nuovo utente deve poter effettuare login, creare una issue e filtrarle entro 10 minuti dal primo utilizzo.

RNF-5 Manutenibilità

- **RNF-M1 (Disaccoppiamento):** Front-end e back-end devono eseguibili indipendentemente.

RNF-6 Sistema

- **RNF-C1 (Architettura):** Il sistema deve essere composto da almeno due componenti distribuiti: back-end (API REST) e front-end che comunica solo tramite API.
- **RNF-C2 (Tecnologia):** Il codice deve essere sviluppato in linguaggio object-oriented.

(ii) Vincoli di dominio

VD-1 Protezione dei Dati Personalini (GDPR): Il sistema gestisce dati personali, trattati nel rispetto del **Regolamento UE 2016/679** e del **D.Lgs. 196/2003**.

e. Formalizzazione Caso d'Uso: CreaIssue

(i) Descrizione testuale strutturata

CreaIssue			
Use Case #1	CreaIssue		
Goal in Context	L'utente loggato vuole creare una nuova issue per un progetto a cui partecipa in modo che il sistema la salvi correttamente.		
Preconditions	Partecipare ad almeno un team di sviluppo di un progetto.		
Success End Conditions	Il sistema tiene traccia della creazione e salva correttamente la issue.		
Failed End Conditions	La issue non viene salvata e viene mostrato un messaggio di errore.		
Primary Actor	LoggedUser		
Trigger	L'utente clicca sul pulsante "Crea Issue" dalla Homepage.		
Main Scenario	Step	LoggedUser	Sistema
	1	Clicca "Crea Issue"	✗
	2	✗	Mostra form creazione issue (M2)
	3	Compila i campi e clicca "Conferma"	✗
	4	✗	Salva la issue e mostra conferma (M3)
	5	Clicca "Ok"	✗
	6	✗	Torna alla homepage (M1)
Extension #1 (Dati non validi)	Step	LoggedUser	Sistema
	3a	Inserisce dati errati o mancanti	✗
	3b	✗	Mostra errore di validazione (M4)
	3c	Corregge i dati e clicca "Ok"	✗
	3d	✗	Ritorna al passo 3 (Main Scenario)

Extension #2 (Errore di sal- vataggio)	Step	LoggedUser	Sistema
	4a	X	Mostra errore di salva- taggio (M5)
	4b	Clicca “Ok”	X
	4c	X	Ritorna al passo 4 (Main Scenario)
Extension #3 (Annullamen- to)	Step	LoggedUser	Sistema
	2a	Clicca “Annulla”	X
	2b	X	Mostra homepage (M1)
Notes	Campi obbligatori per ogni issue: Titolo, Descrizione, Tipo e Progetto. M1: Homepage M2: Form creazione issue M3: Schermata di avvenuto salvataggio M4: Schermata di errore validazione M5: Schermata di errore di salvataggio.		

(ii) Prototipazione visuale via Mock-up

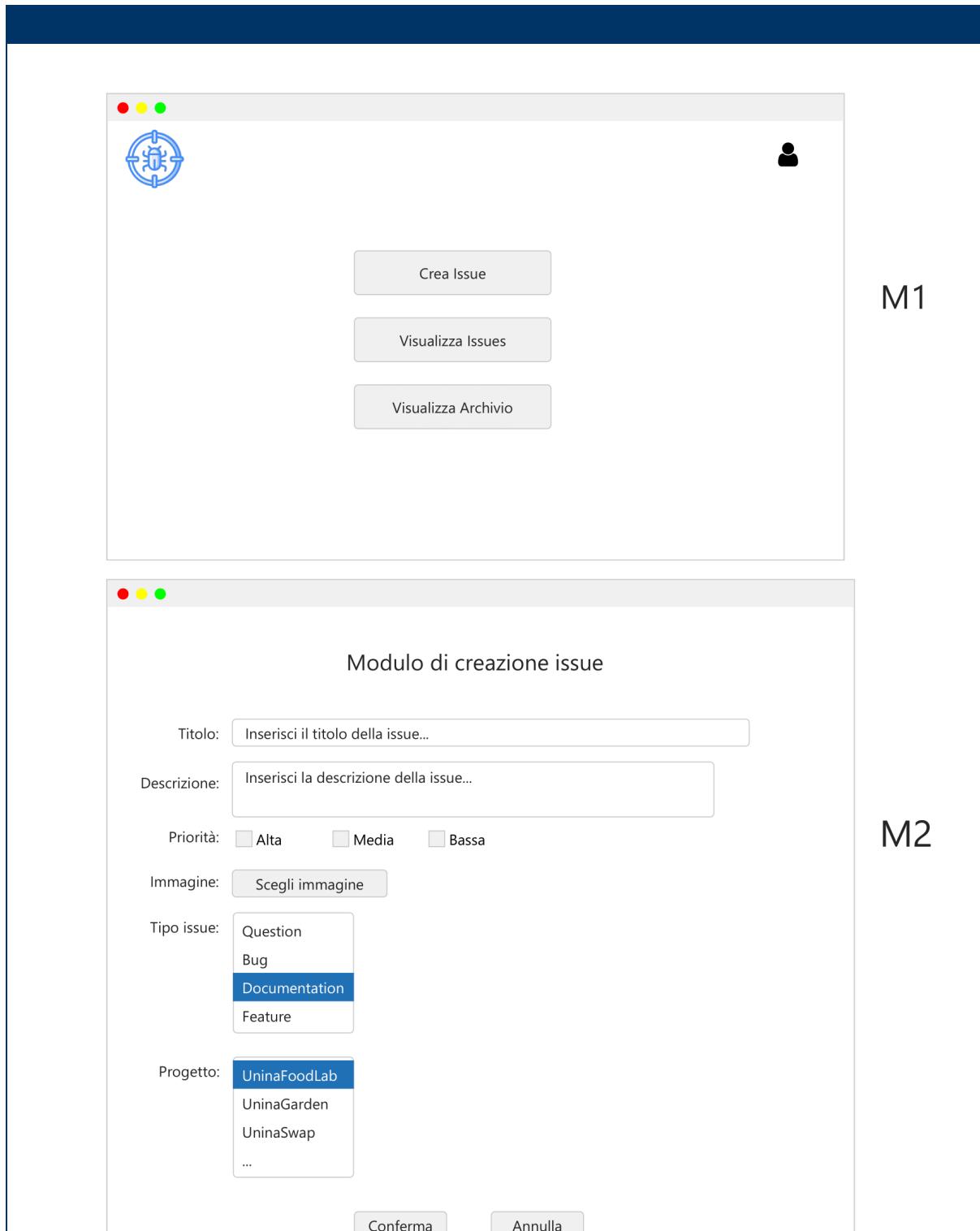


Figura 2: Interfaccia di creazione issue - Homepage e Form creazione

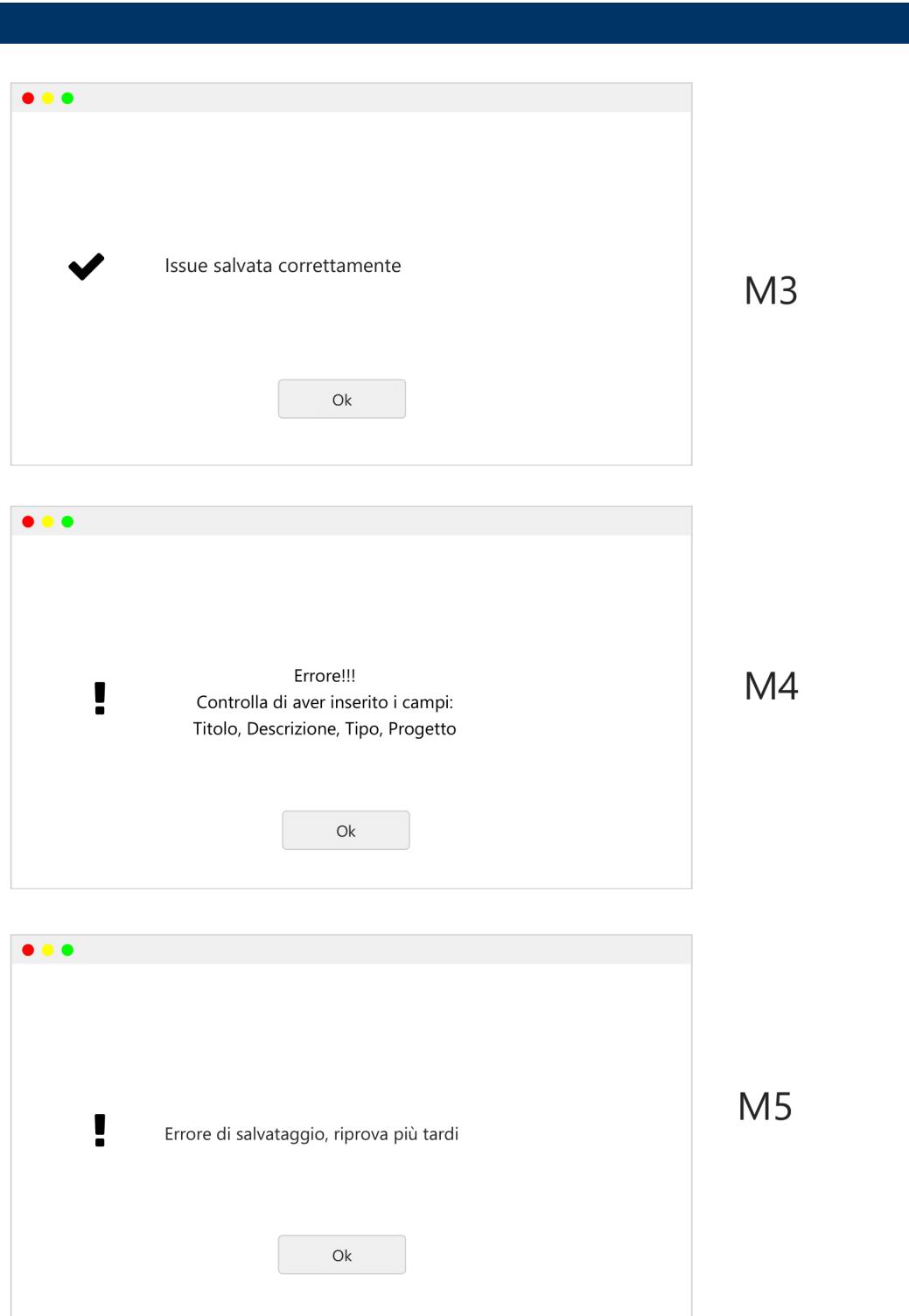


Figura 3: Interfaccia di creazione issue - Notifiche varie