BUGBOARD26

INGEgneria del SOFTWARE

**Specifica dei requisiti del sistema**

|  |  |  |  |
| --- | --- | --- | --- |
| **versione** | **REDAZIONE** | | **DESCRIZIONE** |
| **NOME** | **DATA** |
| V.01 | Federica mele | 10.10.2025 | Prima Stesura |
|  |  |  |  |

**INDICE**

Sommario

[Sommario 2](#_Toc211637384)

[1. Introduzione 4](#_Toc211637385)

[1.1. Scopo del documento 4](#_Toc211637386)

[1.2. Obiettivi 4](#_Toc211637387)

[1.3. Definizioni, Acronimi e Abbreviazioni 4](#_Toc211637388)

[1.3.1. Definizioni 4](#_Toc211637389)

[1.3.2. Sinonimi Adottati 4](#_Toc211637390)

[1.3.3. Abbreviazioni 5](#_Toc211637391)

[1.4. Riferimenti 5](#_Toc211637392)

[2. Descrizione Generale 5](#_Toc211637393)

[2.1. Prospettive del Prodotto 5](#_Toc211637394)

[2.2. Funzionalità del Prodotto 5](#_Toc211637395)

[2.3. Caratteristiche degli Utenti 6](#_Toc211637396)

[2.4. Vincoli 6](#_Toc211637397)

[2.5. Assunzioni e Dipendenze 6](#_Toc211637398)

[3. Requisiti Specifici del Sistema 7](#_Toc211637399)

[3.1. Requisiti Funzionali 7](#_Toc211637400)

[3.1.1 Autenticazione 7](#_Toc211637401)

[3.1.2 Creazione Utenza 8](#_Toc211637402)

[3.1.3 Segnalazione/Creazione Issue 10](#_Toc211637403)

[3.1.4 Filtrare/Ordinare Issue 11](#_Toc211637404)

[3.1.5 Assegnazione Bug 12](#_Toc211637405)

[3.1.6 Visualizzazione Dashboard bug 14](#_Toc211637406)

[3.1.7 Cronologia Bug 16](#_Toc211637407)

[3.2. Requisiti Non Funzionali 17](#_Toc211637408)

[3.2.1 Prestazioni 17](#_Toc211637409)

[3.2.2 Affidabilità 17](#_Toc211637410)

[3.2.3 Sicurezza 17](#_Toc211637411)

[3.2.4 Usabilità 17](#_Toc211637412)

[3.2.5 Scalabilità 17](#_Toc211637413)

[3.2.6 Manutenibilità 18](#_Toc211637414)

[3.2.7 Portabilità 18](#_Toc211637415)

[3.3. Requisiti delle Interfacce Esterne 18](#_Toc211637416)

[3.3.1 Interfaccia Utente 18](#_Toc211637417)

[3.3.2 Interfaccia Hardware 18](#_Toc211637418)

[3.3.3 Interfaccia Software 18](#_Toc211637419)

**INDICE delle figure**

[Figura 1 - UC diagram Autenticazione 8](#_Toc211636944)

[Figura 2 - UC diagram Creazione utente 9](#_Toc211636945)

[Figura 3 - UC diagram Creazione Issue 11](#_Toc211636946)

[Figura 4 - UC diagram: Ricerca issue 12](#_Toc211636947)

[Figura 5 - UC diagram Assegna bug 14](#_Toc211636948)

[Figura 6 - UC diagram Visualizza dashboard bug 16](#_Toc211636949)

[Figura 7 - UC diagram Visualizza cronologia bug 17](#_Toc211636950)

# Introduzione

## Scopo del documento

Il presente documento contiene il dettaglio delle specifiche dei requisiti per la progettazione di una piattaforma di gestione collaborativa di issue. Il software verrà utilizzato da amministratori e sviluppatori semplici, al fine di segnalare problemi relativi a un progetto, monitorarne lo stato, assegnarli a membri del team e tenere traccia delle attività di risoluzione.

## Obiettivi

La piattaforma da realizzare prende il nome di “BugBoard26”. Questa dovrà consentire le seguenti funzionalità:

* L’amministratore già attivo deve poter accedere alla piattaforma e deve poter predisporre la creazione di nuove utenze amministrative o normali che potranno anch’esse accedere all’applicativo .
* Gli utenti devono poter segnalare gli issue.
* Il sistema deve offrire una vista riepilogativa degli issue.
* L’utente amministratore deve poter visualizzare informazioni su bug.
* Il sistema deve visualizzare per ciascun bug la cronologia di modifiche apportate, con le relative informazioni.
* L’utente amministratore deve poter assegnare ciascun bug ad uno specifico utente sviluppatore.

## Definizioni, Acronimi e Abbreviazioni

## Definizioni

* Issue: segnalazione o richiesta registrata nel sistema, che descrive qualcosa da fare, correggere o chiarire. È un contenitore generico di tutti gli altri tipi (question, bug, documentation, feature).
* Question: issue per chiedere spiegazioni o chiarimenti su qualcosa nel progetto.
* Bug: segnala un errore, un malfunzionamento, un difetto o un comportamento anomalo del software.
* Documentation: riguarda errori, mancanze o modifiche necessarie nella documentazione del progetto.
* Feature: rappresenta una proposta di miglioramento o una nuova funzionalità da aggiungere.
* Admin/Amministratore: utente con privilegi elevati, in grado di gestire utenti e avere accesso completo a tutte le funzionalità del sistema.
* Utente normale: membro del team di sviluppo con permessi standard per creare, commentare, e modificare le issue oppure i bug a lui assegnati.
* Stand-alone: oggetto o software capace di funzionare da solo o in maniera indipendente da altri oggetti o software.
* Back-end: parte nascosta di un sito o un’applicazione, quella che gestisce i dati e la logica interna.
* Front-end: parte visibile di un sito o un’applicazione, tutto ciò che l’utente vede e con cui interagisce direttamente.
* API: insieme di regole, metodi e strumenti che permettono a due programmi diversi di comunicare tra loro.
* REST: modo ordinato e standard per far parlare due programmi via Internet usando HTTP.
* CRUD: rappresenta le quattro operazioni basiche e fondamentali che ogni applicazione deve poter fare con i dati.

## Sinonimi Adottati

* Sistema, piattaforma.
* Utenti normali, sviluppatori, membri del team.
* Tabella, archivio.

## Abbreviazioni

* SRS: Software Requirements Specification (questo documento).
* DBMS: Database Management System.
* API: Application Programming Interface.
* REST: Representational State Transfer.
* CRUD: Create, Read, Update, Delete.

## Riferimenti

* Progetto SWE-2025-2026-canale 2\_v1.pdf
* IEEE-SRS-1998.

# Descrizione Generale

## Prospettive del Prodotto

BugBoard26 è un sistema distribuito con front-end e back-end separati. BugBoard26 è un prodotto software indipendente da particolari sistemi Hardware e stand-alone da altri software; necessita della presenza di un DBMS.

## Funzionalità del Prodotto

BugBoard26 deve fornire le funzionalità di seguito riportate:

1. Autenticazione – provvede all'autenticazione dell'utente consentendogli di accedere alla piattaforma per svolgere funzionalità che gli competono.
2. Gestione utenze – consente ad utenti amministratori di creare nuove utenze, su richiesta, specificando il tipo di utente (amministratore/normale).
3. Gestione issue – permette di segnalare, visualizzare, modificare e risolvere issue sulla piattaforma.
4. Gestione bug – permette agli utenti di visualizzare bug con relative informazioni (compresa la cronologia) ed agli amministratori di assegnarli ad utenti specifici.

La seguente tabella mostra i per ogni funzionalità i collegamenti con le altre e gli archivi utilizzati:

|  |  |  |
| --- | --- | --- |
| **Funzionalità** | **Funzionalità collegate** | **Archivi utilizzati** |
| Autenticazione |  | Lista utenze |
| Gestione utenze | Cerca utente  Crea utente | Lista utenze |
| Gestione issue | Cerca issue  Crea issue  Visualizza lista issue | Lista issue  Lista bug |
| Gestione bug | Cerca bug  Visualizza lista bug  Assegna bug | Lista bug  Lista issue  Lista utenze |

## Caratteristiche degli Utenti

Nella seguente tabella si riportano gli Attori del Sistema con le relative attività di competenza, che verranno dettagliate nei singoli casi d’uso descritti nei paragrafi successivi.

|  |  |
| --- | --- |
| **Attori** | **Attività di competenza** |
| Utente Amministratore | Rappresenta il soggetto che ha maggiori privilegi |
| Utente Normale | Rappresenta l’utente che a cui è consentito la visualizzazione degli issue ed in modifica ai bug a lui assegnati |

## Vincoli

L’accesso alla piattaforma BugBoard26 deve essere consentito solo agli utenti autenticati e deve assicurare la riservatezza sui dati degli utenti.

La creazione di utenze è riservata solo agli utenti amministratori.

La visualizzazione della dashboard bug è riservata agli utenti amministratori.

L’assegnazione dei bug è riservata agli utenti amministratori.

I bug possono essere risolti solo dagli utenti a cui sono stati assegnati.

## Assunzioni e Dipendenze

BugBoard26 funziona con Sistema Operativo Windows.

# Requisiti Specifici del Sistema

## Requisiti Funzionali

## Autenticazione

All’avvio di BugBoard26 l’utente deve autenticarsi. Ogni utente dovrà inserire le proprie credenziali: Email e Password, per poter accedere alla piattaforma. Solo se le credenziali risultano valide l’utente potrà utilizzare le funzionalità di BugBoard26.

|  |  |
| --- | --- |
| **Caso d’Uso #1** | **Autenticazione utente** |
| Creato da: | |  |  |  | | --- | --- | --- | |  | Data creazione: |  | |
| Obiettivo: | L’utente vuole accedere al sistema (BugBoard26) |
| Pre-Condizioni: | L’utente deve essere in possesso di credenziali valide per l’accesso al sistema. |
| Post-Condizioni per Successo: | L’utente accede al sistema. |
| Attore coinvolto: | Utente generico. |
| Trigger: | L’utente clicca su “Login”. |
| Scenario principale con successo: | |  |  |  | | --- | --- | --- | | Step n. | Utente | Sistema | | 1 | Apre la piattaforma. |  | | 2 |  | Mostra M0 – Form di  Autenticazione per il  Login. | | 3 | Compila il Form di login con le credenziali e clicca “Login” in M0. |  | | 4 |  | Verifica le credenziali e  mostra l’interfaccia  principale. | |
| Estensione A: Le credenziali non sono valide. | |  |  |  | | --- | --- | --- | | 4.a.1 |  | Mostra M1 – Le  credenziali non sono  valide. | | 4.a.2 | Clicca “ok”. |  | | 4.a.3 |  | Ritorna allo step 2. | |

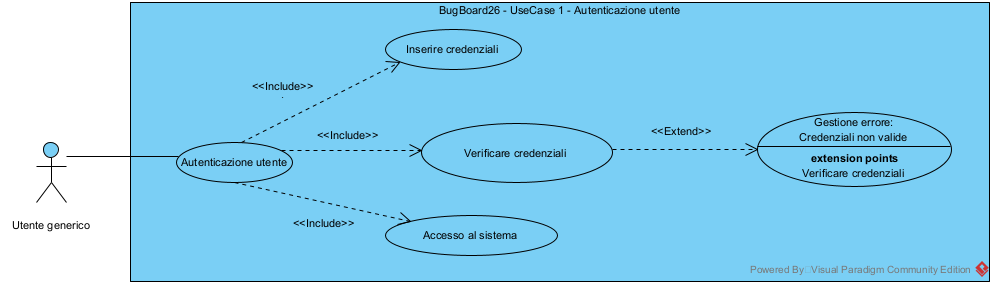


Figura 1 - UC diagram Autenticazione

## Creazione Utenza

L’utente amministratore potrà creare nuove utenze (amministrative o normali) inserendo a sistema le credenziali da assegnare. Il sistema verificherà che l’email sia valida e che non sia già registrata nella piattaforma.

|  |  |
| --- | --- |
| **Caso d’Uso #2** | **Creazione utenza** |
| Creato da: | |  |  |  | | --- | --- | --- | |  | Data creazione: |  | |
| Obiettivo: | L’utente amministratore vuole creare una nuova utenza |
| Pre-Condizioni: | L’utente amministratore deve aver effettuato l’accesso alla piattaforma BugBoard26 |
| Post-Condizioni per Successo: | L’utente è stato creato |
| Attore coinvolto: | Utente amministratore |
| Trigger: | L’utente clicca su “Crea Nuovo Utente” |
| Scenario principale con successo: | |  |  |  | | --- | --- | --- | | Step n. | Amministratore | Sistema | | 1 | Clicca “Crea Nuovo Utente” in M0. |  | | 2 |  | Mostra M1 – Form  di creazione utenza. | | 3 | Inserisce le credenziali necessarie per creare un nuovo utente (email, password e tipologia del nuovo account). Poi clicca “Crea Nuovo Utente” in M1. |  | | 4 |  | Mostra l’interfaccia  principale. | |
| Estensione A: Errore Email | |  |  |  | | --- | --- | --- | | 3.a.1 |  | Mostra M2 –  L’email inserita non  ha il formato  corretto. | | 3.a.2 | Clicca “ok”. |  | | 3.a.3 |  | Ritorna allo step 1. | |
| Estensione B: Utente non amministratore | |  |  |  | | --- | --- | --- | | 4.a.1 |  | Mostra M3 –  Utente non  autorizzato. | | 4.a.2 | Clicca “ok”. |  | | 4.a.3 |  | Ritorna allo step 1. | |
| Estensione C: Email già presente a sistema | |  |  |  | | --- | --- | --- | | 4.b.1 |  | Mostra M4 – Email  già presente nel  sistema. | | 4.b.2 | Clicca “ok”. |  | | 4.b.3 |  | Ritorna allo step 1. | |

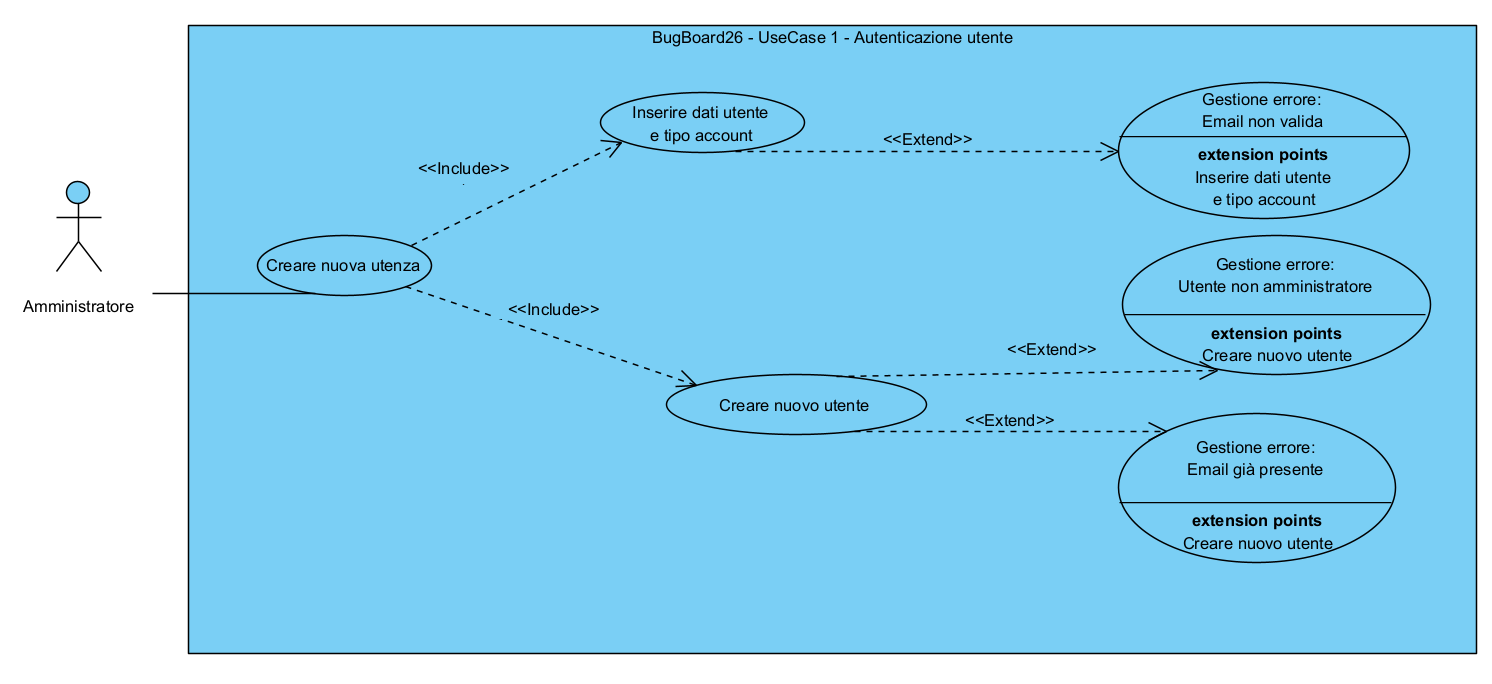


Figura 2 - UC diagram Creazione utente

## Segnalazione/Creazione Issue

Un qualsiasi utente potrà segnalare/creare nuove issue inserendo a sistema titolo, descrizione, tipo ed eventuali informazioni aggiuntive.

|  |  |
| --- | --- |
| **Caso d’Uso #3** | **Segnalazione/Creazione Issue** |
| Creato da: | |  |  |  | | --- | --- | --- | |  | Data creazione: |  | |
| Obiettivo: | L’utente vuole creare una issue |
| Pre-Condizioni: | L’utente deve aver effettuato l’accesso alla piattaforma BugBiard26 |
| Post-Condizioni per Successo: | L’issue è stata creata |
| Attore coinvolto: | Utente |
| Trigger: | L’utente clicca su “Segnala Nuova Issue” |
| Scenario principale con successo: | |  |  |  |  | | --- | --- | --- | --- | | Step n. | Utente | Sistema |  | | 1 | Clicca “Segnala Nuova Issue” in M0. |  |  | | 2 |  | Mostra M1 – Form  di segnalazione issue. |  | | 3 | Inserisci le informazioni obbligatorie relative l’issue da creare (titolo, descrizione e tipologia della nuova issue). Poi clicca “Avanti”. |  |  | | 4 |  | Mostra M2 – Form  di segnalazione issue. |  | | 5 | Inserisce le prossime e opzionali informazioni relative all’issue da creare (priorità e immagine da allegare). Poi clicca “Crea Issue”. |  |  | | 6 |  | Mostra l’interfaccia  principale. |  | |

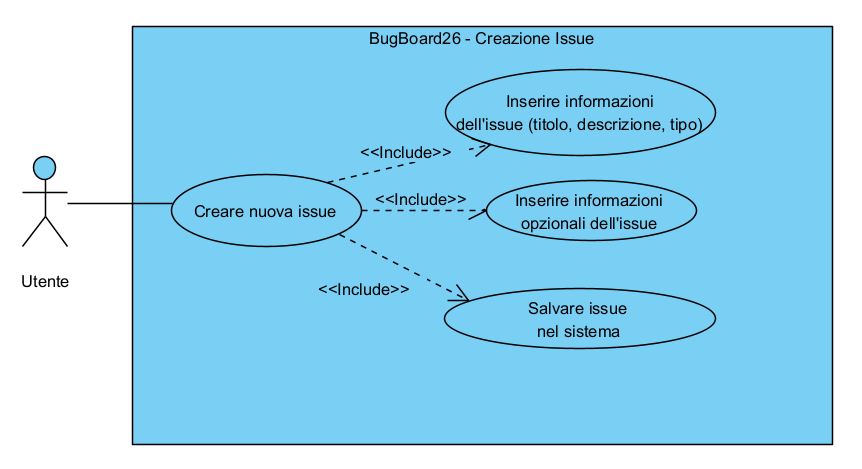


Figura 3 - UC diagram Creazione Issue

## Filtrare/Ordinare Issue

Un qualsiasi utente potrà visualizzare la lista degli issue e filtrare o ordinare in base a criteri come tipologia, stato, priorità o altri parametri rilevanti.

|  |  |
| --- | --- |
| **Caso d’Uso #4** | **Filtrare/Ordinare Issue** |
| Creato da: | |  |  |  | | --- | --- | --- | |  | Data creazione: |  | |
| Obiettivo: | L’utente vuole visualizzare ed ordinare determinate issue |
| Pre-Condizioni: | L’utente deve aver effettuato l’accesso alla piattaforma BugBiard26 |
| Post-Condizioni per Successo: | L’utente visualizza le issue desiderate |
| Attore coinvolto: | Utente |
| Trigger: | L’utente clicca su “Apri Lista Issue” |
| Scenario principale con successo: | |  |  |  | | --- | --- | --- | | Step n. | Utente | Sistema | | 1 | Clicca su “Apri Lista Issue” in M0. |  | | 2 |  | Mostra M1 – Form  di ricerca issue. | | 3 | Visualizza la lista completa degli issue. Filtra ed ordina in base a determinati criteri la lista e mostra il risultato cliccando “Applica Filtri o Ordine”. |  | | 4 |  | Mostra M2 – Form  di ricerca issue. | |

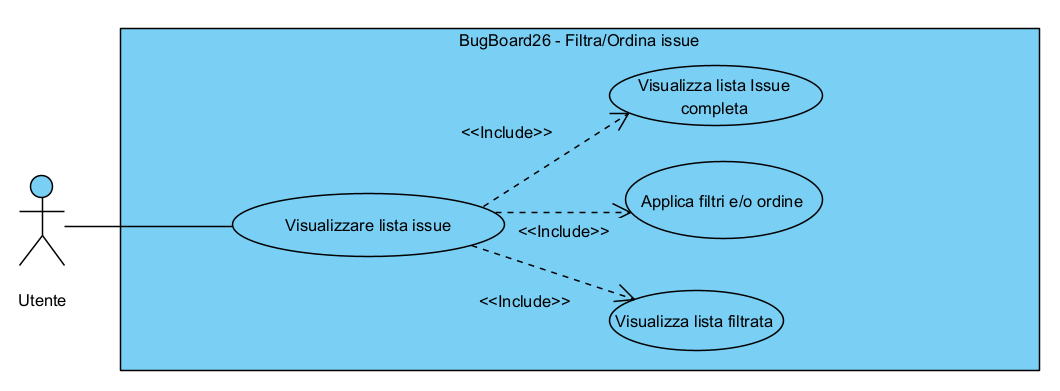


Figura 4 - UC diagram: Ricerca issue

## Assegnazione Bug

L’amministratore può assegnare uno specifico issue (bug) ad un determinato utente presente sulla piattaforma.

|  |  |
| --- | --- |
| **Caso d’Uso #5** | **Assegnazione bug** |
| Creato da: | |  |  |  | | --- | --- | --- | |  | Data creazione: |  | |
| Obiettivo: | L’amministratore vuole assegnare un bug |
| Pre-Condizioni: | L’amministratore deve aver effettuato l’accesso alla piattaforma BugBiard26 |
| Post-Condizioni per Successo: | L’amministratore ha assegnato il bug. |
| Attore coinvolto: | Utente amministratore |
| Trigger: | L’amministratore clicca su “assegna bug” |
| Scenario principale con successo: | |  |  |  | | --- | --- | --- | | Step n. | Amministratore | Sistema | | 1 | Clicca su “Apri Lista Issue” in M0. |  | | 2 |  | Mostra M1 – Form  di ricerca issue. Gli issue  di tipo bug che non sono  stati già assegnati possono  essere selezionati. | | 3 | Seleziona un bug e clicca su “Avanti”. |  | | 4 |  | Mostra M2 – Lista di utenti  sviluppatori che possono  lavorare al bug. La lista è  ordinata in base al carico  di lavoro corrente di ogni  utente. | | 5 | Seleziona un utente presente sulla lista e clicca “Assegna bug”. |  | | 6 |  | Mostra l’interfaccia  principale. | |
| Estensione A: Errore Bug selezionato | |  |  |  | | --- | --- | --- | | 3.a.1 |  | Mostra M3 – Nessun Bug  selezionato. | | 3.a.2 | Clicca “ok”. |  | | 3.a.3 |  | Ritorna allo step 2. | |
| Estensione B: Utente non amministratore | |  |  |  | | --- | --- | --- | | 5.a.1 |  | Mostra M4 - Utente non  autorizzato. | | 5.a.2 | Clicca “ok”. |  | | 5.a.3 |  | Ritorna allo step 2. | |
| Estenzione C: Nessun Utente selezionato. | |  |  |  | | --- | --- | --- | | 3.a.1 |  | Mostra M5 – Nessun  Utente selezionato. | | 3.a.2 | Clicca “ok”. |  | | 3.a.3 |  | Ritorna allo step 2. | |
| Estensione D: Bug già assegnato | |  |  |  | | --- | --- | --- | | 6.a.1 |  | Mostra M6 – Bug non più  Disponibile. Già  assegnato da un altro  amministratore. | | 6.a.2 | Clicca “ok”. |  | | 6.a.3 |  | Ritorna allo step 2. | |

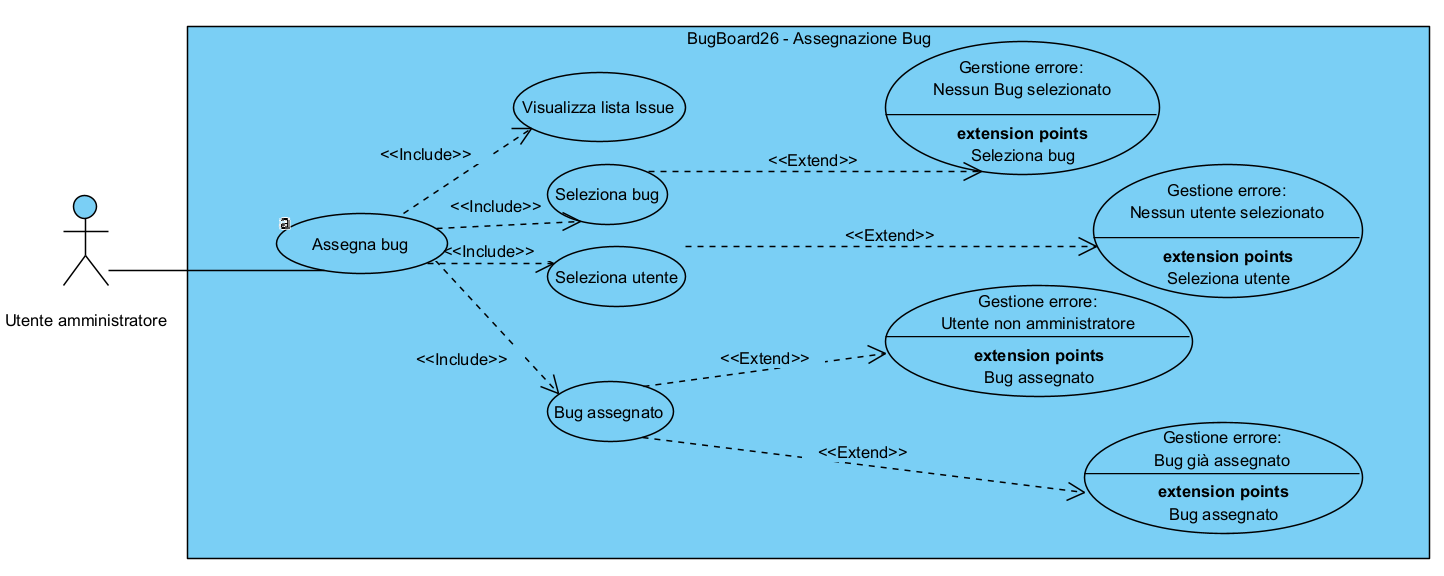


Figura 5 - UC diagram Assegna bug

## Visualizzazione Dashboard bug

L’amministratore può visualizzare una dashboard che mostra informazioni aggregate sui bug, come il numero di bug aperti, quelli assegnati per utente, il tempo medio di risoluzione (aggregato e per utente) e altri dati utili.

|  |  |
| --- | --- |
| **Caso d’Uso #6** | **Visualizzazione dashboard bug** |
| Creato da: | |  |  |  | | --- | --- | --- | |  | Data creazione: |  | |
| Obiettivo: | L’amministratore vuole visualizzare la dashboard dei bug |
| Pre-Condizioni: | L’amministratore deve aver effettuato l’accesso alla piattaforma BugBiard26 |
| Post-Condizioni per Successo: | L’amministratore visualizza la dashboard dei bug. |
| Attore coinvolto: | Utente amministratore |
| Trigger: | L’amministratore clicca su “visualizza dashboard bug” |
| Scenario principale con successo: | |  |  |  | | --- | --- | --- | | Step n. | Amministratore | Sistema | | 1 | Clicca su “Visualizza Dashboard Bug” in M0. |  | | 2 |  | Mostra M1 – Form  dashboard bug. | | 3 | Clicca su “Visualizza per Utente” per visualizzare le informazioni sui bug relative ad un determinato utente. |  | | 4 |  | Mostra M2 – Form Lista  Utenti. | | 5 | Seleziona l’utente per il quale vuole le informazioni sui bug. |  | | 6 |  | Mostra M3 – Form  Utente con bug. | |
| Estensione A: Utente non amministratore | |  |  |  | | --- | --- | --- | | 1.a.1 |  | Mostra M4 - Utente non  autorizzato. | | 1.a.2 | Clicca “ok”. |  | | 1.a.3 |  | Ritorna allo step 1. | |
| Estensione B: Nessun Utente selezionato. | |  |  |  | | --- | --- | --- | | 3.a.1 |  | Mostra M5 – Nessun  Utente selezionato. | | 3.a.2 | Clicca “ok”. |  | | 3.a.3 |  | Ritorna allo step 4. | |
| Estensione C: Utente scelto non ha bug. | |  |  |  | | --- | --- | --- | | 3.b.1 |  | Mostra M6 – Utente  scelto non ha bug. | | 3.b.2 | Clicca “ok”. |  | | 3.b.3 |  | Ritorna allo step 4. | |

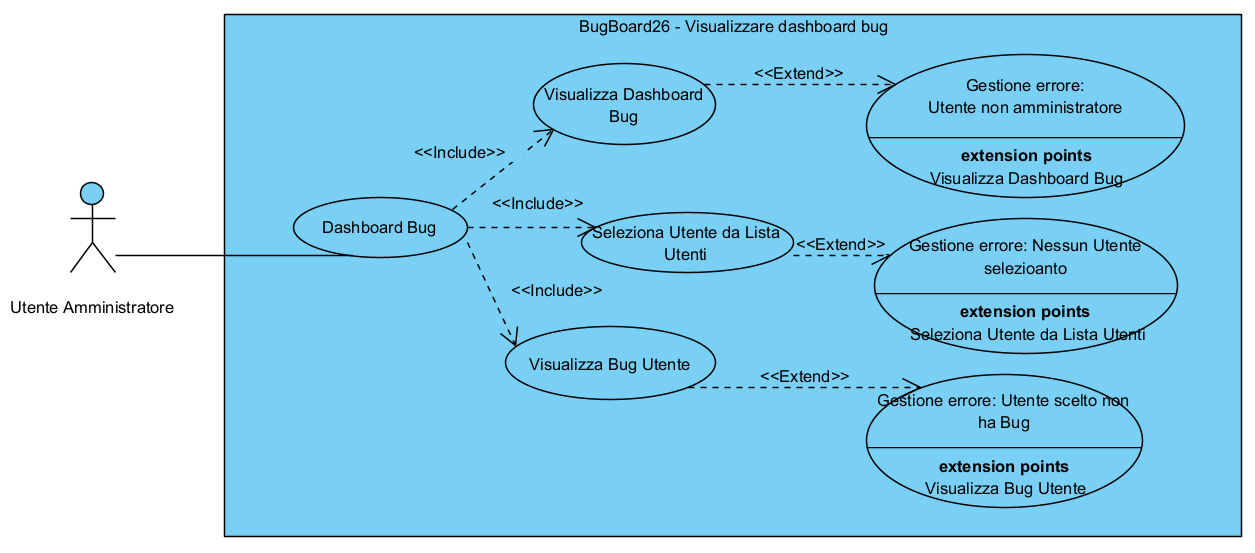


Figura 6 - UC diagram Visualizza dashboard bug

## Cronologia Bug

L’utente può visualizzare la cronologia delle modifiche di uno specifico bug.

|  |  |
| --- | --- |
| **Caso d’Uso #7** | **Cronologia bug** |
| Creato da: | |  |  |  | | --- | --- | --- | |  | Data creazione: |  | |
| Obiettivo: | L’utente vuole visualizzare una cronologia delle modifiche (inclusa la creazione), che mostri chi ha effettuato cambiamenti e quando. |
| Pre-Condizioni: | L’utente deve aver effettuato l’accesso alla piattaforma BugBiard26. |
| Post-Condizioni per Successo: | L’utente visualizza la cronologia delle modifiche del bug. |
| Attore coinvolto: | Utente |
| Trigger: | L’utente clicca su “Cronologia Bug” |
| Scenario principale con successo: | |  |  |  | | --- | --- | --- | | Step n. | Utente | Sistema | | 1 | Clicca su “Cronologia Bug” in M0. |  | | 2 |  | Mostra M1 – Form Lista  Bug. | | 3 | Seleziona sul bug desiderato e poi clicca “Mostra Cronologia”. |  | | 4 |  | Mostra M2 – Form di  Cronologia Bug. | |
| Estensione A: Nessun Bug Selezionato. | |  |  |  | | --- | --- | --- | | 3.a.1 |  | Mostra M3 – Nessun Bug  selezionato. | | 3.a.2 | Clicca “ok”. |  | | 3.a.3 |  | Ritorna allo step 2. | |

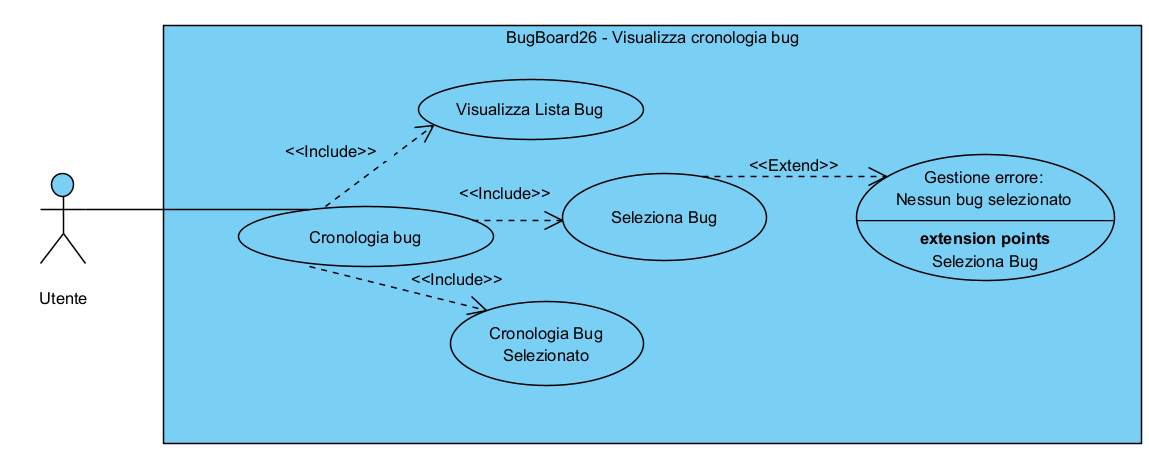


Figura 7 - UC diagram Visualizza cronologia bug

## Requisiti Non Funzionali

## Prestazioni

* Il sistema deve garantire che le operazioni svolte dagli utenti siano eseguite in modo performante.

## Affidabilità

* Il rilascio dell’applicativo avverrà dopo una serie di opportuni test case eseguiti con successo.

## Sicurezza

* L’accesso alla piattaforma deve essere consentito solo tramite autenticazione (email e password).

## Usabilità

* L’interfaccia deve essere intuitiva, con un layout coerente su tutte le schermate.
* Ogni messaggio di errore deve essere chiaro e informativo.
* L’applicazione deve essere compatibile con i browser più diffusi.

## Scalabilità

* Il sistema gestirà l’aumento del numero di utenti e di issue senza perdita significativa di prestazioni.

## Manutenibilità

* Il codice sarà modulare e documentato, in modo da facilitare aggiornamenti e correzioni.
* Il sistema includerà log di sistema e strumenti di monitoraggio errori.
* Ogni rilascio sarà versionato.

## Portabilità

* L’applicativo sarà accessibile sia da browser desktop che mobile.

## Requisiti delle Interfacce Esterne

## Interfaccia Utente

Il sistema fornirà un interfaccia web responsive, accessibile tramite browser desktop e mobile. L’interfaccia comprenderà:

* Schermata di Login, con campi Email e Password.
* Home Page, con riepilogo Issue e pulsanti.
* Form di Creazione Utente.
* Form di Creazione Issue.
* Form di Ricerca Issue.
* Dashboard Bug.
* Form di Cronologia Bug.

Tutti i messaggi saranno chiari, con modifiche di conferma o errore in formato testuale e visivo.

Il linguaggio dell’interfaccia è italiano, con possibilità futura di traduzione multilingua.

## Interfaccia Hardware

Il sistema può essere eseguito su qualsiasi dispositivo dotato di:

* Browser preferibilmente aggiornato con l’ultima versione disponibile.
* L’applicazione non richiede componenti Hardware dedicati.
* L’ambiente server ha uno spazio sul disco necessario per la memorizzazione di utenti, issue e log .

## Interfaccia Software

Il sistema dovrà interagire con:

* Database relazionale per la gestione persistente dei dati.
* Server web per la gestione dell’interfaccia.
* API REST interne per le operazioni CRUD.