

Software Requirements Specification (S.R.S.)

BugBoard26

Una piattaforma collaborativa per la gestione efficiente delle issue nei progetti software

Indice

1	Introduzione	2
1.1	Scopo	2
1.2	Obiettivi	2
1.3	Definizioni, Acronimi e Abbreviazioni	3
1.3.1	Definizioni	3
1.4	Requisiti Specifici	4
1.4.1	3.1 Requisiti Funzionali	4
1.4.2	3.2 Requisiti Non Funzionali	8
1.5	Modellazione dei Casi d'Uso	9
1.5.1	Autenticazione	9
1.5.2	Segnala issue	10
1.5.3	Visualizza elenco issue	10
1.5.4	Visualizza sezione commenti	11
1.5.5	Modifica issue	11
1.5.6	Assegna scadenza	12
1.5.7	UCD Generale	13
1.6	Individuazione e caratterizzazione degli attori del sistema	14
2	Formalizzazione di un Caso d'Uso Significativo	15
2.1	4.1 – Tabella di Cockburn	15
2.2	4.2 – Prototipazione Visuale	16

1. Introduzione

1.1 Scopo

Questo documento si rivolge ai membri del team di sviluppo incaricati della realizzazione del sistema BugBoard26. Lo scopo è illustrare il funzionamento e gli obiettivi principali della piattaforma software per la gestione collaborativa delle *issue* nei progetti software. Il sistema consente ai team di segnalare, assegnare e monitorare problemi, richieste o miglioramenti relativi a un progetto, in modo strutturato, efficiente e collaborativo.

1.2 Obiettivi

Il prodotto software **BugBoard26** deve svolgere le seguenti funzionalità:

1. BugBoard26 provvede all'autenticazione dell'utente consentendogli di accedere solo a quelle funzionalità che gli competono.
2. Creare ulteriori utenti (da parte degli account amministratori) indicando se queste sono utenze "normali" o "di amministrazione"
3. Segnalare e creare issue con: titolo, descrizione, tipologia (question, bug, documentation, feature), e in modo opzionale priorità e allegato immagine.
4. Fornire una vista riepilogativa delle issue con la possibilità di filtrare o ordinare i risultati in base a criteri come tipologia, stato, priorità o altri parametri rilevanti.
5. Assegnare issue ai membri del team con notifica all'assegnatario.
6. Fornire una sezione per commenti associata a ciascun bug, in cui tutti gli utenti possano lasciare messaggi, aggiornamenti o richieste di chiarimento.
7. Fornire funzionalità per la gestione di una issue: un utente può modificare le caratteristiche di una issue.
8. Gestire l'elenco delle issue.
9. Fornire una cronologia delle modifiche per ogni bug (inclusa la creazione) che mostri chi ha effettuato cambiamenti e quando..
10. Fornire la possibilità di impostare scadenze a determinate issue (solo da parte degli amministratori)

BugBoard26 ha come obiettivo principale quello di supportare i team di sviluppo nella gestione collaborativa delle issue legate ai progetti software, fornendo un ambiente unico e organizzato per il monitoraggio, l'assegnazione e la risoluzione dei problemi. La piattaforma si propone di semplificare il lavoro quotidiano degli sviluppatori e dei project manager, fungendo: a) da sistema centralizzato per la segnalazione e il tracciamento delle issue (bug, feature, question, documentation); b) da strumento di comunicazione interna, grazie alla possibilità di commentare e documentare ogni segnalazione in modo strutturato.

1.3 Definizioni, Acronimi e Abbreviazioni

1.3.1 Definizioni

- **Issue** – Segnalazione o richiesta effettuata da un utente all'interno del sistema. Può riguardare un bug, una domanda, una nuova funzionalità o la documentazione del progetto.
- **Tipologia di Issue** – Classificazione di un'issue in base alla sua natura: *bug*, *feature*, *question*, oppure *documentation*.
- **Bug** – Malfunzionamento o comportamento anomalo del sistema che compromette il corretto svolgimento delle operazioni previste.
- **Feature** – Funzionalità richiesta o prevista dal sistema, finalizzata a migliorare o estendere le capacità dell'applicazione.
- **Admin** – Utente con privilegi di amministrazione che può gestire, modificare e monitorare le risorse e gli utenti del sistema.
- **Priorità** – Livello di importanza assegnato a un'issue. Può assumere tre valori: *bassa*, *media* o *alta*, in base all'urgenza e all'impatto sull'attività del progetto.
- **Stato** – Indica il livello di avanzamento di un'issue. Gli stati possibili sono: *todo*, *in progress*, e *done*.
- **Commento** – Messaggio testuale associato a un'issue, utilizzato per fornire chiarimenti, aggiornamenti o osservazioni da parte dei membri del team.
- **Cronologia** – Registro cronologico degli eventi relativi a un'issue o ad altre attività del sistema, utile per tracciare chi ha compiuto una determinata azione, cosa è stato fatto e quando.
- **Mock-up** – Bozza grafica non interattiva che rappresenta l'aspetto e la struttura delle interfacce utente, utilizzata in fase di progettazione per visualizzare l'organizzazione degli elementi.
- **Utente autenticato** – Utente registrato nel sistema che accede mediante credenziali valide, ottenendo l'accesso alle funzionalità riservate.
- **Autenticazione** – Procedura di verifica delle credenziali (email e password) necessarie per accedere al sistema.
- **Credenziali** – Insieme di dati personali, tipicamente composti da email e password, utilizzati per l'autenticazione dell'utente.
- **Use Case** – Descrizione di un'interazione tra un attore (utente o sistema esterno) e il sistema stesso, finalizzata al raggiungimento di un obiettivo specifico.

1.4 Requisiti Specifici

1.4.1 3.1 Requisiti Funzionali

BugBoard26 prevede le seguenti funzionalità.

3.1.1 Autenticazione

3.1.1.1 Introduzione All'avvio, l'utente non autenticato deve effettuare il login con email e password. Il sistema è fornito con un account amministratore predefinito (credenziali di default). Dopo il login, l'utente resta autenticato fino al logout. Il *Sistema di Autenticazione* valida le credenziali.

3.1.1.2 Input

- **Email:** indirizzo dell'utente (obbligatorio).
- **Password:** credenziale (obbligatoria).

3.1.1.3 Elaborazione

- L'utente inserisce le credenziali e richiede l'autenticazione.
- Il sistema verifica la correttezza della coppia email/password tramite il Sistema di Autenticazione.
- Se valide, apre la sessione e abilita le funzionalità in base al ruolo (utente o amministratore); altrimenti richiede nuovo tentativo.
- L'utente autenticato può cambiare la propria password.
- L'amministratore può creare nuove utenze (email, password iniziale, ruolo).

3.1.1.4 Output

- **Messaggi di errore:** autenticazione non riuscita (credenziali errate), password troppo debole, email già esistente.
- **Messaggi di conferma:** autenticazione riuscita; utente creato.
- **Menu** associato al ruolo dell'utente autenticato.

3.1.2 Segnala Issue

3.1.2.1 Introduzione L'utente autenticato può segnalare una nuova issue inserendo almeno titolo e descrizione. La issue può avere tipologia *question*, *bug*, *documentation*, *feature*. Priorità e immagine allegata sono opzionali. Le nuove issue partono dallo stato *TODO*.

3.1.2.2 Input

- **Titolo:** testo breve (obbligatorio).
- **Descrizione:** testo descrittivo (obbligatorio).
- **Tipologia:** {question, bug, documentation, feature} (obbligatoria).
- **Priorità:** {bassa, media, alta} (facoltativa).
- **Immagine allegata:** file immagine (facoltativo).

3.1.2.3 Elaborazione

- L'utente inserisce i dati e richiede la creazione.
- Il sistema valida i campi obbligatori (titolo, descrizione, tipologia).
- Il sistema crea l'issue con stato iniziale *TODO* e associa l'autore.
- Il sistema registra l'evento nella **Cronologia modifiche** (creazione).

3.1.2.4 Output

- **Messaggi di errore:** campo obbligatorio mancante, formato file non valido.
- **Messaggi di conferma:** issue creata con identificativo univoco.

3.1.3 Visualizza Elenco Issue (con filtro e ordinamento)

3.1.3.1 Introduzione Il sistema offre una vista riepilogativa delle issue con possibilità di filtro e ordinamento.

3.1.3.2 Input

- **Criteri di filtro** (facoltativi): tipologia, stato, priorità, autore, data.
- **Criteri di ordinamento** (facoltativi): data di creazione, priorità, stato.

3.1.3.3 Elaborazione

- Il sistema carica l'elenco delle issue.
- Se presenti criteri, applica filtro e/o ordinamento.
- L'utente può selezionare una issue per aprire la **Vista dettagli**.

3.1.3.4 Output

- Elenco delle issue con: titolo, tipologia, stato, priorità, autore, data, eventuale scadenza.
- Messaggi di errore: criteri non validi.

3.1.4 Vista Dettagli Issue e Commenti

3.1.4.1 Introduzione Per ogni issue è disponibile una vista dettagliata e una sezione commenti per messaggi, aggiornamenti o richieste di chiarimento.

3.1.4.2 Input

- **ID Issue**: identificativo della issue (obbligatorio).
- **Testo commento**: contenuto testuale (obbligatorio per aggiunta/modifica).
- **Allegato commento**: file (facoltativo).

3.1.4.3 Elaborazione

- Il sistema mostra i dettagli dell'issue (titolo, descrizione, tipologia, stato, priorità, autore, data, allegati, scadenza).
- L'utente può aggiungere un commento; il sistema valida e salva.
- (Se previsto) L'utente può modificare o eliminare i propri commenti.
- Il sistema registra gli eventi nella **Cronologia modifiche**.

3.1.4.4 Output

- **Messaggi di conferma**: commento aggiunto/modificato/eliminato.
- **Messaggi di errore**: campi mancanti, permessi insufficienti, formato file non valido.
- **Dettagli issue** aggiornati con elenco commenti.

3.1.5 Cronologia delle Modifiche

3.1.5.1 Introduzione Ogni issue mantiene una cronologia (inclusa la creazione) che mostra chi ha effettuato modifiche e quando.

3.1.5.2 Input

- **ID Issue:** identificativo della issue (obbligatorio).
- **Criteri facoltativi:** filtro/ordine eventi.

3.1.5.3 Elaborazione

- Il sistema registra in modo automatico: creazione, modifiche, cambi stato, commenti, scadenze.
- Su richiesta, recupera e visualizza la lista eventi con autore, timestamp, tipo azione, eventuali valori prima/dopo.

3.1.5.4 Output

- Elenco eventi della cronologia (non modificabile dagli utenti).
- Messaggi di errore: issue inesistente, permessi insufficienti.

3.1.6 Gestione Scadenze (solo amministratore)

3.1.6.1 Introduzione L'amministratore può impostare una scadenza opzionale per la risoluzione di una issue, modificarla o rimuoverla.

3.1.6.2 Input

- **ID Issue** (obbligatorio).
- **Data scadenza** (facoltativa per rimozione; obbligatoria per impostazione/modifica).

3.1.6.3 Elaborazione

- L'amministratore seleziona la issue e imposta/modifica/rimuove la scadenza.
- Il sistema valida la data (formato, coerenza temporale).
- Il sistema registra l'evento nella **Cronologia modifiche**.

3.1.6.4 Output

- **Messaggi di conferma:** scadenza impostata/modificata/rimossa.
- **Messaggi di errore:** data non valida, permessi insufficienti, issue inesistente.

1.4.2 3.2 Requisiti Non Funzionali

3.2.1 Sicurezza

- Accesso al sistema mediante autenticazione (email, password).
- Password memorizzate in forma cifrata;
- Permessi differenziati per ruolo solo gli amministratori gestiscono utenti e scadenze.

3.2.2 Prestazioni

- Tempo medio di risposta breve per operazioni comuni (login, lista issue, commento).
- Filtri/ordinamenti eseguiti in tempo reale.

3.2.3 Usabilità

- Interfaccia coerente, responsiva e navigabile in pochi passaggi.
- Messaggi di errore chiari e indicazioni puntuali sui campi mancanti.

3.2.4 Affidabilità

- Ogni modifica ai dati deve essere registrata nella cronologia delle issue, includendo l'autore, la data e l'ora dell'operazione.
- Integrità dei dati garantita anche in caso di crash.
- Disponibilità del servizio $\geq 99\%$ (obiettivo).

3.2.5 Manutenibilità

- Architettura modulare, codice documentato e testabile.
- Aggiunta di nuove tipologie di issue o campi senza modifiche invasive. Il codice sorgente deve seguire standard di programmazione leggibili e documentati.

1.5 Modellazione dei Casi d'Uso

La modellazione dei casi d'uso descrive le interazioni tra gli attori e il sistema **BugBoard26**. I seguenti diagrammi rappresentano le principali funzionalità e le relazioni attore-sistema:

1.5.1 Autenticazione

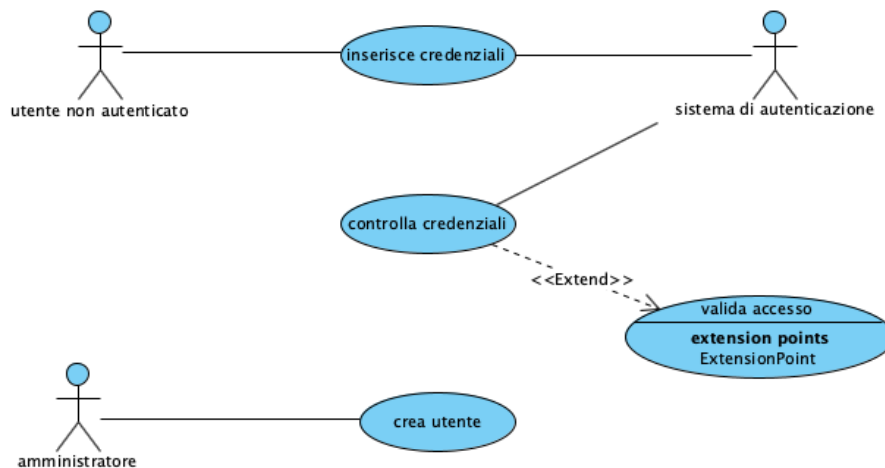


Figura 1.1: Use Case Diagram – Autenticazione

Attori:

- **Utente non autenticato:** inserisce le credenziali
- **Sistema di Autenticazione:** controlla le credenziali e se queste sono corrette può validare l'accesso.
- **Amministratore:** può creare ulteriori utenze, specificando una email, una password, e indicando se quell'utenza sarà "normale" oppure "di amministrazione".

1.5.2 Segnala issue

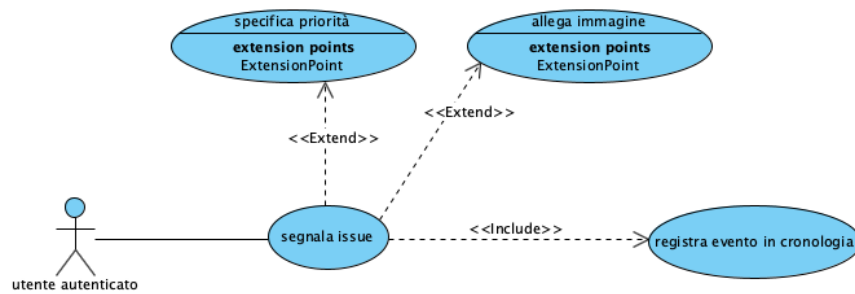


Figura 1.2: Use Case Diagram – segnala una issue

Attori:

- **Utente autenticato:** segnala una nuova issue con almeno titolo e descrizione e sceglie se inserire priorit  e/o immagine allegata

1.5.3 Visualizza elenco issue

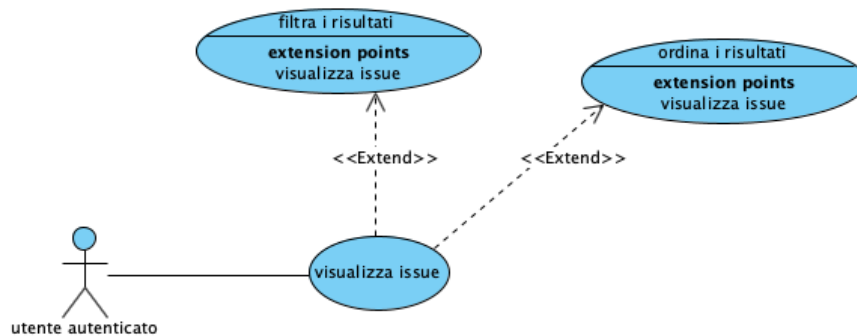


Figura 1.3: Use Case Diagram – visualizza elenco issue

Attori:

- **Utente autenticato:** visualizza l'elenco issue scegliendo se filtrare o ordinare i risultati

1.5.4 Visualizza sezione commenti

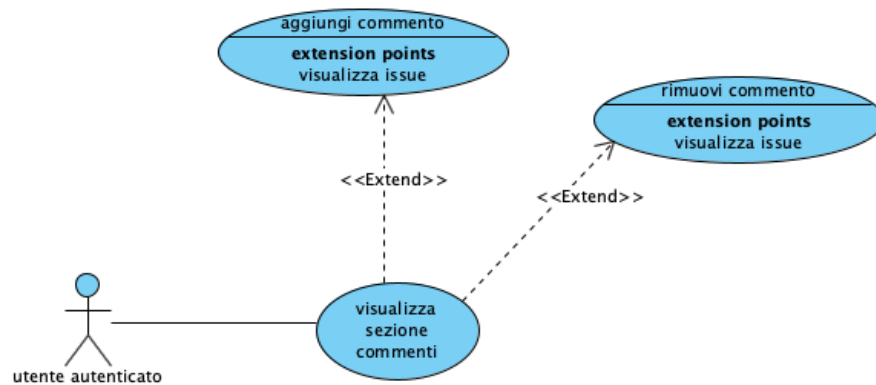


Figura 1.4: Use Case Diagram – visualizza sezione commenti

Attori:

- **Utente autenticato:** visualizza la sezione commenti di una issue con la possibilità di aggiungere un commento o rimuoverlo (la rimozione verrà salvata nella cronologia delle modifiche)

1.5.5 Modifica issue

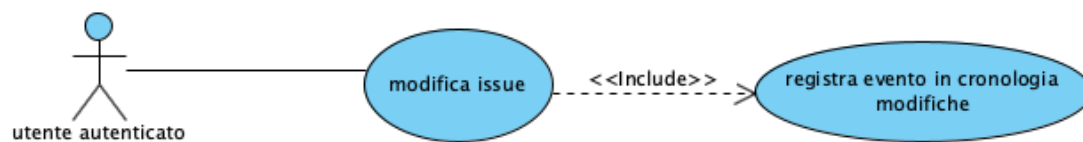


Figura 1.5: Use Case Diagram –modifica issue

Attori:

- **Utente autenticato:** modifica un issue, scegliendo di cambiare la descrizione e/o lo stato

1.5.6 Assegna scadenza

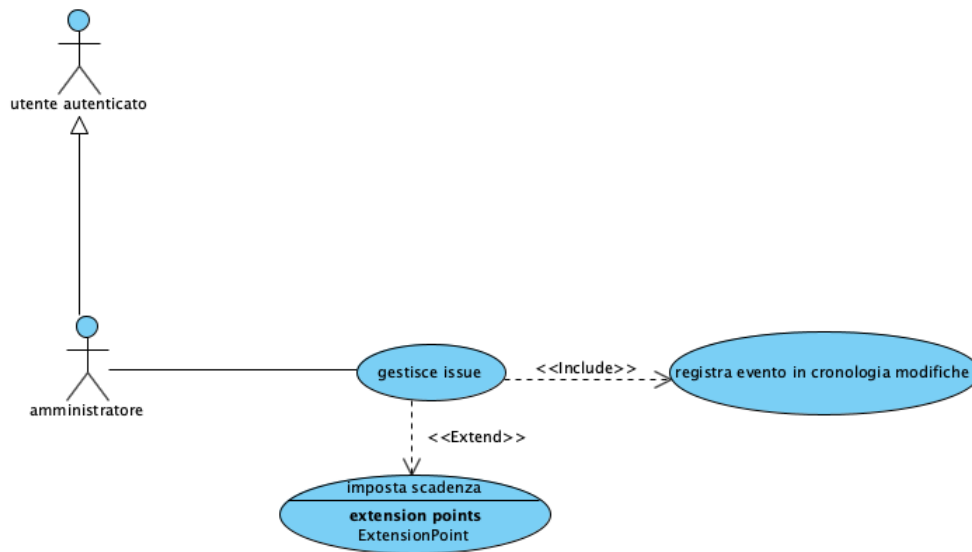


Figura 1.6: Use Case Diagram assegna scadenza

Attori:

- **Amministratore:** Gli amministratori possono impostare scadenze opzionali per la risoluzione di una certa issue.

1.5.7 UCD Generale

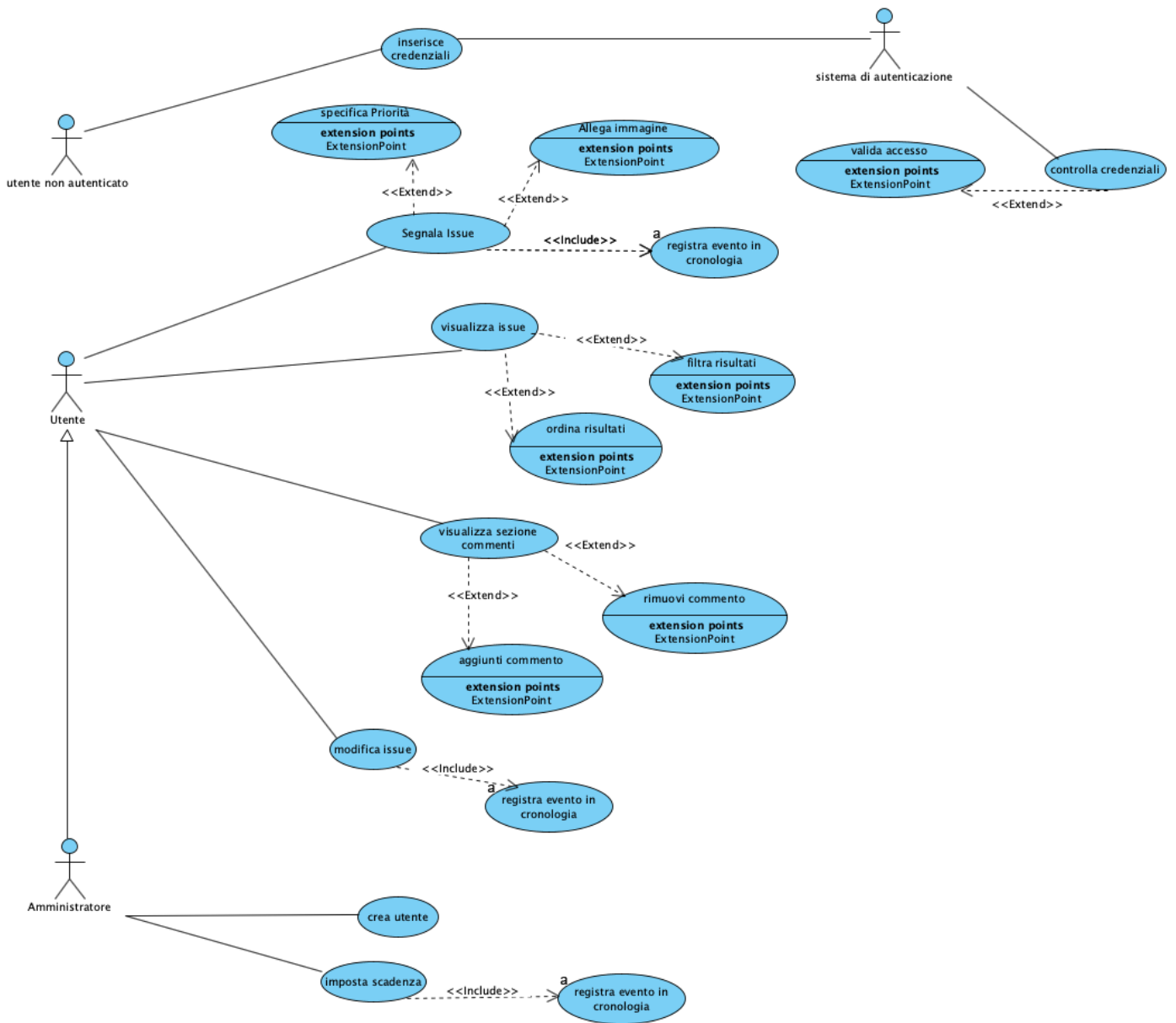


Figura 1.7: Use Case Diagram – generale

1.6 Individuazione e caratterizzazione degli attori del sistema

Utente non autenticato

- **Descrizione:** visitatore che non ha ancora effettuato l'accesso.
- **Obiettivi:** autenticarsi per utilizzare le funzionalità applicative.
- **Responsabilità principali:** inserimento credenziali; eventuale richiesta di cambio password.
- **Permessi:** accesso limitato alla sola schermata di login.

Utente autenticato

- **Descrizione:** utente regolarmente autenticato.
- **Obiettivi:** creare e consultare issue, collaborare tramite commenti, ricercare e filtrare.
- **Responsabilità principali:** segnalazione di nuove issue, consultazione elenco e dettagli, aggiunta/modifica dei propri commenti, lettura della cronologia.
- **Permessi:** accesso a tutte le funzionalità operative non riservate agli amministratori.

Amministratore

- **Descrizione:** utente con ulteriori privilegi..
- **Obiettivi:** gestire utenti e scadenze; garantire integrità e sicurezza operativa.
- **Responsabilità principali:** creazione di nuove utenze (email, password iniziale, ruolo), impostazione/modifica/rimozione di scadenze sulle issue, eventuale gestione avanzata.
- **Permessi:** tutte le azioni dell'Utente autenticato + funzioni amministrative (gestione utenti, scadenze).

Sistema di Autenticazione

- **Descrizione:** servizio esterno che verifica credenziali e applica le politiche di accesso.
- **Interazioni con BugBoard26:** validazione di email e password durante il login; accertamento del ruolo (utente/amministratore); supporto al cambio password.

2. Formalizzazione di un Caso d'Uso Significativo

2.1 4.1 – Tabella di Cockburn

la politica adottata in questo documento segue la rappresentazione di un **Main Scenario**, nel quale uno o più attori interagiscono tra loro mediante l'invocazione di trigger e la descrizione, passo per passo, degli eventi che compongono il flusso.

Use Case #02	Segnalazione nuova issue nella piattaforma
Goal in Context	L'utente autenticato vuole segnalare un nuovo bug o richiesta di funzionalità all'interno del sistema BugBoard26 .
Preconditions	L'utente deve essere autenticato nel sistema.
Success End Condition	L'issue è stata creata correttamente e appare nella lista delle issue con stato "todo".
Failed End Condition	L'issue non viene creata per errore di validazione o problema tecnico.
Primary Actor	Utente autenticato.
Trigger	L'utente seleziona l'opzione "Nuova Issue" dal menu principale.

DESCRIPTION

Step n°	Utente Autenticato	Sistema
1	Clicca sul pulsante "Segnala nuova issue".	Mostra la schermata di creazione issue.
2	Inserisce il titolo dell'issue.	
3	Inserisce la descrizione dell'issue.	
4	Seleziona la tipologia (<i>bug, feature, question, documentation</i>).	
5	(Opzionale) Imposta la priorità (<i>bassa, media, alta</i>).	
6	(Opzionale) Allega un'immagine a supporto della segnalazione.	Verifica che il file caricato sia in un formato valido (.jpg, .png).
7	Clicca su "Conferma" o "Crea".	Valida i dati inseriti (titolo e descrizione obbligatori).
8		Se la validazione è superata, salva i dati nel database.
9		Registra l'evento nella cronologia e invia notifica di conferma all'utente.
10		Mostra messaggio "Issue creata con successo".

Alternative Flow:

- **A1 – Dati mancanti:** se titolo o descrizione non sono compilati, il sistema mostra un messaggio d'errore e non procede con la creazione.
- **A2 – Errore di connessione:** se il database non è raggiungibile, il sistema mostra un messaggio “Errore durante il salvataggio” e consente di riprovare.

Postcondizioni:

- La nuova issue compare nell'elenco con stato iniziale *todo*.
- L'evento “Creazione issue” viene registrato nella cronologia.

2.2 4.2 – Prototipazione Visuale

La seguente figura mostra i mock-up realizzati per il caso d'uso “Segnala Issue”. Il primo rappresenta l'interfaccia web desktop, mentre il secondo illustra la versione mobile, ottimizzata per dispositivi smartphone.

The image shows a desktop mock-up of a web interface for creating a new issue. At the top, there is a blue header bar with the text "BugBoard26" on the left and "utente@azienda.it" on the right. The main content area is light gray and contains a white form titled "Nuova Issue". Below the title, a subtitle reads "Compila i campi obbligatori per creare una nuova segnalazione." The form has several fields: a "Titolo *" field with a placeholder "Es. Crash all'avvio su Windows 11"; a "Descrizione *" field with a placeholder "Descrivi il problema, i passi per riprodurlo, il risultato atteso..."; a "Tipologia" dropdown menu set to "Bug"; a "Priorità (opz.)" dropdown menu set to "Media"; and an "Allegati (opz.)" section with a button "Allega immagine". Below the attachments section, it says "Formati supportati: PNG/JPG fino a 5MB." At the bottom of the form are two buttons: "Crea Issue" (blue) and "Annulla" (white).

Figura 2.1: Mock-up desktop dell'interfaccia utente per la creazione di una nuova issue.

The image is a mobile app mockup for 'BugBoard26'. It features a blue header bar with the app name and a menu icon. The main content is a white card titled 'Nuova Issue' with a subtitle 'Compila i campi obbligatori.' Below this are several form fields: 'Titolo *' with an example 'Es. Crash all'avvio', 'Descrizione *' with a placeholder 'Descrivi problema e passi per riprodurlo...', 'Tipologia' with a dropdown set to 'Bug', 'Priorità (opz.)' with a dropdown set to 'Media', and 'Allegati (opz.)' with a button 'Allega immagine'. A note specifies 'PNG/JPG fino a 5MB.' At the bottom are two buttons: 'Crea Issue' (blue) and 'Annulla' (white).

BugBoard26 ...

Nuova Issue

Compila i campi obbligatori.

Titolo *

Es. Crash all'avvio

Titolo obbligatorio.

Descrizione *

Descrivi problema e passi per riprodurlo...

Descrizione obbligatoria.

Tipologia

Bug

Priorità (opz.)

Media

Allegati (opz.)

Allega immagine

PNG/JPG fino a 5MB.

Crea Issue

Annulla

Figura 2.2: Mock-up mobile dell'interfaccia utente per la creazione di una nuova issue.