Indice

**1 Specifica dei Requisiti 1**

1.1 Modellazione dei casi d’uso 2

1.2 Caratterizzazione degli attori 2

1.3 Requisiti non funzionali attori 3   
 1.4 Formalizzazione di un caso d'uso significativo 4

Link al nostro repository di github: https://github.com/Mariiiii739/BugBoard26

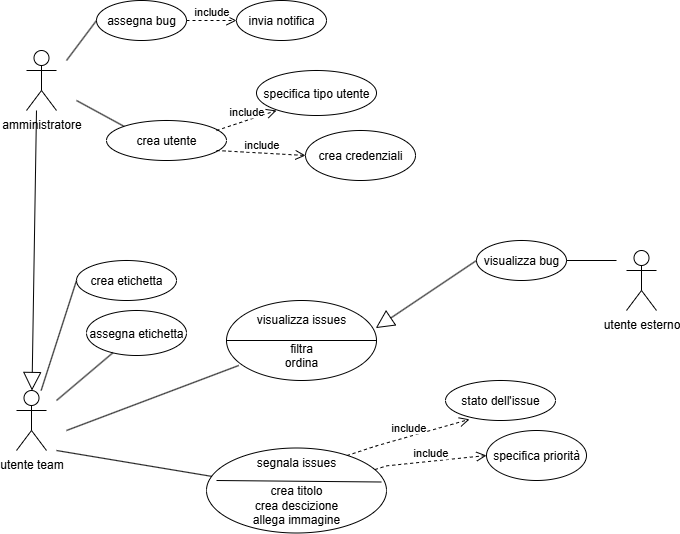
# Specifica dei requisiti

Verranno descritte nelle seguenti pagine la prima stesura di tutte le funzionalità del nostro software: BugBoard26.

Questo sistema permette di gestire issue, che verranno categorizzate in: question, bug, feature, e altri; al fine di permettere la gestione di queste in modo ordinato e compatto, puntando soprattutto al miglioramento delle prestazioni di lavoro di un team specializzato nella risoluzione di questi problemi.

A capo dei gruppi ci saranno degli amministratori che avranno come compito principale la creazione degli account degli utenti e la maggior parte delle questioni organizzative; gli utenti del gruppo potranno effettivamente lavorare sui compiti loro assegnati. Infine, si permetterà a degli utenti esterni di poter visualizzare i bug a scopo formativo

## 1.1 Modellazione dei casi d’uso



## 1.2 Caratterizzazione degli attori

|  |  |  |  |
| --- | --- | --- | --- |
| Attori | Casi d’uso | Descrizione | Livello d’accesso |
| Amministratore | Assegna il bug all’utente; Crea le credenziali per gli utenti. | È il responsabile della gestione delle issue e del personale. Può accedere agli stessi servizi dell’Utente Team. | Alto |
| Utente Team | Crea e assegna le etichette per i bug; visualizza e segnale le issue. | Utilizza i servizi del sistema. Non può accedere ai servizi dell’amministratore. | Medio |
| Utente Esterno | Visualizza i bug e le loro descrizioni. | Non può accedere a nessun servizio, né dell’Amministratore, né dell’Utente Team. Non può interagire direttamente con il sistema. | Basso |

## 1.3 Requisiti non funzionali

Il sistema deve garantire che utenti che non facciano parte di un determinato gruppo non possano modificare o interagire direttamente con la piattaforma, gli è concessa solo la possibilità di visualizzare ciò che i membri del gruppo scrivono.

Si garantisce che personale autorizzato, quindi con una licenza, può creare i profili per altri utenti, questo sarà possibile grazie a dei controlli sul livello d’accesso dell’utente, al fine di assicurare la sicurezza per utente del sistema, inoltre, prima di poter accedere all’applicazione, ogni utente sarà autenticato.

Garantisce inoltre che una volta fatto il primo accesso, tutti gli utenti possano cambiare la propria password, inserendone una personalizzata che rispetti i seguenti criteri:

* Massimo 16 caratteri;
* Almeno una lettera maiuscola
* Almeno un simbolo (come i seguenti: ! @ # ?).

Deve inoltre garantire la totale protezione dei dati di ogni utente, questo avverrà tramite:

* l’implementazione di un database che offrirà la cifratura dei dati sensibili e accessi al database tramite connessioni sicure;
* controllo degli accessi, nel dettaglio tramite autenticazione e autorizzazione e gestione sicura delle credenziali.

Infine, si garantisce che il sistema sarà intuitivo e facile da utilizzare, questo tramite una sezione di aiuti che fornirà maggiori informazioni sulle varie funzionalità del sistema.

1.4 Formalizzazione di un caso d'uso significativo

a) Descrizione testuale strutturata con formalismo tabellare di A.Cockburn

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE #1** | **Assegna un bug** | | |
| **Goal in Context** | Un Amministratore assegna un bug ad un utente del Team | | |
| **Preconditions** | Deve essere segnalata una issue che sia un bug | | |
| **Success End Condition** | Il sistema che tiene traccia della segnalazione di un bug, notifica l'amministratore e assegna il bug ad un utente del Team | | |
| **Failed End Condition** | Il sistema non tiene traccia della segnalazione del bug per un diverso tipo di issue, non notifica l'amministratore | | |
| **Primary Actor** | Amministratore | | |
| **Trigger** | Assegna bug | | |
| **Main Scenario** | **Step n.** | **Amministratore** | **System** | |
| 1 | Assegna il bug, clicca in M0 |  | |
| 2 |  | Mostra M1, verifica se ci sono dei bug | |
| 3 | Sceglie “si” o “no” |  | |
| 4 |  | Se sì, e ci sono dei bug, vai in M3. | |
| 5 | Seleziona il bug dalla lista dei bug disponibili |  | |
| 6 |  | Controlla se tutti i bug sono stati assegnati, se no, va in M4 | |
| 7 | Seleziona l’utente dalla lista degli utenti |  | |
| 8 |  | Il bug è assegnato correttamente, va in M5 | |
| Extension A: Assenza bug | 4A |  | Se sì, e non ci sono bug, vai in M2. | |
| 5A | Clicca OK |  | |
| 6A |  | Torna in M1 | |
| Extension B: Assegnazione dei bug | | 6B |  | Se tutti i bug sono già assegnati, va in M6 | |
| 7B | Clicca OK |  | |
| 8B |  | Torna in M3 | |
| Extension C: Assegnazione dei bug con gli utenti | | 8C |  | Controlla se il bug è stato già assegnato ad un utente, va in M7; | |
| 9C | Clicca OK |  | |
| 10C |  | Torna in M4 | |
| Extension D: Assegnazione degli utenti con i bug | | 8D |  | controlla se l’utente è già associato ad un bug, va in M8; | |
| 9D | Clicca OK |  | |
| 10D |  | Torna in M4 | |

B) Prototipo visuale via Mock-up delle relative interfacce utente Immagine che contiene diagramma, testo, Piano, linea

Il contenuto generato dall'IA potrebbe non essere corretto.