



UNIVERSITÀ DEGLI STUDI DI NAPOLI  
**FEDERICO II**

DOCUMENTO DI SPECIFICA DEI REQUISITI  
INGEGNERIA DEL SOFTWARE  
CdL Triennale in Informatica

AGOSTINO SORRENTINO  
N86005123  
MARIATERESA PRINCIPATO  
N86005284

ANNO ACCADEMICO: 2025/2026

# Sommario

<b><u>INTRODUZIONE</u></b>	<b>3</b>
<b>SCOPO</b>	<b>3</b>
<b>AMBITO</b>	<b>3</b>
<b><u>OBIETTIVI</u></b>	<b>3</b>
<b>REQUISITI DI SISTEMA</b>	<b>3</b>
<b>REQUISITI NON-FUNZIONALI</b>	<b>4</b>
<b><u>DEFINIZIONI</u></b>	<b>5</b>
<b>ACCOUNT</b>	<b>5</b>
<b>ACCOUNT DI AMMINISTRAZIONE (ADMIN)</b>	<b>5</b>
<b>ACCOUNT NORMALE (UTENTE)</b>	<b>5</b>
<b>ATTORE (ACTOR)</b>	<b>5</b>
<b>ALLEGATO (ATTACHMENT)</b>	<b>5</b>
<b>ASSEGNAZIONARIO (ASSIGNEE)</b>	<b>5</b>
<b>BUG</b>	<b>5</b>
<b>CARICO DI LAVORO (WORKLOAD)</b>	<b>5</b>
<b>COMMENTO (COMMENT)</b>	<b>5</b>
<b>FILTRO (FILTER)</b>	<b>5</b>
<b>ISSUE / TICKET</b>	<b>6</b>
<b>LABEL / TAG</b>	<b>6</b>
<b>NOTIFICHE (NOTIFICATION)</b>	<b>6</b>
<b>PRIORITÀ (PRIORITY)</b>	<b>6</b>
<b>REPORTER</b>	<b>6</b>
<b>STATI</b>	<b>6</b>
<b>ALTRO</b>	<b>6</b>
<b><u>INDIVIDUAZIONE E CARATTERIZZAZIONE DEGLI ATTORI DEL SISTEMA</u></b>	<b>7</b>
<b>AMMINISTRATORE (ADMIN)</b>	<b>7</b>
<b>UTENTE REGISTRATO (REGISTERED USER)</b>	<b>7</b>
<b>SISTEMA DI NOTIFICA (ATTORE NON UMANO)</b>	<b>7</b>

# Introduzione

## Scopo

Lo scopo del documento è quello di definire i requisiti funzionali e non-funzionali del sistema BugBoard26, una piattaforma per la gestione di issue in progetti software. Il documento segue il modello IEEE/ISO per la Specifica dei Requisiti Software.

## Ambito

Applicazione desktop (Java + Swing) che permette a team di segnalare, assegnare, tracciare e risolvere issue; include ricerca, filtri, notifiche e assegnazione intelligente.

## Obiettivi

### Requisiti Di Sistema

- Autenticazione e gestione utenti
  - Login tramite email+password.
  - Account amministratore iniziale con credenziali di default.
  - Admin può creare utenti, assegnare ruolo (normale/administration).
- Segnalare issue.
  - Allegare immagine. Supportare jpeg/png.
- Visualizzare elenco issue. Filtri per tipologia, stato, priorità, assegnatario.
- Assegnare issue. Admin con permessi possono assegnare manualmente.
  - Modificare stato e notifiche. Assegnatario può cambiare stato; se risolto -> notifica reporter.
- Ricerca per parole chiave
  - Funzione di ricerca su titolo e descrizione.
- Suggerimento automatico assegnatario. Sistema suggerisce assegnatario.

## Requisiti non-funzionali

1. Portata piattaforma: Desktop Java, Swing UI (NetBeans/IntelliJ form designer). DB relazionale (es. PostgreSQL).
2. Performance: Liste paginate; risposta per filtri e ricerca < 2s per dataset.
3. Affidabilità: Backup giornaliero del DB. Transazioni ACID per operazioni critiche (assegnazione/stato).
4. Sicurezza: Password, connessione al DB protetta, access control basato sui ruoli. Protezione upload (scansione tipi file, size limit).
5. Usabilità: Interfaccia con workflow in massimo 3 click per segnalare issue. Layout coerente con Swing.
6. Manutenibilità: Architettura MVC, codice documentato.
7. Scalabilità: Progettato per essere migrabile su architettura client-server se si rende necessario.

# Definizioni

## Account

Utilizzato per l'accesso di un utente nel sistema. Contiene credenziali (email, password), ruolo, e dati come data di creazione, nome e cognome del proprietario dell'account. I tipi di account sono divisi in “Normale” e “di Amministrazione”.

## Account di Amministrazione (Admin)

Account con privilegi avanzati di amministrazione: oltre alle azioni permesse agli utenti base, può creare/modificare/eliminare utenti e assegnare manualmente issue ad Account.

## Account Normale (Utente)

Account con privilegi base: creazione/ricerca/modifica/eliminazione issue.

## Attore (Actor)

Entità esterna che interagisce con il sistema (es. Utente, Admin, Sistema di Notifica).

## Allegato (Attachment)

File (immagine) allegato a una issue.

## Assegnatario (Assignee)

Utente al quale è assegnato il bug da risolvere.

## Bug

Tipo di issue che segnala un malfunzionamento del software.

## Carico di lavoro (Workload)

Numero di issue attive assegnate a un utente; parametro usato per il suggerimento automatico di assegnatario.

## Commento (Comment)

Messaggio testuale aggiunto dagli utenti all'interno della discussion di una issue.

## Filtro (Filter)

Criterio applicato alla visualizzazione dell'elenco issue.

## Issue / Ticket

Elemento centrale del sistema che rappresenta una segnalazione/compito/bug/feature request. Attributi tipici: id, title, description, type, priority, status, reporter\_id, assignee\_id, created\_at, updated\_at.

## Label / Tag

Parola chiave assegnabile a una issue per categorizzazione libera (es. ui, backend, performance).

## Notifiche (Notification)

Meccanismo per avvisare utenti di eventi (es. assegnazione, risoluzione). Possono essere email, popup in-app o entrambi.

## Priorità (Priority)

LOW / MEDIUM / HIGH / CRITICAL.

## Reporter

Utente che segnala l'issue.

## Stati

e.g. TODO, IN\_PROGRESS, REVIEW, RESOLVED, CLOSED.

## Altro

Issue, Bug, Ticket sono spesso usati come sinonimi; nel glossario si preferisce usare Issue come termine generico e Bug come sottotipo (type = bug).

Reporter = Creator della issue; Assignee = Owner responsabile per la soluzione.

Admin può essere indicato anche come Administrator o System Administrator.

# Individuazione e caratterizzazione degli attori del sistema

Individuazione e caratterizzazione di tutti gli utenti (attori) del sistema.

## Amministratore (Admin)

Privilegi: creare utenti, impostare ruoli, modificare impostazioni globali.

Requisiti: possibilità di forzare reset password, abilitare/disabilitare account.

## Utente registrato (Registered User)

Ruoli concreti: Reporter, Assignee, Developer, Tester (possono coincidere).

Funzionalità: segnalare issue, commentare, risolvere issue, modificare stato se assegnato, consultare dashboard.

## Sistema di Notifica (attore non umano)

Invia notifiche email/in-app quando cambia stato di issue o viene assegnata.