

28/08/2023

Elaborato
”TicketMaster”

Corso di Basi di Dati
Ingegneria e Scienze Informatiche Unibo - Campus di Cesena

Magnani Leonardo David Matteo
leonardo.magnani6@studio.unibo.it
0001021308

Analisi dei requisiti

Intervista

Estrazione concetti principali

Progettazione concettuale

Schema Scheletro

Utenti

Creazione ticket

Messaggistica nei ticket

Cambio di stato del ticket

Segnalazione Bug

Schema Finale

Progettazione logica

Stima del volume dei dati

Descrizione delle operazioni principali e stima della loro frequenza

Schemi di navigazione e tabelle degli accessi

Analisi delle ridondanze

Raffinamento dello schema

Schema relazionale finale

Traduzione delle operazioni in query

Progettazione dell'applicazione

Backend

Frontend

Esposizione

Analisi dei requisiti

Si vuole sviluppare un sistema di ticket di supporto tecnico per un software.

Intervista

Una software house desidera implementare un sistema di ticketing per il supporto tecnico al fine di gestire in modo efficiente i bug e i problemi segnalati dai clienti, offrendo loro un adeguato supporto. Per accedere al sistema, i clienti dovranno registrarsi fornendo le loro informazioni personali, come email, nome, cognome e password.

Il sistema sarà accessibile solo agli utenti registrati, che dovranno effettuare il login per accedere all'interfaccia. Attraverso l'interfaccia, i clienti potranno visualizzare tutti i ticket presenti nel sistema effettuare ricerche filtrate e aprire facilmente nuovi ticket di supporto fornendo informazioni dettagliate. Queste informazioni includeranno un titolo descrittivo del problema, una completa descrizione, il tipo di problema, il sistema operativo utilizzato e il livello di urgenza. Inoltre, gli utenti avranno la possibilità di eliminare i propri account se necessario.

Una volta aperto un ticket, il sistema assegnerà automaticamente il ticket all'operatore con meno ticket in carico al momento. Gli operatori potranno visualizzare i ticket assegnati e aggiungere commenti contenenti testo. La data e l'ora di scrittura saranno automaticamente registrate per ogni commento. I clienti, a loro volta, potranno visualizzare i commenti degli operatori e rispondere direttamente tramite l'interfaccia, permettendo così una comunicazione bidirezionale.

Quando un operatore visualizza un ticket assegnato, il suo stato passa da "aperto" a "in analisi". Una volta che il problema è risolto, l'operatore può chiudere il ticket, specificando se il problema è stato risolto o irrisolvibile. Ogni cambiamento di stato verrà registrato in uno storico per tenere traccia dell'evoluzione del ticket. Nel caso in cui il problema sia stato risolto, l'operatore può anche fornire uno o più tutorial come soluzione al ticket, che includerà un nome e un URL per un video o un PDF. Gli operatori hanno inoltre la possibilità di segnalare i bug riscontrati agli sviluppatori, fornendo una descrizione dettagliata e indicando l'urgenza per consentire una correzione tempestiva.

Alla chiusura del ticket, il cliente che l'ha aperto avrà la possibilità di valutare l'operatore attraverso un voto da 1 a 10 e un commento opzionale, fornendo così un feedback sul lavoro svolto e sull'assistenza ricevuta.

Gli sviluppatori avranno accesso alla lista dei bug riscontrati e potranno modificare lo stato di risoluzione dei bug. Ogni modifica dello stato verrà inserito in uno storico, insieme all'identità dello sviluppatore, per individuare eventuali responsabilità in caso di avanzamenti non corretti.

I responsabili della software house potranno monitorare diverse statistiche riguardanti il sistema di ticket di supporto tecnico. Questi dati statistici comprenderanno, per quanto riguarda gli operatori, il tempo medio impiegato da ciascun operatore per completare la chiusura di un ticket, il numero totale di ticket chiusi da ciascun operatore e la media delle valutazioni dell'operatore da parte dei clienti riguardo alla sua utilità. Per quanto concerne i ticket, saranno monitorati il numero di ticket in stato aperto, la quantità di ticket in fase di analisi, il numero di ticket effettivamente chiusi, il tempo medio di chiusura dei ticket, la percentuale di ticket chiusi rispetto al totale e, inoltre, il numero di ticket correlati a specifici tipi di problemi o sistemi operativi.

Le statistiche riguardanti i ticket potranno essere visualizzate selezionando il periodo di tempo desiderato. Questa funzionalità fornirà alla software house una panoramica completa delle prestazioni del supporto tecnico nel corso di un mese o di un anno, consentendo di individuare tendenze,

analizzare i risultati e prendere decisioni informate sulla gestione dei ticket e sulla risoluzione dei problemi segnalati dai clienti.

Inoltre, gli amministratori avranno il ruolo di gestire gli account degli operatori e degli sviluppatori, potendoli creare ed eliminare a loro discrezione.

Complessivamente, il sistema di ticket di supporto tecnico offrirà un metodo strutturato ed efficiente per gestire i problemi segnalati dai clienti, facilitando la comunicazione tra clienti e operatori, identificando le priorità di risoluzione dei problemi e fornendo alla software house informazioni dettagliate sulle prestazioni del supporto tecnico.

Estrazione concetti principali

Termine	Descrizione	Sinonimo
Utente	Utilizzatore del sistema. Può essere un cliente, operatore, sviluppatore e amministratore.	
Amministratore	Utente responsabile della gestione e della supervisione del sistema di ticket di supporto tecnico.	Responsabili della Software House
Cliente	Utente che utilizza il software che può visualizzare tutti i ticket e aprire nuovi ticket di supporto per segnalare problemi riscontrati durante l'utilizzo del software.	
Operatore	Utente membro del personale della software house responsabile di gestire e risolvere i ticket di supporto tecnico aperti dai clienti.	
Sviluppatore	Utente responsabile della risoluzione dei bug segnalati dai operatori.	
Ticket	Richiesta di assistenza aperta dai clienti attraverso l'interfaccia del sistema di ticket di supporto.	
Messaggio	Messaggio scritto da un utente in un ticket.	Commento
Valutazione Operatore	Valutazione effettuata da un cliente su un operatore.	Feedback
Creazione Ticket	L'azione che il cliente effettua per creare un nuovo ticket.	Apertura ticket
Fase del ticket	Stato in cui si trova un ticket durante il suo percorso di gestione.	Evoluzione del ticket
Avanzamento del sviluppo del bug	Stato in cui si trova un bug durante il suo percorso di gestione.	Modificare lo stato di risoluzione dei bug, Avanzamento

Riassunto concetti principali rimuovendo le ambiguità:

- Gli **Utenti** registrati nel sistema di supporto tecnico devono fornire email, nome, cognome e password per accedere al sistema.

Sono presenti diversi tipi di **Utenti** ognuno con privilegi specifici:

- **Clienti**
 - Hanno accesso alla visualizzazione di tutti i ticket presenti nel sistema, inclusi quelli creati da loro stessi.
 - Possono creare dei ticket per richiedere supporto.
 - Mentre il ticket è aperto, i clienti potranno scrivere messaggi all'interno dello stesso per comunicare con il team di supporto e fornire ulteriori dettagli o chiarimenti.
 - Una volta che il loro ticket è stato risolto e chiuso, i clienti avranno la possibilità di valutare l'operatore assegnato, assegnandogli un voto da 1 a 10 e lasciando un commento opzionale riguardante l'esperienza di supporto ricevuta.
- **Operatori**
 - Hanno accesso alla visualizzazione di tutti i ticket assegnati a loro.
 - Quando viene creato un ticket, il sistema assegnerà automaticamente il ticket all'operatore con il minor ticket in carico al momento.
 - Durante la fase aperta del ticket, gli operatori potranno scrivere messaggi all'interno del ticket per comunicare con il cliente e fornire aggiornamenti sullo stato di avanzamento.
 - Quando visualizzano un ticket, avanzeranno lo stato del ticket da "aperto" a "in analisi", e al termine della gestione, avranno la possibilità di chiudere il ticket assegnandogli lo stato di "chiuso - risolto" o "chiuso - non risolvibile".
 - Possono segnalare bug riguardante un problema riscontrato in un ticket agli sviluppatori.
 - Nel caso in cui il ticket venga risolto con successo (stato "chiuso - risolto"), l'operatore avrà la possibilità di associare un o più **tutorial** come soluzione al ticket, fornendo un nome e un link URL a un video o PDF.
- **Sviluppatori**
 - Hanno accesso alla visualizzazione di tutti i bug segnalati nel sistema.
 - Possono avanzare lo stato di sviluppo del bug attraverso diverse fasi di sviluppo: da "non in correzione" a "in revisione" e, infine, a "chiuso - risolto" o "chiuso - non riproducibile".
- **Amministratori**
 - Hanno accesso alle statistiche. Inoltre, avranno la possibilità di gestire gli operatori e gli sviluppatori del sistema, compresa la visualizzazione, eliminazione e creazione di nuovi account per gli stessi.
- **Ticket**
 - Hanno un titolo, una descrizione dettagliata, la priorità, il tipo di problema, il sistema operativo e stato. Un identificatore univoco verrà assegnato automaticamente al ticket.
 - Per ogni cambio di stato viene memorizzata la fase in uno storico per tenere traccia dell'evoluzione del ticket.
 - Ogni cliente può creare dei ticket.
 - Ogni ticket appena aperto è assegnato ad un operatore.
 - Contiene dei messaggi.

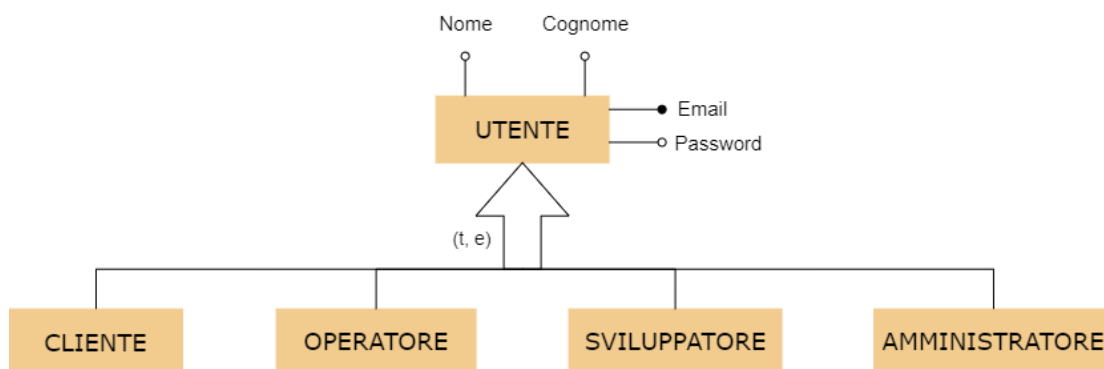
- Può contenere più tutorial come soluzioni.
- **Bug**
 - Hanno una descrizione dettagliata, priorità e stato. Un identificatore univoco verrà assegnato automaticamente al bug.
 - Per ogni cambio di stato viene memorizzato lo sviluppo in uno storico insieme al sviluppatore che l'ha effettuato per individuare eventuali responsabilità in caso di avanzamenti non corretti.
 - Riguarda un ticket.
- **Tutorial**
 - Hanno un nome e link URL a un video o PDF.
 - Vengono utilizzati come soluzioni per i ticket.
- **Messaggio**
 - Hanno del testo e una data di scrittura.
 - Può essere scritto da due tipi diversi di utenti, cliente e operatore.
 - E' contenuto in un ticket.

Progettazione concettuale

Schema Scheletro

Utenti

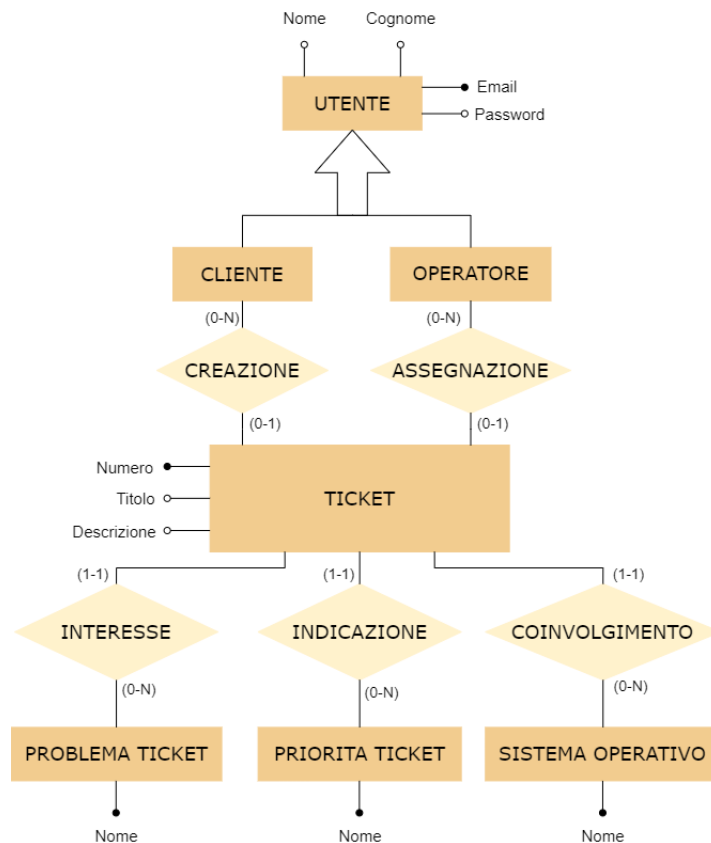
Le entità di **cliente**, **operatore**, **sviluppatore** e **amministratore** derivano dall'entità padre **utente** identificata tramite l'email.



Creazione ticket

I **clienti** hanno la possibilità di creare più **ticket** all'interno del sistema. Ogni ticket (identificato da un codice univoco) richiede la selezione di un Titolo, una Descrizione, un **Problema**, una **Priorità** e un **Sistema Operativo**.

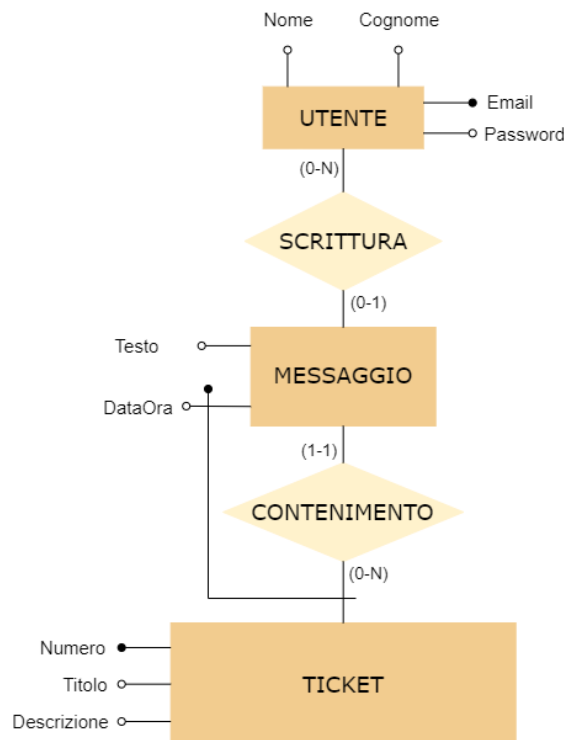
Alla creazione di un ticket, questo viene assegnato all'**operatore** più disponibile.



È importante notare che un ticket può essere creato anche senza essere associato a un cliente o operatore. Questa scelta è stata adottata per evitare di perdere i ticket nel caso in cui i dati dei clienti o operatori vengano eliminati dal sistema. In questo modo, i ticket possono rimanere salvati e gestiti indipendentemente dallo stato dei dati dei clienti o operatori.

Messaggistica nei ticket

Gli utenti hanno la possibilità di scrivere **messaggi** (identificati dalla data e ora di scrittura nel ticket) all'interno dei propri ticket associati.



Poiché solo gli operatori e i clienti sono tecnicamente associati ai ticket, saranno solo loro a poter scrivere nei ticket nell'implementazione. Tuttavia, è stata presa la decisione di consentire a tutti i tipi di utenti di inviare messaggi ai ticket. Questa scelta è stata fatta per semplificare non solo la gestione dei messaggi tramite query, ma anche per favorire possibili estensioni future, come la possibilità di avere chat private tra utenti.

È importante sottolineare che, analogamente ai ticket, anche i messaggi possono essere creati senza un utente associato. Questa scelta è stata adottata con l'obiettivo di mantenere tutti i dati di un ticket, garantendo che nessun messaggio venga perso, indipendentemente dalla presenza o dall'assenza di un utente.

Tuttavia, è importante tenere presente che se un ticket viene eliminato, essendo un contenitore per i messaggi, anche tutti i messaggi contenuti al suo interno verranno eliminati insieme ad esso. Questo permette di mantenere l'integrità dei dati e di evitare la presenza di messaggi "orfani" nel sistema.

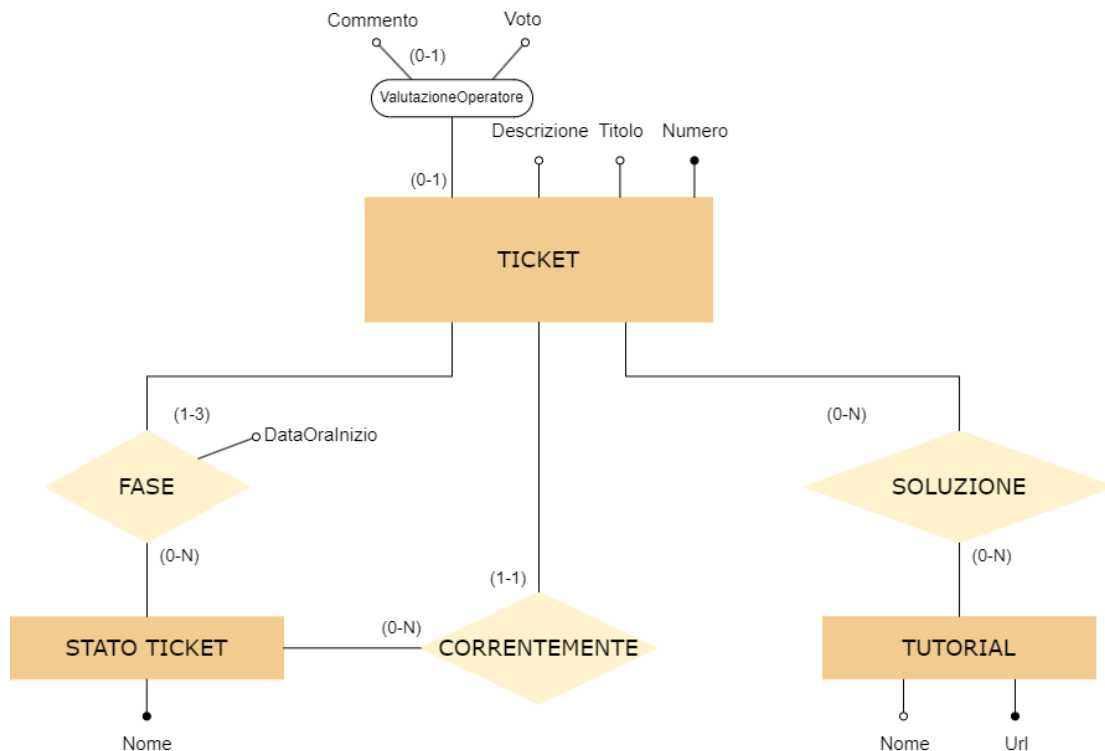
Cambio di stato del ticket

Alla creazione del ticket, il suo **stato** iniziale è "aperto". Successivamente, quando l'operatore visualizza il ticket, lo stato cambia in "analisi". L'operatore ha la facoltà di decidere se chiudere il ticket con una risoluzione ("chiuso - risolto") o chiuderlo senza risoluzione ("chiuso - non risolvibile").

Ogni volta che avviene un cambiamento di stato, viene memorizzata una **fase del ticket** nello storico, tenendo traccia della data di inizio dello stato. Dopo la chiusura, non sarà più possibile cambiare lo stato del ticket, quindi ogni ticket avrà almeno uno ("aperto") e al massimo tre stati ("aperto" → "analisi" → "chiuso").

In caso di chiusura con risoluzione di un ticket, l'operatore ha anche la possibilità di inserire dei **tutorial** come soluzione del ticket per il cliente.

Una volta che il ticket è stato chiuso, il cliente avrà la possibilità di **valutare l'operatore** assegnato, esprimendo un voto e, se lo desidera, aggiungendo un commento opzionale.



Nello storico, viene memorizzata solo la data di inizio della fase. Per ricavare la data di fine della fase, è sufficiente accedere alla fase successiva poiché l'avanzamento è unidirezionale e predefinito, passando da "aperto" a "analisi" e poi a "chiuso".

Inserire una data di fine aggiuntiva sarebbe ridondante e non necessaria. Accedere alla fase successiva della fase corrente richiederebbe 2 accessi in lettura, ma ciò è comunque preferibile rispetto all'aggiornamento della fase precedente con una data di fine quando si inserisce una nuova fase, il quale richiederebbe 2 accessi in scrittura. Le operazioni di scrittura sono generalmente più costose rispetto a quelle di lettura, quindi ridurre il numero di accessi in scrittura è una scelta generalmente più efficiente.

Inoltre, ho incluso la "valutazione dell'operatore" come un attributo composto del ticket. Questo è stato fatto per evidenziare che ogni operatore può ricevere una valutazione per ogni ticket che ha gestito e chiuso.

Si è aggiunta una associazione ridondante "Correntemente" la motivazione di questa scelta si potrà vedere in "Analisi delle Ridondanze"

Segnalazione Bug

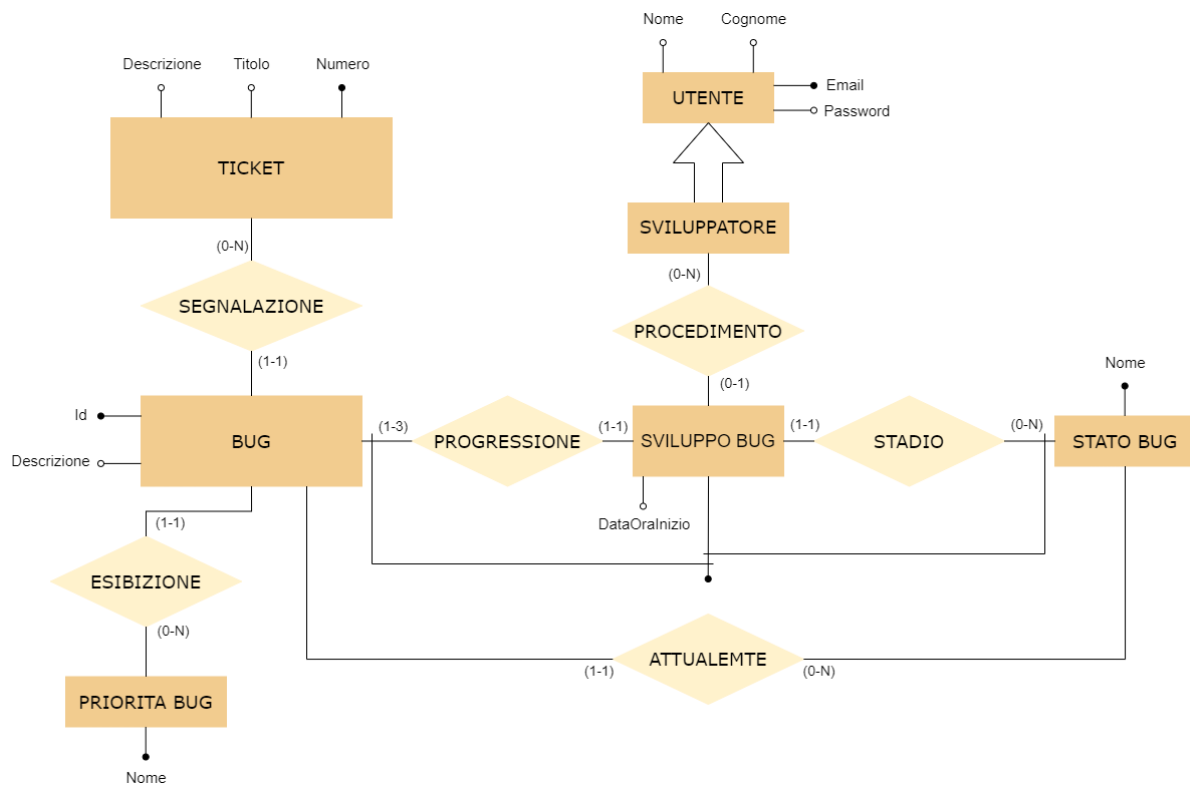
Gli operatori hanno la possibilità di segnalare i **bug** ai **sviluppatori**, e ogni bug è identificato da un codice univoco. Ogni segnalazione richiede una descrizione e una **priorità** per la risoluzione del problema.

Al momento della segnalazione, il bug assume uno **stato** iniziale di "non in correzione".

Successivamente, il sviluppatore può avanzare lo stato a "in revisione" e poi chiuderlo con lo stato di

"chiuso - non riproducibile" o "chiuso - risolto". Ogni volta che avviene un cambiamento di stato, viene registrato l'**avanzamento del sviluppo del bug** nello storico, tenendo traccia della data di inizio dello stato e del sviluppatore che ha effettuato la modifica.

Dopo la chiusura, non sarà più possibile modificare lo stato del bug. Di conseguenza, ogni bug avrà almeno uno stato ("non in correzione") e al massimo tre stati ("non in correzione" → "in revisione" → "chiuso").



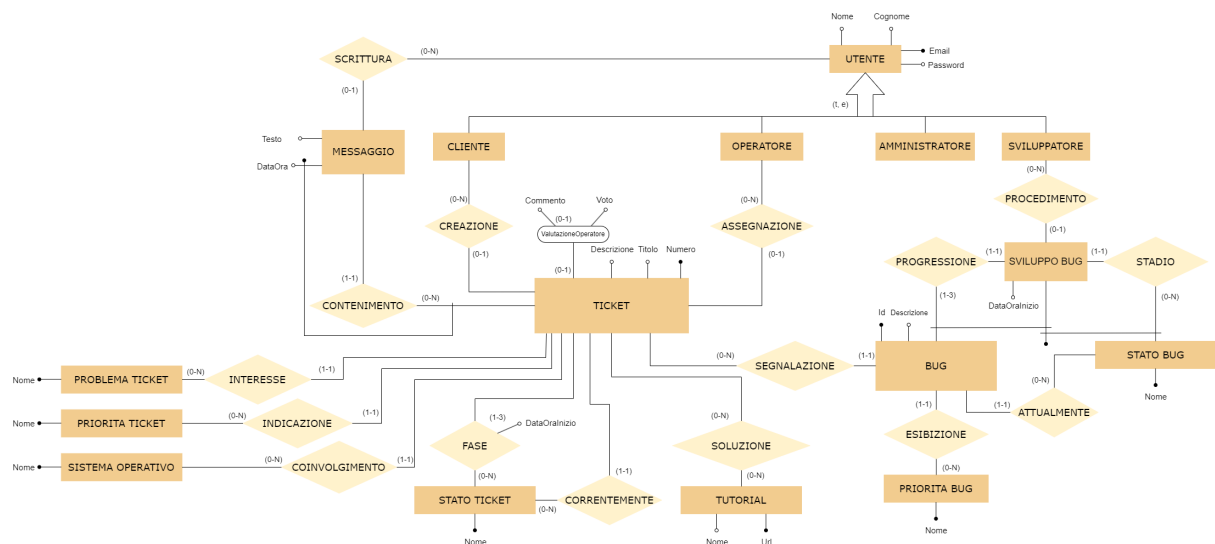
"Sviluppo Bug" è un'associazione ternaria che è stata rappresentata come entità con 3 associazioni per rendere più comprensibile l'associazione stessa. Questo approccio è stato scelto per migliorare la leggibilità dello schema e rendere più chiara la relazione tra le diverse entità coinvolte.

"Sviluppo Bug", come l'entità "Fase", ha solo il campo data di inizio.

È importante sottolineare che un procedimento di sviluppo del bug può essere creato anche senza essere associato a un sviluppatore. Questa scelta è stata adottata per evitare la perdita di informazioni riguardanti gli stadi di sviluppo del bug nel caso in cui un sviluppatore venga rimosso dal sistema.

Si è aggiunto una associazione ridondante "Attualmente" la motivazione di questa scelta si potrà vedere in "Analisi delle Ridondanze"

Schema Finale



Progettazione logica

Stima del volume dei dati

Concetto	Costrutto	Volume
Cliente	E	700.000
Creazione	R	960.000 (clienti possono essere eliminati)
Operatore	E	500
Assegnazione	R	920.000 (ticket chiuso associato ad un operatore eliminato non viene riassegnato $((1.000.000 \times 4/5) / 500) \times 50 = 80.000$)
Messaggio	E	3.000.000 (3x Ticket)
Scrittura	R	2.600.000 (operatori e clienti possono essere eliminati)
Ticket	E	1.000.000
Contenimento	R	3.000.000 (3x Ticket)
Problema Ticket	E	6
Interesse	R	1.000.000
Priorita Ticket	E	3
Indicazione	R	1.000.000
Sistema Operativo	E	3
Coinvolgimento	R	1.000.000
Stato Ticket	E	4
Fase	R	2.800.000 $(1.000.000 \times 4/5 \times 3 + 1.000.000 \times 1/5 \times 2)$
Correntemente	R	1.000.000
Tutorial	E	10

Concetto	Costrutto	Volume
Soluzione	R	300.000 (circa 1/3 dei ticket avrà 1 o più soluzioni)
Bug	E	36.500
Segnalazione	R	36.500
Priorita Bug	E	3
Esibizione	R	36.500
Sviluppo Bug	E	100.000 ($3/4 \times 36.500 \times 3 + 1/4 \times 36.500 \times 2$)
Progressione	R	100.000 ($3/4 \times 36.500 \times 3 + 1/4 \times 36.500 \times 2$)
Stato Bug	E	4
Stadio	R	100.000 ($3/4 \times 36.500 \times 3 + 1/4 \times 36.500 \times 2$)
Attualmente	R	36.500
Sviluppatore	E	100
Procedimento	R	90.000 (sviluppatori che hanno fatto un procedimento del sviluppo possono essere eliminati)
Amministratore	E	1

Descrizione delle operazioni principali e stima della loro frequenza

Presupposizioni effettuate:

- 2/3 dei clienti che hanno appena creato un account saranno anche propensi a creare un ticket
- Tutti i ticket creati nel giorno vengono aperti dai operatori e all'apertura il ticket da aperto va in analisi
- 4/5 dei Ticket vengono chiusi al giorno
- 1/5 dei Ticket non vengono chiusi al giorno
- 3/4 dei Ticket chiusi vengono risolti
- 1/4 dei Ticket chiusi non vengono risolti
- 3/4 dei Bug vengono chiusi al giorno
- 1/4 dei Bug non vengono chiusi al giorno

Segue una tabella riportante la descrizione delle operazioni e la loro relativa frequenza:

Codice	Operazione	Frequenza	Motivazione
1	Creare un nuovo ticket	3.000 al giorno	Guardando online il sistema di ticketing "Zendesk", ho scoperto che vengono creati in media 3.000 ticket al giorno.

Codice	Operazione	Frequenza	Motivazione
2	Cercare all'operatore più disponibile da poi assegnare ad un ticket	3.000 al giorno	Per ogni ticket creato c'è un'assegnazione quindi 3.000 al giorno.
3	Registrare un nuovo cliente	2.000 al giorno	Ho deciso di utilizzare come numero di registrazione dei clienti i 2/3 della creazione giornaliera di ticket. La mia supposizione si basa sul fatto che almeno due terzi dei clienti che hanno appena creato un account saranno anche propensi a creare un ticket. Di conseguenza, il numero di registrazione dei clienti al giorno sarà di $(2/3 \times 3.000)$ 2.000 al giorno.
4	Eliminare un cliente	100 al giorno	L'eliminazione dell'account non è un avvenimento comune ma per la grandezza del sistema mi sembra che 100 sia un numero appropriato.
5	Autenticare l'accesso ad un utente	4.600 al giorno	Per calcolare il numero totale di autenticazioni al giorno, dobbiamo considerare sia le nuove registrazioni giornaliere (2.000) che il numero di accessi degli utenti già registrati (2.000). Considerando i clienti, gli operatori e gli sviluppatori, stimiamo che il totale di accessi giornalieri sarà di circa $(2.000 + 2.000 + 500 + 100)$ 4.600 al giorno. Tuttavia, gli amministratori sono esclusi da questa stima poiché accedono solo 1 volta a settimana.
6	Visualizzare i dettagli di un ticket e i suoi messaggi	10.800 al giorno	Suppongo che ogni cliente che ha effettuato l'accesso che non ha eliminato il suo account (3.900) visualizzerà i dettagli di almeno 2 ticket, mentre gli operatori (500) ne visualizzeranno almeno $(3.000/500)$ 6 diversi. Quindi, il numero totale di visualizzazioni dei dettagli di un ticket sarà approssimativamente $(3.900 \times 2 + 500 \times 6)$ 10.800 al giorno.

Codice	Operazione	Frequenza	Motivazione
7	Visualizzare i ticket	18.600 al giorno	Dopo il login di un cliente, la visualizzazione dei ticket avviene automaticamente, basandomi sulla supposizione di codice 6 , dove ogni cliente guarda almeno i dettagli di 2 ticket, quando un cliente visualizza i dettagli di un ticket poi ritorna alla schermata dei ticket. Questo avviene per 1 volta oltre al primo accesso ($3.900 * (1+1)$), quindi i clienti vedranno i ticket per un totale di (3.900×2) 7.800 volte al giorno. Oltre alla visualizzazione automatica, i clienti possono effettuare una ricerca filtrata, e suppongo che ogni cliente effettuerà tale ricerca almeno 2 volte. Questo porta a un totale di (3.900×2) 7.800 ricerche filtrate al giorno. Per quanto riguarda gli operatori, essi possono visualizzare i ticket assegnati a loro, e ipotizzo che apriranno tutti i ticket a loro assegnati. Seguendo lo stesso ragionamento precedente, ogni operatore visualizzerà i dettagli di ($3.000/500$) 6 ticket più la visualizzazione all'accesso quindi vedranno i ticket per un totale di ($500 \times (5+1)$) 3.000 volte al giorno. Pertanto, il numero totale di visualizzazioni dei ticket sarà di ($7.800+7.800+3.000$) 18.600 volte al giorno.
8	Scrittura di un messaggio	9.000 al giorno	Suppongo che ogni operatore risponda ai ticket il giorno stesso in cui gli vengono assegnati. Di solito, c'è una media di 3 messaggi per ogni ticket, che include il primo messaggio dell'operatore, la risposta del cliente e la successiva risposta dell'operatore. Quindi, ci sarà un totale di circa (3.000×3) 9.000 messaggi al giorno.
9	Cambiare stato di un ticket	8.400 al giorno	Il cambio di stato avviene in tre fasi distinte. Innanzitutto, c'è un cambio di stato alla creazione del ticket (3.000). Successivamente, avviene un cambio di stato quando l'operatore apre il ticket (3.000). Infine, il terzo e ultimo cambio di stato si verifica quando l'operatore chiude il ticket, la chiusura avverrà per 4/5 dei ticket ($4/5 \times 3.000$). In totale, ci saranno circa ($3.000 \times 2 + 4/5 \times 3.000$) 8.400 cambi di stato al giorno.
10	Associazione dei tutorial ad un ticket	900 al giorno	Suppongo che almeno 3/4 dei ticket chiusi venga risolto, il che equivale a ($3/4 \times 4/5 \times 3.000$) 1.800 ticket risolti al giorno. Di questi ticket risolti, presumo che almeno la metà riceva un tutorial, quindi circa ($1.800 / 2$) 900 ticket con tutorial al giorno.

Codice	Operazione	Frequenza	Motivazione
11	Segnalazione di un bug	100 al giorno	Si presuppone che circa 1/6 dei ticket non risolvibili non sia risolvibile a causa di bug. Quindi, ci saranno circa $(1/4 \times 4/5 \times 3.000 \times 1/6)$ 100 segnalazioni di bug al giorno.
12	Visualizzare i bug	100 al giorno	Proseguendo con lo stesso ragionamento del codice 6 , si stima che gli sviluppatori visualizzino un numero equo di bug assegnati a loro. Supponendo che aprirebbero tutti i bug loro assegnati e valutando che ci sono circa 100 segnalazioni di bug al giorno più la visualizzazione al primo accesso, gli operatori visualizzeranno i bug all'incirca $(100 \times (100/100 - 1 + 1))$ 100 al giorno.
13	Visualizzare i dettagli di un bug	100 al giorno	Supponendo che il carico di lavoro sia suddiviso in modo equo tra gli operatori, ognuno di loro visualizzerà i dettagli di circa $(100 \times (100/100))$ 100 bug al giorno.
14	Cambiare stato di un bug	550 al giorno	Il cambio di stato avviene in tre fasi distinte. Innanzitutto, c'è un cambio di stato alla creazione del bug (200). Successivamente, avviene un cambio di stato quando un sviluppatore prende in carico il bug (200). Infine, il terzo e ultimo cambio di stato si verifica quando il sviluppatore chiude il bug che accadrà per almeno 3/4 dei bug. In totale, ci saranno circa $(200 \times 2 + 200 \times 3/4)$ 550 cambi di stato al giorno.
15	Visualizzare le statistiche dei ticket in un dato periodo	1 a settimana	I dati del sistema subiscono frequenti cambiamenti, pertanto è necessario effettuare controlli tempestivi delle statistiche.
16	Visualizzare prestazioni degli operatori	1 a settimana	I dati del sistema subiscono frequenti cambiamenti, pertanto è necessario effettuare controlli tempestivi delle statistiche.
17	Creazione di un operatore	50 all'anno	Stima di assunzioni di operatori annui.
18	Eliminazione di un operatore	50 all'anno	Stima di abbandoni o licenziamenti di operatori annui. Equivale al numero di assunzioni poiché si cerca di mantenere lo stesso numero di operatori annui.
19	Creazione di un sviluppatore	10 all'anno	Stima di assunzioni di sviluppatori annui.
20	Eliminazione di un sviluppatore	10 all'anno	Stima di abbandoni o licenziamenti di sviluppatori annui. Equivale al numero di assunzioni poiché si cerca di mantenere lo stesso numero di operatori annui.

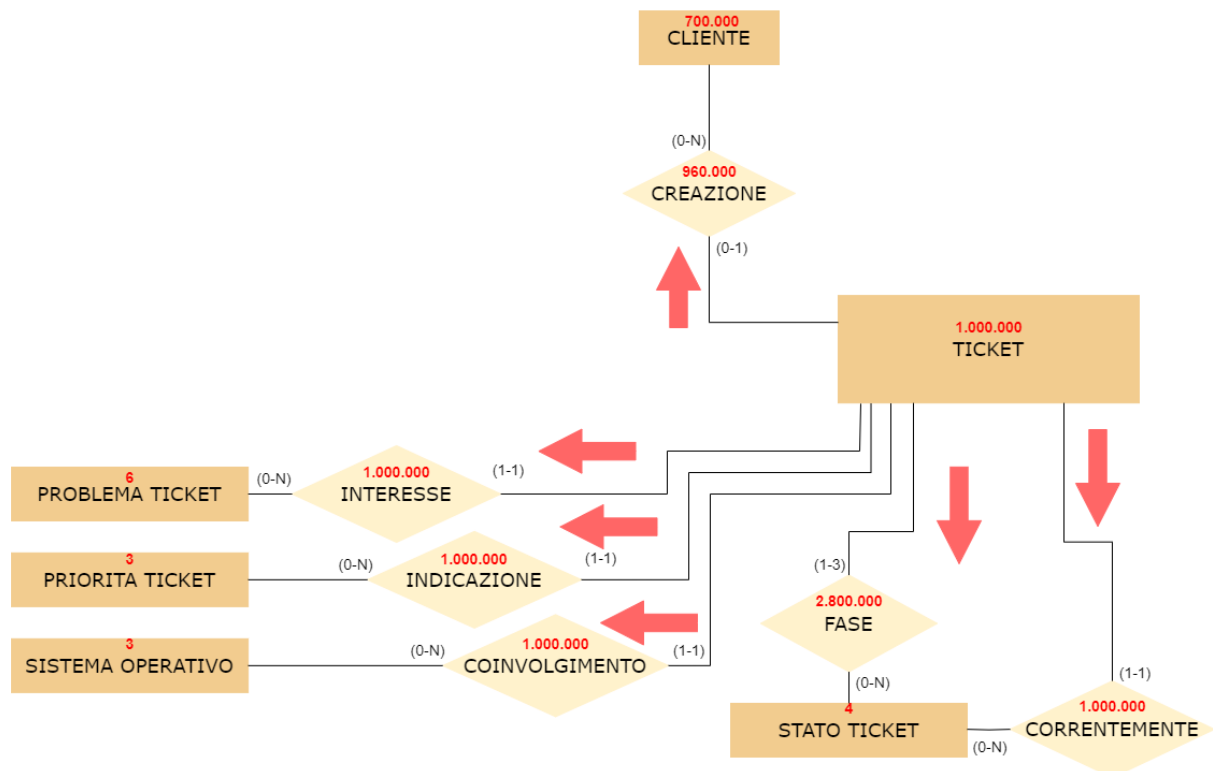
Codice	Operazione	Frequenza	Motivazione
21	Visualizzare tutti gli operatori	1 a settimana	Dato che si eliminano 50 operatori all'anno 365/50 si hanno 7 giorni cioè 1 settimana. Dato che per effettuare l'eliminazione bisogna visualizzare gli operatori questo sarà la frequenza.
22	Visualizzare tutti gli sviluppatori	1 al mese	Dato che si eliminano 10 sviluppatori all'anno 365/10 si hanno 36 giorni facciamo cioè 1 mese. Dato che per effettuare l'eliminazione bisogna visualizzare gli sviluppatori questo sarà la frequenza.
23	Valutazione Operatore	2.400 al giorno	La valutazione si può fare solo su ticket chiusi al giorno si chiudono (4/5 x 3.000) 2.400 quindi circa 2.400 al giorno

Schemi di navigazione e tabelle degli accessi

Sono riportate in seguito le tabelle degli accessi delle operazioni sopra riportate.

Al fine del calcolo dei costi, si considerano di peso doppio gli accessi in scrittura rispetto a quelli in lettura.

OP1 - Creare un nuovo ticket

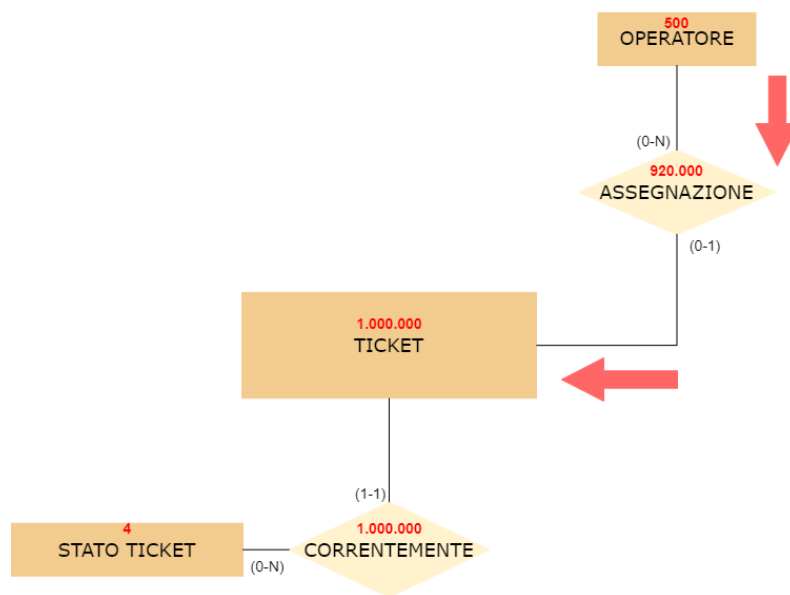


Concetto	Costrutto	Accessi	Tipo
Ticket	E	1	S
Creazione	R	1	S
Fase	R	1	S

Concetto	Costrutto	Accessi	Tipo
Correntemente	R	1	S
Interesse	R	1	S
Indicazione	R	1	S
Coinvolgimento	R	1	S

Totale: 7S → 42.000 al giorno (14 x 3.000 al giorno)

OP2 - Cercare all'operatore più disponibile da poi assegnare ad un ticket



Si leggono le assegnazioni attuali degli operatori per ottenere i ticket che sono loro stati assegnati. Successivamente, si leggono le ultime fasi dei ticket e si ricavano solo quelli che sono ancora aperti o in fase di analisi.

Poiché solo 1/5 dei ticket non è ancora stato chiuso e accedo solo ai ticket ancora aperti, il numero di ticket a cui accedo è equivalente a 1/5 di quelli assegnati all'operatore.

“Stato Ticket” ha un unico campo che è la chiave primaria “nome” quindi non si farà alcun accesso perché il nome dello stato sarà già presente nel “Ticket”

Concetto	Costrutto	Accessi	Tipo
Operatore	E	1	L
Assegnazione	R	368 (920.000/500 x 1/5)	L
Ticket	E	368 (920.000/500 x 1/5)	L

Totale: 737L → 2.211.000 al giorno (737 x 3.000 al giorno)

OP3 - Registrare un nuovo cliente

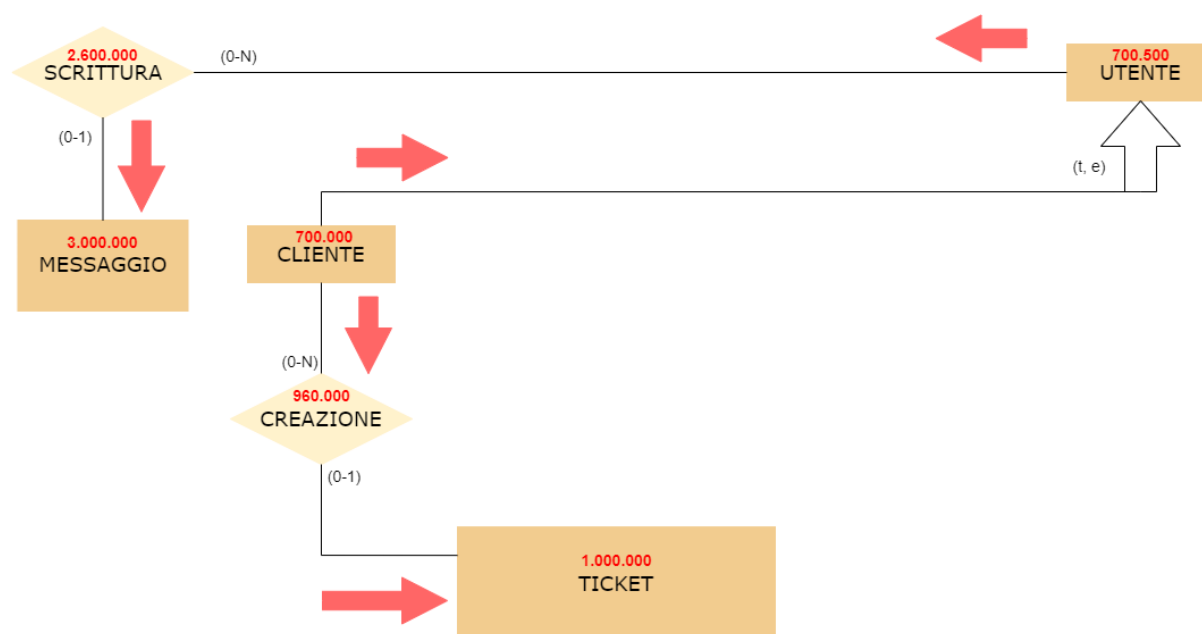
Per creare un nuovo cliente, prima si effettua un controllo per verificare se l'indirizzo email inserito è già presente nel sistema.

Successivamente, una volta verificato che l'email non esiste già, si procede aggiungendo il cliente come nuovo utente e lo si associa al gruppo dei clienti.

Concetto	Costrutto	Accessi	Tipo
Utente	E	1	L
Utente	E	1	S
Cliente	E	1	S

Totale: 1L + 2S → 10.000 al giorno (5 x 2.000 al giorno)

OP4 - Eliminare un cliente



Concetto	Costrutto	Accessi	Tipo
Cliente	E	1	S
Utente	E	1	S
Scrittura	R	3,7 (2.600.000/700.500)	S
Messaggio	E	3,7 (2.600.000/700.500)	S
Creazione	R	1,4 (960.000/700.000)	S
Ticket	E	1,4 (960.000/700.000)	S

Totale: 12,2S → 2.440 al giorno (24,4 x 100 al giorno)

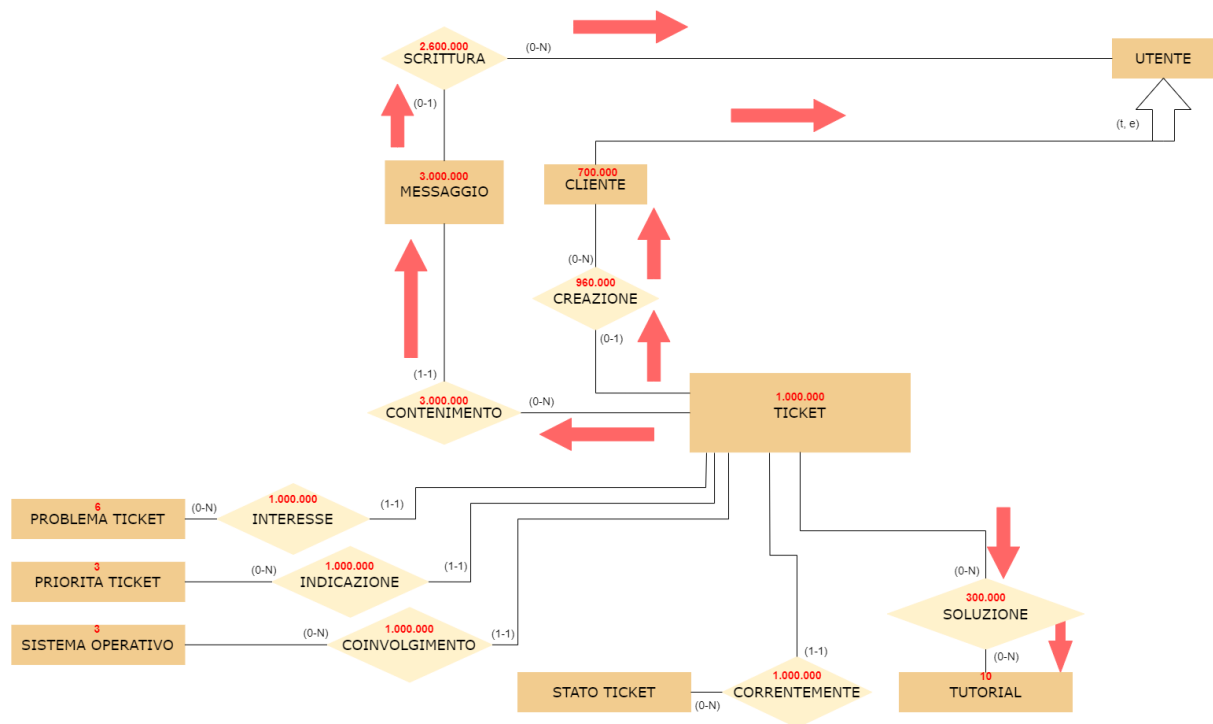
OP5 - Autenticare l'accesso ad un utente

Per autenticare l'utente, sarà sufficiente effettuare un controllo per verificare se l'email e la password inserite corrispondono a quelle di un utente registrato.

Concetto	Costrutto	Accessi	Tipo
Utente	E	1	L

Totale: 1L → 4.600 al giorno (1 x 4.600 al giorno)

OP6 - Visualizzare i dettagli di un ticket e i suoi messaggi



Alla lettura del cliente si fa un accesso in lettura anche all'utente poiché il suo nome e cognome (dati che prendo per mostrare i dettagli) sono solo all'interno di utente.

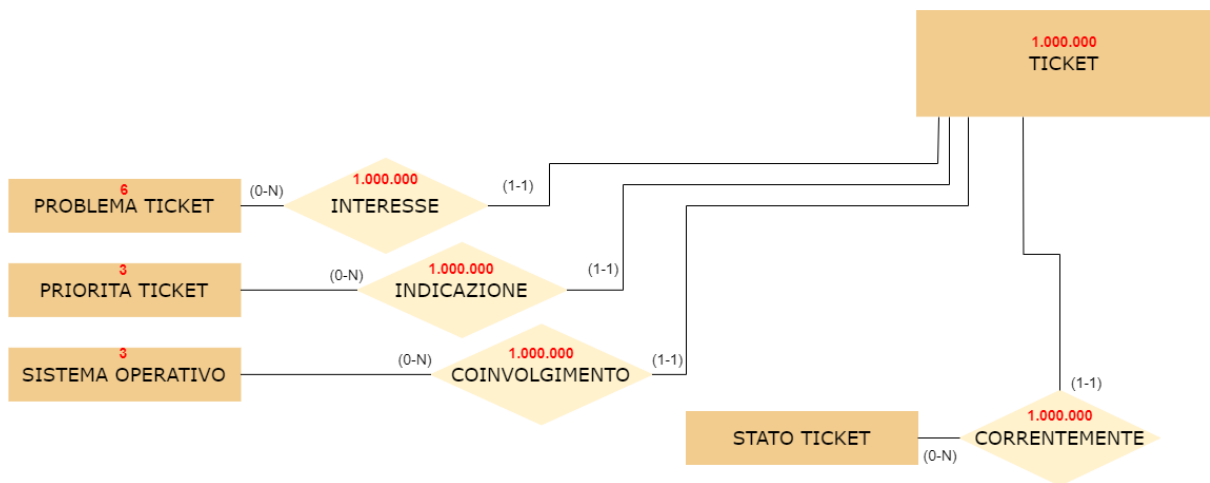
"Problema Ticket", "Priorita Ticket", "Sistema Operativo" e "Stato Ticket" hanno solo un campo che è la chiave primaria "nome" essendo che è contenuto in "Ticket" non richiederanno accesso.

Concetto	Costrutto	Accessi	Tipo
Ticket	E	1	L
Creazione	R	0,96 (960.000/1.000.000)	L
Cliente	E	0,96 (960.000/1.000.000)	L
Utente	E	0,96 (960.000/1.000.000)	L
Contenimento	R	3 (3.000.000 / 1.000.000)	L
Messaggio	E	3 (3.000.000 / 1.000.000)	L
Scrittura	R	2,6 (2.600.000 / 3.000.000 x 3)	L

Concetto	Costrutto	Accessi	Tipo
Utente	E	2,6 (2.600.000 / 3.000.000 x 3)	L
Soluzione	R	0,3 (300.000 / 1.000.000)	L
Tutorial	E	0,3 (300.000 / 1.000.000)	L

Totale: 15,68L → 169.344 al giorno (15,68 x 10.800 al giorno)

OP7 - Visualizzare i ticket



“Problema Ticket”, “Priorita Ticket”, “Sistema Operativo” e “Stato Ticket” hanno solo un campo che è la chiave primaria “nome” essendo che è contenuto in “Ticket” non richiederanno accesso.

Concetto	Costrutto	Accessi	Tipo
Ticket	E	1.000.000	L

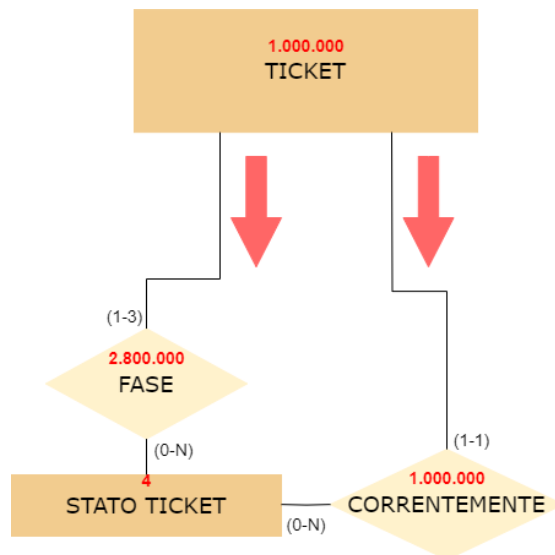
Totale: 1.000.000L → 18.600.000.000 al giorno (1.000.000 x 18.600 al giorno)

OP8 - Scrittura di un messaggio

Concetto	Costrutto	Accessi	Tipo
Messaggio	E	1	S
Scrittura	R	1	S
Contenimento	R	1	S

Totale: 3S → 54.000 al giorno (6 x 9.000 al giorno)

OP9 - Cambiare stato di un ticket



Concetto	Costrutto	Accessi	Tipo
Ticket	E	1	S
Fase	R	1	S
Correntemente	R	1	S

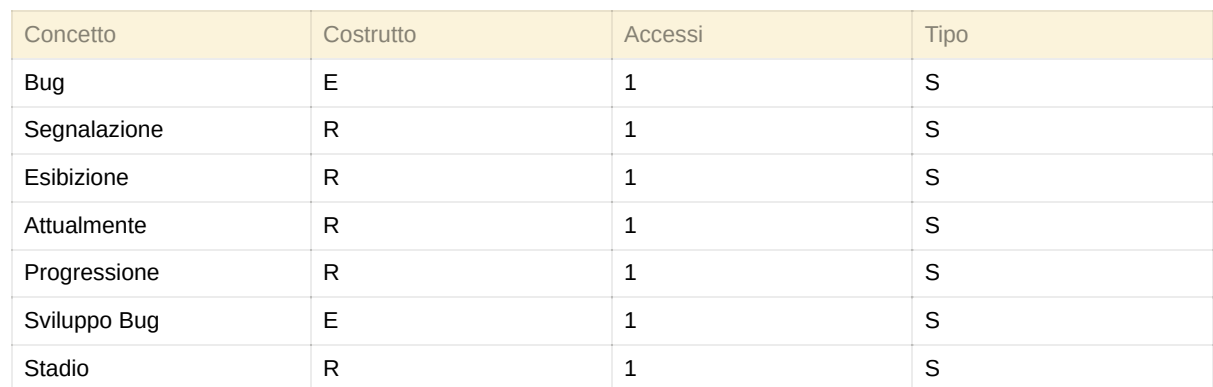
Totale: 3S → 50.400 al giorno (6 x 8.400 al giorno)

OP10 - Associazione dei tutorial ad un ticket

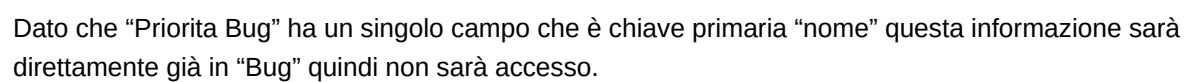
Concetto	Costrutto	Accessi	Tipo
Soluzione	R	0,3 (300.000/1.000.000)	S

Totale: 1S → 1.800 al giorno (2 x 900 al giorno)

OP11 - Segnalazione di un bug



OP12 - Visualizzare i bug



Dato che “Stato Bug” ha un singolo campo che è chiave primaria “nome” questa informazione sarà direttamente già in “Bug” quindi non sarà accesso.

Concetto	Costrutto	Accessi	Tipo
Bug	E	36.500	L

Totale: 36.500L → 3.650.000 al giorno (36500 x 100 al giorno)

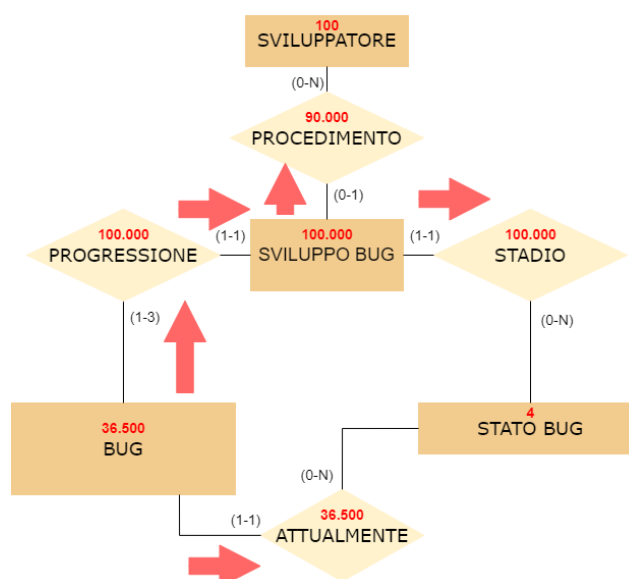
OP13 - Visualizzare i dettagli di un bug

Equivalente a visualizzare i bug nella **OP12** ma solo per 1 bug.

Concetto	Costrutto	Accessi	Tipo
Bug	E	1	L

Totale: 1L → 100 al giorno (1 x 100 al giorno)

OP14 - Cambiare stato di un bug



Concetto	Costrutto	Accessi	Tipo
Bug	E	1	S
Attualmente	R	1	S
Progressione	R	1	S
Sviluppo Bug	E	1	S
Stadio	R	1	S
Procedimento	R	1	S

Totale: 6S → 6.600 al giorno (12 x 550 al giorno)

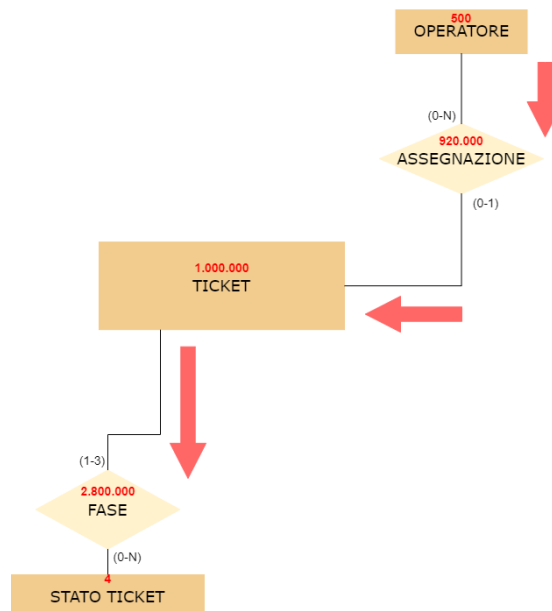
OP15 - Visualizzare le statistiche dei ticket in un dato periodo

Gli accessi a "Fase" sono il doppio dei "Ticket" che sono chiusi poiché si accede alla prima e ultima "Fase" del "Ticket" per calcolarne il tempo trascorso.

Concetto	Costrutto	Accessi	Tipo
Ticket	E	1.000.000	L
Fase	R	1.600.000 (1.000.000 x 4/5 x 2)	L

Totale: 2.600.000L → 2.600.000 a settimana (2.600.000 x 1 a settimana)

OP16 - Visualizzare prestazioni degli operatori



Le prestazioni degli operatori vengono valutate in base ai ticket chiusi, e il numero di ticket chiusi è pari al 4/5 del totale dei ticket gestiti dall'operatore. Pertanto, per valutare le prestazioni di un operatore, è sufficiente accedere alle fasi finali e iniziali (per vedere durata totale) dei ticket che sono stati chiusi.

Concetto	Costrutto	Accessi	Tipo
Operatore	E	500	L
Assegnazione	R	736.000 (500x920.000/500 x 4/5)	L
Ticket	E	736.000 (500x920.000/500 x 4/5)	L
Fase	R	1.472.000 (500x920.000/500 x 4/5 x 2)	L

Totale: 2.944.500L → 2.944.500 a settimana (2.944.500 x 1 a settimana)

OP17 - Creazione di un operatore

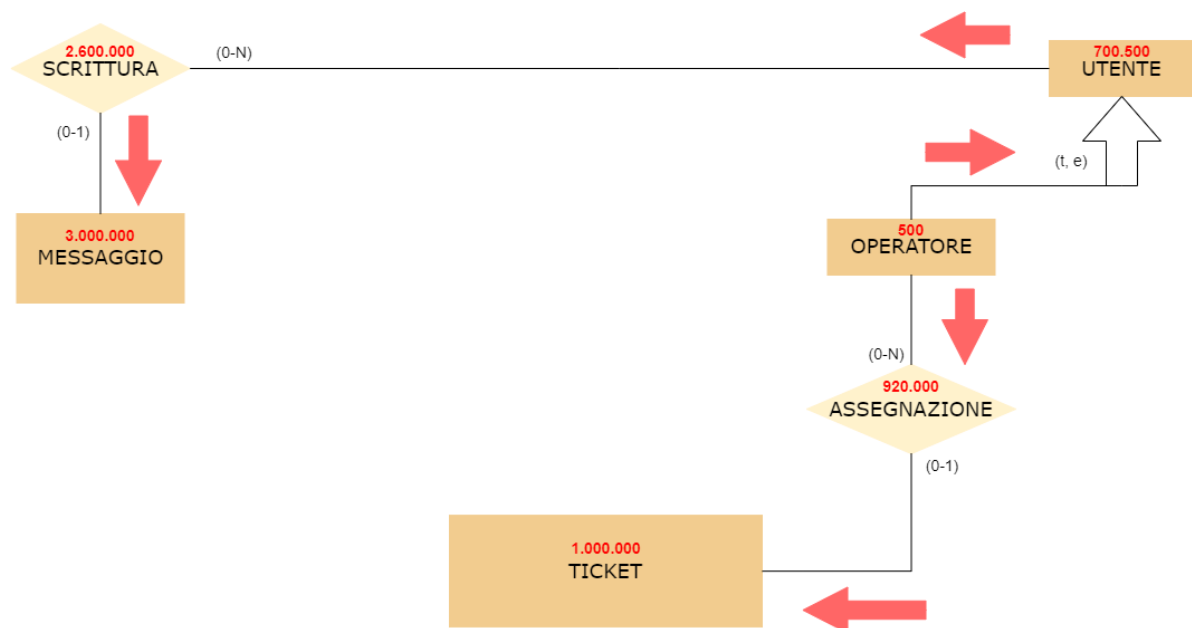
Per creare un nuovo operatore, prima si effettua un controllo per verificare se l'indirizzo email inserito è già presente nel sistema.

Successivamente, una volta verificato che l'email non esiste già, si procede aggiungendo il operatore come nuovo utente e lo si associa al gruppo dei operatori.

Concetto	Costrutto	Accessi	Tipo
Utente	E	1	L
Utente	E	1	S
Operatore	E	1	S

Totale: 1L + 2S → 250 all'anno (5 x 50 all'anno)

OP18 - Eliminazione di un operatore



Concetto	Costrutto	Accessi	Tipo
Operatore	E	1	S
Utente	E	1	S
Scrittura	R	3,7 (2.600.000/700.500)	S
Messaggio	E	3,7 (2.600.000/700.500)	S
Assegnazione	R	1.840 (920.000/500)	S
Ticket	E	1.840 (920.000/500)	S

Totale: 3.689,4S → 368.940 all'anno (7.378,8 x 50 all'anno)

OP19 - Creazione di un sviluppatore

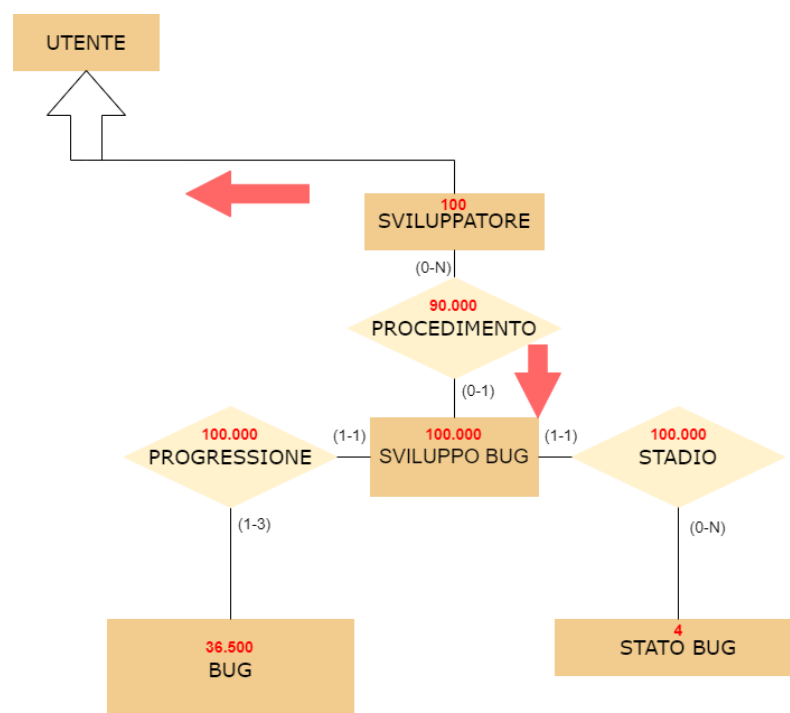
Per creare un nuovo sviluppatore, prima si effettua un controllo per verificare se l'indirizzo email inserito è già presente nel sistema.

Successivamente, una volta verificato che l'email non esiste già, si procede aggiungendo il sviluppatore come nuovo utente e lo si associa al gruppo dei sviluppatori.

Concetto	Costrutto	Accessi	Tipo
Utente	E	1	L
Utente	E	1	S
Sviluppatore	E	1	S

Totale: 1L + 2S → 50 all'anno (5 x 10 all'anno)

OP20 - Eliminazione di un sviluppatore



Concetto	Costrutto	Accessi	Tipo
Sviluppatore	E	1	S
Utente	E	1	S
Procedimento	R	900 (90.000/100)	S
Sviluppo Bug	E	900 (90.000/100)	S

Totale: 1.802S → 36.040 all'anno (3.604 x 10 all'anno)

OP21 - Visualizzare tutti gli operatori

Concetto	Costrutto	Accessi	Tipo
Operatore	E	500	L
Utente	E	500	L

Totale: 1.000L → 1.000 a settimana (1.000 x 1 a settimana)

OP22 - Visualizzare tutti gli sviluppatori

Concetto	Costrutto	Accessi	Tipo
Sviluppatore	E	100	L
Utente	E	100	L

Totale: 200L → 200 al mese (200 x 1 al mese)

OP23 - Valutazione Operatore

Concetto	Costrutto	Accessi	Tipo
Ticket	E	1	S

Totale: 1S → 4.800 al giorno (2 x 2.400 al giorno)

Analisi delle ridondanze

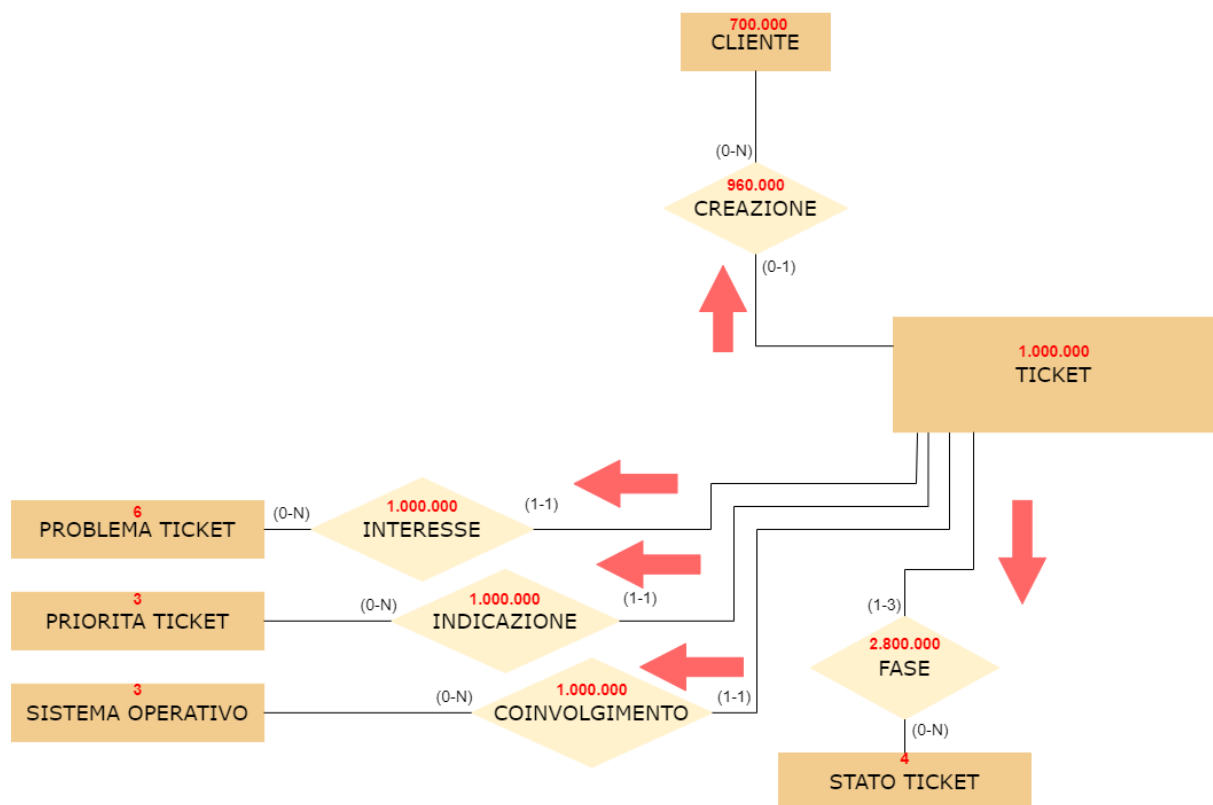
Sono state introdotte alcune ridondanze nel sistema al fine di ridurre il numero complessivo di accessi alle tabelle "Fase" e "Sviluppo Bug". Con questa strategia, quando un "Ticket" o un "Bug" desidera accedere al proprio stato corrente, può farlo direttamente tramite l'informazione già presente nella rispettiva entità.

Questo approccio aumenta il numero di accessi in alcune parti del sistema, ma allo stesso tempo riduce il numero complessivo di accessi nelle tabelle "Fase" e "Sviluppo Bug" durante la lettura dei dettagli di un "Ticket" o "Bug".

È importante valutare attentamente il trade-off tra gli accessi aggiuntivi introdotti dalle ridondanze e i risparmi ottenuti nei punti critici del sistema. Se il guadagno complessivo di prestazioni supera la perdita dovuta alle ridondanze, allora questa scelta è giustificata e può portare a un sistema più efficiente ed ottimizzato.

OP1 - Creare un nuovo ticket

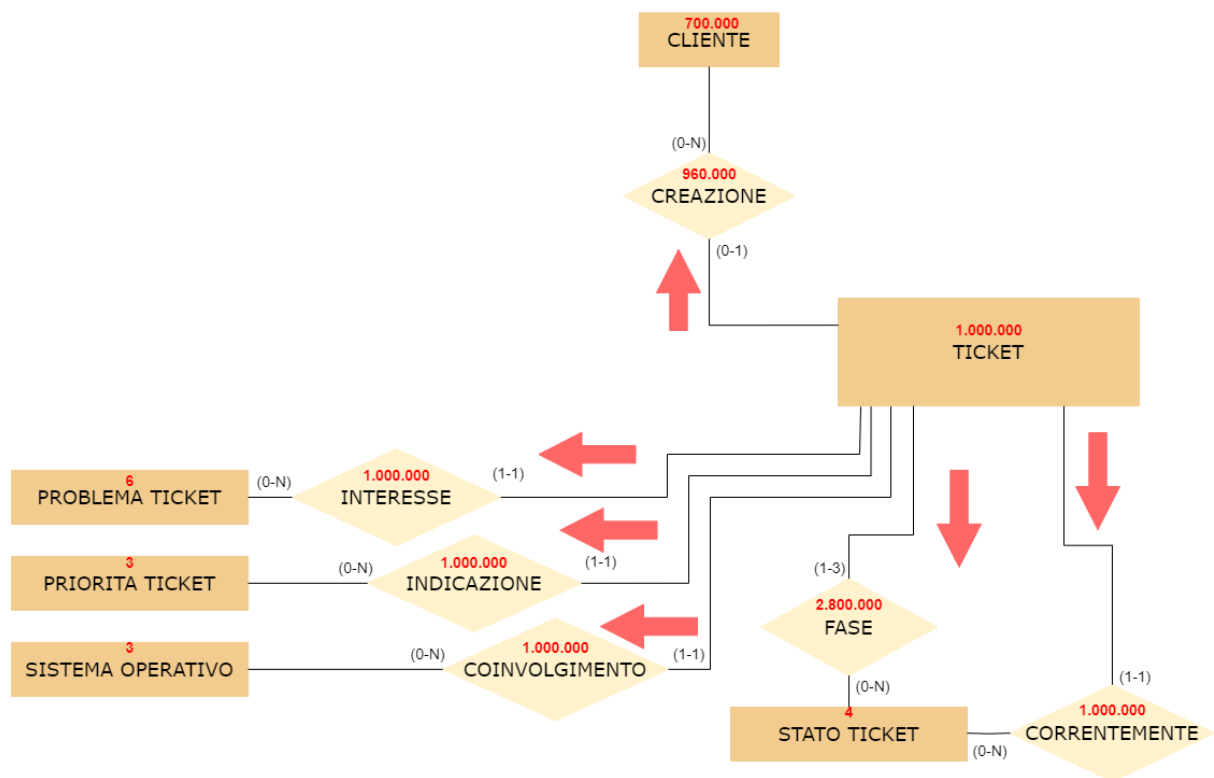
Senza ridondanza



Concetto	Costrutto	Accessi	Tipo
Ticket	E	1	S
Creazione	R	1	S
Fase	R	1	S
Interesse	R	1	S
Indicazione	R	1	S
Coinvolgimento	R	1	S

Totale: 6S → 36.000 al giorno (12 x 3.000 al giorno)

Con ridondanza

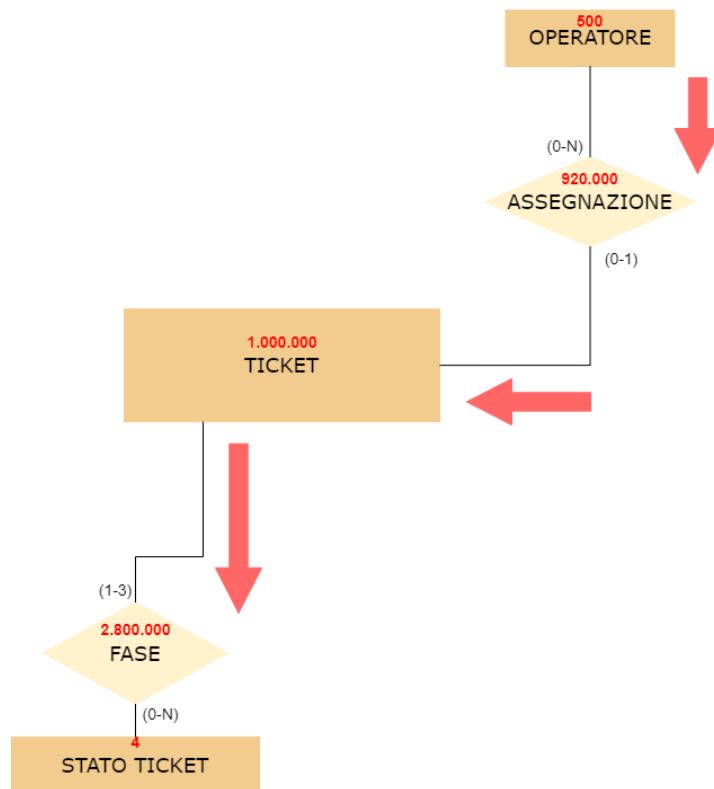


Concetto	Costrutto	Accessi	Tipo
Ticket	E	1	S
Creazione	R	1	S
Fase	R	1	S
Correntemente	R	1	S
Interesse	R	1	S
Indicazione	R	1	S
Coinvolgimento	R	1	S

Totale: 7S → 42.000 al giorno (14 x 3.000 al giorno)

OP2 - Cercare all'operatore più disponibile da poi assegnare ad un ticket

Senza ridondanza



Si leggono le assegnazioni attuali degli operatori per ottenere i ticket che sono loro stati assegnati. Successivamente, si leggono le ultime fasi dei ticket e si ricavano solo quelli che sono ancora aperti o in fase di analisi.

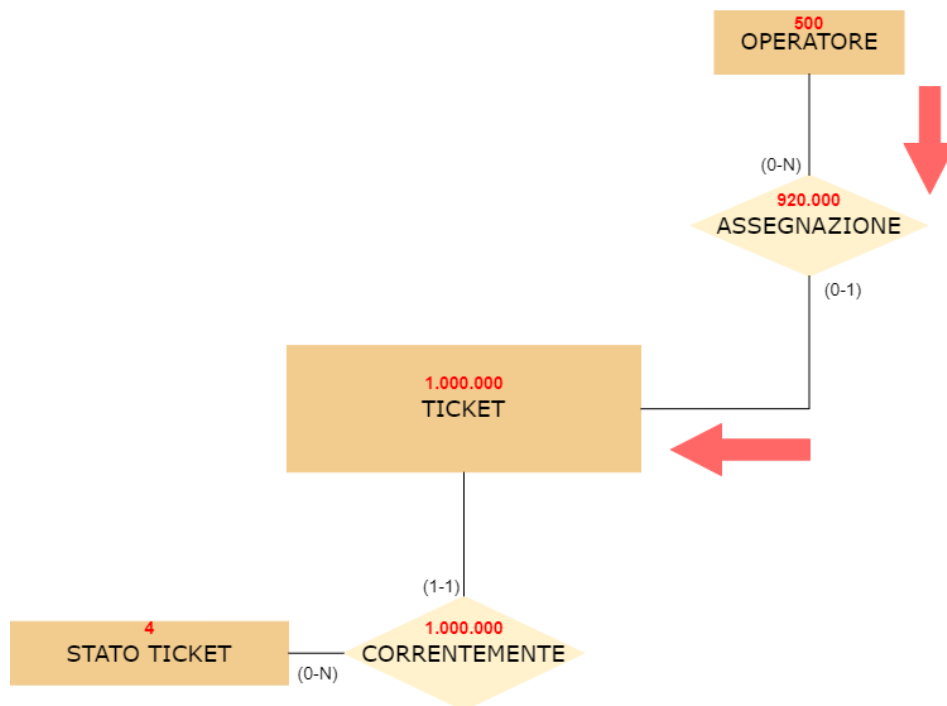
Poiché solo 1/5 dei ticket non è ancora stato chiuso e accedo solo ai ticket ancora aperti, il numero di ticket a cui accedo è equivalente a 1/5 di quelli assegnati all'operatore.

Dato che si legge solo l'ultima "Fase" dei "Ticket" ancora aperti gli accessi equivalgono a quelli dei "Ticket".

Concetto	Costrutto	Accessi	Tipo
Operatore	E	1	L
Assegnazione	R	368 $(920.000/500 \times 1/5)$	L
Ticket	E	368 $(920.000/500 \times 1/5)$	L
Fase	R	368 $(920.000/500 \times 1/5)$	L

Totale: 1105L → 3.315.000 al giorno (1105 x 3.000 al giorno)

Con ridondanza

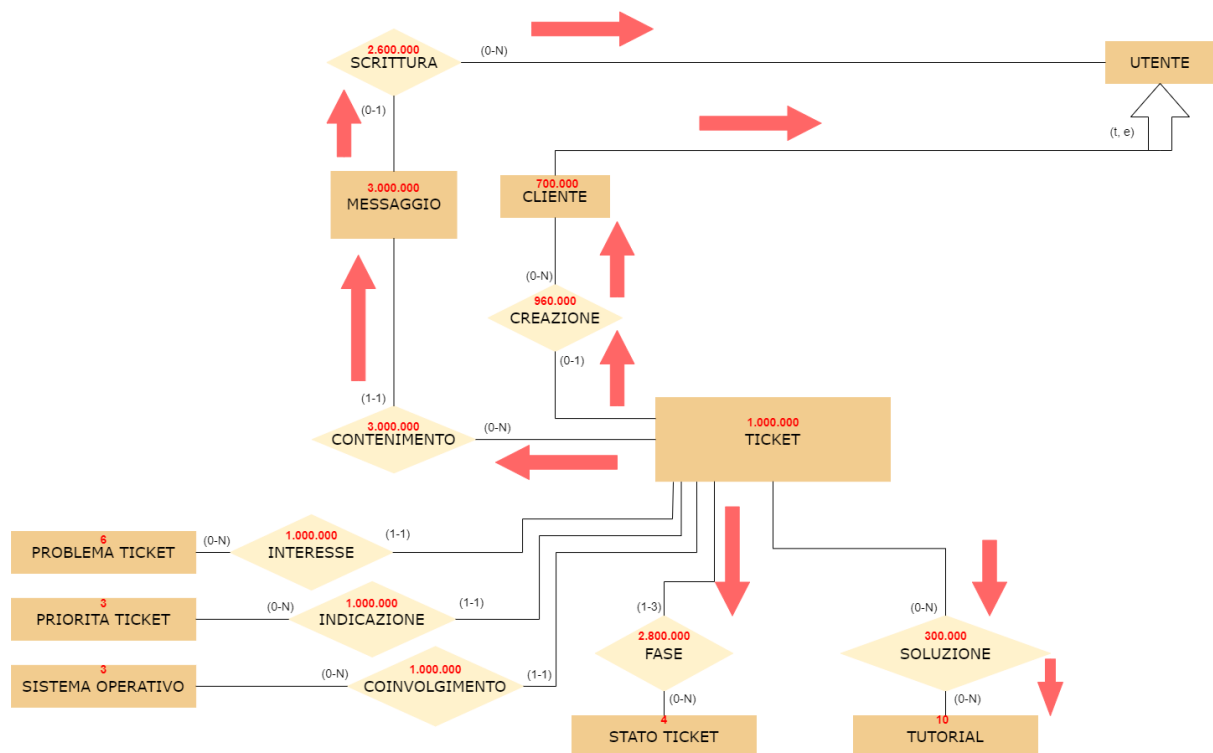


Concetto	Costrutto	Accessi	Tipo
Operatore	E	1	L
Assegnazione	R	368 (920.000/500 x 1/5)	L
Ticket	E	368 (920.000/500 x 1/5)	L

Totale: 737L → 2.211.000 al giorno (737 x 3.000 al giorno)

OP6 - Visualizzare i dettagli di un ticket e i suoi messaggi

Senza ridondanza



Alla lettura del cliente si fa un accesso in lettura anche all'utente poiché il suo nome e cognome (dati che prendo per mostrare i dettagli) sono solo all'interno di utente.

Inoltre, poiché leggiamo solo l'ultima fase del ticket per ricavarne lo stato corrente, avremo un solo accesso alla tabella "Fase". Questo perché la tabella "Stato Ticket" ha come unico campo e chiave primaria il suo nome, e tale informazione è già contenuta all'interno della tabella "Fase". Pertanto, non è necessario fare un accesso aggiuntivo alla tabella "Stato Ticket" per ottenere lo stato corrente del ticket.

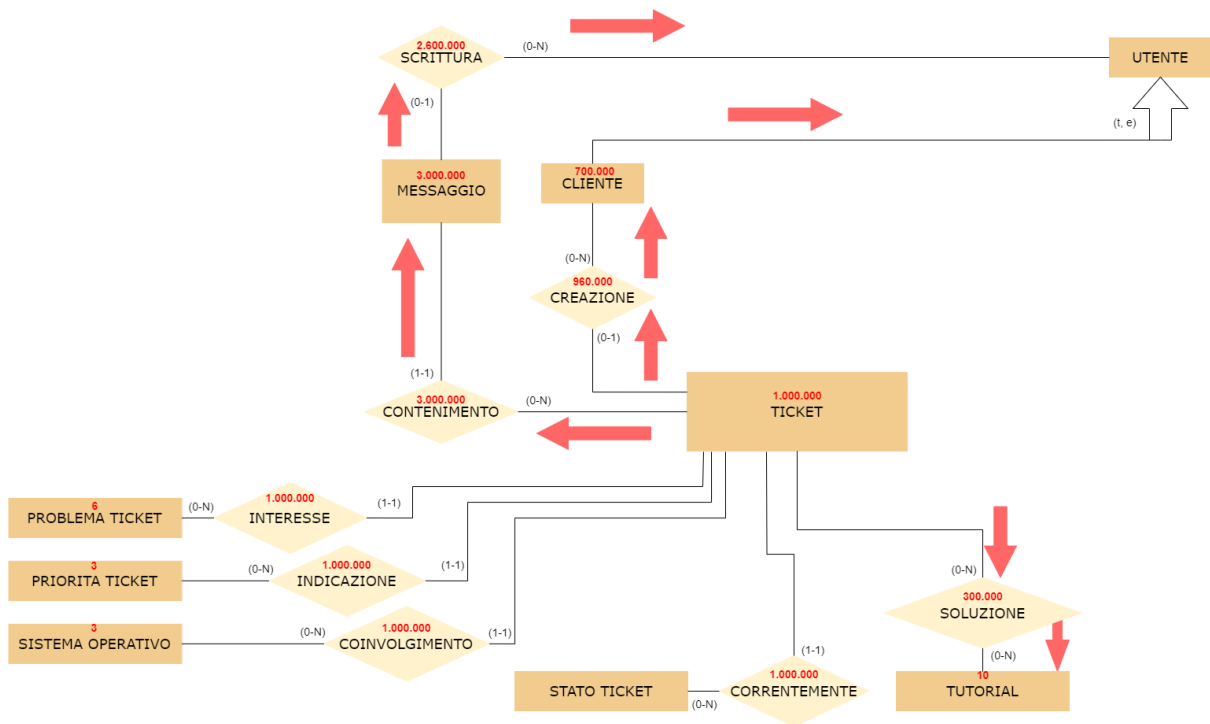
"Problema Ticket", "Priorita Ticket" e "Sistema Operativo" hanno solo un campo che è la chiave primaria "nome" essendo che è contenuto in "Ticket" non richiederanno accesso.

Concetto	Costrutto	Accessi	Tipo
Ticket	E	1	L
Creazione	R	0,96 (960.000/1.000.000)	L
Cliente	E	0,96 (960.000/1.000.000)	L
Utente	E	0,96 (960.000/1.000.000)	L
Contenimento	R	3 (3.000.000 / 1.000.000)	L
Messaggio	E	3 (3.000.000 / 1.000.000)	L
Scrittura	R	2,6 (2.600.000 / 3.000.000 x 3)	L
Utente	E	2,6 (2.600.000 / 3.000.000 x 3)	L
Fase	R	1 (si legge solo l'ultima fase del ticket)	L
Soluzione	R	0,3 (300.000 / 1.000.000)	L

Concetto	Costrutto	Accessi	Tipo
Tutorial	E	0,3 (300.000 / 1.000.000)	L

Totale: 16,68L → 180.144 al giorno (16,68 x 10.800 al giorno)

Con ridondanza

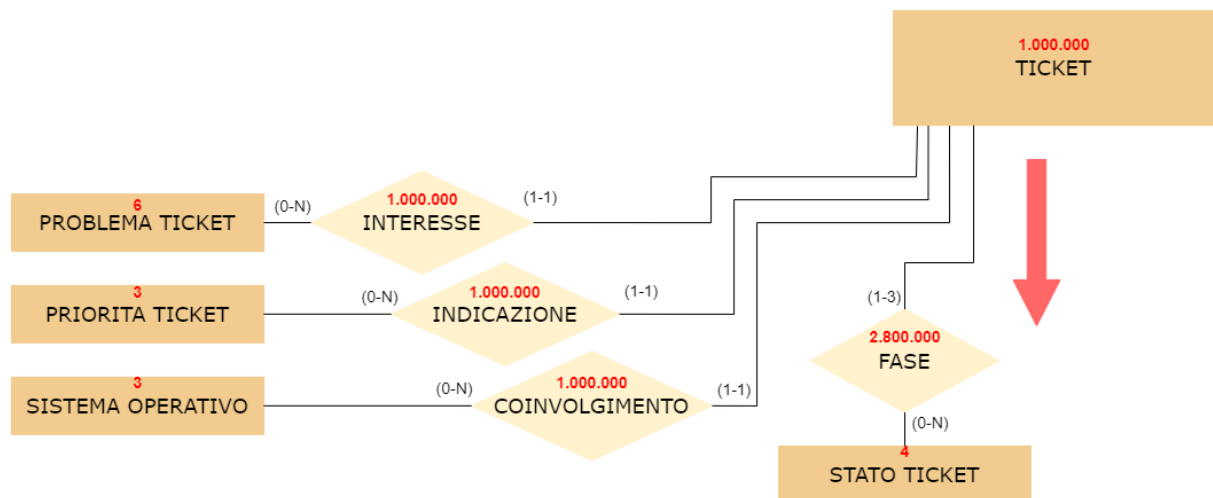


Concetto	Costrutto	Accessi	Tipo
Ticket	E	1	L
Creazione	R	0,96 (960.000/1.000.000)	L
Cliente	E	0,96 (960.000/1.000.000)	L
Utente	E	0,96 (960.000/1.000.000)	L
Contenimento	R	3 (3.000.000 / 1.000.000)	L
Messaggio	E	3 (3.000.000 / 1.000.000)	L
Scrittura	R	2,6 (2.600.000 / 3.000.000 x 3)	L
Utente	E	2,6 (2.600.000 / 3.000.000 x 3)	L
Soluzione	R	0,3 (300.000 / 1.000.000)	L
Tutorial	E	0,3 (300.000 / 1.000.000)	L

Totale: 15,68L → 169.344 al giorno (15,68 x 10.800 al giorno)

OP7 - Visualizzare i ticket

Senza ridondanza



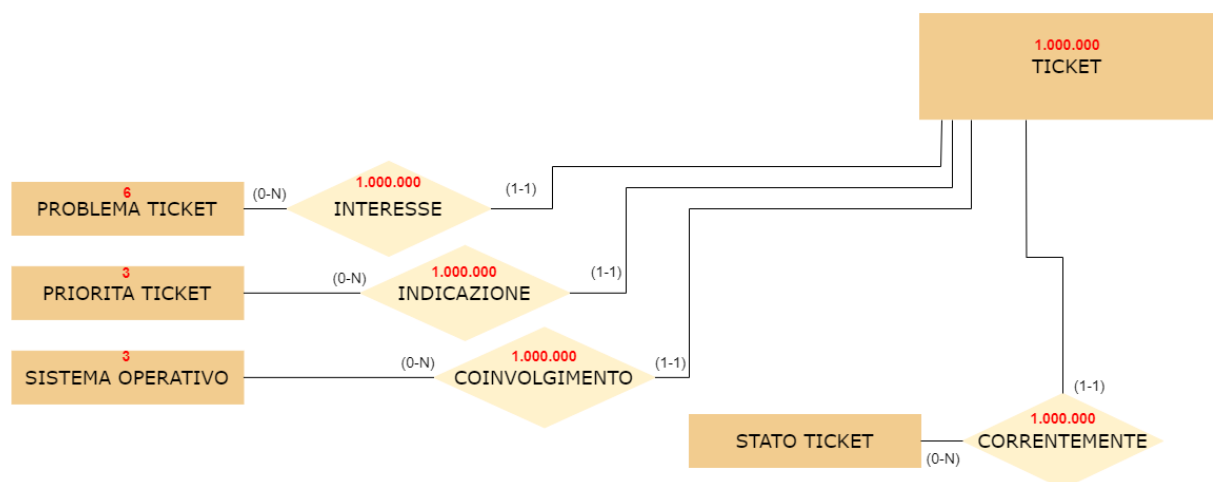
“Problema Ticket”, “Priorita Ticket” e “Sistema Operativo” hanno tutti come unico campo e chiave primaria il “nome”, quindi queste informazioni saranno già presenti nel “Ticket”.

Per ottenere lo stato corrente del “Ticket”, basterà fare un accesso in lettura all'ultima “Fase” del ticket. Poiché anche la tabella “Stato Ticket” ha come unico campo e chiave primaria il “nome”, queste informazioni saranno contenute all'interno della tabella “Fase”.

Concetto	Costrutto	Accessi	Tipo
Ticket	E	1.000.000	L
Fase	R	1.000.000 (poiché si legge solo l'ultima fase)	L

Totale: 2.000.000L → 37.200.000.000 al giorno (2.000.000 x 18.600 al giorno)

Con ridondanza



Concetto	Costrutto	Accessi	Tipo
Ticket	E	1.000.000	L

Totale: 1.000.000L → 18.600.000.000 al giorno (1.000.000 x 18.600 al giorno)

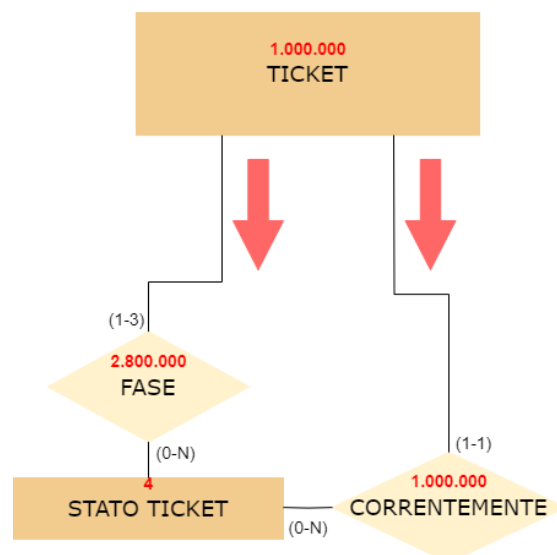
OP9 - Cambiare stato di un ticket

Senza ridondanza

Concetto	Costrutto	Accessi	Tipo
Fase	R	1	S

Totale: 1S → 16.800 al giorno (2 x 8.400 al giorno)

Con ridondanza



Concetto	Costrutto	Accessi	Tipo
Ticket	E	1	S
Fase	R	1	S
Correntemente	R	1	S

Totale: 3S → 50.400 al giorno (6 x 8.400 al giorno)

COSTO TOTALE TICKET

Senza ridondanza

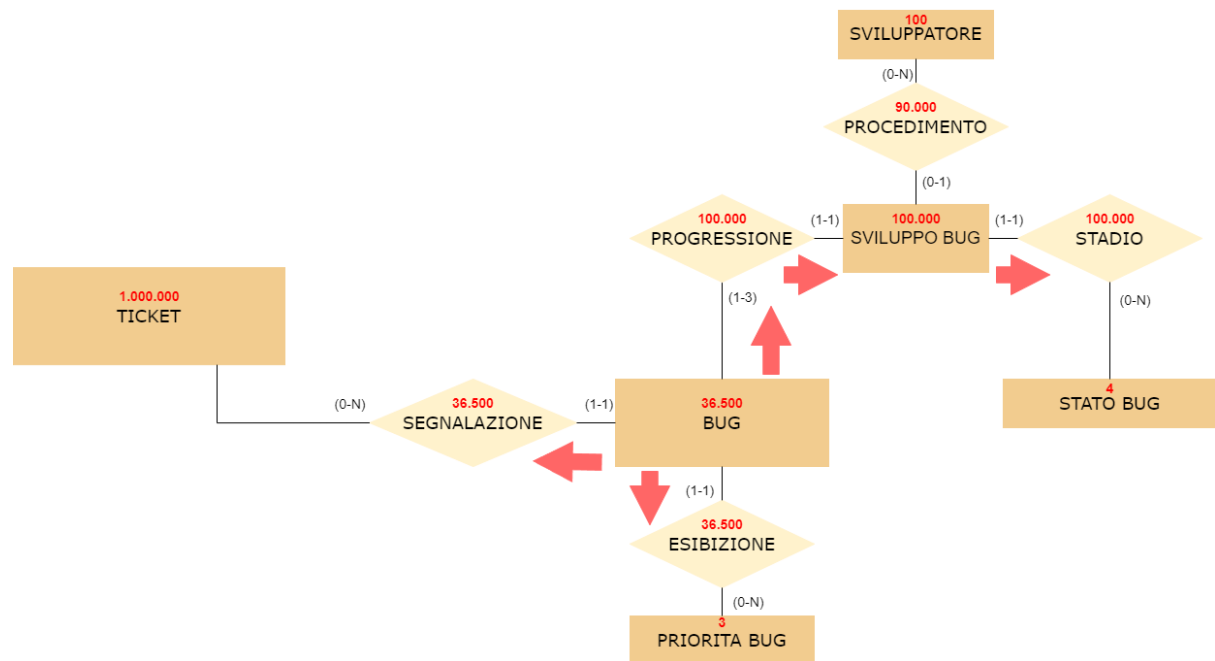
Totale: 37.203.547.940 al giorno

Con ridondanza

Totale: 18.602.472.740 al giorno

OP11 - Segnalazione di un bug

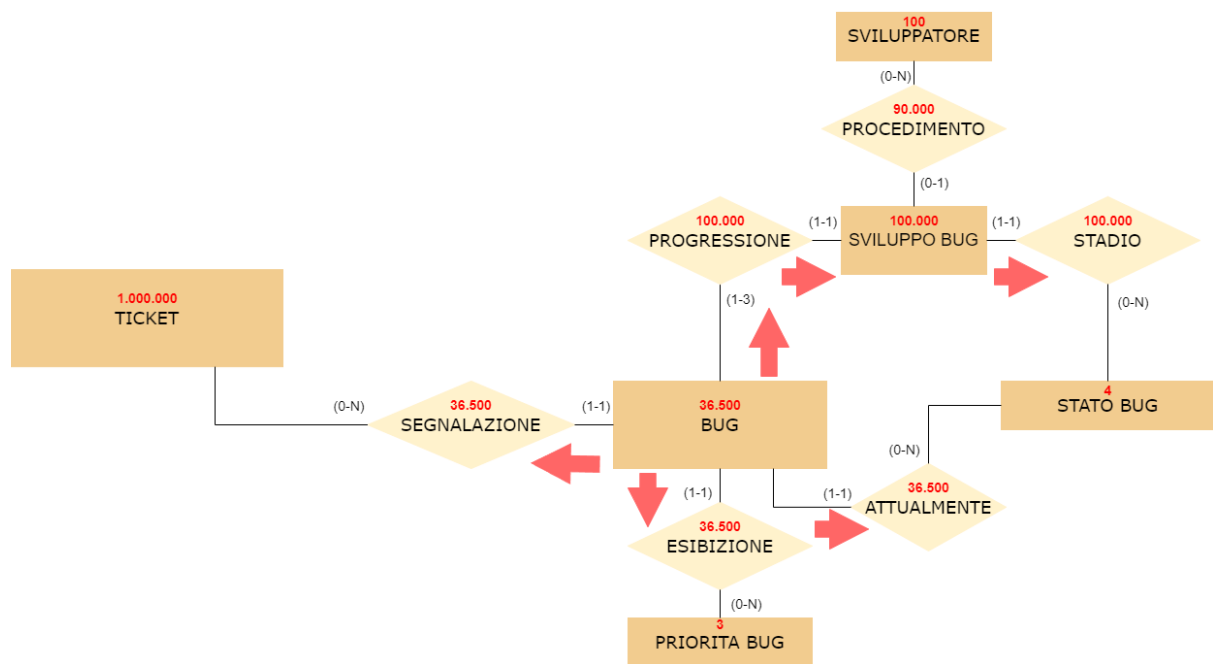
Senza ridondanza



Concetto	Costrutto	Accessi	Tipo
Bug	E	1	S
Segnalazione	R	1	S
Esibizione	R	1	S
Progressione	R	1	S
Sviluppo Bug	E	1	S
Stadio	R	1	S

Totale: 6S → 1.200 al giorno (12 x 100 al giorno)

Con ridondanza

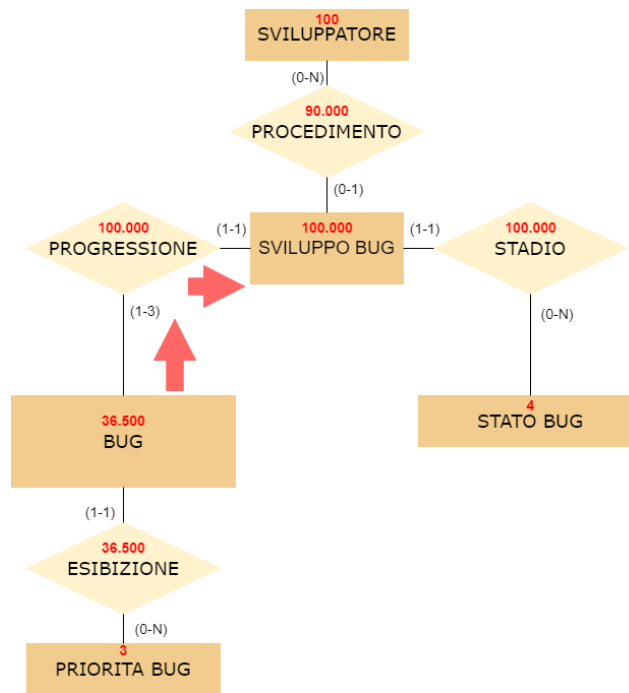


Concetto	Costrutto	Accessi	Tipo
Bug	E	1	S
Segnalazione	R	1	S
Esibizione	R	1	S
Attualmente	R	1	S
Progressione	R	1	S
Sviluppo Bug	E	1	S
Stadio	R	1	S

Totale: 7S → 1.400 al giorno (14 x 100 al giorno)

OP12 - Visualizzare i bug

Senza ridondanza



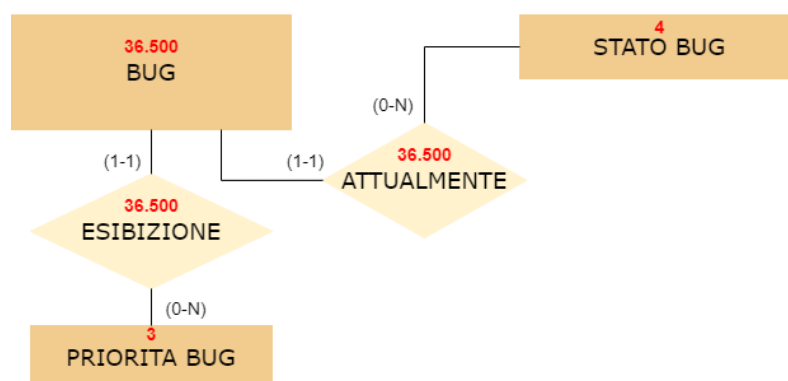
Dato che “Priorita Bug” ha un singolo campo che è chiave primaria “nome” questa informazione sarà direttamente già in “Bug” quindi non sarà accesso.

Dato che “Stato Bug” ha un singolo campo che è chiave primaria “nome” questa informazione sarà direttamente già in “Sviluppo Bug” quindi non sarà accesso.

Concetto	Costrutto	Accessi	Tipo
Bug	E	36.500	L
Progressione	R	36.500 (poiché si legge solo l'ultimo sviluppo)	L
Sviluppo Bug	E	36.500 (poiché si legge solo l'ultimo sviluppo)	L

Totale: 109.500L → 10.950.000 al giorno (109.500 x 100 al giorno)

Con ridondanza



Concetto	Costrutto	Accessi	Tipo
Bug	E	36.500	L

Totale: 36.500L → 3.650.000 al giorno (36.500 x 100 al giorno)

OP13 - Visualizzare i dettagli di un bug

Senza ridondanza

Equivalente a visualizzare i bug nella **OP12** ma solo per 1 bug.

Concetto	Costrutto	Accessi	Tipo
Bug	E	1	L
Progressione	R	1 (poiché si legge solo l'ultimo sviluppo)	L
Sviluppo Bug	E	1 (poiché si legge solo l'ultimo sviluppo)	L

Totale: 3L → 300 al giorno (3 x 100 al giorno)

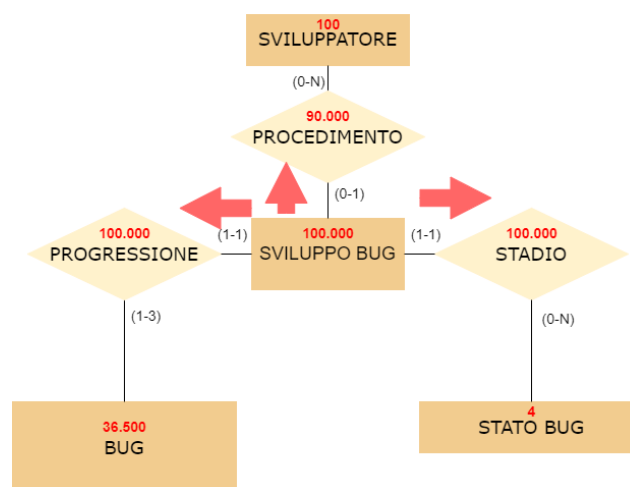
Con ridondanza

Concetto	Costrutto	Accessi	Tipo
Bug	E	1	L

Totale: 1L → 100 al giorno (1 x 100 al giorno)

OP14 - Cambiare stato di un bug

Senza ridondanza

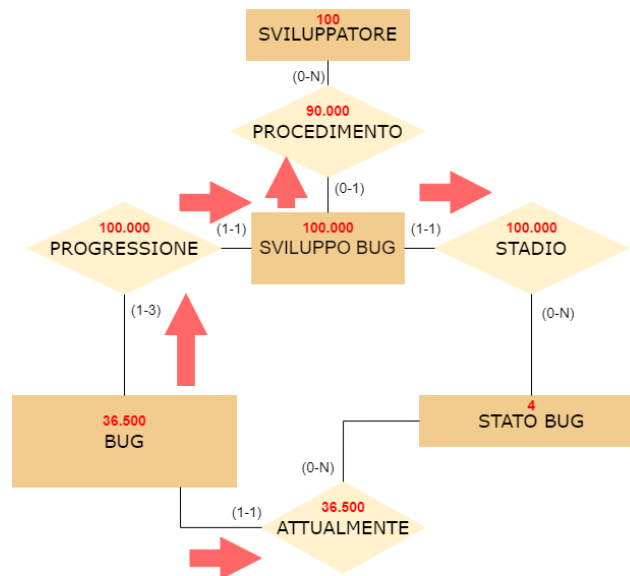


Concetto	Costrutto	Accessi	Tipo
----------	-----------	---------	------

Concetto	Costrutto	Accessi	Tipo
Sviluppo Bug	E	1	S
Progressione	R	1	S
Stadio	R	1	S
Procedimento	R	1	S

Totale: 4S → 4.400 al giorno (8 x 550 al giorno)

Con ridondanza



Concetto	Costrutto	Accessi	Tipo
Bug	E	1	S
Attualmente	R	1	S
Progressione	R	1	S
Sviluppo Bug	E	1	S
Stadio	R	1	S
Procedimento	R	1	S

Totale: 6S → 6.600 al giorno (12 x 550 al giorno)

COSTO TOTALE BUG

Senza ridondanza

Totale: 10.955.900 al giorno

Con ridondanza

Totale: 3.658.100 al giorno

L'aggiunta di queste due ridondanze si è dimostrata essere una scelta positiva e vantaggiosa per il sistema nel complesso.

Le ridondanze hanno permesso di ridurre significativamente il numero di accessi alle tabelle "Fase" e "Sviluppo Bug" durante la lettura dei dettagli dei ticket e dei bug. Questo ha portato a una drastica riduzione dei costi complessivi del sistema, come evidenziato nel calcolo dei costi giornalieri.

Raffinamento dello schema

Gestione delle gerarchie

Nello schema E/R è presente una gerarchia in cui l'entità "Utente" funge da entità padre per "Sviluppatore", "Amministratore", "Cliente" e "Operatore".

La scelta di mantenere tutte le entità separate è stata fatta perché gli accessi e le operazioni effettuate sull'entità "Utente" possono differire da quelli dei figli, ossia "Sviluppatore", "Amministratore", "Cliente" e "Operatore". Mantenendo le entità separate, è possibile gestire in modo più flessibile e preciso i privilegi di accesso e le operazioni consentite per ciascun tipo di utente.

Eliminazione degli attributi composti

Nello schema E/R è presente l'attributo composto "Valutazione Operatore". Invece di inserire direttamente gli attributi della valutazione all'interno dell'entità "Ticket", si è optato per creare una tabella separata con chiave primaria "NumeroTicket" per gestire queste informazioni. Questa scelta è stata fatta per migliorare la struttura del database e rendere più efficiente la gestione delle valutazioni degli operatori. Utilizzando una tabella dedicata, è possibile gestire più facilmente le valutazioni correlate a ciascun ticket e mantenere un design del database più ordinato e coerente.

Scelta delle chiavi primarie

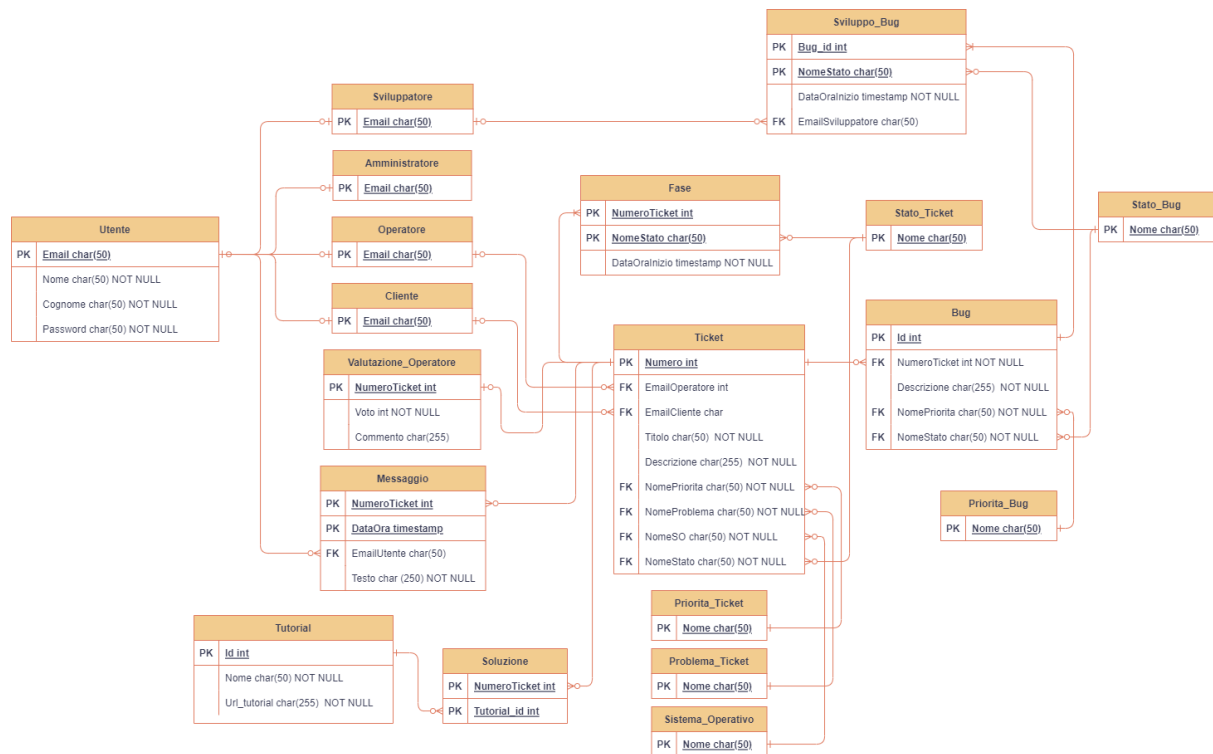
Nell'attuale schema E/R, sono state mantenute tutte le chiavi primarie mostrate, ad eccezione di "Tutorial". Poiché il campo "URL" potrebbe raggiungere lunghezze significative, rallentando le query e risultando scomodo da utilizzare, si è optato per aggiungere un campo "Id" come chiave primaria per l'entità "Tutorial".

Chiavi esterne

Nel sistema, dove necessario per garantire la semplicità e la chiarezza delle relazioni tra le tabelle, vengono utilizzate chiavi esterne con il nome "chiave primaria + nome entità" o "nome entità + chiave primaria" a cui fanno riferimento. Questo formato aiuta a identificare facilmente la relazione tra le tabelle e a comprenderne la struttura senza ambiguità.

Tuttavia, in alcuni casi specifici si è utilizzato direttamente il nome della chiave primaria della tabella di riferimento, nel nostro caso nell'ereditarietà.

Schema relazionale finale



Traduzione delle operazioni in query

OP1 - Creare un nuovo ticket

Creazione del ticket inserendo l'operatore più disponibile

INSERT INTO Ticket (email_operatore, email_cliente, titolo, descrizione, nome_priorita, nome_problema, nome_so, nome_stato) **VALUES** (?, ?, ?, ?, ?, ?, ?, 'aperto')

Creazione della fase del ticket appena creato

INSERT INTO Fase (numero_ticket, nome_stato, data_ora_inizio) **VALUES** (?, 'aperto', now)

OP2 - Cercare all'operatore più disponibile da poi assegnare ad un ticket

SELECT o.email **FROM** Operatore o
LEFT JOIN Ticket t **ON** t.email_operatore = o.email
AND (t.nome_stato = 'aperto' **OR** t.nome_stato = 'analisi')
GROUP BY o.email
ORDER BY COUNT(o.email) **ASC**
LIMIT 1

OP3 - Registrare un nuovo cliente

Controllo se l'email è già stata utilizzata

SELECT EXISTS (
 SELECT 1 **FROM** Utente u
 WHERE lower(?) = lower(u.email)
)

Se la email non è stata già utilizzata inserisco l'utente nel database

INSERT INTO Utente (email, nome, cognome, password) **VALUES** (?, ?, ?, ?, ?)

INSERT INTO Cliente (email) **VALUES** (?)

OP4 - Eliminare un cliente

Rimuovo tutte le referenze al cliente:

UPDATE Ticket t

SET t.email_cliente = null

WHERE t.email_cliente = ?

UPDATE Messaggio m

SET m.email_utente = null

WHERE m.email_utente = ?

Elimino il cliente

DELETE FROM Cliente

WHERE email = ?

DELETE FROM Utente

WHERE email = ?

OP5 - Autenticare l'accesso ad un utente

SELECT EXISTS (

SELECT 1 **FROM** Utente u

WHERE u.email = ?

AND u.password = ?

)

OP6 - Visualizzare i dettagli di un ticket e i suoi messaggi

Ricavo i dettagli del ticket

SELECT t.email_cliente, t.titolo, t.descrizione, t.nome_priorita, t.nome_problema, t.nome_so,
t.nome_stato **FROM** Ticket t

WHERE t.numero = ?

Ricavo i il testo e il nome e cognome dei scrittori dei messaggi del ticket.

SELECT u.nome, u.cognome, m.testo **FROM** Messaggio m

LEFT JOIN Utente u **ON** u.email = m.email_utente

WHERE m.numero_ticket = ?

ORDER BY m.data_scrittura **ASC**

Ricavo i tutorial associati al ticket.

SELECT tu.nome, tu.url **FROM** Tutorial tu, Soluzione s

WHERE tu.id = s.tutorial_id

AND s.numero_ticket = ?

OP7 - Visualizzare i ticket

Visualizzazione dei ticket con filtro

SELECT t.numero, t.titolo, t.nome_problema, t.nome_so, f.nome_stato **FROM** Ticket t

WHERE (? IS NULL **OR** lower(t.titolo) **LIKE** lower(concat('%', ?, '%')))

AND (? IS NULL **OR** t.nome_priorita = ?)

AND (? IS NULL **OR** t.nome_problema = ?)

AND (? IS NULL **OR** t.nome_so = ?)
AND (? IS NULL **OR** t.nome_stato = ?)
AND (? IS NULL **OR** t.email_cliente = ? **OR** t.email_operatore = ?)

OP8 - Scrittura dei messaggi

INSERT INTO Messaggio (numero_ticket, data_ora, email_utente, testo) **VALUES** (?, now, ?, ?)

OP9 - Cambiare stato di un ticket

Aggiorno stato del ticket

UPDATE Ticket t
SET t.nome_stato = ?
WHERE t.numero = ?

Aggiungo la nuova fase

INSERT INTO Fase f (numero_ticket, nome_stato, data_ora_inizio) **VALUES** (?, ?, now)

OP10 - Associazione tutorials ad un ticket

Ripetuto per quanti tutorial si vuole aggiungere al singolo ticket

INSERT INTO Soluzione (numero_ticket, tutorial_id) **VALUES** (?, ?)

OP11 - Segnalazione di un bug

Ripetuto per quanti bug si vogliono segnalare per quel singolo ticket

INSERT INTO Bug (numero_ticket, descrizione, nome_priorita, nome_stato) **VALUES** (?, ?, ?, 'non in correzione')

Creazione dello stato di sviluppo del bug

INSERT INTO Sviluppo_Bug sb (bug_id, nome_stato, data_ora_inizio) **VALUES** (?, 'non in correzione', now)

OP12 - Visualizzare dei bug

SELECT b.id, b.descrizione, b.nome_priorita, sb.nome_stato **FROM** Bug b
WHERE (? IS NULL **OR** lower(b.descrizione) **LIKE** lower(concat('%', ?, '%')))
AND (? IS NULL **OR** b.nome_priorita = ?)
AND (? IS NULL **OR** b.nome_stato = ?)

OP13 - Visualizzare i dettagli di un bug

SELECT b.descrizione, b.nome_priorita, b.nome_stato **FROM** Bug b
WHERE b.id = ?

OP14 - Cambiare stato di un bug

Aggiorno stato del bug

UPDATE Bug b
SET b.nome_stato = ?
WHERE b.id = ?

Aggiungo il nuovo sviluppo

INSERT INTO Sviluppo_Bug sb (bug_id, nome_stato, data_ora_inizio, email_sviluppatore) **VALUES** (?, ?, now, ?)

OP15 - Visualizzare le statistiche dei ticket in un dato periodo

Conteggio dei numeri di ticket

```
SELECT COUNT(DISTINCT t.numero) FROM Ticket t
JOIN Fase f ON f.numero_ticket = t.numero
AND (? IS NULL OR f.data_ora_inizio >= ?)
AND (? IS NULL OR f.data_ora_inizio <= ?)
```

Numero di ticket con uno stato

```
SELECT COUNT(DISTINCT t.numero) FROM Ticket t
JOIN Fase f ON f.numero_ticket = t.numero
AND f.nome_stato = ?
AND f.dataOrainizio = (
    SELECT MAX(f2.dataOrainizio) FROM Fase f2
    WHERE f2.numero_ticket = t.numero
    AND (f2.dataOrainizio BETWEEN COALESCE(?, f2.dataOrainizio) AND
COALESCE(?, f2.dataOrainizio))
)
```

COALESCE restituisce il primo valore non nullo valido nella lista.

Numero di ticket con un problema, operazione uguale per "Numero di ticket su un sistema operativo"

```
SELECT COUNT(DISTINCT t.numero) FROM Ticket t
JOIN Fase f ON f.numero_ticket = t.numero
AND t.nome_problema = ?
AND (? IS NULL OR f.data_ora_inizio >= ?)
AND (? IS NULL OR f.data_ora_inizio <= ?)
```

Tempo medio di durata dei ticket da aperto a chiuso

```
SELECT AVG(TIMESTAMPDIFF(HOUR, f2.data_ora_inizio, f.data_ora_inizio)) FROM Ticket t
JOIN Fase f ON f.numero_ticket = t.numero
JOIN Fase f2 ON f2.numero_ticket = t.numero
AND (f.nome_stato = 'chiuso - risolto' OR f.nome_stato = 'chiuso - non risolto')
AND f2.nome_stato = 'aperto'
AND (? IS NULL OR f.data_ora_inizio >= ?)
AND (? IS NULL OR f.data_ora_inizio <= ?)
```

OP16 - Visualizzare prestazioni degli operatori

```
SELECT u.email, u.nome, u.cognome,
COUNT(t.numero) AS numeroTickets,
AVG(TIMESTAMPDIFF(HOUR, f2.data_ora_inizio, f.data_ora_inizio)) AS tempoMedio,
AVG(vo.voto) AS votoMedio
FROM Ticket t
LEFT JOIN Valutazione_Operatore vo ON vo.numero_ticket = t.numero
JOIN Utente u ON u.email = t.email_operatore
JOIN Fase f ON f.numero_ticket = t.numero
JOIN Fase f2 ON f2.numero_ticket = t.numero
AND (f.nome_stato = 'chiuso - risolto' OR f.nome_stato = 'chiuso - non risolvibile')
AND f2.nome_stato = 'aperto'
```

GROUP BY t.email_operatore

ORDER BY votoMedio **DESC**

OP17 - Creazione di un operatore

Equivale a **OP3 - Registrare un nuovo cliente** ma Operatore al posto di cliente

OP18 - Eliminazione di un operatore

Equivale a **OP4 - Eliminare un cliente** ma Operatore al posto di cliente

OP19 - Creazione di un sviluppatore

Equivale a **OP3 - Registrare un nuovo cliente** ma Operatore al posto di cliente

OP20 - Eliminazione di un sviluppatore

Rimuovo tutte le referenze al sviluppatore:

UPDATE Sviluppo_Bug sb

SET sb.email_sviluppatore = null

WHERE sb.email_sviluppatore = ?

Elimino lo sviluppatore:

DELETE FROM Sviluppatore

WHERE email = ?

DELETE FROM Utente

WHERE email = ?

OP21 - Visualizzare tutti gli operatori

SELECT o.email, o.nome, o.cognome **FROM** Operatore o

WHERE ? IS NULL

OR (

lower(o.email) LIKE lower(concat('%', ?, '%'))

OR lower(o.nome) LIKE lower(concat('%', ?, '%'))

OR lower(o.cognome) LIKE lower(concat('%', ?, '%'))

)

OP22 - Visualizzare tutti gli sviluppatori

Equivale a **OP21 - Visualizzare tutti gli operatori** ma Sviluppatore al posto di Operatore.

OP23 - Valutazione operatore

INSERT INTO Valutazione_Operatore (numero_ticket, voto, commento) **VALUES** (?, ?, ?)

Progettazione dell'applicazione

Il sistema di ticketing è stato implementato utilizzando Java sia per il front-end che per il back-end.

Backend

Il backend è stato implementato utilizzando il framework Spring Boot, che è una libreria Java per la creazione di servizi RESTful e applicazioni web.

Spring Boot semplifica il processo di configurazione fornendo gestione delle eccezioni, la sicurezza e l'integrazione con i database.

La parte fondamentale del backend riguarda la modellazione delle entità del dominio utilizzando l'annotazione `@Entity` di JPA (Java Persistence API). Questo permette di definire le entità dell'applicazione come oggetti Java, i quali possono essere mappati direttamente su tabelle all'interno di un database relazionale. Il framework Hibernate, incluso di default in Spring Boot, si occupa della gestione della persistenza delle entità nel database, semplificando il processo di memorizzazione e recupero dei dati.

E' stato utilizzato MySQL come database principale, garantendo l'efficienza e l'affidabilità nella gestione dei dati.

MySQL è un sistema di gestione di database relazionale molto popolare e ampiamente utilizzato. La scelta di utilizzare MySQL come database principale offre diversi vantaggi, tra cui:

1. **Efficienza:** MySQL è noto per le sue prestazioni veloci e ottimizzate, che consentono di gestire un elevato numero di query in modo rapido ed efficiente.
2. **Affidabilità:** MySQL è stabile e affidabile, garantendo che i dati siano gestiti in modo sicuro e senza perdite.
3. **Compatibilità con Spring Boot e Hibernate:** MySQL è pienamente compatibile con Spring Boot e Hibernate, semplificando l'integrazione e l'interazione tra l'applicazione e il database.
4. **Community e supporto:** MySQL ha una vasta community di sviluppatori e un supporto attivo, il che rende più facile trovare risorse e soluzioni per eventuali problemi o domande.

Frontend

Il frontend del sistema è stato realizzato utilizzando il web framework open-source chiamato Vaadin. Vaadin semplifica notevolmente lo sviluppo del frontend grazie a un approccio basato sugli oggetti e alle componenti grafiche predefinite.

Vaadin permette di creare facilmente il frontend utilizzando oggetti predefiniti:

Vaadin offre una vasta gamma di componenti grafiche predefinite, come pulsanti, tabelle e campi di input, che possono essere utilizzate per creare interfacce utente interattive senza la necessità di scrivere manualmente codice HTML, CSS o JavaScript. Questo approccio basato sugli oggetti rende lo sviluppo del frontend più rapido ed efficiente.

Applicazione a singola pagina:

Vaadin permette di gestire le interazioni dell'utente attraverso eventi e metodi, consentendo di aggiornare dinamicamente l'interfaccia utente senza ricaricare l'intera pagina web. Ciò permette di creare un'esperienza utente fluida e reattiva, con interfacce che si adattano automaticamente a diverse dimensioni di schermo (responsiveness).

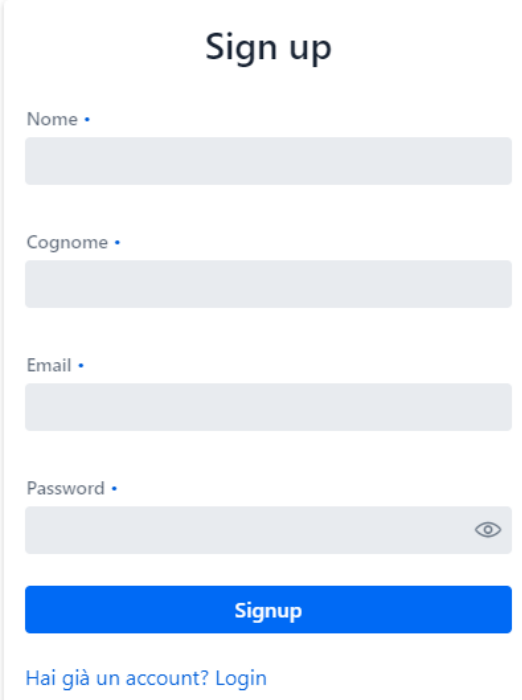
Gestione delle sessioni utente e sicurezza:

Vaadin offre funzionalità per gestire le sessioni utente e garantire che solo gli utenti autorizzati possano accedere a determinate pagine o funzionalità. In caso di accesso a pagine non permesse, Vaadin può reindirizzare automaticamente l'utente alla pagina di login.

Integrazione con Spring Boot per l'avvio delle query e gli accessi al database:

Vaadin si integra perfettamente con Spring Boot, il quale offre supporto per l'esecuzione delle query al database tramite il framework JPA con Hibernate. Ciò semplifica notevolmente il recupero e l'aggiornamento dei dati dal database. Vaadin offre inoltre componenti speciali per la creazione di tabelle e form per l'interazione con il database in modo intuitivo e sicuro.

Esposizione



Sign up

Nome •

Cognome •

Email •

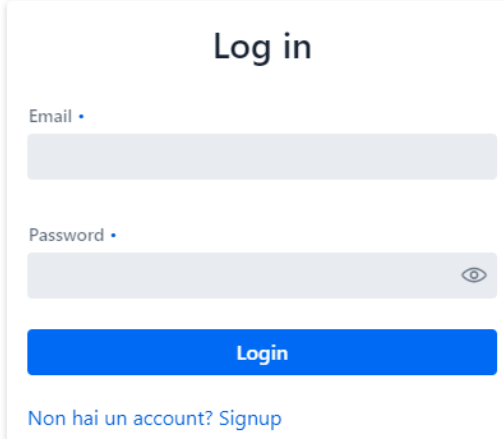
Password •

Signup

[Hai già un account? Login](#)



Schermata di registrazione per la creazione del cliente.



Log in

Email •

Password •

Login

[Non hai un account? Signup](#)



Schermata di log-in per l'autenticazione dell'utente.

All'avvio della web-app, la prima schermata visualizzata sarà quella di log-in. Per garantire la sicurezza e l'accesso appropriato alle diverse pagine, ho implementato un sistema di autenticazione. Una volta effettuato il log-in con successo, la sessione viene memorizzata per consentire l'accesso alle pagine consentite.

In caso si tenti di accedere a una pagina non autorizzata tramite link diretto, il sistema rileva tale tentativo e disconnette l'utente, richiedendo di effettuare nuovamente il log-in.

Per garantire la correttezza dei dati inseriti, la web-app effettua una serie di controlli a livello locale senza coinvolgere il database. Ad esempio, se vengono lasciati campi vuoti durante l'inserimento, l'applicazione segnalerà immediatamente l'errore senza interagire con il database. Questo aiuta a prevenire inserimenti errati e semplifica il processo di interazione con l'utente.

Ticket Master
Logout

Popo Pipo

Home
Miei Ticket
Nuovo Ticket
Elimina Account

Popo Pipo

Titolo
Problemi di connessione

Descrizione
Il software non riesce a connettersi al server per sincronizzare i dati.
72/500

Urgenza
alta
Problema
rete
Sistema Operativo
windows
Stato
chiuso - risolto

Valuta Operatore

Chat

GiulioCesare:
Potrebbe essere un problema di configurazione delle impostazioni di rete. Stiamo indagando.

PopoPipo:
Ho notato che il problema si verifica solo con il vostro software, gli altri programmi funzionano correttamente.

GiulioCesare:
Grazie per le informazioni. Stiamo cercando una soluzione al problema di connessione.

Come installare app
<https://www.youtube.com/watch?v=1y0-7HbKkX8>

Come utilizzare app
<https://www.youtube.com/watch?v=1y0-8UJBD678>



Visualizzazione di un ticket chiuso, associato al cliente.

Voto

Descrizione
0/500

Valuta



Schermata di valutazione dell'operatore.

Marco Magnani

Home

Miei Ticket

Nuovo Ticket

Elimina Account

Logout

Nuovo ticket

Titolo *

Descrizione *

0/500

Urgenza * Problema * SO *

Crea



Creazione di un ticket.

Il cliente nella sua schermata può effettuare il "logout" per uscire dalla sua sessione, eliminare il suo account, visualizzare i suoi ticket e creare un nuovo ticket.

Quando seleziona un ticket aperto, se questo è associato al cliente, avrà la possibilità di scrivere nella chat, altrimenti potrà solamente visualizzarla.

Se il ticket selezionato è chiuso e il cliente è associato a esso, avrà la possibilità di valutare l'operatore.

La schermata di visualizzazione dei suoi ticket è simile a quella home, ma mostra solo i ticket associati al cliente.

Giulio Cesare

Home

Logout

Numero	Titolo	Priorità	Problema	SO	Stato
1	Il software si blocca	alta	software	windows	chiuso - risolto
4	Non si aggiorna	bassa	aggiornamento	linux	chiuso - non risolvibile
7	Errore durante l'installazione	alta	installazione	windows	chiuso - non risolvibile
10	Problemi di connessione	alta	rete	windows	analisi
13	Crash durante l'aggiornamento	alta	aggiornamento	windows	analisi



Schermata principale dell'operatore.

Ticket Master - Operatore
Logout

Giulio Cesare
Home

Popo Pipò

Titolo
Problemi di connessione

Descrizione
Il software non riesce a connettersi al server per sincronizzare i dati.

72/500

Urgenza
alta
Problema
rete
Sistema Operativo
windows
Stato
analisi

Cambia Stato
chiuso - risolto
Chiudi Ticket
Segnala Bug
Tutorial

Chat
GiulioCesare:
Potrebbe essere un problema di configurazione delle impostazioni di rete. Stiamo indagando.
PopoPipo:
Ho notato che il problema si verifica solo con il vostro software, gli altri programmi funzionano correttamente.
GiulioCesare:
Grazie per le informazioni. Stiamo cercando una soluzione al problema di connessione.

Message
Send



Visualizzazione di un ticket aperto, associato all'operatore.

La schermata principale dell'operatore è simile a quella dei clienti, mostrando tutti i ticket associati all'operatore. Tuttavia, è presente un filtro per la priorità dei ticket, che consente all'operatore di organizzare e gestire meglio le richieste.

Quando l'operatore visualizza un ticket ancora aperto, avrà la possibilità di scrivere messaggi, modificare lo stato del ticket e, a seconda dello stato, aggiungere anche dei tutorial o indicazioni utili per il cliente.

Ticket Master - Amministratore
Logout

Capo Capi
Home
Operatori
Nuovo Operatore
Sviluppatori
Nuovo Sviluppatore

From Date
To Date




Reset
Genera Statistiche

Ticket Statistics
Ticket Aperti
1
Ticket In Analisi
4
Ticket Chiusi
10
Tickets Totali
15
Percentuale Ticket Chiusi
66.66666666666666
%
Tempo Medio di Risoluzione
93.66670227050781
h

Tipologia Ticket
Software
6
Rete
2
Installazione
2
Aggiornamento
2
Sicurezza
1
Performance
2

SO dei Ticket
Windows
Linux
Mac

Operator Statistics

Operatore	Tickets Chiusi	Tempo Medio	Voto Medio
 Lizmo Lizmato p3@gmail.com	3	15.3333	9.0
 Gizmo Gizmato p2@gmail.com	3	81.0	7.3333
 Giulio Cesare p1@gmail.com	4	128.75	5.3333

Valutazione Media

7.2221999168396



Schermata principale dell'amministratore.

Ticket Master - Amministratore

Logout

Capo Capi

Home

Operatori




Nuovo Operatore

Sviluppatori

Nuovo Sviluppatore

Titolo

Filter by name...

Email ↕	Nome ↕	Cognome ↕	Elimina
p1@gmail.com	Giulio	Cesare	
p2@gmail.com	Gizmo	Gizmato	
p3@gmail.com	Lizmo	Lizmato	



Visualizzazione degli operatori.


Crea Operatore

Nome •

Cognome •

Email •

Password •



Crea



Creazione nuovo operatore.

L'amministratore può accedere alle varie statistiche riguardanti i ticket e gli operatori nella schermata principale. Selezionando un periodo specifico, può ottenere statistiche riferite a quel periodo. Se viene scelta solo una data tra le due del periodo, verranno mostrate tutte le statistiche a partire da quella data o prima di essa, al posto di un periodo definito.

Inoltre, l'amministratore ha la possibilità di visualizzare gli operatori e gli sviluppatori, creare nuovi account per loro e, se necessario, eliminarli. Questo consente all'amministratore di gestire in modo flessibile il personale dell'applicazione e garantire un'efficace distribuzione delle responsabilità.