

TICKETEASE

*Applicazione software per la gestione delle richieste di supporto tecnico
all'interno di aziende o organizzazioni.*

*Progetto di
Ingegneria del
Software*

Sommario

| | | |
|----|---|----|
| 1. | Ideazione | 3 |
| - | Posizionamento | 3 |
| | Opportunità di business..... | 3 |
| | Formulazione del problema | 3 |
| | Formulazione della posizione del prodotto | 3 |
| - | Descrizione delle parti interessate..... | 3 |
| - | Riepilogo delle caratteristiche del sistema..... | 3 |
| - | Elenco regole..... | 4 |
| - | Definizioni del glossario | 4 |
| - | Requisiti | 4 |
| - | Obiettivi e casi d'uso..... | 5 |
| - | Modello dei casi d'uso | 6 |
| | UC3: Creazione di un ticket di assistenza..... | 6 |
| | UC6. Chiusura del ticket..... | 8 |
| | UC7. Assegnazione ticket a un tecnico IT | 9 |
| | UC1: Registrazione utente | 10 |
| | UC2: Login utente..... | 10 |
| | UC4: Visualizzazione notifiche | 10 |
| | UC5: Aggiornamento dello stato di un ticket | 10 |
| | UC8. Visualizzazione storico stato ticket | 10 |
| 2. | Iterazione 1..... | 11 |
| - | Introduzione | 11 |
| - | Analisi orientata agli oggetti | 11 |
| | Modello di Dominio..... | 11 |
| | Diagramma di sequenza di sistema..... | 12 |
| | Contratti delle operazioni..... | 14 |
| - | Progettazione..... | 15 |
| | Diagramma delle classi | 16 |
| | Diagrammi di Sequenza | 16 |
| - | Refactoring..... | 17 |
| 3. | Iterazione 2..... | 18 |
| - | Introduzione | 18 |
| - | Analisi orientata agli Oggetti | 18 |

| | |
|---|----|
| Modello di Dominio..... | 18 |
| Diagramma di sequenza di sistema..... | 19 |
| Contratti delle operazioni..... | 20 |
| - Progettazione..... | 21 |
| Diagramma delle classi | 22 |
| Diagrammi di Sequenza | 22 |
| - Refactoring..... | 24 |
| 4. Iterazione 3..... | 25 |
| - Introduzione | 25 |
| UC9. Visualizzazione dei ticket assegnati | 25 |
| - Analisi orientata agli Oggetti | 25 |
| Modello di dominio..... | 26 |
| Diagrammi di sequenza di sistema..... | 26 |
| Contratti delle operazioni..... | 28 |
| - Progettazione..... | 29 |
| Diagramma delle classi | 29 |
| Diagrammi di sequenza..... | 29 |
| - Refactoring..... | 31 |
| 5. Iterazione 4..... | 32 |
| - Introduzione | 32 |
| - Analisi orientata agli Oggetti | 32 |
| Diagrammi di sequenza di sistema..... | 32 |
| Contratti delle operazioni..... | 33 |
| - Progettazione..... | 33 |
| Diagramma delle classi | 34 |
| Diagrammi di sequenza..... | 34 |
| - Refactoring..... | 35 |
| 6. Testing..... | 36 |
| Registrazione utente | 36 |
| Login utente | 36 |
| Creazione ticket..... | 36 |
| Assegnazione ticket..... | 36 |

Applicazione TicketEase

1. Ideazione

Il sistema di gestione ticket IT mira a fornire un'infrastruttura per la gestione delle richieste di supporto tecnico. Gli utenti possono segnalare problemi attraverso la creazione di ticket, gli amministratori possono anch'essi creare ticket e assegnarli, mentre i tecnici possono aggiornare e chiudere i ticket.

- *Posizionamento*

Opportunità di business

La digitalizzazione della gestione delle richieste IT riduce i tempi di risposta e migliora la tracciabilità. L'assegnazione automatica ai tecnici competenti ottimizza le risorse, mentre le notifiche automatiche tengono aggiornati utenti e operatori, garantendo efficienza e trasparenza.

Formulazione del problema

Molte aziende non hanno un sistema centralizzato per gestire le richieste di assistenza IT, causando inefficienze nella gestione delle segnalazioni. Questo sistema mira a ottimizzare il processo, migliorando la comunicazione tra utenti e tecnici e tracciando lo stato delle richieste in tempo reale.

Formulazione della posizione del prodotto

Il sistema è rivolto alle aziende e ai reparti IT che necessitano di un modo efficiente per gestire le richieste di assistenza.

- *Descrizione delle parti interessate*

- I. Utente: può aprire e monitorare ticket di assistenza.
- II. Tecnico IT: prende in carico e gestisce i ticket assegnati.
- III. Amministratore: assegna i ticket ai tecnici e supervisiona il sistema.

- *Riepilogo delle caratteristiche del sistema*

Il sistema permetterà di:

- Creare ticket di assistenza con dettagli sul problema.
- Assegnare i ticket ai tecnici IT
- Aggiornare lo stato del ticket
- Notificare automaticamente gli utenti sulle modifiche dello stato

- Archiviare e ricercare i ticket risolti

- *Elenco regole*

| ID | REGOLA | MODIFICABILITA' | SORGENTE |
|----|--|-----------------|------------------------|
| R1 | Ogni utente può aprire più ticket ma solo uno per problema. | Media | Politica di TicketEase |
| R2 | Solo i tecnici assegnati possono aggiornare lo stato di un ticket. | Bassa | Politica di TicketEase |
| R3 | Gli utenti ricevono notifiche automatiche per ogni aggiornamento del ticket. | Alta | Politica di TicketEase |
| R4 | Gli amministratori possono riassegnare un ticket a un altro tecnico. | Media | Politica di TicketEase |

- *Definizioni del glossario*

| Termine | Definizione |
|-------------------------|---|
| <i>Ticket</i> | Segnalazione di un problema IT aperta da un utente e gestita dai tecnici. |
| <i>Tecnico IT</i> | Professionista che prende in carico e risolve i ticket assegnati. |
| <i>Utente</i> | Persona che utilizza il sistema per aprire e monitorare i ticket. |
| <i>Amministratore</i> | Supervisore del sistema che assegna i ticket ai tecnici. |
| <i>Stato del Ticket</i> | Indica la fase in cui si trova il ticket (Aperto, In lavorazione, Risolto, Chiuso). |
| <i>Notifica</i> | Messaggio automatico inviato agli utenti e ai tecnici sugli aggiornamenti dei ticket. |

- *Requisiti*

Il sistema da realizzare è un'applicazione software per la gestione delle richieste di supporto tecnico (ticketing) all'interno di un'azienda o organizzazione. Gli utenti potranno creare ticket di assistenza IT per segnalare problemi tecnici, mentre i tecnici IT saranno incaricati di risolverli.

Gli amministratori avranno il compito di supervisionare il sistema e assegnare i ticket ai tecnici competenti.

TicketEase

Ogni ticket deve contenere una descrizione dettagliata del problem. Per ogni ticket sarà tracciato lo stato di avanzamento e verrà registrata la cronologia degli aggiornamenti.

Gli utenti possono aprire nuovi ticket e ricevere notifiche sugli aggiornamenti relativi ai propri ticket. I tecnici assegnati devono poter modificare lo stato del ticket e fornire aggiornamenti dettagliati. Gli amministratori possono gestire l'assegnazione dei ticket e monitorare l'andamento delle richieste.

I tecnici ricevono notifiche automatiche all'assegnazione di un nuovo ticket. Il sistema tiene traccia di tutti i ticket chiusi per future analisi e reportistica.

Il sistema deve possedere:

- Un'area di autenticazione con login per utenti, tecnici e amministratori.
- Un'interfaccia per la creazione e gestione dei ticket.
- Una sezione dedicata alla modifica dello stato dei ticket da parte dei tecnici.
- Un sistema di notifiche per informare gli utenti sugli aggiornamenti.
- Un'area di amministrazione per la gestione dei ticket e la loro assegnazione ai tecnici.

L'applicazione deve permettere di:

- Registrare nuovi utenti, tecnici e amministratori.
- Effettuare il login con ruoli differenziati.
- Creare un nuovo ticket.
- Assegnare ticket ai tecnici disponibili.
- Modificare lo stato di avanzamento dei ticket.
- Inviare notifiche automatiche agli utenti e ai tecnici.
- Consultare la cronologia dei ticket aperti e chiusi.

- Obiettivi e casi d'uso

Analizzando i requisiti riportati nel paragrafo precedente, sono stati individuati gli attori principali a cui è destinato il sistema e gli obiettivi che si intende portare a termine; da queste informazioni sono stati ricavati i casi d'uso principali.

| <i>Attore</i> | Obiettivo | Caso d'uso |
|---------------|---------------------------|--------------------------------|
| <i>Utente</i> | Registrarsi al sistema | UC1. Registrazione utente |
| <i>Utente</i> | Effettuare il login | UC2. Login utente |
| <i>Utente</i> | Creare un nuovo ticket di | UC3. Creazione di un ticket di |

TicketEase

| | assistenza. | assistenza |
|---------------------------|--|---|
| <i>Utente, Tecnico IT</i> | Visualizzare le notifiche ricevute all'interno del sistema | UC4. Visualizzazione notifiche |
| <i>Tecnico IT</i> | Gestire e aggiornare i ticket assegnati. | UC5. Aggiornamento dello stato del ticket |
| <i>Tecnico IT</i> | Chiudere un ticket risolto. | UC6. Chiusura del ticket |
| <i>Amministratore</i> | Assegnare i ticket ai tecnici disponibili. | UC7. Assegnazione ticket a un tecnico IT |
| <i>Amministratore</i> | Visualizzare lo storico e lo stato dei ticket. | UC8. Visualizzazione storico ticket |

- Modello dei casi d'uso

Tra tutti i casi d'uso individuati, si è scelto di fornire una descrizione in formato dettagliato per i seguenti casi d'uso:

- UC3: Creazione di un Ticket di Assistenza
- UC6. Chiusura di un Ticket
- UC7: Assegnazione di un Ticket a un Tecnico IT

Per i restanti casi d'uso si fornisce una descrizione in formato breve.

UC3: Creazione di un ticket di assistenza

| NOME DEL CASO D'USO | UC3. Creazione di un ticket di assistenza |
|--------------------------------------|--|
| Portata | Applicazione TicketEase |
| Livello | Obiettivo utente |
| Attore primario | Utente |
| Parti interessate e interessi | Utente registrato: vuole segnalare un problema per ricevere assistenza. Reparto IT: vuole ricevere segnalazioni dettagliate per una risoluzione efficace. |
| Precondizioni | L'utente deve essere registrato nel sistema. |
| Garanzia di successo | Il ticket viene creato con successo e registrato |

| | |
|--|--|
| | nel sistema. |
| Scenario principale di successo | <ol style="list-style-type: none">1. L'utente accede alla propria area personale.2. L'utente seleziona la voce "Nuovo Ticket".3. Il sistema fornisce un form per la creazione del ticket.4. L'utente compila il form inserendo il titolo, la descrizione del problema e la categoria di appartenenza.5. Il sistema acquisisce i dati e assegna un ID univoco al ticket.6. Il ticket viene registrato e inviato automaticamente al reparto IT. |
| Estensioni | <p>*a. Il sistema fallisce e si arresta in maniera improvvisa</p> <ol style="list-style-type: none">1. Viene visualizzato un messaggio di errore.2. Viene chiesto all'utente di riprovare. <p>4a. L'utente non inserisce tutti i campi obbligatori:</p> <ol style="list-style-type: none">1. Il sistema mostra un messaggio di errore evidenziando i campi mancanti.2. L'utente ripete il passo 4 fino al completamento corretto del form. |
| Requisiti speciali | <ol style="list-style-type: none">1. Il sistema deve garantire la persistenza del ticket anche in caso di errore temporaneo.2. I ticket devono essere classificati per priorità. |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Ogni volta che un utente ha necessità di assistenza. |

| | |
|--------------|--|
| Varie | |
|--------------|--|

UC6. Chiusura del ticket

| NOME DEL CASO D'USO | UC6. Chiusura del ticket |
|--|--|
| Portata | Applicazione TicketEase |
| Livello | Obiettivo Tecnico IT |
| Attore primario | Tecnico IT |
| Parti interessate e interessi | Tecnico IT: vuole chiudere un ticket risolto. Utente: vuole essere informato sulla risoluzione del problema. |
| Precondizioni | Il ticket deve essere stato assegnato a un tecnico. |
| Garanzia di successo | Il ticket viene chiuso con successo e l'utente viene informato. |
| Scenario principale di successo | <ol style="list-style-type: none">1. Il tecnico accede al sistema.2. Il tecnico visualizza i ticket assegnati.3. Il tecnico seleziona un ticket risolto.4. Il tecnico inserisce una descrizione della soluzione adottata.5. Il sistema chiude il ticket e lo archivia.6. L'utente riceve una notifica sulla risoluzione del problema. |
| Estensioni | 4a. Il tecnico non inserisce una descrizione della soluzione: <ol style="list-style-type: none">1. Il sistema mostra un messaggio di errore.2. Il tecnico ripete il passo 4. |
| Requisiti speciali | Il sistema deve mantenere uno storico dei ticket risolti. |
| Elenco delle varianti tecnologiche e dei dati | |
| Frequenza di ripetizioni | Ogni volta che un problema viene risolto. |
| Varie | |

UC7. Assegnazione ticket a un tecnico IT

| NOME DEL CASO D'USO | | UC7. Assegnazione ticket a un tecnico IT |
|---|--|--|
| Portata | | Applicazione TicketEase |
| Livello | | Obiettivo amministratore |
| Attore primario | | Amministratore |
| Parti interessate e interessi | | Amministratore: vuole assegnare i ticket ai tecnici disponibili per una gestione efficiente delle segnalazioni. |
| Precondizioni | | Il ticket deve essere stato creato e deve essere in stato "Nuovo". |
| Garanzia di successo | | Il ticket viene assegnato con successo a un tecnico disponibile. |
| Scenario principale di successo | | <ol style="list-style-type: none">1. L'amministratore accede alla piattaforma.2. L'amministratore visualizza la lista dei ticket in attesa di assegnazione.3. L'amministratore seleziona un ticket da assegnare.4. Il sistema mostra la lista dei tecnici disponibili.5. L'amministratore sceglie un tecnico e conferma l'assegnazione.6. Il sistema assegna il ticket e notifica il tecnico. |
| Estensioni | | 5a. Se non ci sono tecnici disponibili: <ol style="list-style-type: none">1. Il sistema mostra un messaggio di avviso e lascia il ticket in sospeso. |
| Requisiti speciali | | Il sistema deve permettere il riassegnamento di un ticket se necessario. |
| Elenco delle varianti tecnologiche e dei dati | | |
| Frequenza di ripetizioni | | Ogni volta che viene aperto un ticket. |
| Varie | | Il sistema deve mantenere un log delle assegnazioni per tracciabilità. |

UC1: Registrazione utente

Scenario di successo

1. L'utente accede alla pagina di registrazione.
2. Compila il modulo con i dati richiesti.
3. Il sistema valida i dati e crea l'account.
4. L'utente riceve una conferma di registrazione.

UC2: Login utente

Scenario di successo

1. L'utente accede alla pagina di login.
2. Inserisce username e password nei campi richiesti.
3. Il sistema verifica le credenziali dell'utente.
4. Se le credenziali sono corrette, l'utente viene autenticato.
5. L'utente viene reindirizzato alla propria area personale.

UC4: Visualizzazione notifiche

Scenario di successo

1. Il cliente o il tecnico IT accede all'area personale.
2. Seleziona la sezione "Visualizza notifiche"
3. Il sistema mostra le notifiche dell'utente corrente.

UC5: Aggiornamento dello stato di un ticket

Scenario di successo

1. Il tecnico IT accede alla piattaforma.
2. Visualizza la lista dei ticket assegnati.
3. Seleziona un ticket per aggiornarne lo stato.
4. Il sistema mostra le opzioni disponibili per l'aggiornamento dello stato (es. "In lavorazione", "Risolto", "Chiuso").
5. Il tecnico seleziona il nuovo stato e conferma la modifica.
6. Il sistema salva l'aggiornamento e invia una notifica all'utente.

UC8. Visualizzazione storico stato ticket

Scenario di successo

1. L'amministratore o il tecnico IT accede all'area personale.
2. Seleziona la sezione "Storico stato Ticket".
3. Il sistema mostra i vari cambiamenti di stato di un ticket.

2. Iterazione 1

- *Introduzione*

Dopo la fase di ideazione, si procede con l'elaborazione e le varie iterazioni.

Con le varie iterazioni si intende raffinare il modello di Visione, implementare l'architettura del software proposto in maniera iterativa, risolvere le problematiche legate ai rischi maggiori, definire in maniera più chiara i requisiti e la portata, fornendo inoltre stime più realistiche del piano di lavoro e delle risorse necessarie.

Nella prima iterazione si è scelto di concentrarsi sui seguenti casi d'uso:

- UC1: Registrazione Utente
- UC2: Login Utente
- UC3: Creazione di un ticket di assistenza

Questa iterazione è fondamentale per permettere agli utenti di accedere al sistema e, successivamente, abilitare la gestione dei ticket nelle iterazioni successive. Il focus è sull'implementazione delle funzionalità di registrazione, autenticazione con validazione delle credenziali e creazione di un ticket.

In questa fase iniziale, si considera esclusivamente la registrazione e l'accesso del cliente, ovvero l'utente che richiede assistenza. Il cliente può registrarsi, autenticarsi e creare un ticket per segnalare un problema.

Nelle iterazioni successive, verranno introdotte le funzionalità per gli amministratori e i tecnici IT, che saranno responsabili della gestione e della risoluzione dei ticket assegnati.

- *Analisi orientata agli oggetti*

L'analisi orientata agli oggetti verte sulla generazione di una descrizione del dominio da un punto di vista ad oggetti. A questo proposito, vengono presentati i seguenti strumenti:

- Modello di dominio
- Diagramma di sequenza di sistema (SSD)
- Contratti delle operazioni

Modello di Dominio

La modellazione del business è una disciplina che, in termini di UP (processo unificato), si occupa di fornire dettagli sul dominio. Nello specifico, essa è costituita da:

- Modello di dominio
- Elaborato grafico dei concetti
- Attributi e associazioni significative

Relativamente al caso d'uso scelto (UC3), dop una valutazione dello scenario

TicketEase

principale di successo è stato possibile identificare le seguenti classi concettuali:

- TicketEase
- Utente
- Cliente
- Ticket
- Prodotti

Da queste classi, tenendo conto di associazioni e attributi, è stato ricavato il seguente modello di dominio:

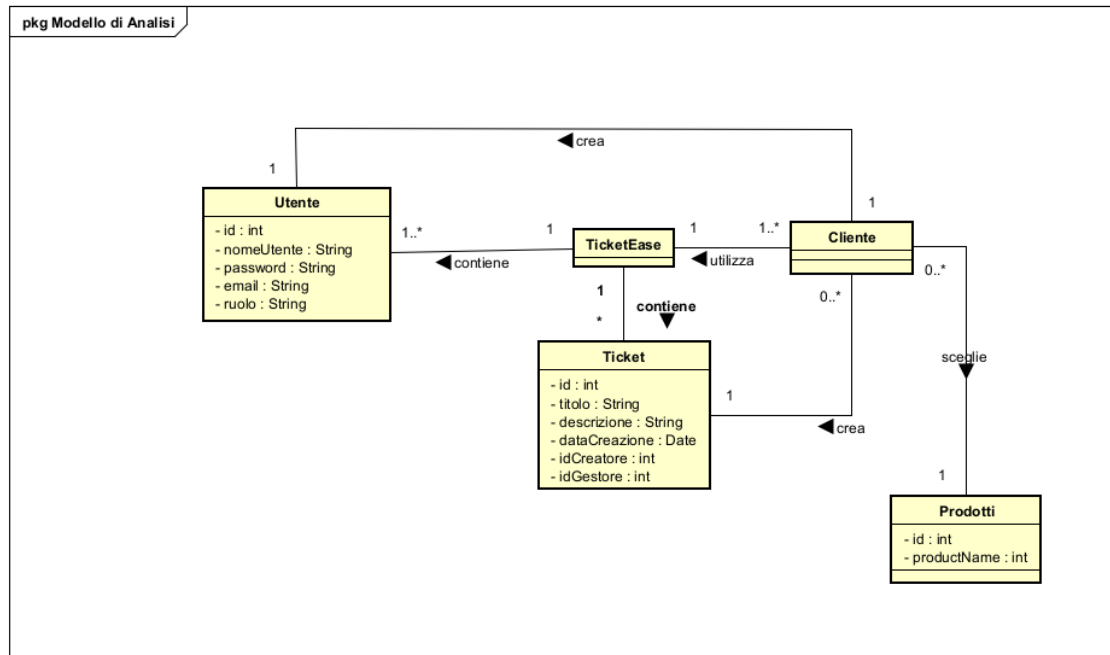
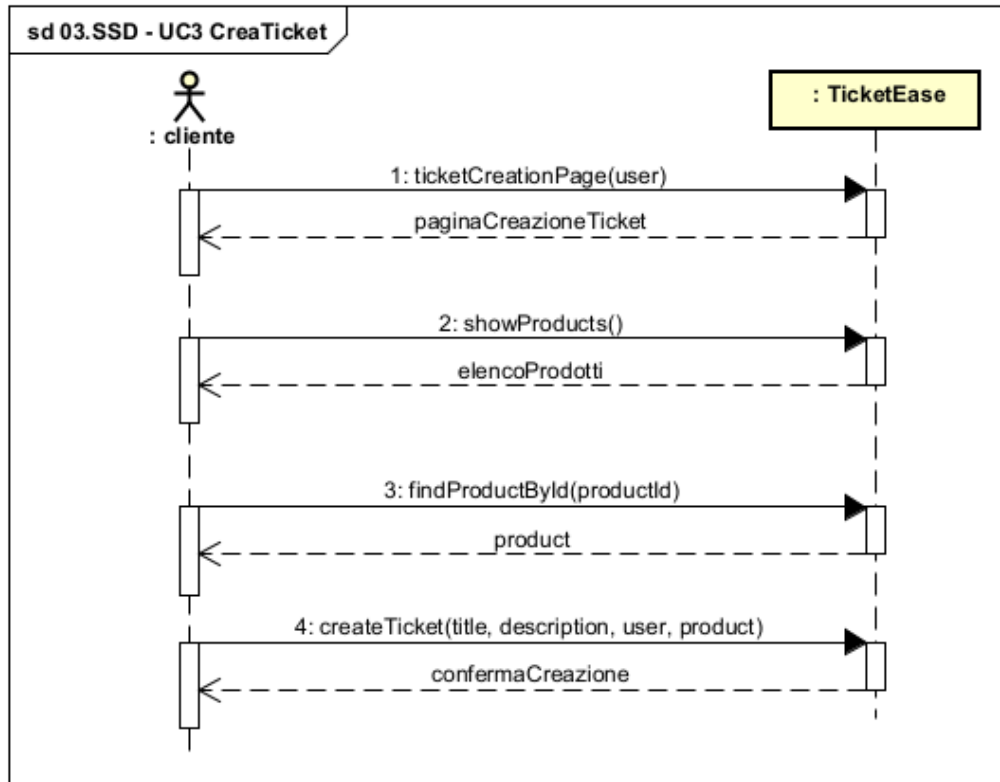


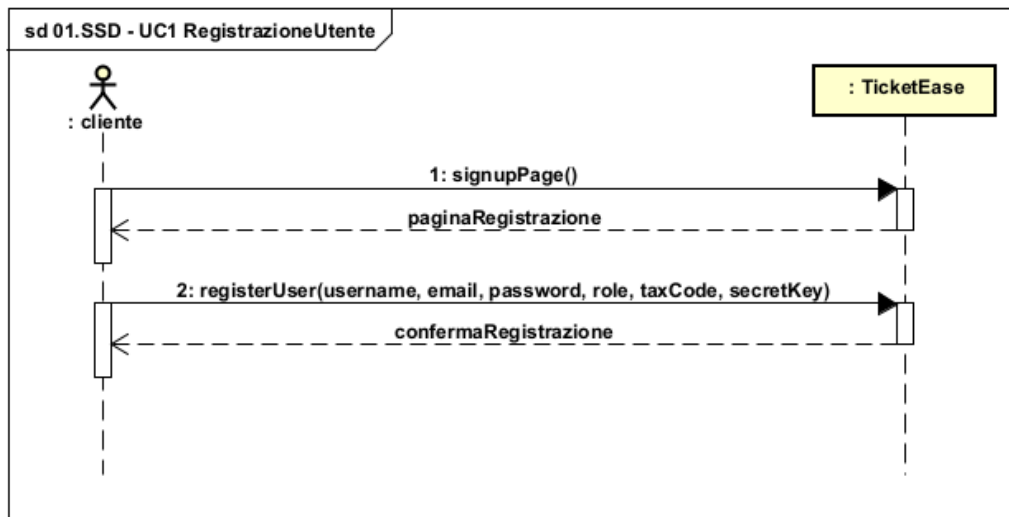
Diagramma di sequenza di sistema

Il passo successivo all'analisi OO è la creazione del diagramma di sequenza (SSD) con lo scopo di mostrare il succedersi degli eventi di input e output per lo scenario principale di successo del caso d'uso UC3. Si avrà dunque:

TicketEase

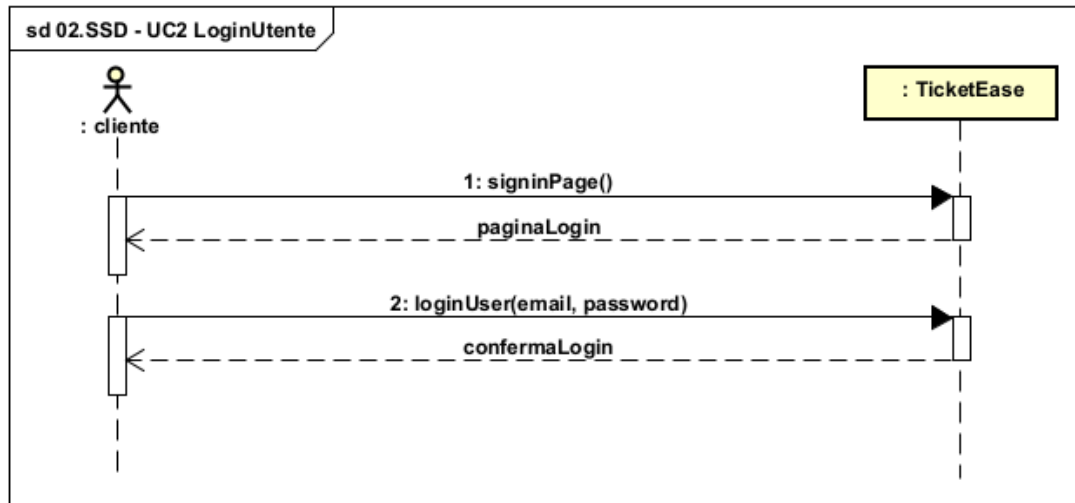


Per quanto riguarda il caso d'uso UC1, il diagramma di sequenza di sistema è il seguente:



Mentre, per il caso d'uso UC2, necessario per lo svolgimento di UC3, il diagramma di sequenza di sistema è il seguente:

TicketEase



Contratti delle operazioni

Attraverso i Contratti vengono adesso illustrate le principali operazioni di sistema che si occupano di gestire gli eventi di sistema individuati negli SSD.

CONTRATTO CO1 – Pagina di registrazione

| | |
|-----------------|---|
| Operazione | signupPage() |
| Riferimenti | UC1 |
| Pre-condizioni | - |
| Post-condizioni | L'utente viene reindirizzato alla pagina di registrazione |

CONTRATTO CO2 – Registrazione utente

| | |
|-----------------|---|
| Operazione | registerUser(username : String, email : String, password : String, role : String, taxCode : String, secretKey : String) |
| Riferimenti | UC1 |
| Pre-condizioni | L'utente inserisce correttamente le credenziali. |
| Post-condizioni | L'utente viene reindirizzato alla pagina di menu del cliente |

CONTRATTO CO3 – Pagina di login

| | |
|-----------------|---|
| Operazione | signinPage(email : String, password : String) |
| Riferimenti | UC2 |
| Pre-condizioni | - |
| Post-condizioni | L'utente viene reindirizzato alla pagina di login |

| CONTRATTO CO4 - Login Utente | |
|------------------------------|--|
| Operazione | loginUser(email : String, password : String) |
| Riferimenti | UC2 |
| Pre-condizioni | L'utente è registrato |
| Post-condizioni | L'utente viene reindirizzato al menu utente |

| CONTRATTO CO3 – Pagina di creazione ticket | |
|--|--|
| Operazione | createTicket(user : User) |
| Riferimenti | UC3 |
| Pre-condizioni | L'utente ha effettuato il login. |
| Post-condizioni | <ul style="list-style-type: none">- Vengono mostrati i prodotti selezionabili per l'apertura del ticket- Sono mostrati i vari campi compilabili per la creazione del ticket |

| CONTRATTO CO4 – Mostra prodotti | |
|---------------------------------|--|
| Operazione | showProducts() |
| Riferimenti | UC3 |
| Pre-condizioni | È in corso la creazione del ticket |
| Post-condizioni | <ul style="list-style-type: none">- Viene stampata a video la lista dei prodotti presenti su Ticketease. |

| CONTRATTO CO5 – Selezione prodotto | |
|------------------------------------|------------------------------------|
| Operazione | findProductById(productId: int) |
| Riferimenti | UC3 |
| Pre-condizioni | È in corso la creazione del ticket |
| Post-condizioni | - |

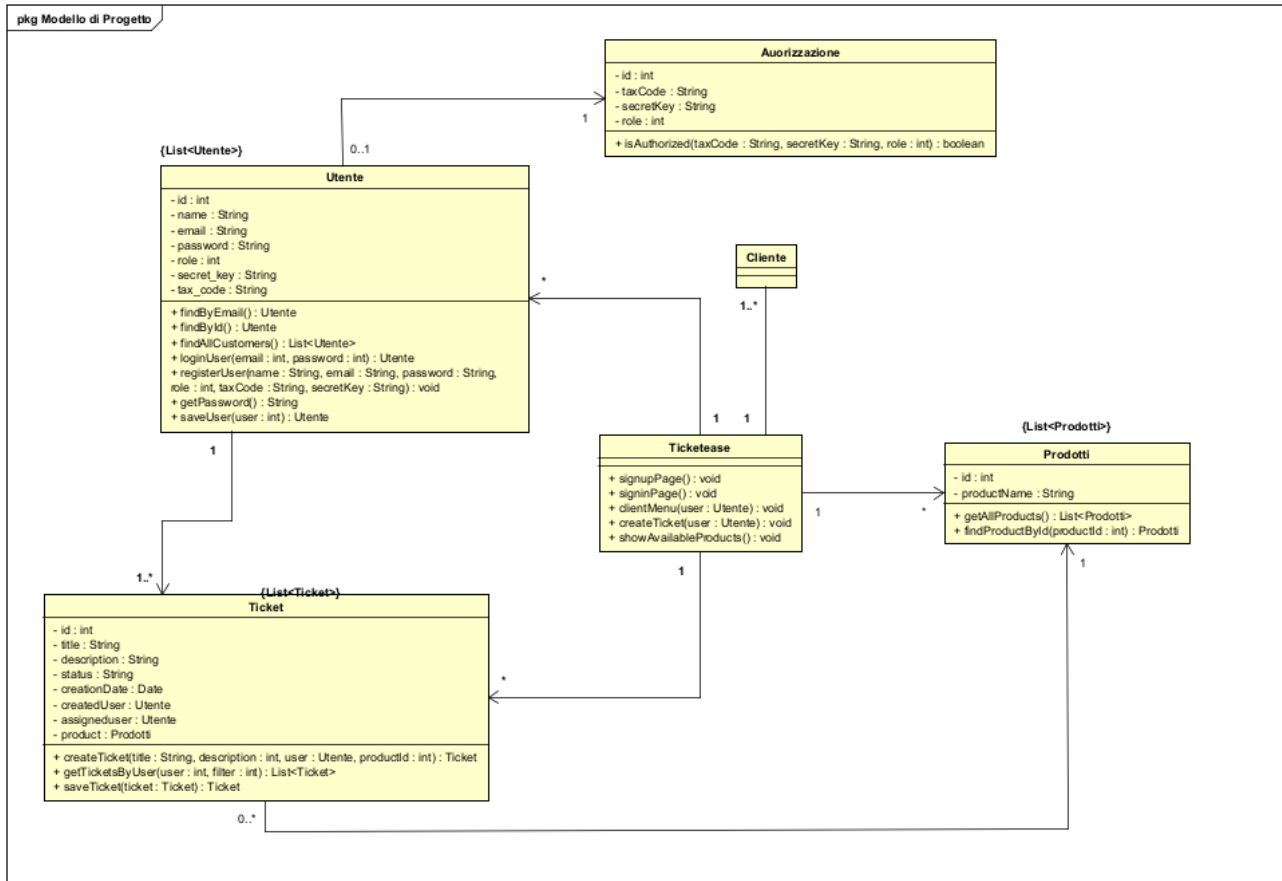
| CONTRATTO CO6 – Creazione del ticket | |
|--------------------------------------|---|
| Operazione | createTicket(title : String, description : String, user |
| Riferimenti | UC3 |
| Pre-condizioni | È in corso la creazione del ticket |
| Post-condizioni | <ul style="list-style-type: none">- Il ticket di assistenza viene creato all'interno del database.- Viene restituito un messaggio di conferma. |

- Progettazione

La disciplina di up che si occupa della definizione degli oggetti software, delle loro responsabilità e del modo in cui questi agiscono per soddisfare i requisiti individuati nei punti precedenti, è la

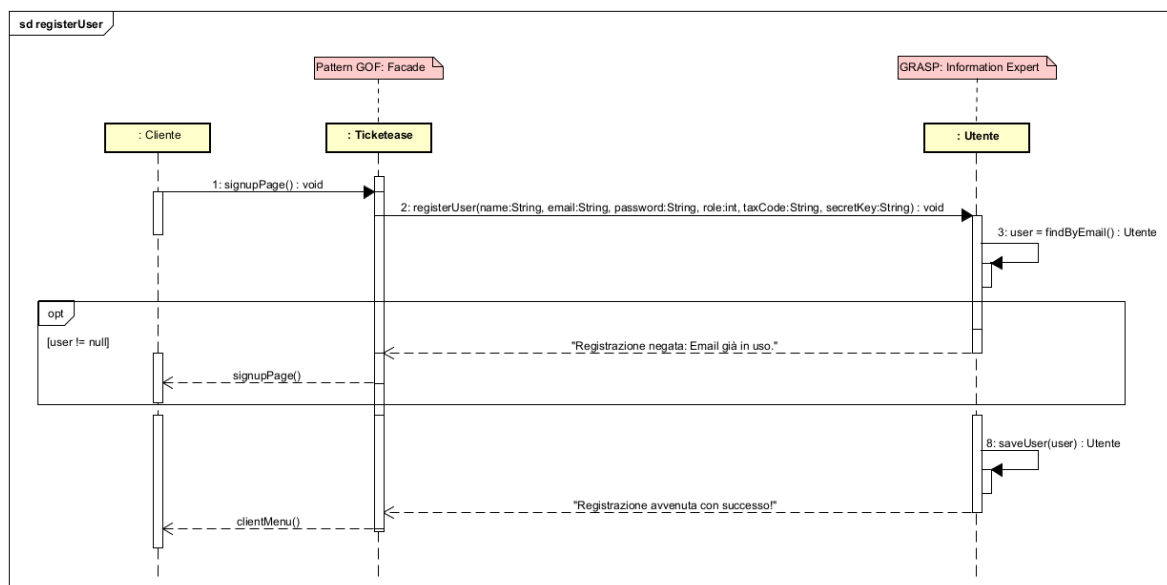
progettazione orientata agli oggetti. Il modello di progetto è l'insieme dei diagrammi che descrivono la progettazione logica sia da un punto di vista dinamico (Diagrammi di interazione) che statico (Diagramma delle Classi). A seguire, figurano i diagrammi di iterazione più significativi e il diagramma delle classi relativi al caso d'uso UC7 realizzato dopo uno studio scrupoloso degli elaborati composti in precedenza.

Diagramma delle classi

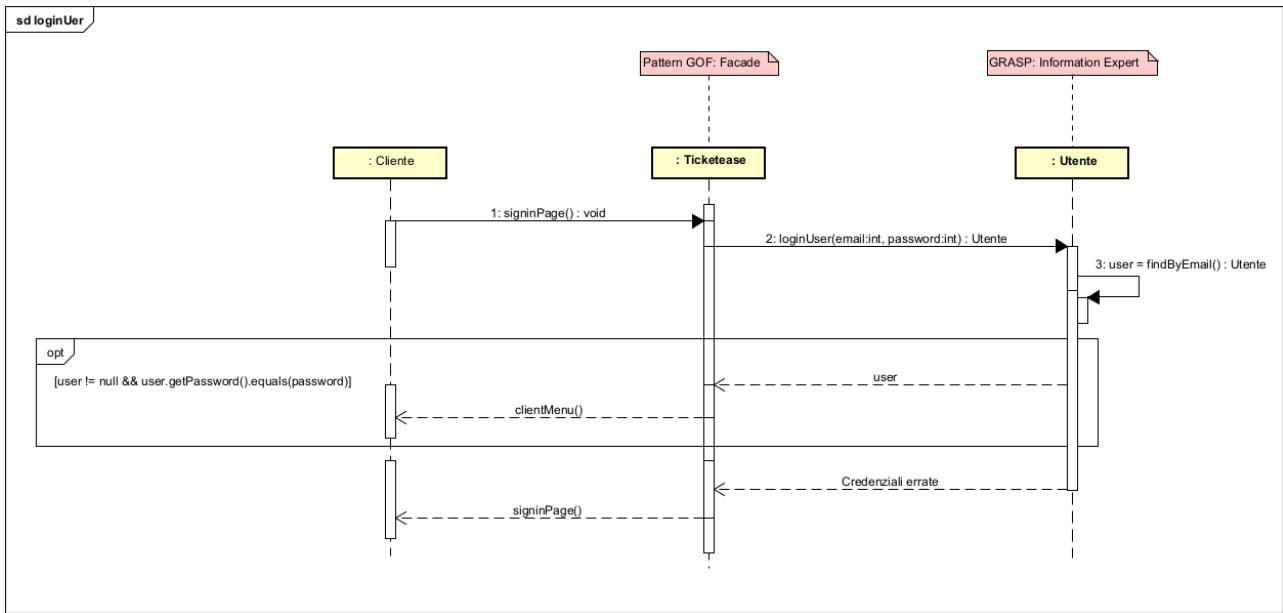


Diagrammi di Sequenza

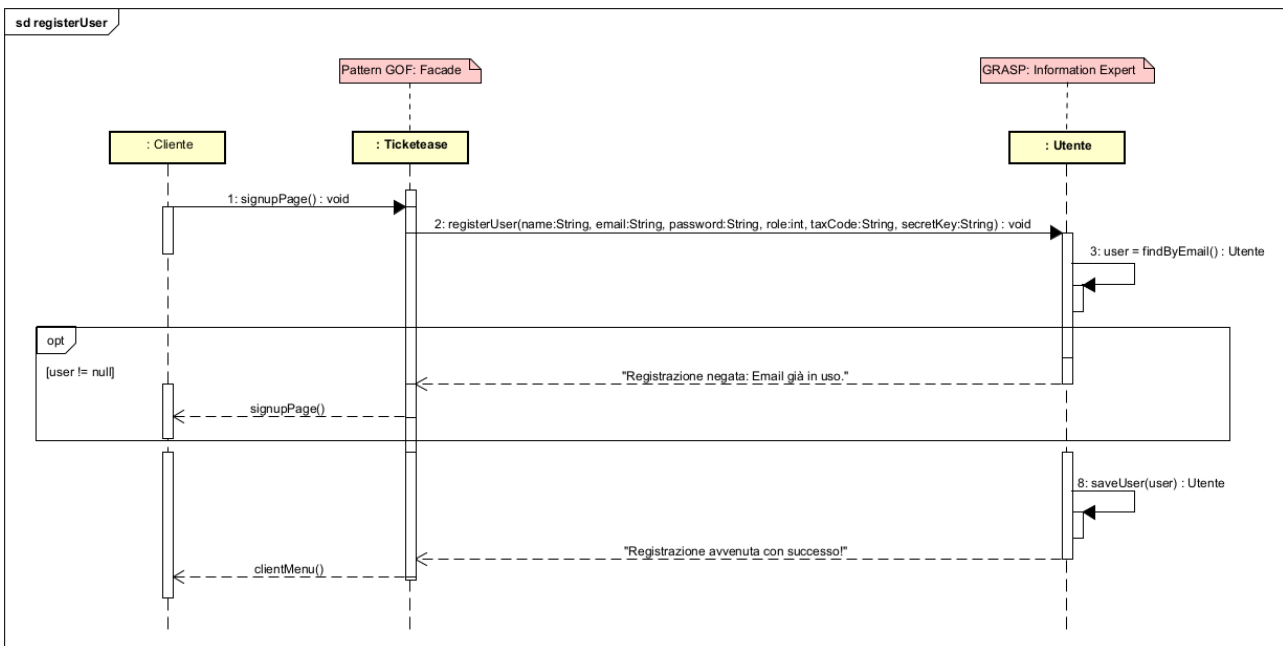
UC1:



UC2:



Per quanto riguarda il caso d'uso UC3, il diagramma SD è il seguente:



- Refactoring

In fase di refactoring non sono state apportate modifiche al modello di dominio. È stato tuttavia migliorato il codice dal punto di vista di performance e leggibilità.

3. Iterazione 2

- *Introduzione*

Dopo la prima iterazione, si procede con la seconda iterazione. Durante questa fase, il focus sarà incentrato sullo scenario di successo del caso d'uso UC7. Inoltre, verrà discusso anche lo scenario di successo del caso d'uso UC1, in quanto è stato implementato un meccanismo di riconoscimento per gli amministratori e per i tech.

Un altro aspetto chiave di questa iterazione riguarda lo scenario di successo del caso d'uso UC4, poiché è stata introdotta la funzionalità di notifica ai tech dopo l'assegnazione di un ticket. Questa implementazione migliora il flusso di lavoro, garantendo una comunicazione tempestiva ed efficace all'interno del sistema.

- *Analisi orientata agli Oggetti*

In questa iterazione, come nella precedente saranno utilizzati strumenti come: Modello di Dominio, Sequence System Diagram (SSD), e contratti delle operazioni. Verranno analizzati anche i cambiamenti rispetto alla fase precedente.

Modello di Dominio

Analizzando il caso d'uso UC7 i nuovi attori e una nuova classe concettuale, rispetto all'iterazione precedente:

Rispetto alla prima iterazione, sono ora presenti gli attori "**amministratore**" e "**tecnico**", i quali tuttavia sono solo figurativi, poiché la classe concettuale "**Utente**" identifica, attraverso l'attributo "**ruolo**", le diverse figure di utilizzo della piattaforma.

Inoltre, è stata introdotta la nuova classe concettuale "**Notifica**" utilizzata per salvare le notifiche che vengono inviate al tecnico dopo l'assegnazione di un ticket. Per implementare questa funzionalità, è stato adottato il pattern GOF Observer, che consente di gestire in modo efficace l'invio automatico delle notifiche in risposta a determinati eventi nel sistema.

TicketEase

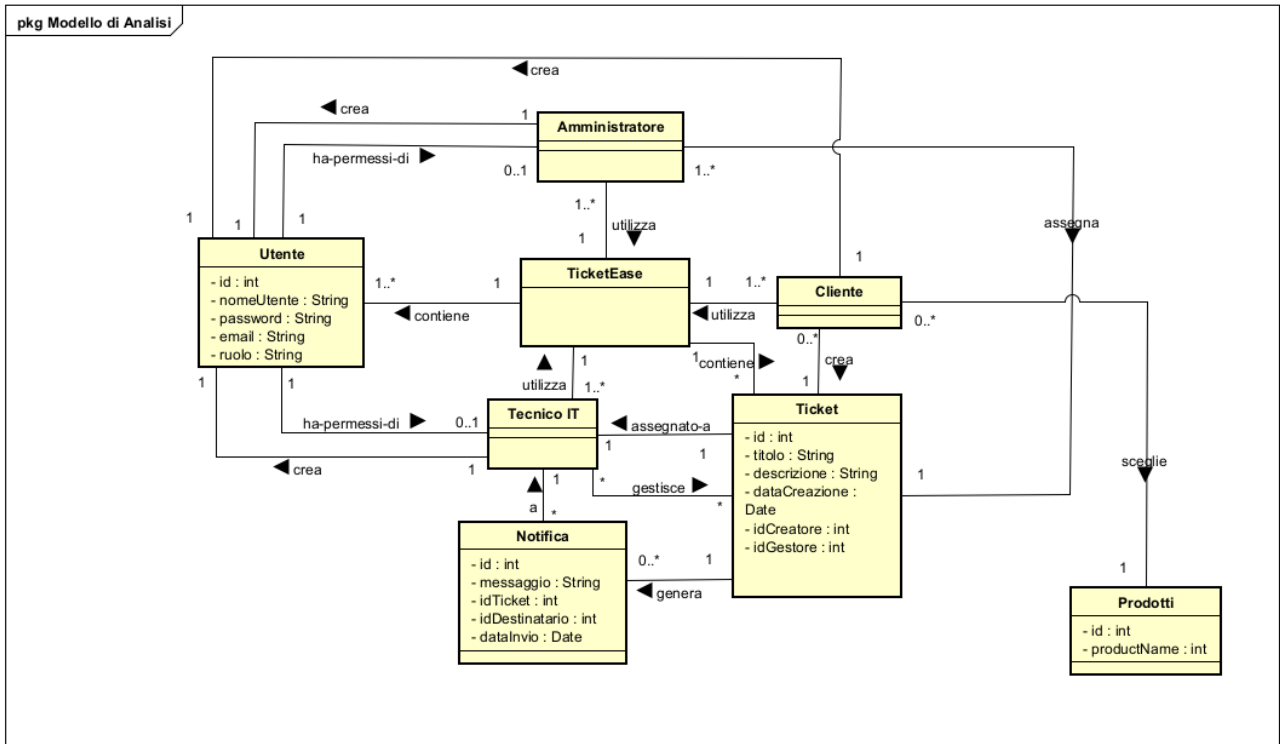
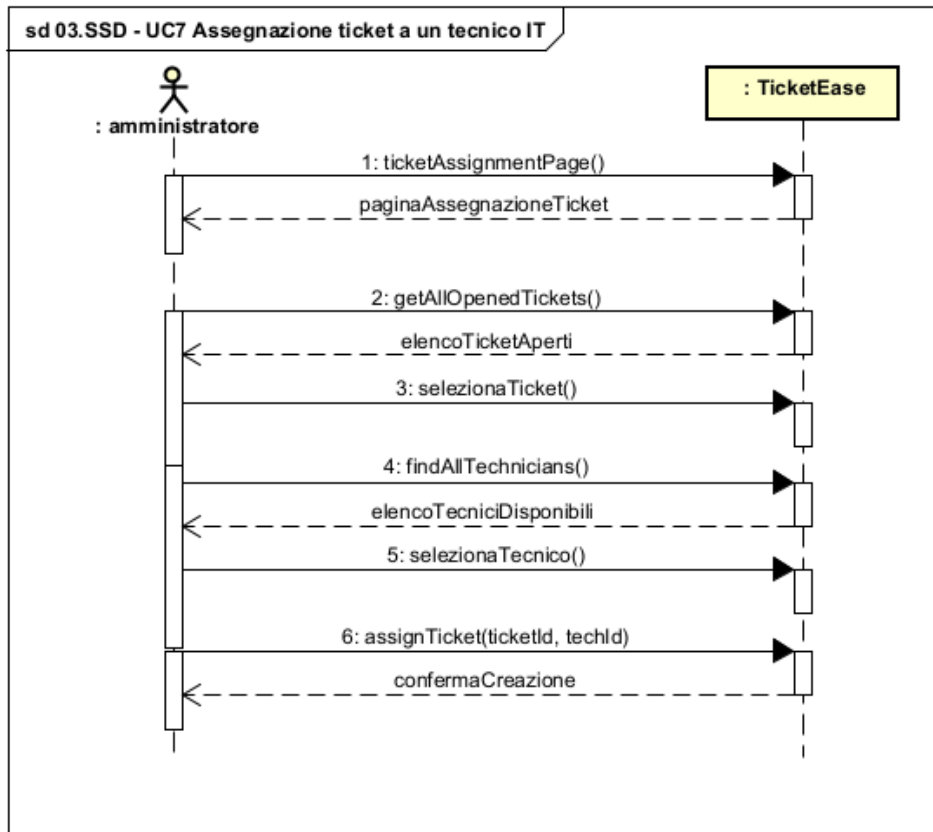


Diagramma di sequenza di sistema

In seguito, è stato creato il diagramma di sequenza di sistema (SSD) per il caso d'uso UC7.



Non è stato realizzato il diagramma del caso d'uso UC4 in quanto tale caso non prevede un'interazione diretta tra l'utente e il sistema. Le operazioni coinvolte avvengono interamente tra i vari metodi chiamati nel contesto del caso d'uso UC7.

Tuttavia, il comportamento di UC4 verrà analizzato successivamente nel diagramma di sequenza, dove saranno evidenziate le interazioni tra le componenti coinvolte.

Anche l'aggiornamento del caso d'uso UC1 verrà analizzato in dettaglio successivamente.

Contratti delle operazioni

Attraverso i Contratti vengono adesso illustrate le principali operazioni di sistema che si occupano di gestire gli eventi di sistema individuati nell'SSD.

| CONTRATTO CO1 – Pagina di registrazione | |
|---|---|
| <i>Operazione</i> | signupPage() |
| <i>Riferimenti</i> | UC1 |
| <i>Pre-condizioni</i> | - |
| <i>Post-condizioni</i> | L'utente viene reindirizzato alla pagina di registrazione |

| CONTRATTO CO2 – Registrazione utente | |
|--------------------------------------|---|
| <i>Operazione</i> | registerUser(username : String, email : String, password : String, role : String, taxCode : String, secretKey : String) |
| <i>Riferimenti</i> | UC1 |
| <i>Pre-condizioni</i> | <ul style="list-style-type: none">- L'utente inserisce correttamente le credenziali.- L'utente inserisce il proprio codice fiscale e chiave segreta e questi sono presenti nella tabella degli utenti autorizzati. |
| <i>Post-condizioni</i> | L'utente viene reindirizzato alla pagina di menu dell'amministratore |

| CONTRATTO CO3 – Pagina di assegnazione ticket | |
|---|--|
| <i>Operazione</i> | assignTicket () |
| <i>Riferimenti</i> | UC7 |
| <i>Pre-condizioni</i> | L'utente ha effettuato il login. |
| <i>Post-condizioni</i> | <ul style="list-style-type: none">- Vengono mostrati all'amministratore i ticket d'assistenza aperti- Sono mostrati i vari campi compilabili per la l'assegnazione del ticket- Viene inviata una notifica al tecnico |

| | |
|--|-------------------------|
| | responsabile del ticket |
|--|-------------------------|

| CONTRATTO CO4 – Elenco dei ticket aperti | |
|--|--|
| Operazione | getAllOpenedTickets() |
| Riferimenti | UC7 |
| Pre-condizioni | È in corso la creazione del ticket |
| Post-condizioni | Viene stampata a video la lista dei prodotti presenti su Ticketease. |

| CONTRATTO CO5 – Selezione prodotto | |
|------------------------------------|---|
| Operazione | findAllTechnicians() |
| Riferimenti | UC7 |
| Pre-condizioni | È in corso la creazione del ticket |
| Post-condizioni | Viene stampata a video la lista dei tecnici presenti su Ticketease. |

| CONTRATTO CO6 – Creazione del ticket | |
|--------------------------------------|---|
| Operazione | assignTicket(title : String, description : String, user |
| Riferimenti | UC3 |
| Pre-condizioni | È in corso la creazione del ticket |
| Post-condizioni | <ul style="list-style-type: none">- Il ticket di assistenza viene creato all'interno del database.- Viene restituito un messaggio di conferma. |

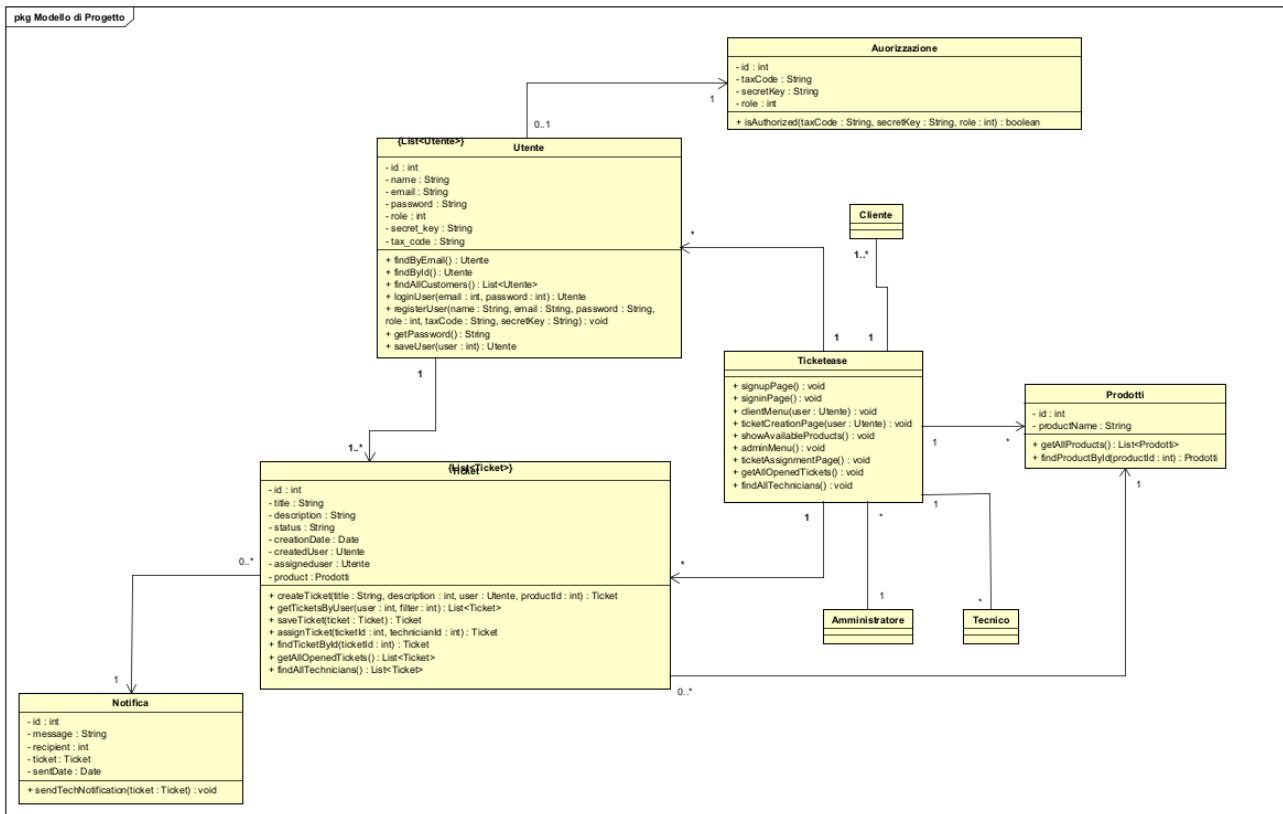
- *Progettazione*

Nella seconda iterazione è stato introdotto per la prima volta il pattern **Observer (GoF)** per gestire l'invio di notifiche automatiche nel sistema. L'obiettivo iniziale era notificare i tecnici ogni volta che un ticket veniva assegnato, migliorando la reattività e la comunicazione interna del sistema.

Per questo scopo è stata progettata e implementata un'interfaccia comune che consente di definire una logica di notifica generica, applicabile a diverse categorie di destinatari. Questa interfaccia si è rivelata utile anche per le iterazioni successive, dove è stata estesa per includere altri osservatori, come i clienti.

Inoltre, come nell'iterazione precedente, vengono mostrati tutti i diagrammi che servono a descrivere la progettazione logica da un punto di vista dinamico (diagrammi di interazione) e da un punto di vista statico (diagramma delle classi).

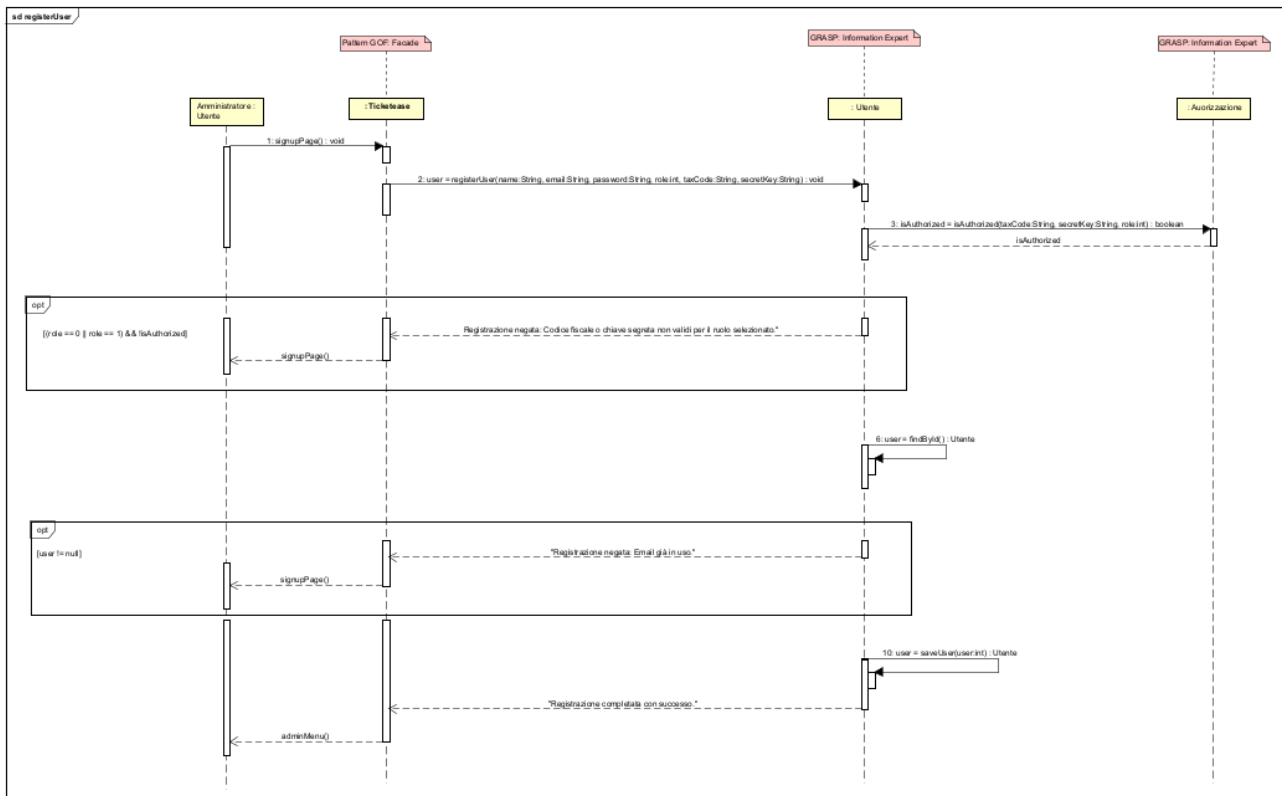
Diagramma delle classi



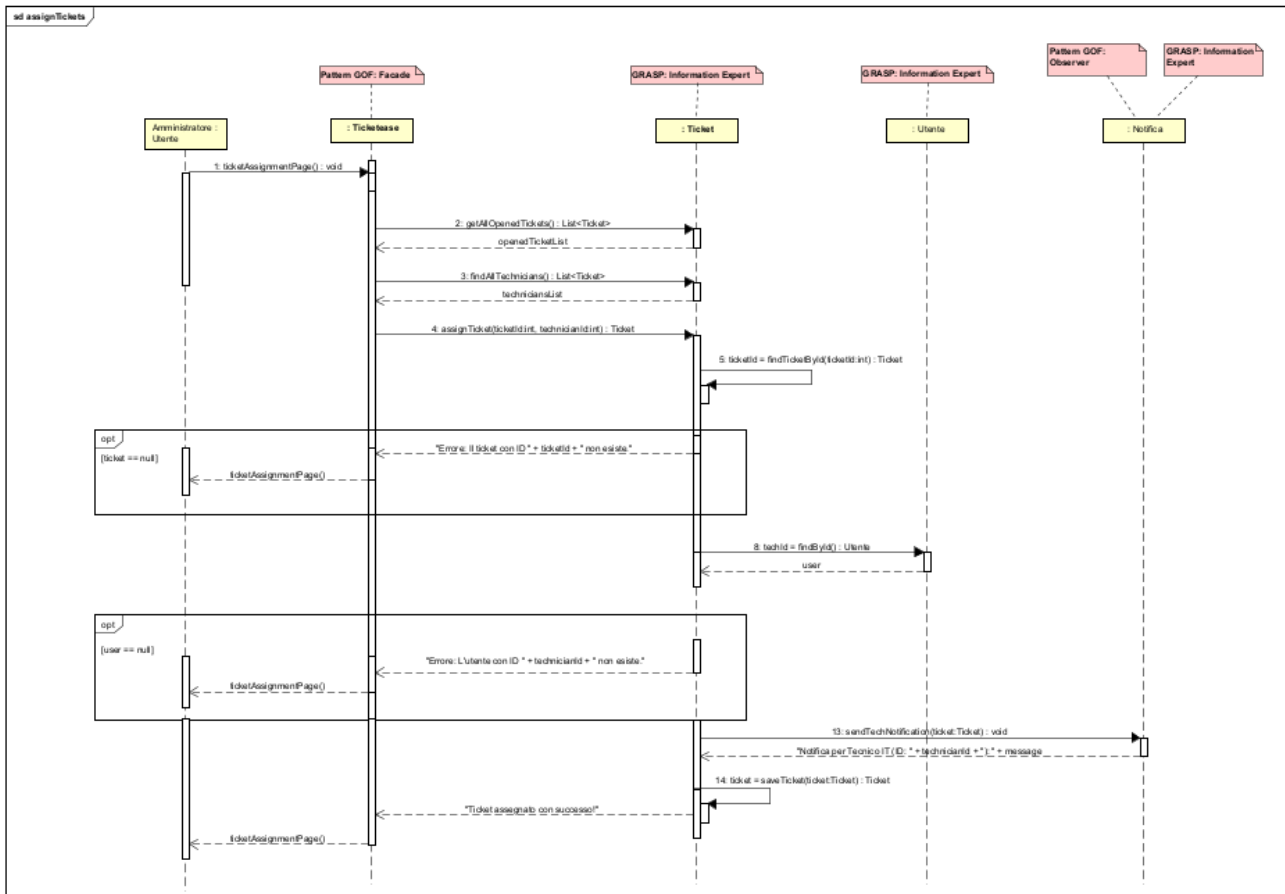
Diagrammi di Sequenza

In questa sezione vengono mostrati gli SD per il caso d'uso UC7. Non vengono mostrati quelli già presentati nell'iterazione 1, tranne per il caso UC1 che presenta l'aggiunta di un meccanismo di identificazione per gli utenti amministratori e tecnici.

SD di UC1 - Registrazione



SD di UC7 – Assegnazione ticket



- *Refactoring*

In fase di refactoring non sono state apportate modifiche al modello di dominio. È stato tuttavia migliorato il codice dal punto di vista di performance e leggibilità e sono state gestite le eccezioni.

4. Iterazione 3

- *Introduzione*

Durante questa terza iterazione, ci si concentrerà sull'analisi e sull'implementazione del caso d'uso **UC6: Chiusura del ticket**, che consente al tecnico IT di aggiornare un ticket modificandone lo stato in "chiuso" e inserendo un messaggio di risoluzione.

Inoltre, durante questa terza iterazione è stato aggiunto il caso d'uso "UC9: Visualizzazione dei ticket assegnati" ai casi d'uso già identificati nella fase di progetto. Tale aggiunta si è resa necessaria poiché il comportamento descritto in UC9 veniva utilizzato implicitamente all'interno dell'iterazione, rendendo quindi opportuno esplicitarlo e formalizzarlo nel modello dei casi d'uso.

| Attore | Obiettivo | Caso d'uso |
|------------|---------------------------------|---|
| Tecnico IT | Visualizzare i ticket assegnati | UC9. Visualizzazione dei ticket assegnati |

UC9. Visualizzazione dei ticket assegnati

Scenario di successo

1. Il tecnico IT accede all'area personale.
2. Seleziona la voce "Visualizza i ticket assegnati".
3. Il sistema mostra i ticket aperti assegnati al tecnico IT.

Saranno quindi presenti anche i due casi d'uso non dettagliati **UC9 e UC4** in cui è descritto solo lo scenario di successo.

Per permettere l'esecuzione di questa funzionalità, è necessario che l'utente acceda al sistema autenticandosi come tecnico IT tramite il caso d'uso **UC2: Login utente**. La modalità di registrazione, invece, è già stata affrontata e implementata durante la seconda iterazione.

- *Analisi orientata agli Oggetti*

Al fine di descrivere il dominio da un punto di vista ad oggetti e gestire ulteriori requisiti, saranno utilizzati nuovamente gli stessi strumenti delle iterazioni precedenti.

Tuttavia, esaminando il caso d'uso **UC6**, si è deciso di modificare il **Modello di Dominio** aggiungendo la classe "**Stato Ticket**", al fine di tenere traccia di tutti i cambiamenti di stato associati a un ticket.

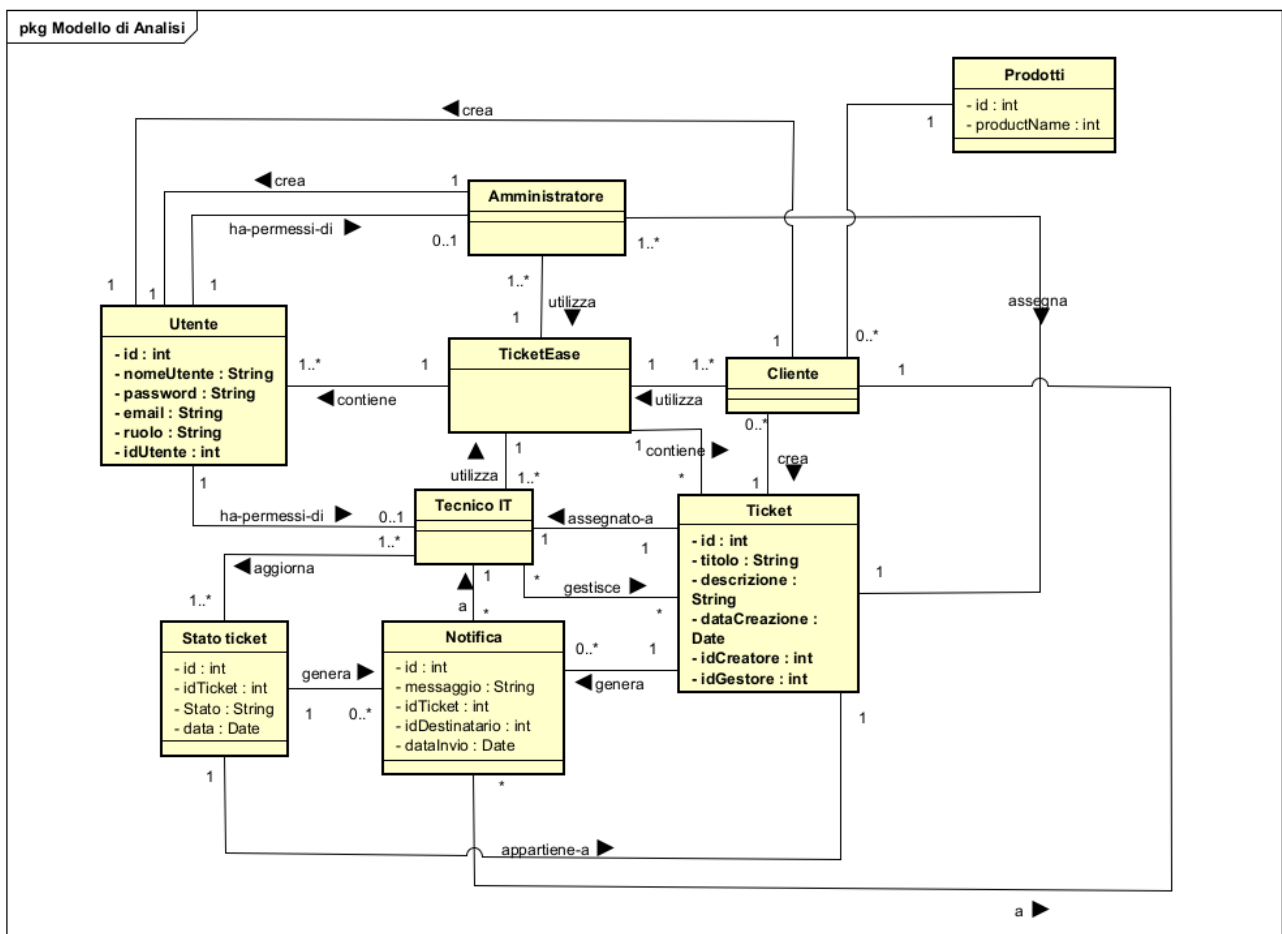
Questa nuova classe è in relazione con la classe "**Ticket**", in quanto ogni cambiamento di stato comporta anche un aggiornamento dello status direttamente all'interno della tabella **Ticket**.

Modello di dominio

Analizzando il caso d'uso UC6 emerge una nuova classe concettuale, rispetto all'iterazione precedente:

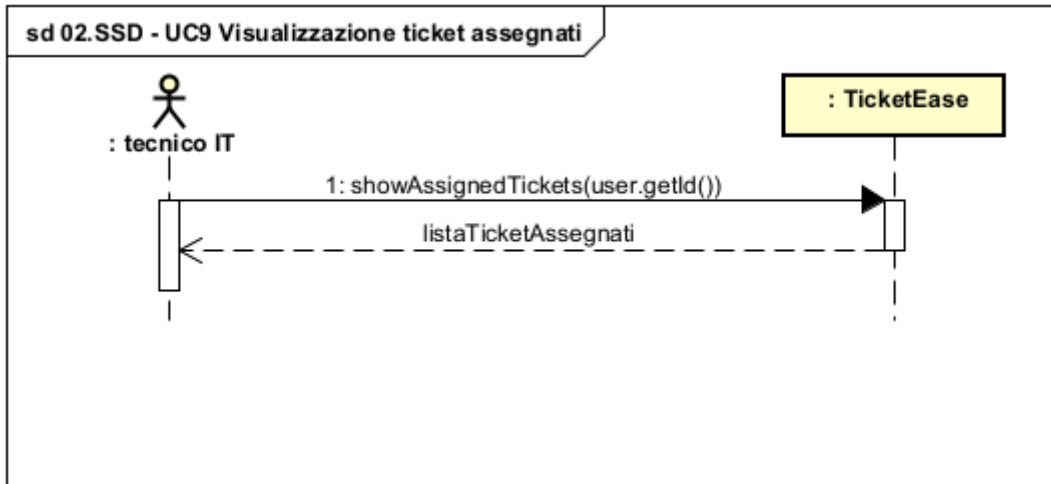
- Stato ticket: contiene tutte le informazioni di un cambio di stato al ticket, come la descrizione dell'aggiornamento e il nuovo che stato.

Dall'integrazione di questa nuova classe, tenendo conto di associazioni e attributi, è stato ricavato il seguente modello di dominio.

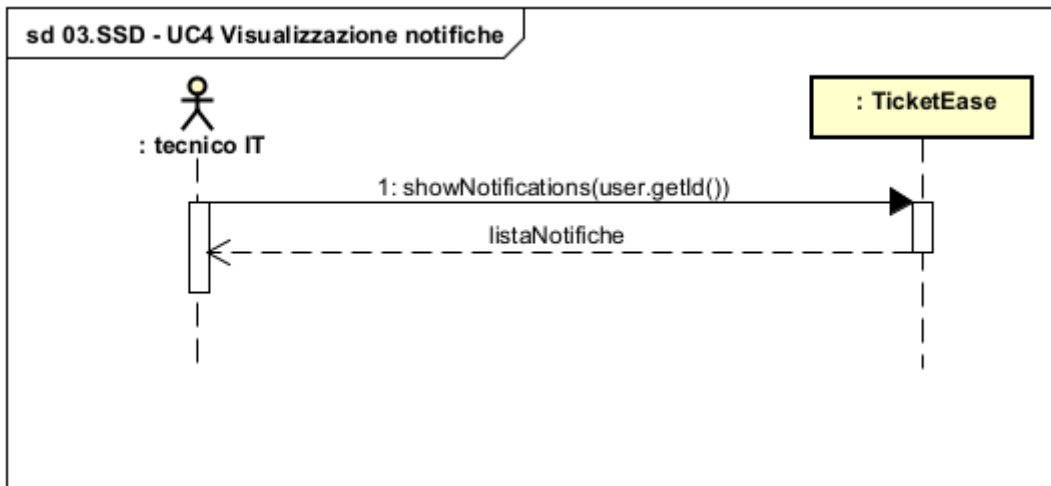


Diagrammi di sequenza di sistema

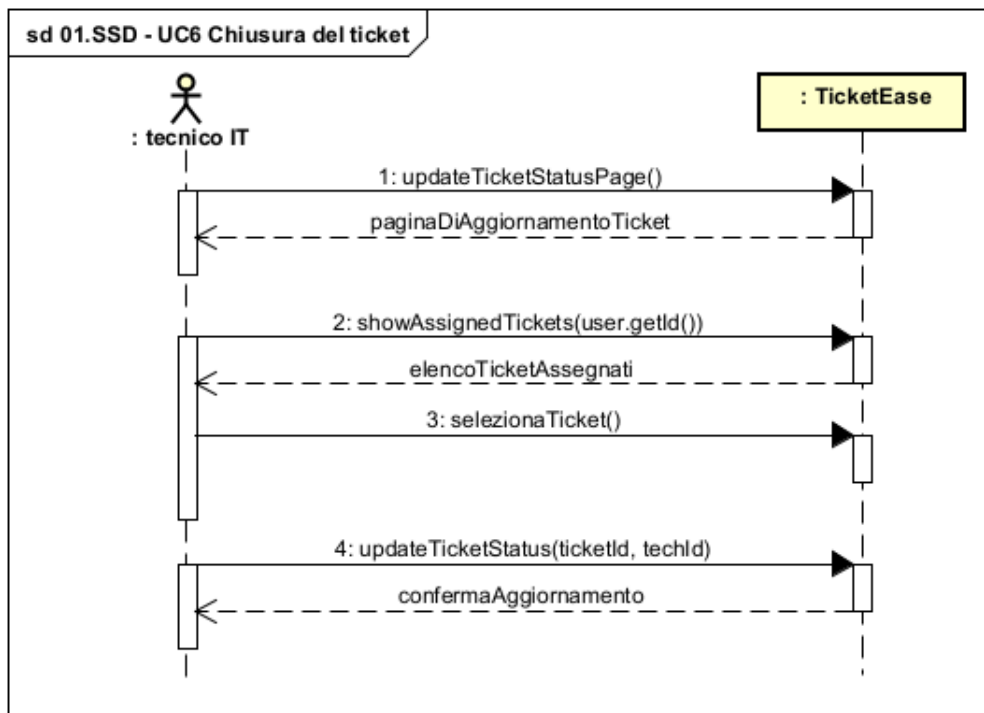
Di seguito il diagramma relativo al caso d'uso UC9.



Caso d'uso UC4:



Caso d'uso UC6:



Contratti delle operazioni

| CONTRATTO CO1 – Lista dei ticket assegnati al tecnico IT | |
|---|--|
| <i>Operazione</i> | showAssignedTickets(techId : int) |
| <i>Riferimenti</i> | UC9 |
| <i>Pre-condizioni</i> | L'utente ha effettuato l'accesso come tecnico IT |
| <i>Post-condizioni</i> | Viene stampata a video la lista dei ticket assegnati al tecnico IT |

| CONTRATTO CO2 – Visualizzazione notifiche | |
|--|---|
| <i>Operazione</i> | showNotifications(userId : int) |
| <i>Riferimenti</i> | UC4 |
| <i>Pre-condizioni</i> | <ul style="list-style-type: none"> - L'utente ha effettuato l'accesso come tecnico IT - Il tecnico IT ha almeno un ticket assegnato |
| <i>Post-condizioni</i> | Viene stampata a video la lista delle notifiche ricevute dal tecnico IT |

| CONTRATTO CO3 – Pagina di aggiornamento stato del ticket | |
|---|--|
| <i>Operazione</i> | updateTicketStatusPage () |
| <i>Riferimenti</i> | UC7 |
| <i>Pre-condizioni</i> | <ul style="list-style-type: none"> - Il tecnico IT ha effettuato il login - Il tecnico IT ha selezionato l'opzione "Aggiorna stato di un ticket" |
| <i>Post-condizioni</i> | Vengono mostrati al tecnico i ticket assegnati con le varie opzioni di cambiamento stato del ticket. |

| CONTRATTO CO4 – Aggiornamento stato e chiusura del ticket | |
|--|--|
| <i>Operazione</i> | updateTicketStatus(ticketId : int, newStatus : String, updatedBy : int, statusDescription : String) |
| <i>Riferimenti</i> | UC7 |
| <i>Pre-condizioni</i> | <ul style="list-style-type: none"> - È in corso l'aggiornamento del ticket - Il tecnico IT ha selezionato l'opzione come stato "3. Chiuso". |
| <i>Post-condizioni</i> | <ul style="list-style-type: none"> - Il ticket viene aggiornato e impostato allo stato "chiuso" - L'utente riceve una notifica che avvisa l'avvenuta chiusura del ticket |

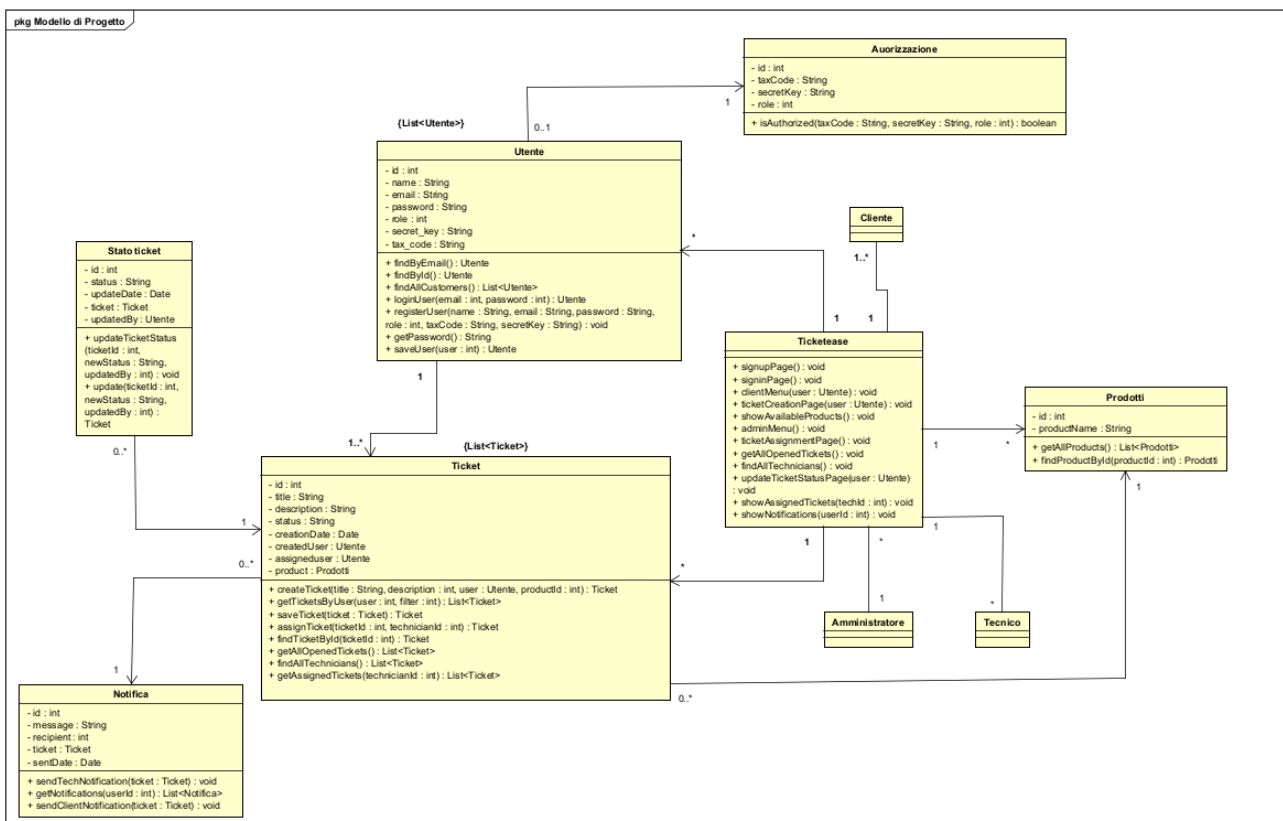
- Progettazione

Nuovamente, gli elaborati principali presi in considerazione sono i diagrammi che descrivono la progettazione logica sia da un punto di vista dinamico (Diagrammi di Interazione) che da un punto di vista statico (Diagramma delle Classi).

Anche in questa terza iterazione è stato implementato il pattern **Observer** (GoF) per la gestione delle notifiche da inviare agli utenti.

L'implementazione sfrutta l'interfaccia definita durante la seconda iterazione, permettendo così di inviare notifiche a classi specifiche che si registrano come osservatori. In questo contesto, è stato gestito l'invio di notifiche verso una particolare categoria di destinatari: i **clienti**.

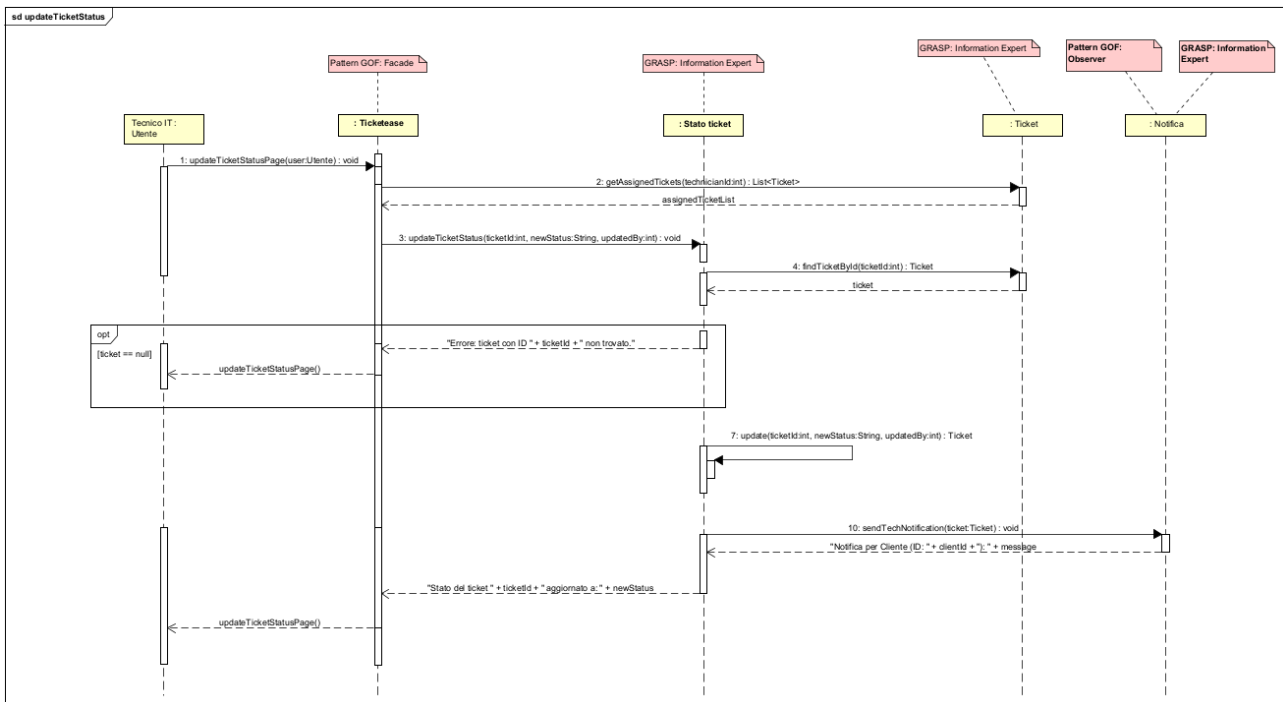
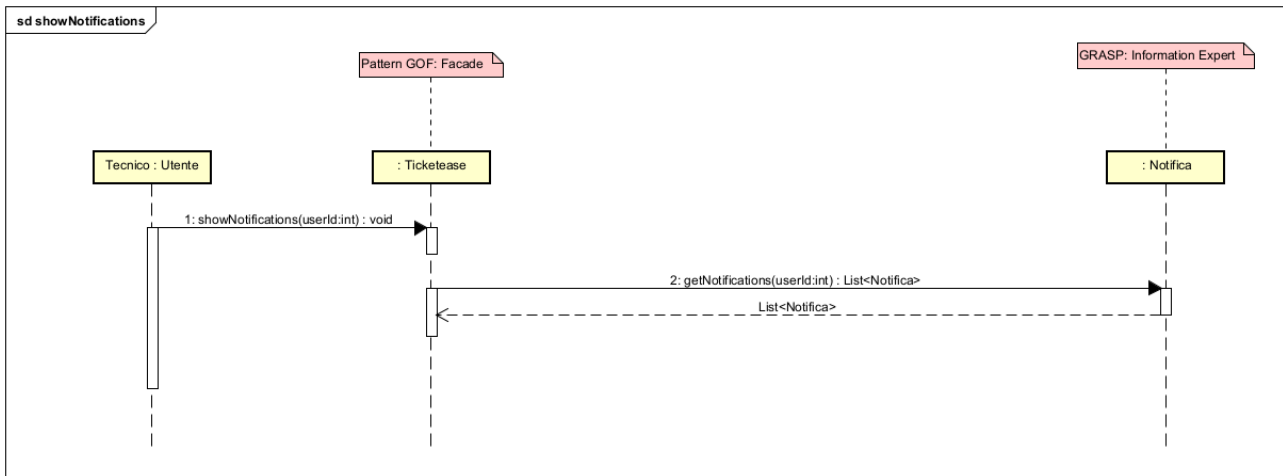
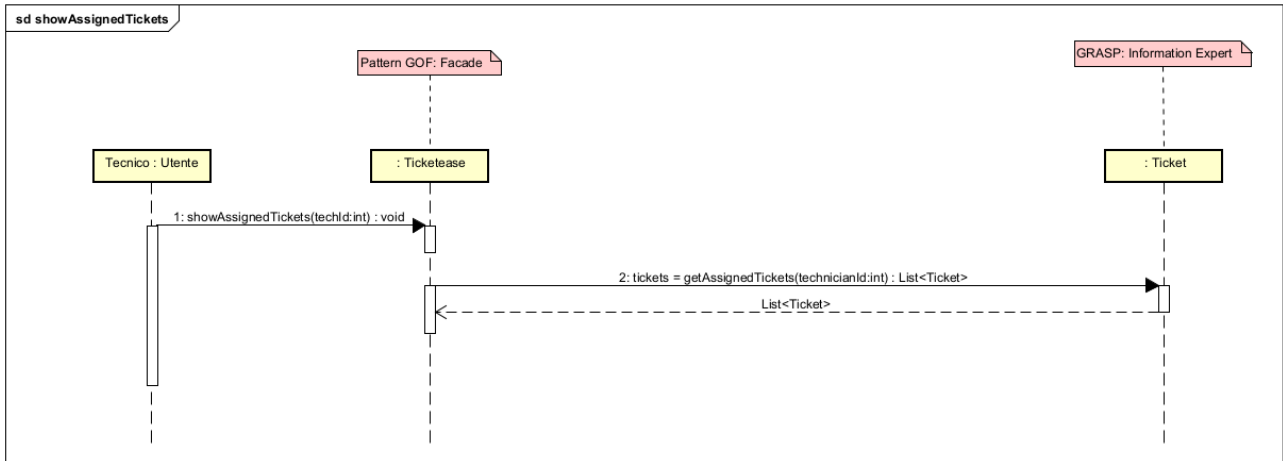
Diagramma delle classi



Diagrammi di sequenza

In questa sezione vengono mostrati gli SD per il caso d'uso UC6. Non vengono mostrati quelli già presentati nelle iterazioni precedenti.

TicketEase



- *Refactoring*

In fase di refactoring sono state apportate modifiche minime al codice esistente, focalizzandosi principalmente su attività di **refactoring mirato**. L'intervento più significativo ha riguardato la **migrazione del livello di persistenza ad Hibernate**, con l'obiettivo di rendere più efficiente e manutenibile la gestione dei dati.

La transizione ha comportato l'adattamento delle classi del dominio attraverso l'aggiunta di annotazioni JPA, la sostituzione delle query SQL manuali con operazioni basate su EntityManager.

Tutte queste modifiche sono state effettuate preservando la logica applicativa preesistente, senza impattare sull'interfaccia utente o sul comportamento esterno del sistema.

5. Iterazione 4

- *Introduzione*

Durante questa quarta iterazione, ci si concentrerà sull'analisi e l'implementazione del caso d'uso UC8: Visualizzazione storico stato ticket

Il caso d'uso è rivolto agli amministratori e ai tecnici, che hanno la necessità di consultare lo storico delle modifiche di stato relative ai ticket presenti nel sistema. Si tratta di una funzionalità accessibile esclusivamente agli utenti autorizzati.

Per semplicità, in questa iterazione non verranno trattati i casi d'uso precedenti relativi all'autenticazione o alla gestione del contesto utente: si presuppone che l'utente sia già autenticato e che abbia accesso all'area riservata. La visualizzazione dello storico rappresenta una delle voci del menu disponibili per l'utente all'interno della sua area funzionale.

Questa funzionalità è particolarmente utile per consultare lo storico delle risoluzioni passate, così da poter tracciare pattern ricorrenti o recuperare informazioni utili in caso di future segnalazioni simili, favorendo una gestione più efficiente e consapevole dei ticket.

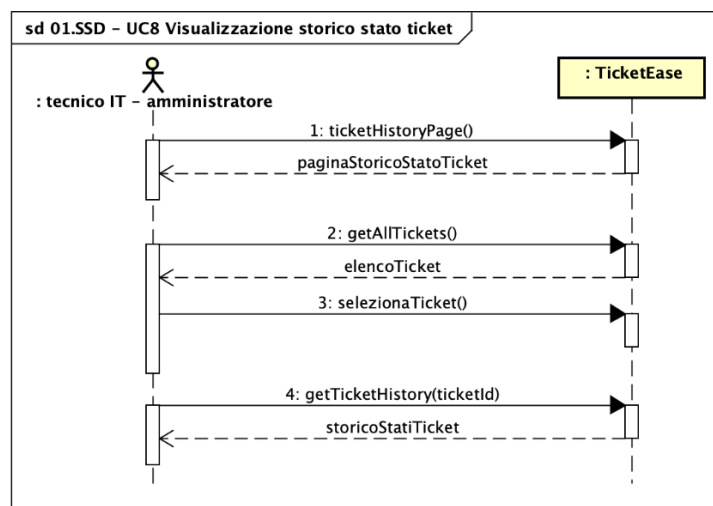
- *Analisi orientata agli Oggetti*

Per questa quarta iterazione non sono previste variazioni significative al modello di dominio rispetto alle iterazioni precedenti.

Gli elementi concettuali identificati finora risultano già sufficienti per rappresentare in modo adeguato le entità coinvolte nel caso d'uso UC8: Visualizzazione storico stato ticket.

Di conseguenza, si è scelto di mantenere invariata la struttura del dominio e di riutilizzare gli stessi strumenti e classi concettuali già introdotti nelle iterazioni precedenti.

Diagrammi di sequenza di sistema



Contratti delle operazioni

| CONTRATTO CO1 – Pagina per lo storico degli stati di un ticket | |
|--|---|
| Operazione | ticketHistoryPage() |
| Riferimenti | UC8 |
| Pre-condizioni | L'utente ha effettuato l'accesso come tecnico IT o come amministratore |
| Post-condizioni | Viene mostrata l'opzione che permette di selezionare uno dei ticket disponibili |

| CONTRATTO CO2 – Lista di tutti i ticket del sistema | |
|---|--|
| Operazione | getAllTickets () |
| Riferimenti | UC8 |
| Pre-condizioni | È presente almeno un ticket nel sistema |
| Post-condizioni | Viene stampata a video la lista di tutti i ticket presenti nel sistema |

| CONTRATTO CO3 – Pagina di aggiornamento stato del ticket | |
|--|--|
| Operazione | getTicketHistory(ticketId : int) |
| Riferimenti | UC8 |
| Pre-condizioni | Il ticket selezionato ha effettuato almeno un cambiamento di stato |
| Post-condizioni | Viene mostrato lo storico di tutti i cambiamenti di stato effettuati per il ticket selezionato |

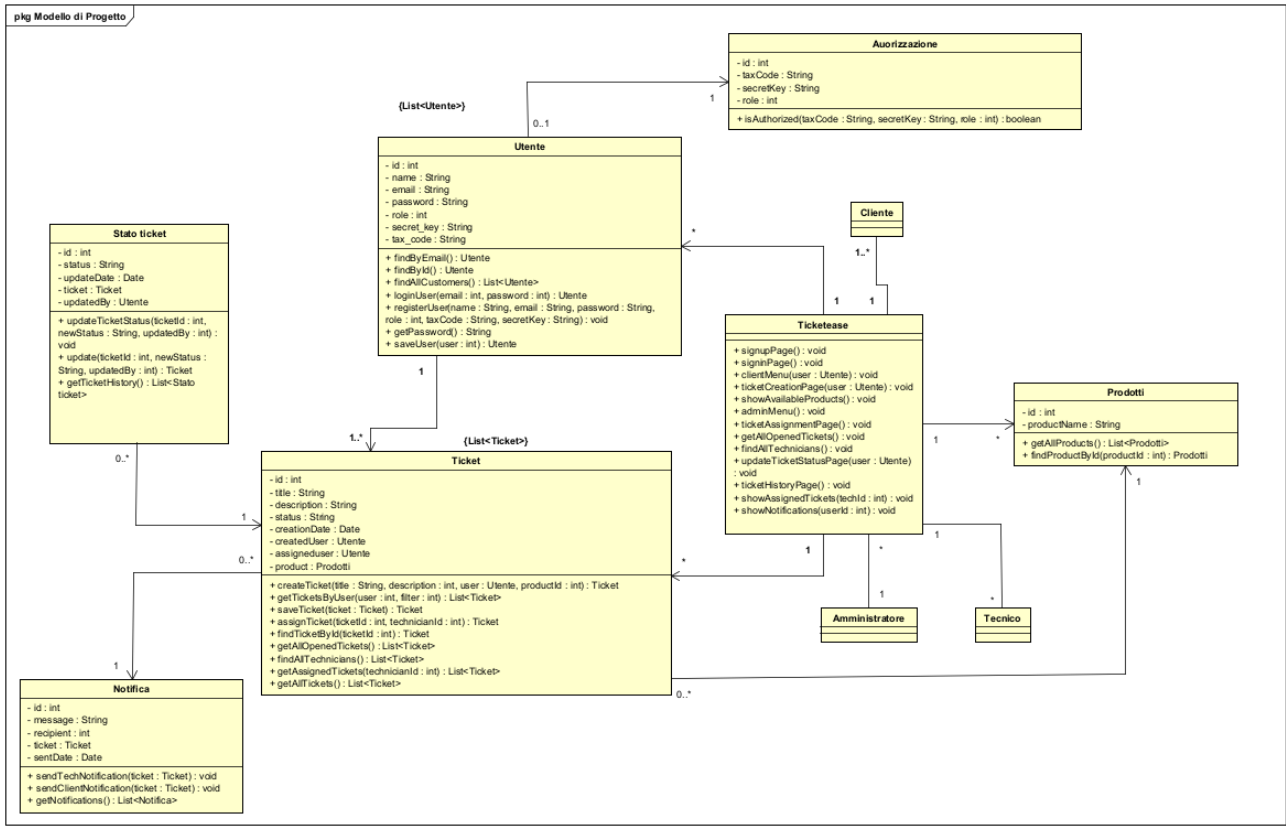
- *Progettazione*

Nuovamente, gli elaborati principali presi in considerazione sono i diagrammi che descrivono la progettazione logica, sia da un punto di vista dinamico (Diagrammi di Interazione), sia da un punto di vista statico (Diagramma delle Classi).

In questa ultima iterazione verranno riutilizzati tutti gli strumenti e le strutture già introdotte nelle iterazioni precedenti, senza introdurre nuove classi o modificare il modello esistente.

Le uniche modifiche riguardano l'aggiunta di nuovi metodi all'interno delle classi responsabili della persistenza e della logica di business, specificamente per supportare il caso d'uso UC8: Visualizzazione storico stato ticket.

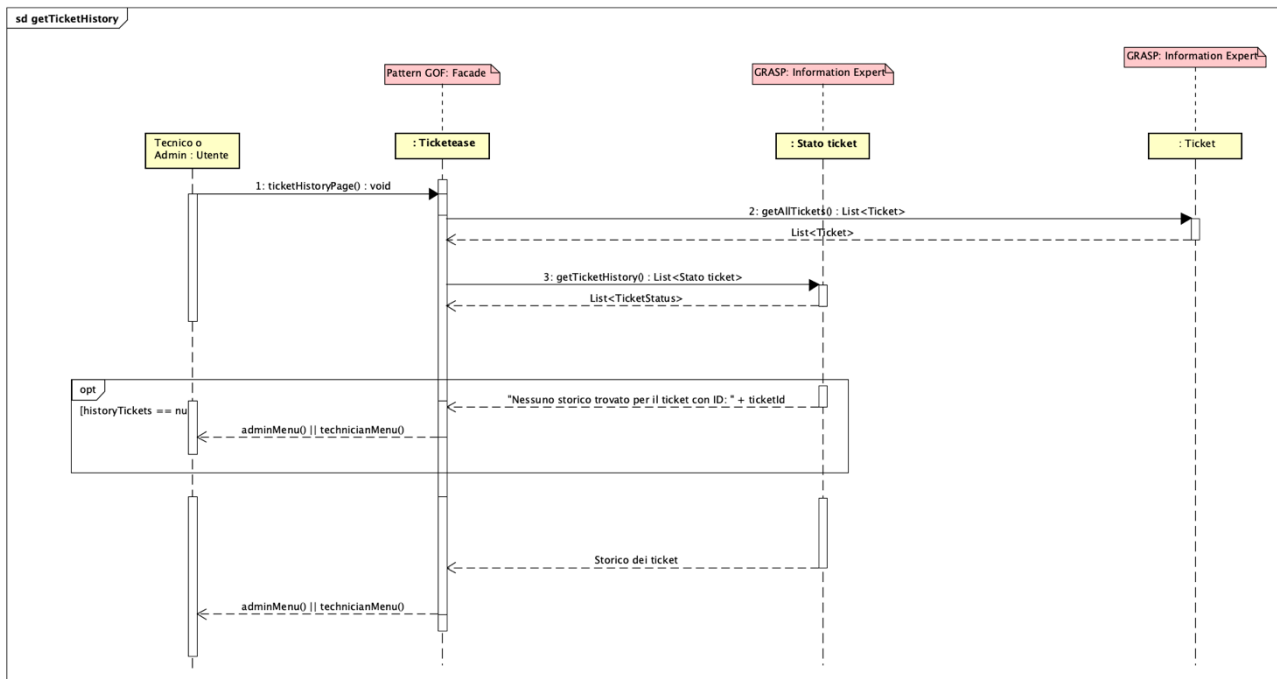
Diagramma delle classi



Diagrammi di sequenza

In questa sezione verrà mostrato l'SD per il caso d'uso UC8. Non vengono mostrati quelli già presentati nelle iterazioni precedenti.

TicketEase



- Refactoring

In fase di refactoring non sono state apportate modifiche al modello di dominio. È stato tuttavia migliorato il codice dal punto di vista di performance e leggibilità e sono state gestite le eccezioni.

6. Testing

Il **testing** è una fase essenziale del processo di sviluppo di un programma solido ed efficiente.

Grazie all'utilizzo del testing è possibile ridurre notevolmente i costi di manutenzione di un'applicazione. Per rilevare un maggior numero di malfunzionamenti è necessario eseguire un numero considerevole di test; tuttavia, è difficile testare un programma completamente.

Un testing progettato accuratamente può rivelare anomalie nell'applicazione che una volta risolte rendono il software realizzato funzionante e funzionale secondo le specifiche del committente.

I test si dividono principalmente in due famiglie:

- **I test unitari (*Unit Test*):** Sono semplici test/prove che vanno a verificare la correttezza direttamente del codice, in ogni sua piccola parte. L'idea dello Unit Test in Java è quella di valutare ogni singolo metodo in funzione dei valori attesi.

- **I test funzionali:** Sono dei test che vanno a verificare che il sistema software nella sua completezza funzioni correttamente. Questi test trattano il sistema come se fosse una scatola nera alla quale danno degli input e verificano la correttezza degli output.

Per garantire l'affidabilità e la correttezza delle funzionalità implementate, sono stati sviluppati diversi test unitari utilizzando JUnit.

Tutti i test estendono la classe `BaseTest`, una classe astratta che gestisce in modo automatico l'inizializzazione e la chiusura del contesto JPA tramite `EntityManager` ed `EntityManagerFactory`.

Sebbene l'approccio ottimale sarebbe utilizzare un'unità di persistenza separata (ad esempio un datasource dedicato o un database in-memory come H2), per semplicità e per sfruttare i dati già presenti, i test utilizzano l'unità di persistenza dello sviluppo.

Registrazione utente

- **testRegisterUser_SuccessfulForClient:** Il test verifica la registrazione riuscita di un utente cliente.
- **testRegisterUser_FailsForUnauthorizedAdmin:** Il test verifica che un tentativo di registrazione come amministratore senza autorizzazione fallisca correttamente.
- **testRegisterUser_SuccessAuthorizedForAdmin:** Il test verifica che un tentativo di registrazione come amministratore con autorizzazione avvenga correttamente.

Login utente

- **testLogin_Successful:** il test verifica un login corretto.
- **testLogin_FailsForWrongPassword:** il test garantisce che credenziali errate non consentano l'accesso.

Creazione ticket

- **testCreateTicket_Successful:** il test verifica che un utente **autenticato** possa creare un ticket, selezionando un prodotto esistente.

Assegnazione ticket

- **testAssignTicketToTechnician:** il test verifica che il ticket venga correttamente assegnato ad un tecnico selezionato.

Aggiornamento stato ticket

- **testUpdateTicketStatusToInProgress:** il test verifica che lo status del ticket venga correttamente aggiornato allo stato “In Corso”.
- **testUpdateTicketStatusToClosed:** il test verifica che lo status del ticket venga correttamente aggiornato allo stato “Chiuso” e che venga, quindi, anche inviata la notifica al cliente.

Invio e visualizzazione notifiche

- **testSendTechNotification:** il test verifica che venga inviata correttamente una notifica al tecnico quando gli viene assegnato un ticket.
- **testSendClientNotification:** il test verifica che venga inviata correttamente una notifica al cliente alla chiusura del ticket.
- **testGetNotifications:** il test verifica che le notifiche associate a un utente possano essere recuperate correttamente dal sistema.

Visualizzazione storico ticket

- **testSendTechNotification:** il test verifica che lo storico degli stati di un ticket venga recuperato correttamente e che tutti gli stati siano associati al ticket specificato.