



Università degli Studi
Mediterranea
di Reggio Calabria

BASI DI DATI I

Progettazione di un DBMS per un negozio di
materiale elettrico

Progetto realizzato da: Iero Demetrio

Matricola: 106857

DESCRIZIONE DELLA REALTA'

Si vuole realizzare un DBMS per la gestione di un negozio di materiale elettrico. In tale DBMS verranno inserite tutte le informazioni riguardanti gli articoli in vendita, i clienti ed i fornitori.

- Ogni articolo in vendita verrà identificato secondo la seguente struttura:

codice prodotto, descrizione, prezzo, pezzi disponibili.

- Ogni cliente verrà identificato secondo la seguente struttura:

codice cliente, nome, cognome, indirizzo (città, via, CAP), numero di telefono, indirizzo e-mail.

- Ogni fornitore verrà identificato secondo la seguente struttura:

partita iva fornitore, nome ditta, indirizzo (città, via, CAP), numero di telefono, indirizzo e-mail.

Ad ogni acquisto, al cliente verrà rilasciata una ricevuta fiscale, la quale potrà essere identificata attraverso i seguenti campi:

codice ricevuta, data, IVA, importo netto.

Al cliente potrà essere rilasciata una tessera fedeltà, mediante la quale potrà ricevere un buono sconto in base a quanti punti avrà accumulato tramite acquisti precedenti.

Ogni prodotto, inoltre, dovrà appartenere ad una determinata categoria, scelta tra le seguenti:

cavi elettrici, batterie, lampadine, ricambi per impianti.

Il sistema dovrà, infine, essere implementato tramite un backup periodico dei dati, al fine di prevenire la perdita dei dati contenuti nel DBMS stesso.

REQUISITI FUNZIONALI

- RF1:** Il sistema dovrà gestire le attività "CRUD" sui prodotti.
- RF2:** Il sistema dovrà gestire le attività "CRUD" sui clienti.
- RF3:** Il sistema dovrà gestire le attività "CRUD" sui fornitori.
- RF4:** Il sistema dovrà gestire le attività "CRUD" sulle tessere fedeltà.
- RF5:** Il sistema dovrà rilasciare una ricevuta fiscale ai clienti.
- RF6:** Il sistema dovrà contenere i dati di un cliente prima di poter rilasciare una tessera fedeltà.
- RF7:** Il sistema dovrà gestire i punti accumulati da un cliente per assegnargli un determinato buono sconto.
- RF8:** Il sistema dovrà fare un backup periodico dei dati.
- RF9:** Il sistema dovrà permettere al titolare dell'attività l'accesso, sia in lettura che in scrittura, a tutti i dati memorizzati nel DBMS.

REQUISITI NON FUNZIONALI

RNF1: Il sistema dovrà essere realizzato in tecnologia java.

RNF2: Il sistema dovrà possedere password di almeno 8 caratteri.

RNF3: Il sistema dovrà proteggere i dati del committente mediante meccanismi di crittografia.

DIAGRAMMA DEI CASI D'USO

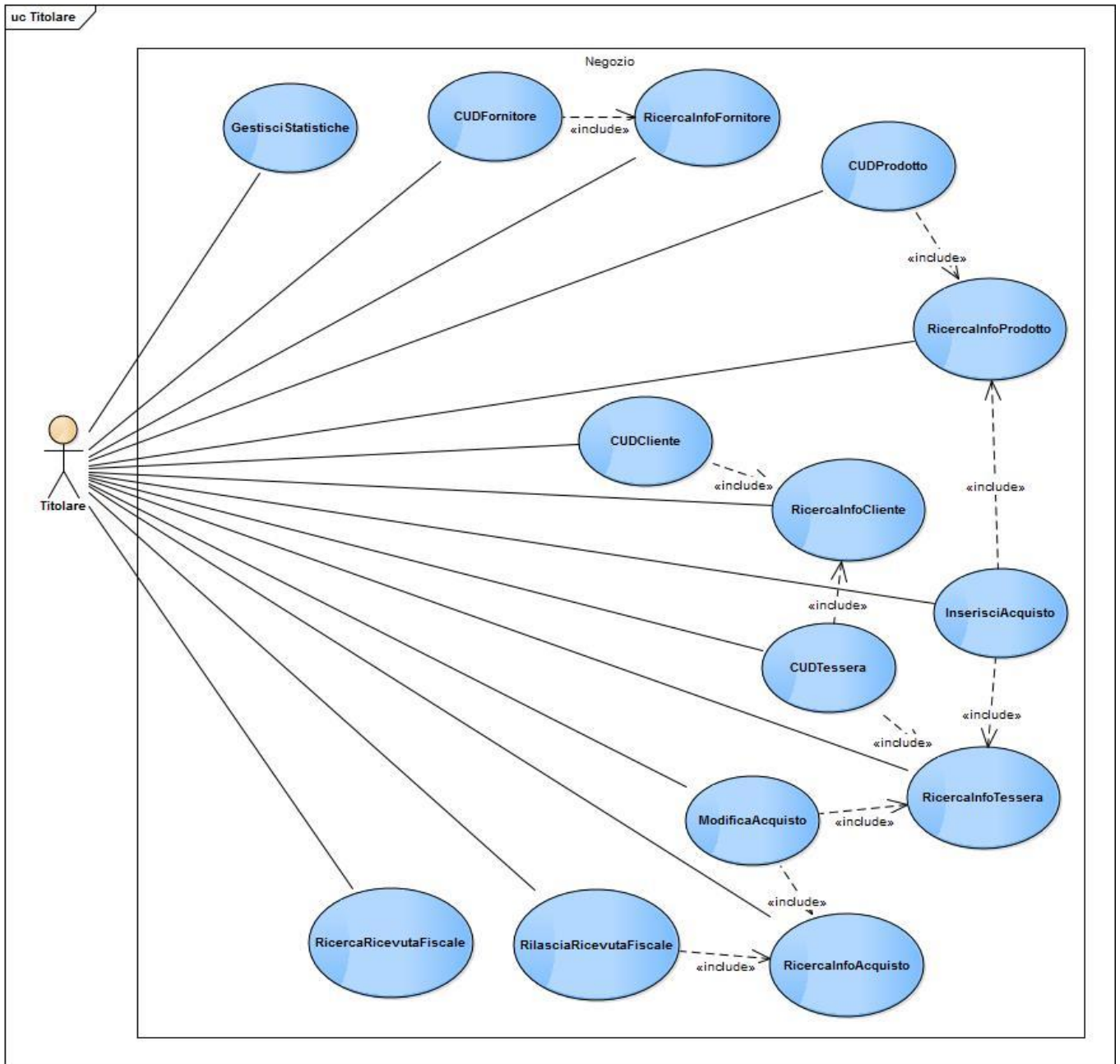
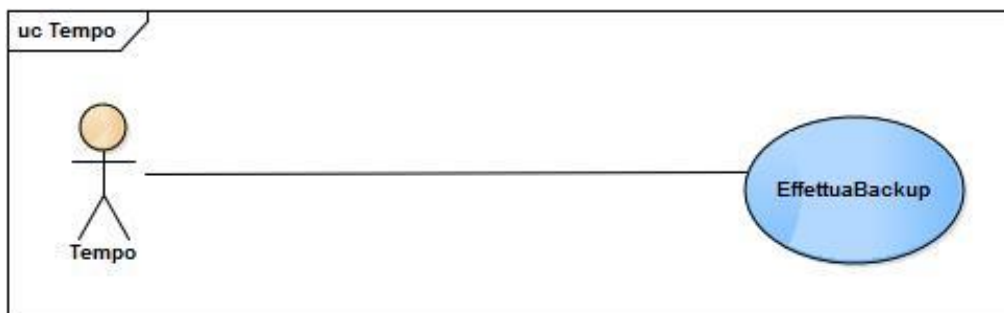
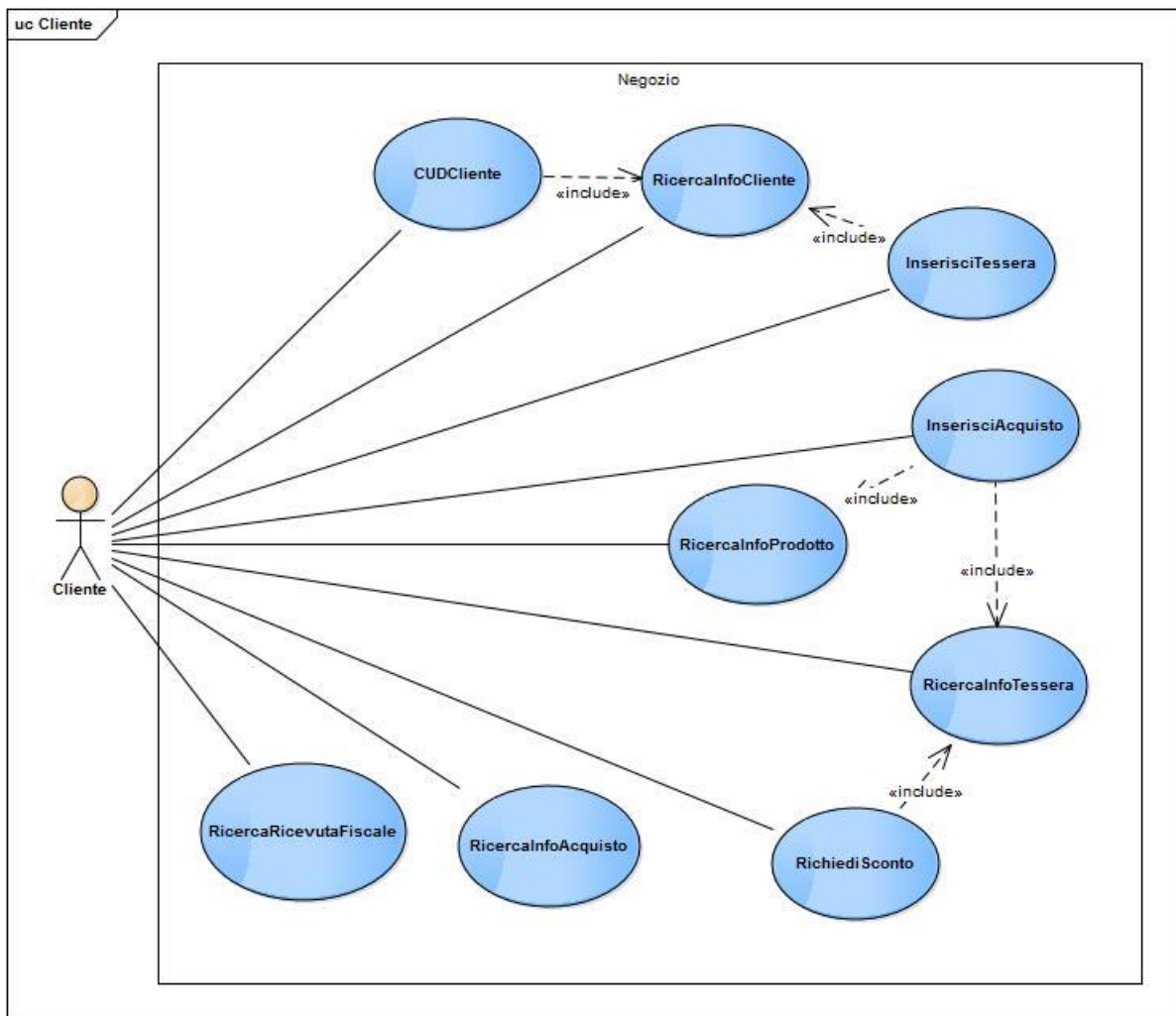


DIAGRAMMA DEI CASI D'USO



CASI D'USO

CU1: CUDFornitore

CU2: RicercaInfoFornitore

CU3: CUDProdotto

CU4: RicercaInfoProdotto

CU5: CUDCliente

CU6: RicercaInfoCliente

CU7: InserisciAcquisto

CU8: ModificaAcquisto

CU9: RicercaInfoAcquisto

CU10: RilasciaRicevutaFiscale

CU11: RicercaRicevutaFiscale

CU12: CUDTessera

CU13: RicercaInfoTessera

CU14: InserisciTessera

CU15: RichiediSconto

CU16: EffettuaBackup

CU17: GestisciStatistiche

DESCRIZIONE DEI CASI D'USO

Caso d'uso: CUDFornitore	
ID: CU1	
Breve descrizione:	Questo caso d'uso permette all'attore primario di inserire, modificare o cancellare i dati relativi ad un fornitore.
Attori primari:	Titolare.
Attori secondari:	Nessuno.
Precondizioni:	L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali:	<ol style="list-style-type: none">1. Il caso d'uso inizia quando l'attore primario vuole effettuare operazioni "CUD" su un fornitore;2. if l'attore primario vuole inserire un nuovo fornitore<ol style="list-style-type: none">2.1. L'attore primario inserisce i dati di un nuovo fornitore;2.2. Il sistema memorizza i dati inseriti nel database;3. else if l'attore primario vuole modificare i dati di un fornitore<ol style="list-style-type: none">3.1. include(RicercaInfoFornitore);3.2. L'attore primario inserisce i nuovi dati;3.3. Il sistema aggiorna i dati relativi al fornitore selezionato;4. else if l'attore primario vuole cancellare un fornitore dal sistema<ol style="list-style-type: none">4.1. include(RicercaInfoFornitore);4.2. Il sistema provvede alla rimozione del fornitore selezionato;
Postcondizioni:	Nessuna.
Sequenza degli eventi alternativa:	Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: RicercaInfoFornitore
ID: CU2
Breve descrizione: Questo caso d'uso permette all'attore primario di ricercare nel sistema i dati un fornitore per visualizzarli a video o per altri usi.
Attori primari: Titolare.
Attori secondari: Nessuno.
Precondizioni: L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali: 1. Il caso d'uso inizia quando l'attore primario vuole ricercare i dati di un fornitore all'interno del sistema; 2. L'attore primario immette nel sistema una chiave di ricerca; 3. if il fornitore è presente nel database 3.1. Il sistema restituisce i dati relativi al fornitore richiesto; 4. else il fornitore non è presente nel database 4.1. Il sistema visualizza un messaggio di errore;
Postcondizioni: Nessuna.
Sequenza degli eventi alternativa: Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: CUDProdotto	
ID: CU3	
Breve descrizione:	Questo caso d'uso permette all'attore primario di inserire, modificare o cancellare i dati relativi ad un prodotto.
Attori primari:	Titolare.
Attori secondari:	Nessuno.
Precondizioni:	L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali:	<ol style="list-style-type: none">1. Il caso d'uso inizia quando l'attore primario vuole effettuare operazioni "CUD" su un prodotto;2. if l'attore primario vuole inserire un nuovo prodotto<ol style="list-style-type: none">2.1. L'attore primario inserisce i dati di un nuovo prodotto;2.2. Il sistema memorizza i dati inseriti nel database;3. else if l'attore primario vuole modificare i dati di un prodotto<ol style="list-style-type: none">3.1. include(RicercaInfoProdotto);3.2. L'attore primario inserisce i nuovi dati;3.3. Il sistema aggiorna i dati relativi al prodotto selezionato;4. else if l'attore primario vuole cancellare un prodotto dal sistema<ol style="list-style-type: none">4.1. include(RicercaInfoProdotto);4.2. Il sistema provvede alla rimozione del prodotto selezionato;
Postcondizioni:	Nessuna.
Sequenza degli eventi alternativa:	Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: RicercaInfoProdotto
ID: CU4
Breve descrizione: Questo caso d'uso permette all'attore primario di ricercare nel sistema i dati un prodotto per visualizzarli a video o per altri usi.
Attori primari: Titolare, Cliente.
Attori secondari: Nessuno.
Precondizioni: L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali: 1. Il caso d'uso inizia quando l'attore primario vuole ricercare i dati di un prodotto all'interno del sistema; 2. L'attore primario immette nel sistema una chiave di ricerca; 3. if il prodotto è presente nel database 3.1. Il sistema restituisce i dati relativi al prodotto richiesto; 4. else il prodotto non è presente nel database 4.1. Il sistema visualizza un messaggio di errore;
Postcondizioni: Nessuna.
Sequenza degli eventi alternativa: Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: CUDCliente
ID: CU5
Breve descrizione: Questo caso d'uso permette all'attore primario di inserire, modificare o cancellare i dati relativi ad un cliente.
Attori primari: Titolare, Cliente.
Attori secondari: Nessuno.
Precondizioni: L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali: <ol style="list-style-type: none">1. Il caso d'uso inizia quando l'attore primario vuole effettuare operazioni "CUD" su un cliente;2. if l'attore primario vuole inserire un nuovo cliente<ol style="list-style-type: none">2.1. L'attore primario inserisce i dati di un nuovo cliente;2.2. Il sistema memorizza i dati inseriti nel database;3. else if l'attore primario vuole modificare i dati di un cliente<ol style="list-style-type: none">3.1. include(RicercaInfoCliente);3.2. L'attore primario inserisce i nuovi dati;3.3. Il sistema aggiorna i dati relativi al cliente selezionato;4. else if l'attore primario vuole cancellare un cliente dal sistema<ol style="list-style-type: none">4.1. include(RicercaInfoCliente);4.2. Il sistema provvede alla rimozione del cliente selezionato;
Postcondizioni: Nessuna.
Sequenza degli eventi alternativa: Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: RicercaInfoCliente
ID: CU6
Breve descrizione: Questo caso d'uso permette all'attore primario di ricercare nel sistema i dati un cliente per visualizzarli a video o per altri usi.
Attori primari: Titolare, Cliente.
Attori secondari: Nessuno.
Precondizioni: L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali: 1. Il caso d'uso inizia quando l'attore primario vuole ricercare i dati di un cliente all'interno del sistema; 2. L'attore primario immette nel sistema una chiave di ricerca; 3. if il cliente è presente nel database 3.1. Il sistema restituisce i dati relativi al cliente richiesto; 4. else il cliente non è presente nel database 4.1. Il sistema visualizza un messaggio di errore;
Postcondizioni: Nessuna.
Sequenza degli eventi alternativa: Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: InserisciAcquisto	
ID: CU7	
Breve descrizione:	Questo caso d'uso permette all'attore primario di effettuare un acquisto.
Attori primari:	Titolare, Cliente.
Attori secondari:	Nessuno.
Precondizioni:	L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali:	<ol style="list-style-type: none">1. Il caso d'uso inizia quando l'attore primario vuole acquistare un prodotto;2. include(RicercaInfoProdotto);3. if il numero di pezzi disponibili del prodotto è maggiore o uguale a 1<ol style="list-style-type: none">3.1. Il sistema calcola il prezzo totale e lo visualizza all'utente;3.2. Il sistema decrementa il numero di pezzi disponibili del prodotto;3.3. Il sistema aggiorna il database con i dati dell'acquisto eseguito;3.4. include(RicercaInfoTessera);3.5. Il sistema aggiorna il totale dei punti del cliente;4. else<ol style="list-style-type: none">4.1. Il sistema visualizza un messaggio di errore;
Postcondizioni:	Nessuna.
Sequenza degli eventi alternativa:	Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: ModificaAcquisto
ID: CU8
Breve descrizione: Questo caso d'uso permette all'attore primario di modificare un acquisto fatto precedentemente.
Attori primari: Titolare.
Attori secondari: Nessuno.
Precondizioni: L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali: <ol style="list-style-type: none">1. Il caso d'uso inizia quando l'attore primario vuole modificare un acquisto fatto precedentemente;2. include(RicercaInfoAcquisto);3. L'attore primario specifica i prodotti che intende cambiare;4. Il sistema aggiorna i dati relativi all'acquisto selezionato;5. Il sistema aggiorna il numero di pezzi disponibili dei prodotti coinvolti nella modifica effettuata;6. include(RicercaInfoTessera);7. Il sistema aggiorna il totale punti del cliente in seguito alle modifiche dell'acquisto;
Postcondizioni: Nessuna.
Sequenza degli eventi alternativa: Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: RicercaInfoAcquisto	
ID: CU9	
Breve descrizione:	Questo caso d'uso permette all'attore primario di ricercare i dati relativi ad un acquisto eseguito nel DBMS.
Attori primari:	Titolare, Cliente.
Attori secondari:	Nessuno.
Precondizioni:	L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali:	<ol style="list-style-type: none">1. Il caso d'uso inizia quando l'attore primario vuole ricercare i dati relativi ad un acquisto all'interno del sistema;2. L'attore primario specifica una chiave di ricerca;3. if l'acquisto è presente nel database<ol style="list-style-type: none">3.1. Il sistema visualizza i dati relativi all'acquisto cercato;4. else<ol style="list-style-type: none">4.1. Il sistema visualizza un messaggio di errore;
Postcondizioni:	Nessuna.
Sequenza degli eventi alternativa:	Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: RilasciaRicevutaFiscale
ID: CU10
Breve descrizione: Questo caso d'uso permette all'attore primario di rilasciare una ricevuta fiscale al cliente.
Attori primari: Titolare.
Attori secondari: Nessuno.
Precondizioni: L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali: 1. Il caso d'uso inizia quando l'attore primario vuole rilasciare una ricevuta fiscale al cliente; 2. include (RicercaInfoAcquisto); 3. Il sistema visualizza e stampa la ricevuta fiscale relativa all'acquisto cercato;
Postcondizioni: Nessuna.
Sequenza degli eventi alternativa: Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: RicercaRicevutaFiscale
ID: CU11
Breve descrizione: Questo caso d'uso permette all'attore primario di ricercare i dati di una ricevuta fiscale rilasciata ad un cliente.
Attori primari: Titolare, Cliente.
Attori secondari: Nessuno.
Precondizioni: L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali: 1. Il caso d'uso inizia quando l'attore primario vuole ricercare nel sistema i dati relativi ad una ricevuta fiscale; 2. L'attore primario specifica una chiave di ricerca; 3. if la ricevuta fiscale cercata è presente nel database 3.1. Il sistema visualizza i dati della ricevuta fiscale cercata; 4. else 4.1. Il sistema visualizza un messaggio di errore;
Postcondizioni: Nessuna.
Sequenza degli eventi alternativa: Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: CUDTessera	
ID: CU12	
Breve descrizione:	Questo caso d'uso permette all'attore primario di inserire, modificare o cancellare i dati relativi ad una tessera fedeltà.
Attori primari:	Titolare.
Attori secondari:	Nessuno.
Precondizioni:	L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali:	<ol style="list-style-type: none">1. Il caso d'uso inizia quando l'attore primario vuole effettuare operazioni "CUD" su una tessera fedeltà;2. if l'attore primario vuole inserire una nuova tessera per un cliente<ol style="list-style-type: none">2.1. include(RicercaInfoCliente);2.2. Il sistema associa una nuova tessera al cliente cercato;2.3. Il sistema memorizza i dati inseriti nel database;3. else if l'attore primario vuole modificare i dati di una tessera<ol style="list-style-type: none">3.1. include(RicercaInfoTessera);3.2. L'attore primario inserisce i nuovi dati;3.3. Il sistema aggiorna i dati relativi alla tessera selezionata;4. else if l'attore primario vuole cancellare una tessera dal sistema<ol style="list-style-type: none">4.1. include(RicercaInfoTessera);4.2. Il sistema provvede alla rimozione della tessera selezionata;
Postcondizioni:	Nessuna.
Sequenza degli eventi alternativa:	Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: RicercaInfoTessera
ID: CU13
Breve descrizione: Questo caso d'uso permette all'attore primario di ricercare i dati di una tessera fedeltà.
Attori primari: Titolare, Cliente.
Attori secondari: Nessuno.
Precondizioni: L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali: 1. Il caso d'uso inizia quando l'attore primario vuole ricercare nel sistema i dati relativi ad una tessera fedeltà; 2. L'attore primario specifica una chiave di ricerca; 3. if la tessera cercata è presente nel database 3.1. Il sistema visualizza i dati relativi alla tessera cercata; 4. else 4.1. Il sistema visualizza un messaggio di errore;
Postcondizioni: Nessuna.
Sequenza degli eventi alternativa: Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: InserisciTessera
ID: CU14
Breve descrizione: Questo caso d'uso permette all'attore primario di inserire i dati relativi ad una nuova tessera fedeltà.
Attori primari: Cliente.
Attori secondari: Nessuno.
Precondizioni: L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali: 1. Il caso d'uso inizia quando l'attore primario vuole inserire nel sistema i dati relativi ad una nuova tessera fedeltà; 2. include (RicercaInfoCliente); 3. Il sistema associa una nuova tessera al cliente selezionato; 4. Il sistema memorizza i dati inseriti nel database;
Postcondizioni: Nessuna.
Sequenza degli eventi alternativa: Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: RichiediSconto	
ID: CU15	
Breve descrizione:	Questo caso d'uso permette all'attore primario di richiedere un buono sconto.
Attori primari:	Cliente.
Attori secondari:	Nessuno.
Precondizioni:	L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali:	<ol style="list-style-type: none">1. Il caso d'uso inizia quando l'attore primario vuole richiedere un buono sconto da usare negli acquisti successivi;2. include(RicercaInfoTessera);3. if il totale dei punti nella tessera è maggiore o uguale a 50<ol style="list-style-type: none">3.1. Il sistema visualizza e stampa il codice e l'importo del buono;3.2. Il sistema decrementa i punti utilizzati dal totale dei punti;4. else<ol style="list-style-type: none">4.1. Il sistema visualizza un messaggio di errore;
Postcondizioni:	Nessuna.
Sequenza degli eventi alternativa:	Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: EffettuaBackup

ID: CU16

Breve descrizione:

Questo caso d'uso permette al sistema di effettuare il backup dei dati presenti all'interno del database.

Attori primari:

Tempo.

Attori secondari:

Titolare.

Precondizioni:

L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.

Sequenza degli eventi principali:

1. Il caso d'uso inizia quando arriva il momento di effettuare il backup dei dati presenti nel database;
2. Il sistema preleva i dati relativi ai fornitori;
3. Il sistema copia su disco i dati prelevati;
4. Il sistema preleva i dati relativi ai prodotti;
5. Il sistema copia su disco i dati prelevati;
6. Il sistema preleva i dati relativi agli acquisti;
7. Il sistema copia su disco i dati prelevati;
8. Il sistema preleva i dati relativi ai clienti;
9. Il sistema copia su disco i dati prelevati;
10. Il sistema preleva i dati relativi alle tessere;
11. Il sistema copia su disco i dati prelevati;
12. Il sistema preleva i dati relativi alle ricevute fiscali;
13. Il sistema copia su disco i dati prelevati;

Postcondizioni:

I dati sono stati copiati correttamente su disco.

Sequenza degli eventi alternativa:

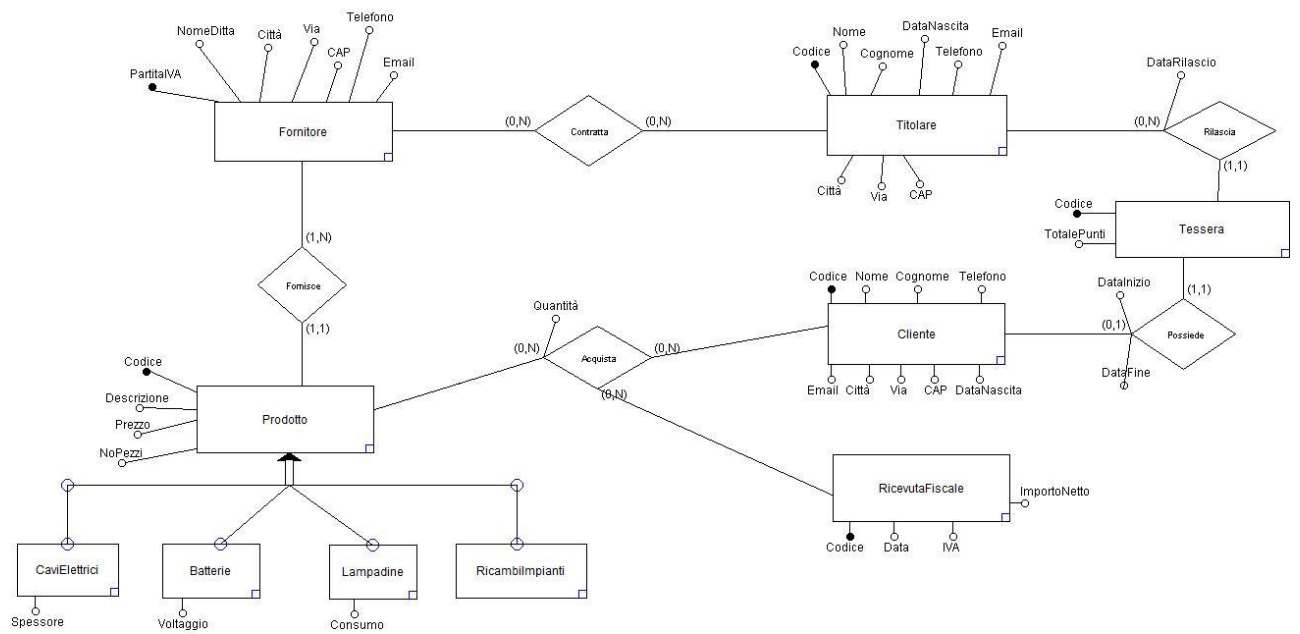
Nessuna.

DESCRIZIONE DEI CASI D'USO

Caso d'uso: GestisciStatistiche	
ID: CU17	
Breve descrizione:	Questo caso d'uso gestisce la visualizzazione di opportune statistiche per il titolare.
Attori primari:	Titolare.
Attori secondari:	Nessuno.
Precondizioni:	L'attore primario deve disporre di un account di utilizzo con appropriati diritti di accesso.
Sequenza degli eventi principali:	<ol style="list-style-type: none">1. Il caso d'uso inizia quando l'attore primario richiede alcune statistiche al sistema;2. if l'attore primario richiede statistiche sui fornitori<ol style="list-style-type: none">2.1. Il sistema preleva i dati sui fornitori;2.2. Il sistema elabora opportunamente i dati prelevati;2.3. Il sistema visualizza i dati elaborati;3. if l'attore primario richiede statistiche sui prodotti<ol style="list-style-type: none">3.1. Il sistema preleva i dati sui prodotti;3.2. Il sistema elabora opportunamente i dati prelevati;3.3. Il sistema visualizza i dati elaborati;4. if l'attore primario richiede statistiche sugli acquisti<ol style="list-style-type: none">4.1. Il sistema preleva i dati sugli acquisti;4.2. Il sistema elabora opportunamente i dati prelevati;4.3. Il sistema visualizza i dati elaborati;5. if l'attore primario richiede statistiche sui clienti<ol style="list-style-type: none">5.1. Il sistema preleva i dati sui clienti;5.2. Il sistema elabora opportunamente i dati prelevati;5.3. Il sistema visualizza i dati elaborati;6. if l'attore primario richiede statistiche sulle tessere<ol style="list-style-type: none">6.1. Il sistema preleva i dati sulle tessere;6.2. Il sistema elabora opportunamente i dati prelevati;6.3. Il sistema visualizza i dati elaborati;7. if l'attore primario richiede statistiche sulle ricevute fiscali<ol style="list-style-type: none">7.1. Il sistema preleva i dati sulle ricevute fiscali;7.2. Il sistema elabora opportunamente i dati prelevati;7.3. Il sistema visualizza i dati elaborati;
Postcondizioni:	Nessuna.
Sequenza degli eventi alternativa:	Nessuna.

PROGETTAZIONE CONCETTUALE

DIAGRAMMA E-R



DIZIONARIO DELLE ENTITA'

ENTITA'	DESCRIZIONE	ATTRIBUTI	IDENTIFICATORE
Fornitore	Colui che fornisce i prodotti per il negozio al titolare	PartitaIVA, NomeDitta, Telefono, Email, Città, Via, CAP	PartitaIVA
Titolare	Colui che gestisce il negozio	Codice, Nome, Cognome, DataNascita, Telefono, Email, Città, Via, CAP	Codice
Prodotto	Ciò che viene esposto e messo in vendita nel negozio	Codice, Prezzo, Descrizione, NoPezzi	Codice
Cliente	Colui che compie acquisti all'interno del negozio; può possedere una tessera fedeltà.	Codice, Nome, Cognome, Telefono, Email, Città, Via, CAP, DataNascita	Codice
RicevutaFiscale	Documento fiscale rilasciato al cliente per ogni singolo acquisto	Codice, Data, IVA, ImportoNetto	Codice
Tessera	Una tessera fedeltà che può essere rilasciata al cliente per poter usufruire di buoni sconto associati alla raccolta punti	Codice, TotalePunti	Codice
CaviElettrici	Sottocategoria dei prodotti in vendita	Codice, Prezzo, Descrizione, NoPezzi, Spessore	Codice
Batterie	Sottocategoria dei prodotti in vendita	Codice, Prezzo, Descrizione, NoPezzi, Voltaggio	Codice
Lampadine	Sottocategoria dei prodotti in vendita	Codice, Prezzo, Descrizione, NoPezzi, Consumo	Codice
RicambiImpianti	Sottocategoria dei prodotti in vendita	Codice, Prezzo, Descrizione, NoPezzi	Codice

DIZIONARIO DELLE RELAZIONI

RELAZIONE	DESCRIZIONE	ENTITA' COINVOLTE	ATTRIBUTI
Contratta	Associa uno o più titolari ad uno o più fornitori	Titolare (0,N) Fornitore (0,N)	-
Fornisce	Associa almeno un fornitore ad ogni prodotto	Fornitore (1, N) Prodotto (1, 1)	-
Acquista	Associa uno o più clienti ad uno o più prodotti e ad una o più ricevuta fiscale	Cliente (0, N) Prodotto (0, N) RicevutaFiscale(0, N)	Quantità
Possiede	Associa una tessera ad uno e un solo cliente	Tessera (1, 1) Cliente (0, 1)	DataInizio, DataFine
Rilascia	Associa uno o più titolari ad una o più tessere	Titolare (0, N) Tessera (1, 1)	DataRilascio

REGOLE

IDENTIFICATORE REGOLA	REGOLA
R1	Il cliente non appena raggiunge almeno 50 punti nella tessera può richiedere un buono sconto
R2	Il prezzo di un prodotto deve essere maggiore di 0
R3	Il numero di pezzi disponibili di un prodotto non può assumere un valore negativo
R4	I prodotti acquistati o usati non possono essere restituiti al negozio
R5	Un acquisto per essere tale deve riferirsi ad almeno un prodotto presente nel negozio, indipendentemente dalla sottocategoria di appartenenza di quest'ultimo
R6	Il campo telefono non può contenere caratteri che non siano numerici
R7	La quantità dei prodotti in un acquisto deve essere maggiore di 0

PROGETTAZIONE LOGICA

RISTRUTTURAZIONE DEL DIAGRAMMA E-R

➤ ANALISI DELLE RIDONDANZE

Nel diagramma E-R realizzato in precedenza non sono presenti casi di ridondanza.

➤ ELIMINAZIONE DELLE GENERALIZZAZIONI

Nel diagramma E-R realizzato in precedenza l'unica generalizzazione presente è riferita all'entità "Prodotto".

Per eliminarla si farà ricorso all'accorpamento delle entità figlie nell'entità padre, aggiungendo a tale entità l'attributo "Categoria" (che rappresenterà quindi il campo in cui andrà specificato il tipo di prodotto al quale ci si riferisce) e gli attributi appartenenti alle entità figlie.

➤ PARTIZIONAMENTO/ACCORPAMENTO DI CONCETTI

Nel diagramma E-R realizzato in precedenza non sono presenti concetti che debbano essere partizionati o accorpati.

➤ ELIMINAZIONE DEGLI ATTRIBUTI MULTIVALORE

Nel diagramma E-R realizzato in precedenza non sono presenti attributi multivalore, tenendo presente che gli attributi "Telefono" e "Email" delle entità "Cliente", "Titolare" e "Fornitore" hanno cardinalità (0,1), quindi sono opzionali e con non più di un valore.

➤ ELIMINAZIONE DEGLI ATTRIBUTI COMPOSTI

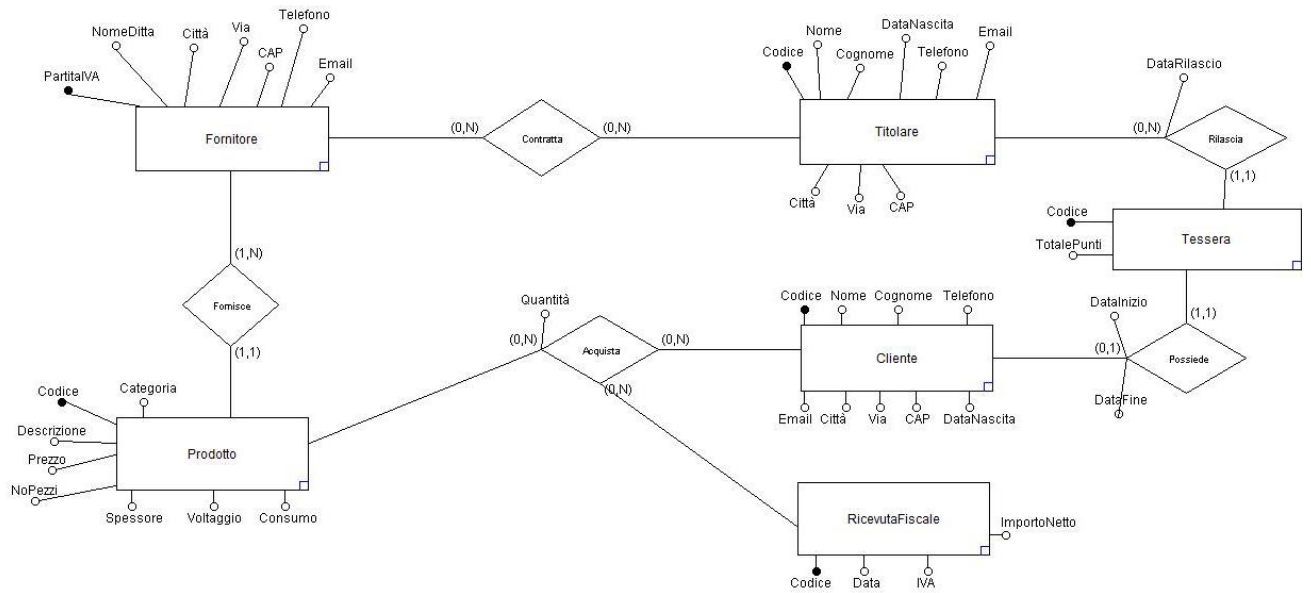
Nel diagramma E-R realizzato in precedenza non sono presenti attributi composti.

➤ SCELTA DEGLI INDICATORI PRIMARI

Nel diagramma E-R realizzato in precedenza sono stati evidenziati i seguenti indicatori primari tra gli attributi disponibili per ogni entità:

- 1) Per l'entità "Fornitore" la chiave sarà "PartitaIVA";
- 2) Per l'entità "Prodotto" la chiave sarà "Codice"; (Chiave Surrogata)
- 3) Per l'entità "Titolare" la chiave sarà "Codice"; (Chiave Surrogata)
- 4) Per l'entità "Tessera" la chiave sarà "Codice";
- 5) Per l'entità "RicevutaFiscale" la chiave sarà "Codice";
- 6) Per l'entità "Cliente" la chiave sarà "Codice"; (Chiave Surrogata)

DIAGRAMMA E-R RISTRUTTURATO



TRADUZIONE VERSO IL MODELLO RELAZIONALE

Il diagramma E-R ristrutturato permette di individuare le seguenti tabelle:

- FORNITORE (PartitaIVA, NomeDitta, Città, Via, CAP, Telefono, Email)
- PRODOTTO (Codice, Categoria, Descrizione, Prezzo, NoPezzi, Spessore, Voltaggio, Consumo)
- CLIENTE (Codice, Nome, Cognome, DataNascita, Città, Via, CAP, Telefono, Email, CodiceTessera)
- TITOLARE (Codice, Nome, Cognome, DataNascita, Città, Via, CAP, Telefono, Email)
- RICEVUTAFISCALE (Codice, Data, ImportoNetto, IVA)
- TESSERA (Codice, TotalePunti, DataEmissione, DataScadenza)
- CONTRATTA (PartitaIVAFornitore, CodiceTitolare)
- FORNISCE (PartitaIVAFornitore, CodiceProdotto)
- ACQUISTA (CodiceProdotto, CodiceCliente, CodiceRicevuta, Quantità)
- RILASCIA (CodiceTitolare, CodiceTessera, DataRilascio)

PROGETTAZIONE FISICA

SCELTA DEGLI INDICI

Per ogni tabella, gli **indici primari**, ovvero quelli che hanno il proprio indexfield associato alla rispettiva chiave, vengono automaticamente creati dal DBMS usato.

Per quanto riguarda gli **indici secondari**, è opportuno che siano presenti:

- Per la tabella *FORNITORE*, appare utile un indice da associare all'attributo *NomeDitta*;
- Per la tabella *PRODOTTO*, appare utile un indice da associare alla coppia di attributi *<Prezzo, NoPezzi>*;
- Per la tabella *CLIENTE*, appare utile un indice associato alla coppia di attributi *<Cognome, Nome>* e un indice associato all'attributo *DataNascita*;
- Per la tabella *RICEVUTAFISCALE*, appare utile un indice associato all'attributo *Data*;

STIMA DELLE DIMENSIONI

Questa stima delle dimensioni per ciascuna tabella viene effettuata considerando un'ipotetica durata di 20 anni di esercizio.

Inoltre la dimensione di ciascuna tabella è data dalla dimensione di ogni singolo attributo moltiplicata per il numero di tuple che la tabella conterrà.

FORNITORE

- PartitaIVA: 11 byte
- NomeDitta: 50 byte
- Città: 30 byte
- Via: 50 byte
- CAP: 5 byte
- Telefono: 15 byte
- Email: 30 byte

Una tupla di questa tabella occupa 191 byte. Ipotizzando in 20 anni la memorizzazione di 30 fornitori, si arriva ad un totale di 5.730 byte.

PRODOTTO

- Codice: 4 byte
- Categoria: 20 byte
- Descrizione: 150 byte
- Prezzo: 7 byte
- NoPezzi: 4 byte
- Spessore: 5 byte
- Voltaggio: 5 byte
- Consumo: 5 byte

Una tupla di questa tabella occupa 200 byte. Ipotizzando in 20 anni la memorizzazione di 5.000 prodotti di tipo diverso, si arriva ad un totale di 1.000.000 byte.

CLIENTE

- Codice: 4 byte
- Nome: 30 byte
- Cognome: 30 byte
- DataNascita: 10 byte
- Città: 30 byte
- Via: 50 byte
- CAP: 5 byte
- Telefono: 15 byte
- Email: 30 byte
- CodiceTessera: 4 byte

Una tupla di questa tabella occupa 208 byte. Ipotizzando in 20 anni di memorizzare 6.000 clienti, si arriva ad un totale di 1.248.000 byte.

TITOLARE

- Codice: 1 byte
- Nome: 30 byte
- Cognome: 30 byte
- DataNascita: 10 byte
- Città: 30 byte
- Via: 50 byte
- CAP: 5 byte
- Telefono: 15 byte
- Email: 30 byte

Una tupla di questa tabella occupa 201 byte. Essendo un negozio a conduzione familiare, si supponga in 20 anni la memorizzazione di 7 titolari, per un totale di 1.407 byte.

RICEVUTAFISCALE

- Codice: 6 byte
- Data: 10 byte
- ImportoNetto: 7 byte
- IVA: 3 byte

Una tupla di questa tabella occupa 26 byte. Ipotizzando una media giornaliera di 50 acquisti, che equivalgono a circa 365.000 acquisti effettuati in 20 anni, dato che per ogni acquisto viene rilasciata conseguente ricevuta fiscale, si arriva ad un totale di 9.490.000 byte.

TESSERA

- Codice: 4 byte
- TotalePunti: 6 byte
- DataEmissione: 10 byte
- DataScadenza: 10 byte

Una tupla di questa tabella occupa 30 byte. Ipotizzando che, nel caso peggiore, ogni cliente voglia possedere una tessera, ed avendo già stimato in 20 anni la registrazione di 6.000 clienti, si arriva ad un totale di 180.000 byte.

CONTRATTA

- PartitaIVAForitore: 11 byte
- CodiceTitolare: 1 byte

Una tupla di questa tabella occupa 12 byte. Ipotizzando in 20 anni, come visto precedentemente, che si memorizzino 30 fornitori e che non ci possano essere 2 tuple con lo stesso fornitore, prescindendo dal titolare che lo ha contrattato, si arriva ad un totale di 360 byte.

FORNISCE

- PartitaIVAForitore: 11 byte
- CodiceProdotto: 4 byte

Una tupla di questa tabella occupa 15 byte. Ipotizzando in 20 anni che, come visto in precedenza, vengano memorizzati 5.000 prodotti di diversa categoria e 30 fornitori, e che uno stesso prodotto non possa essere fornito da 2 diversi fornitori, si arriva ad un totale di 75.000 byte.

ACQUISTA

- CodiceProdotto: 4 byte
- CodiceCliente: 4 byte
- CodiceRicevuta: 6 byte
- Quantità: 4 byte

Una tupla di questa tabella occupa 18 byte. Ipotizzando in 20 anni che, come visto in precedenza, vengano effettuati in media 50 acquisti al giorno, si arriva ad un totale di 6.570.000 byte.

RILASCIA

- CodiceTitolare: 1 byte
- CodiceTessera: 4 byte
- DataRilascio: 10 byte

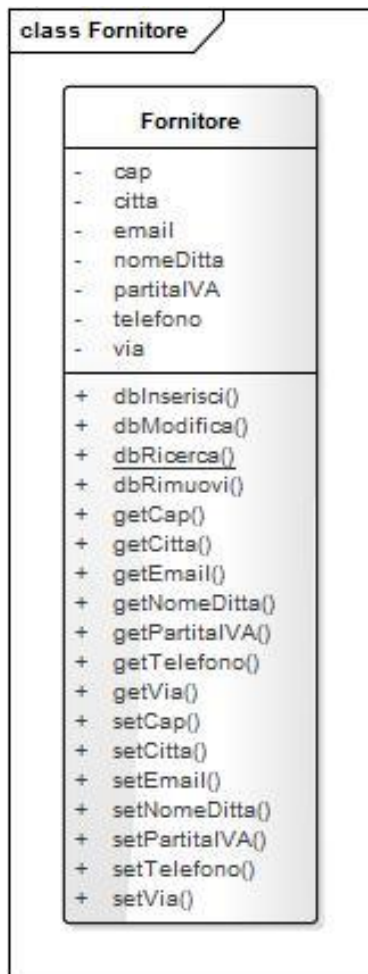
Una tupla di questa tabella occupa 15 byte. Ipotizzando che, nel caso peggiore, ogni cliente voglia possedere una tessera, ed avendo già stimato in 20 anni la registrazione di 6.000 clienti, considerando che 2 titolari diversi non possano rilasciare la stessa tessera, si arriva ad un totale di 90.000 byte.

Lo spazio necessario per l'intero database, pertanto, è pari a 18.660.497 byte, pari a circa 18 MB.

PROGETTAZIONE DELLA COMPONENTE APPLICATIVA

DIAGRAMMA DELLE CLASSI

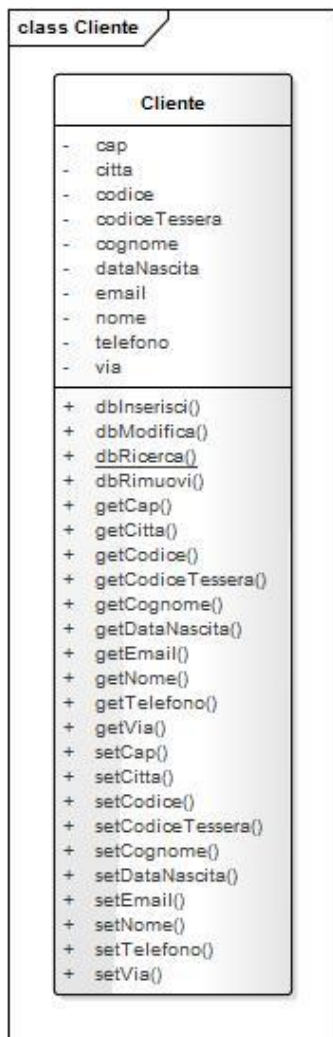
➤ CLASSE FORNITORE



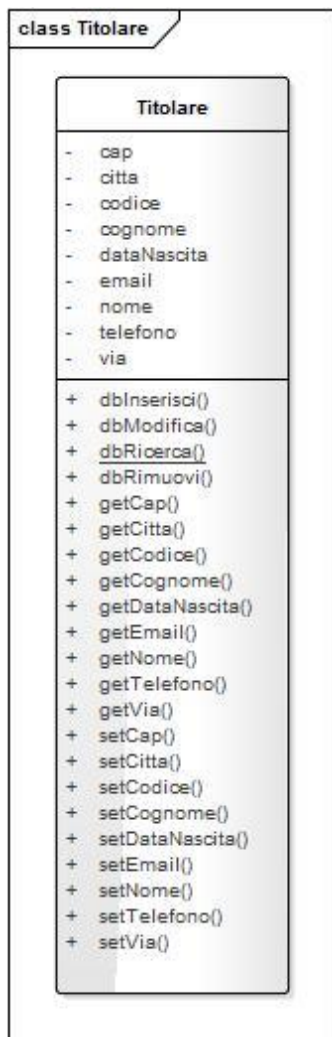
➤ CLASSE PRODOTTO



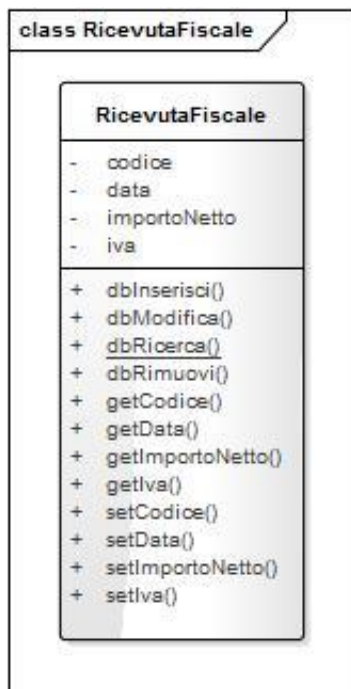
➤ CLASSE CLIENTE



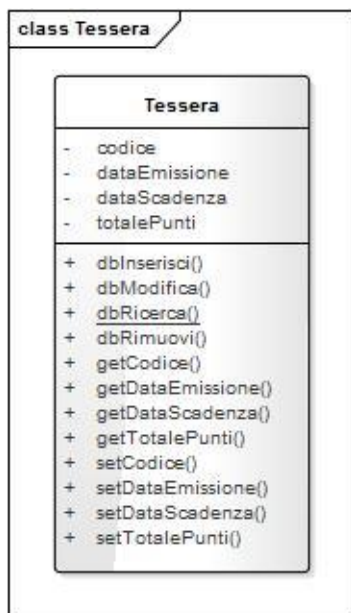
➤ CLASSE TITOLARE



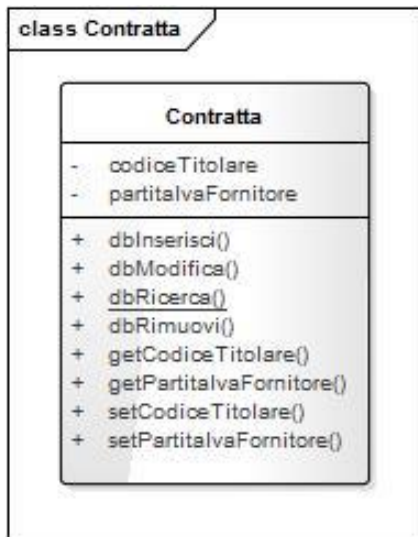
➤ CLASSE RICEVUTAFISCALE



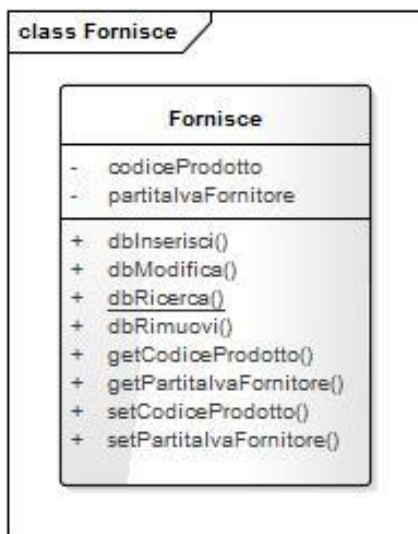
➤ CLASSE TESSERA



➤ CLASSE CONTRATTA



➤ CLASSE FORNISCE



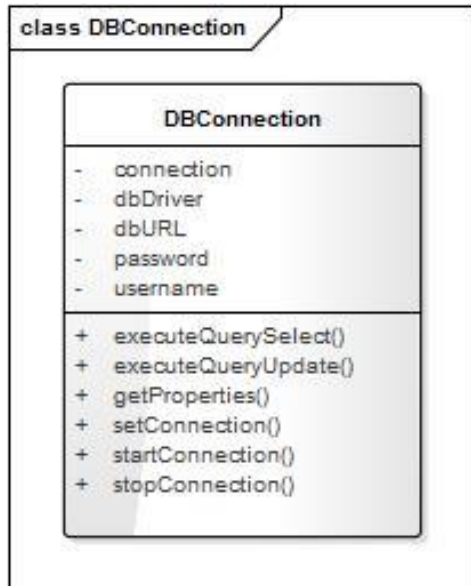
➤ CLASSE ACQUISTA



➤ CLASSE RILASCIA



- E' inoltre opportuno aggiungere una classe per la gestione, la connessione e l'interrogazione del database denominata DBCONNECTION

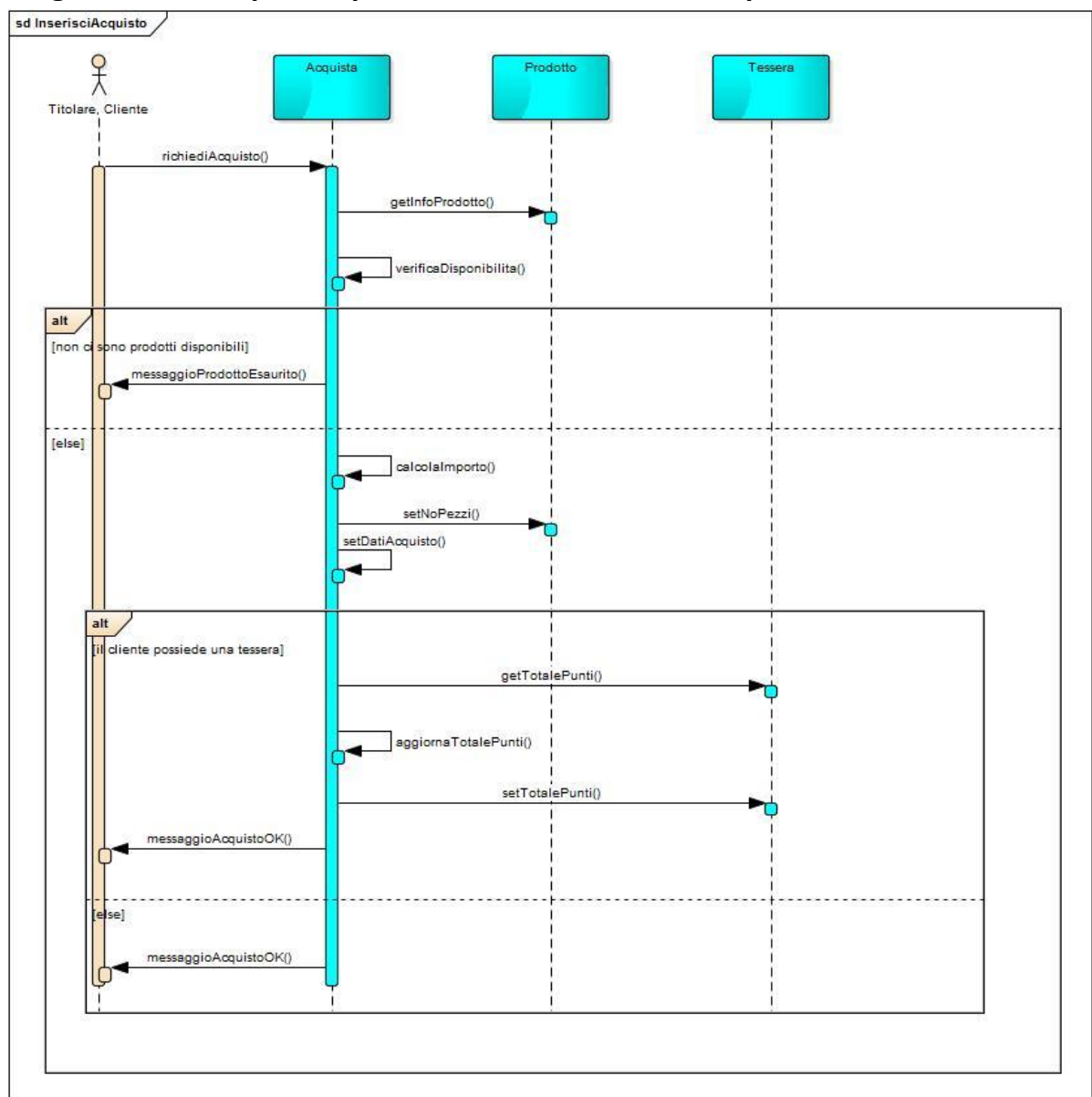


PROGETTAZIONE DELLE FUNZIONALITA' E RAFFINAMENTO DELLE CLASSI

DIAGRAMMA DI SEQUENZA

I casi d'uso **CUDFornitore**, **RicercaInfoFornitore**, **CUDCliente**, **RicercaInfoCliente**, **CUDProdotto**, **RicercaInfoProdotto**, **CUDTessera**, **RicercaInfoTessera**, **RicercaRicevutaFiscale**, **RichiediSconto** coinvolgono una sola classe e non richiedono elaborazioni sui dati, pertanto non risulta necessario costruire i rispettivi diagrammi di sequenza.

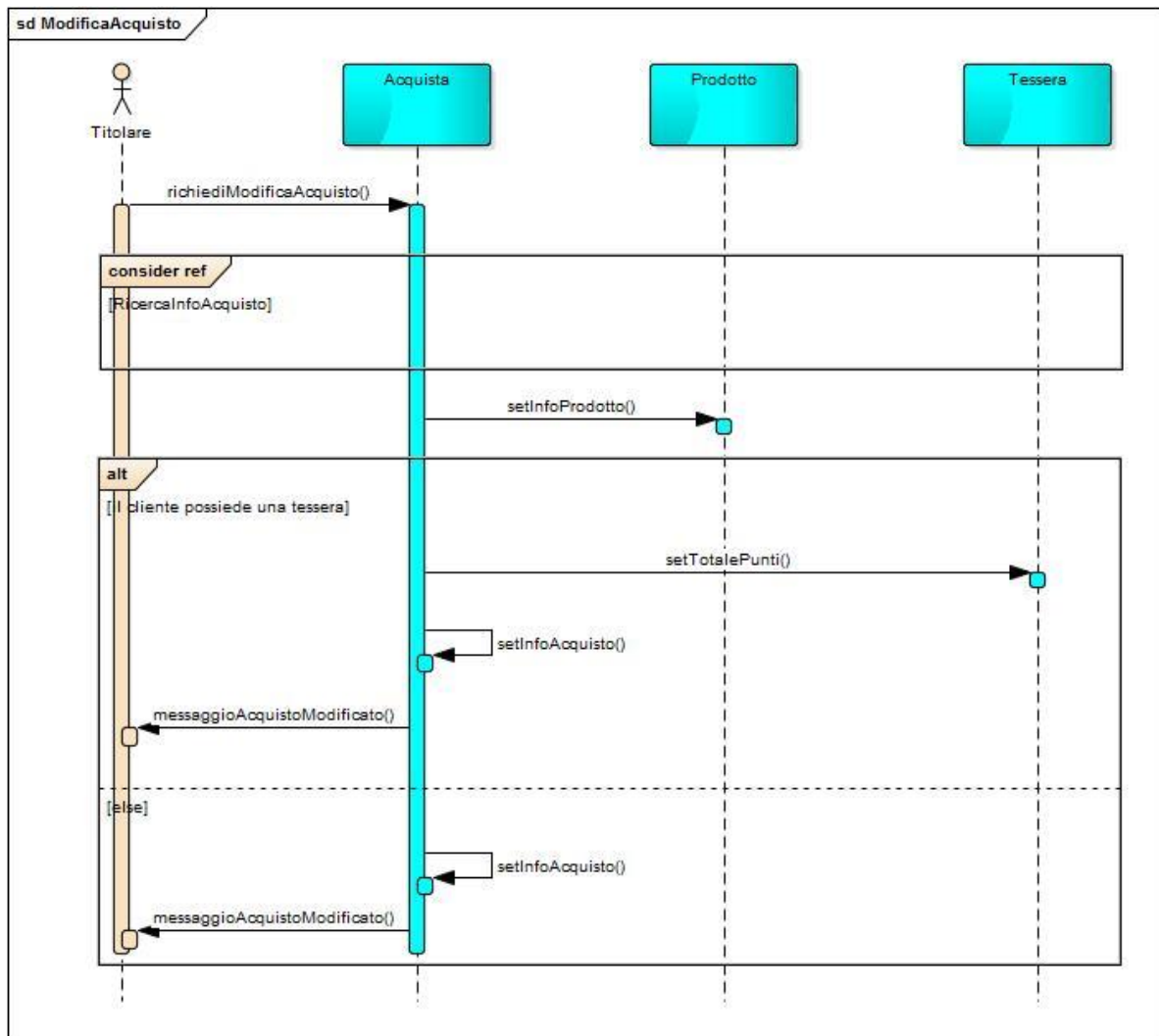
➤ Diagramma di sequenza per il caso d'uso InserisciAcquisto



Dal diagramma di sequenza sopra si evince la necessità di aggiungere i seguenti metodi:

- richiediAcquisto() alla classe Acquista;
- getInfoProdotto() alla classe Prodotto;
- verificaDisponibilita() alla classe Acquista;
- calcolaImporto() alla classe Acquista;
- setDatiAcquisto() alla classe Acquista;
- aggiornaTotalePunti() alla classe Acquista.

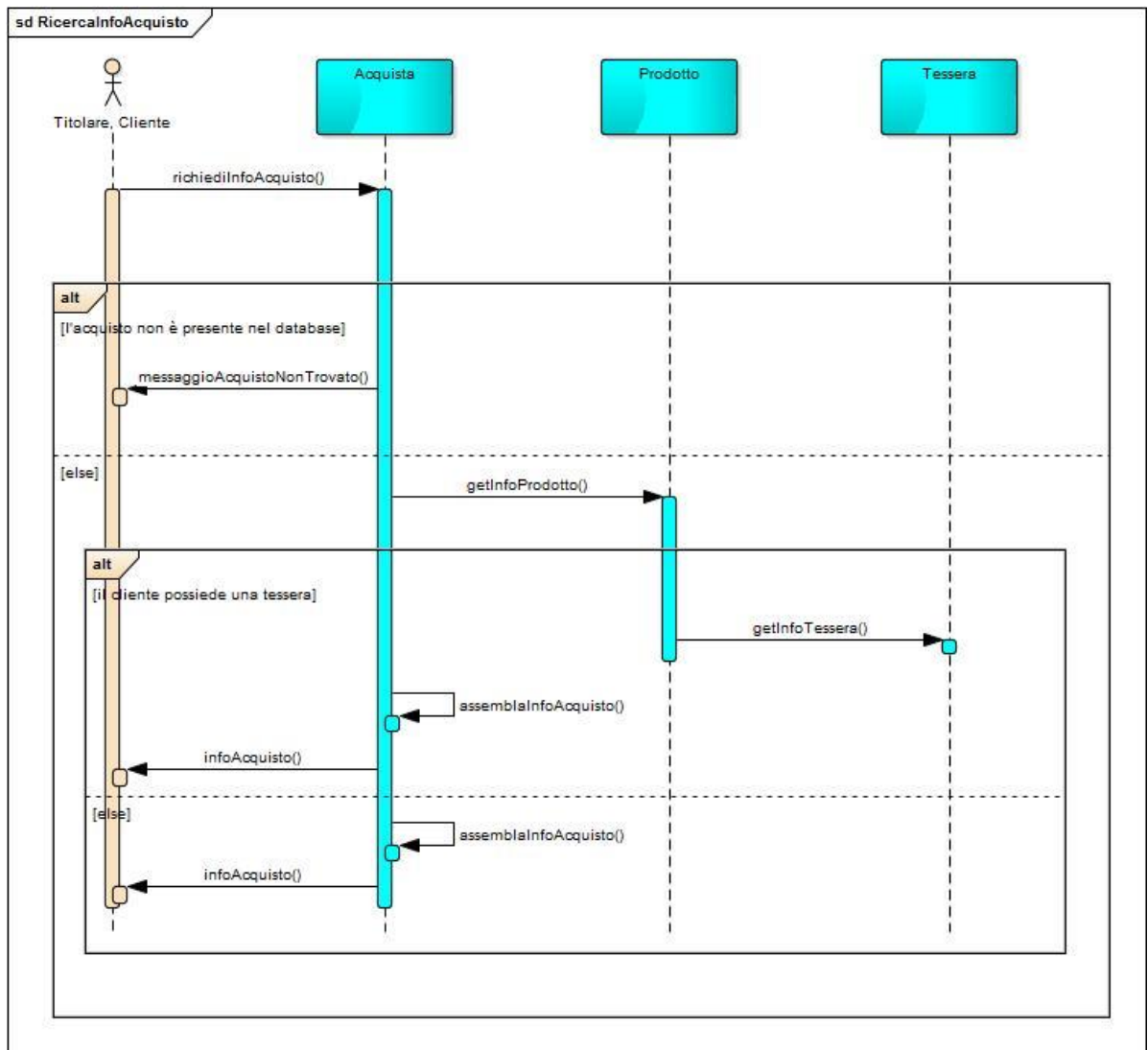
➤ **Diagramma di sequenza per il caso d'uso ModificaAcquisto**



Dal diagramma di sequenza sopra si evince la necessità di aggiungere i seguenti metodi:

- `richiediModificaAcquisto()` alla classe `Acquista`;
- `setInfoProdotto()` alla classe `Prodotto`;
- `setInfoAcquisto()` alla classe `Acquista`.

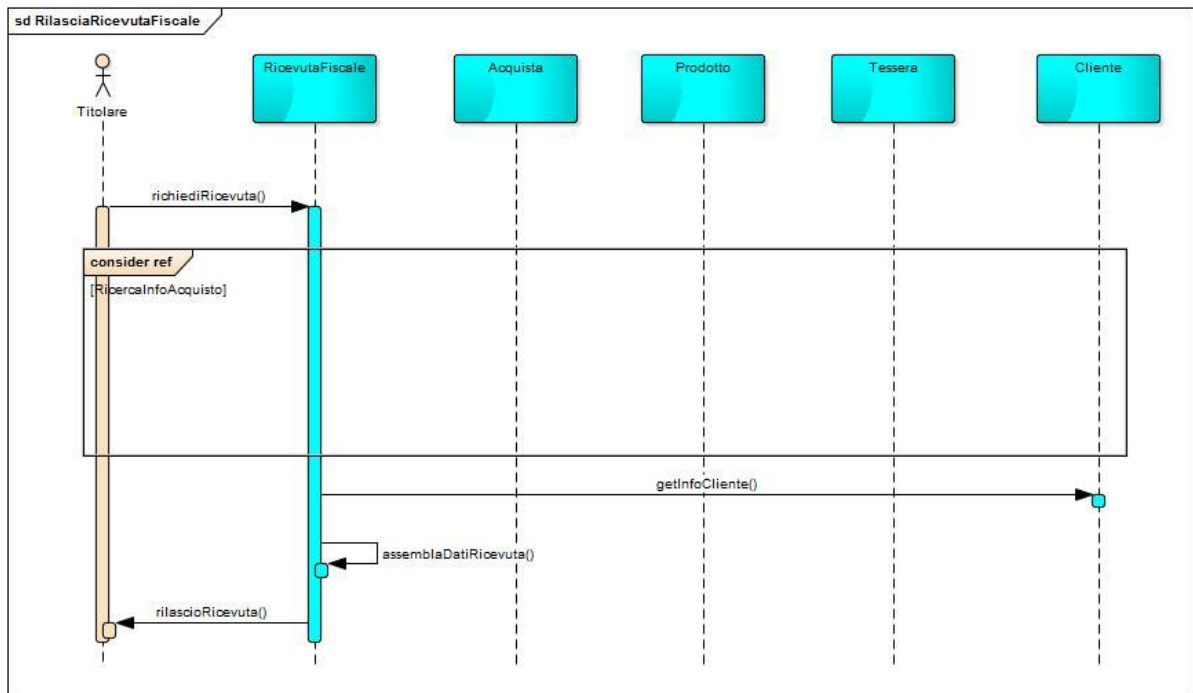
➤ **Diagramma di sequenza per il caso d'uso RicercaInfoAcquisto**



Dal diagramma di sequenza sopra si evince la necessità di aggiungere i seguenti metodi:

- `richiediInfoAcquisto()` alla classe `Acquista`;
- `getInfoProdotto()` alla classe `Prodotto`;
- `getInfoTessera()` alla classe `Tessera`;
- `assembalaInfoAcquisto()` alla classe `Acquista`.

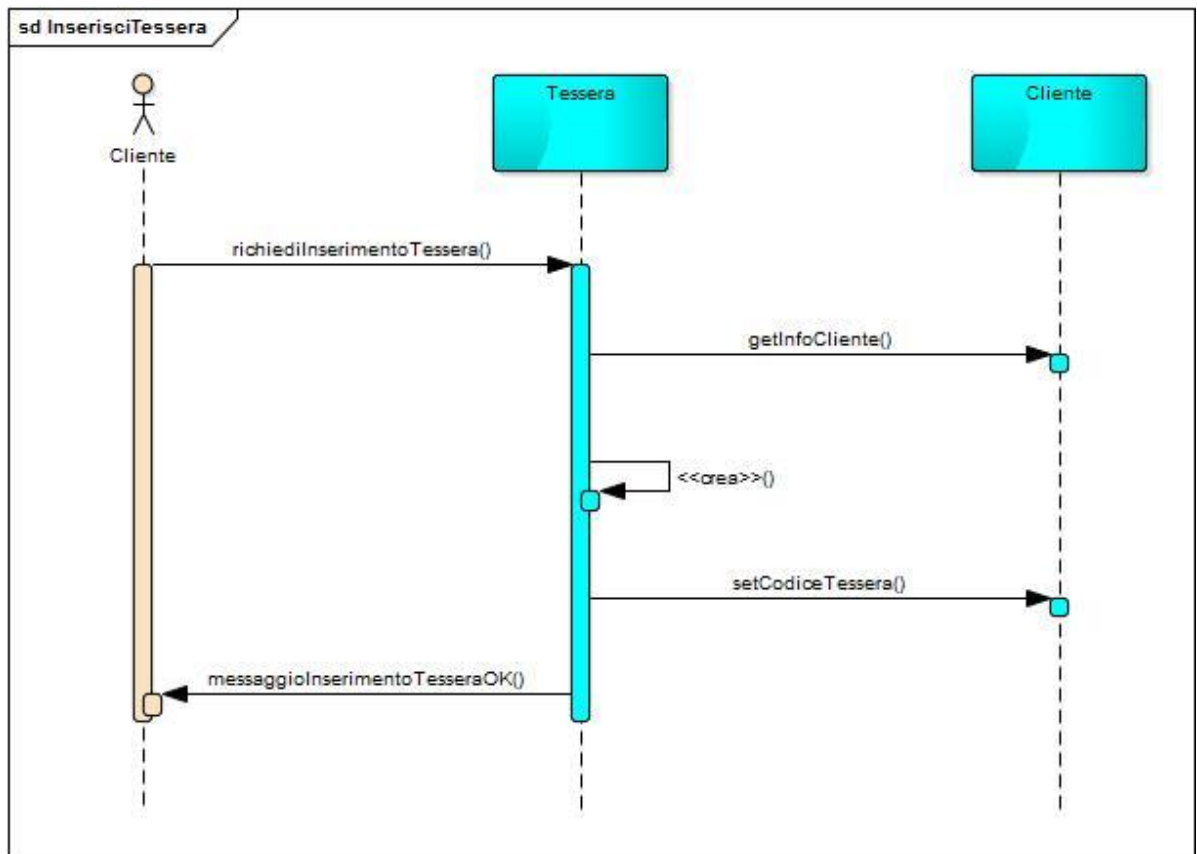
➤ **Diagramma di sequenza per il caso d'uso RilasciaRicevutaFiscale**



Dal diagramma di sequenza sopra si evince la necessità di aggiungere i seguenti metodi:

- richiediRicevuta() alla classe RicevutaFiscale;
- getInfoCliente() alla classe Cliente;
- assemblaDatiRicevuta() alla classe RicevutaFiscale;

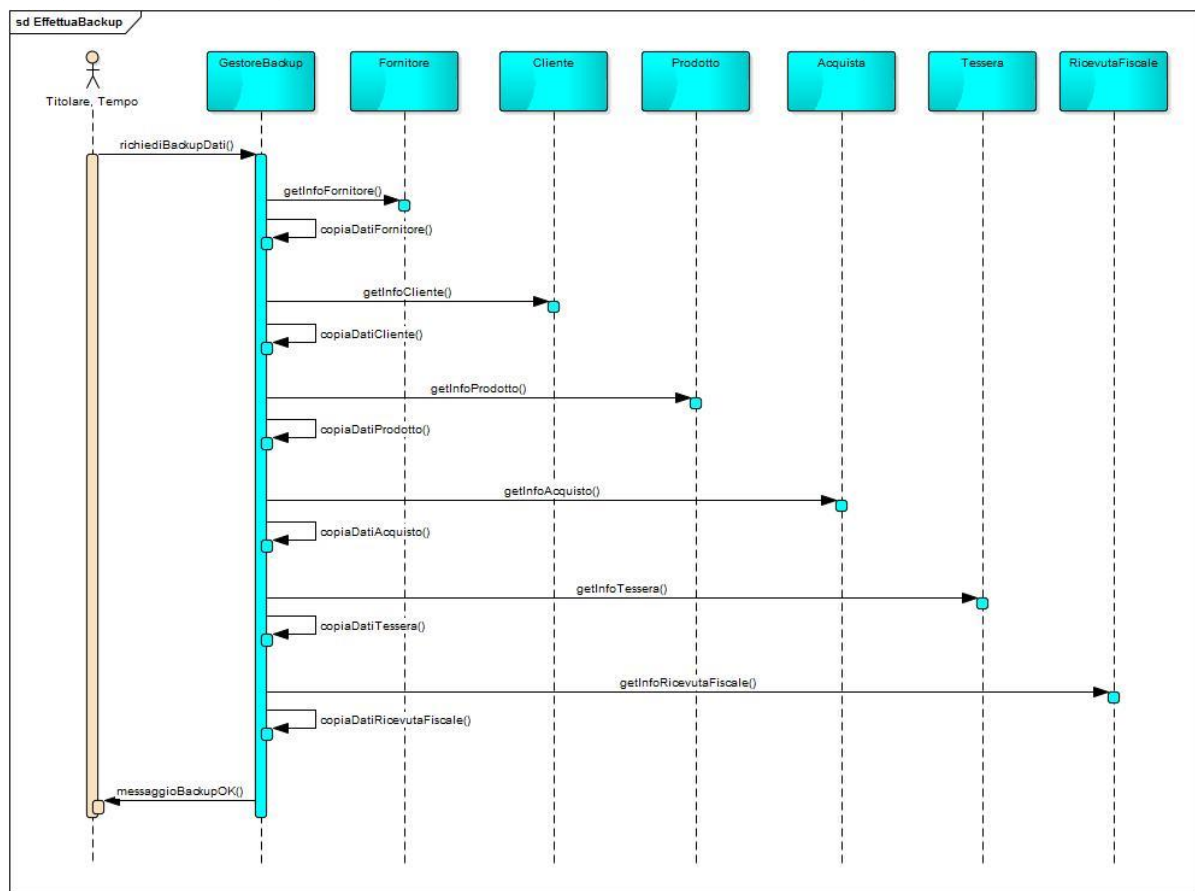
➤ **Diagramma di sequenza per il caso d'uso InserisciTessera**



Dal diagramma di sequenza sopra si evince la necessità di aggiungere i seguenti metodi:

- richiediInserimentoTessera() alla classe Tessera;
- getInfoCliente() alla classe Cliente.

➤ Diagramma di sequenza per il caso d'uso EffettuaBackup

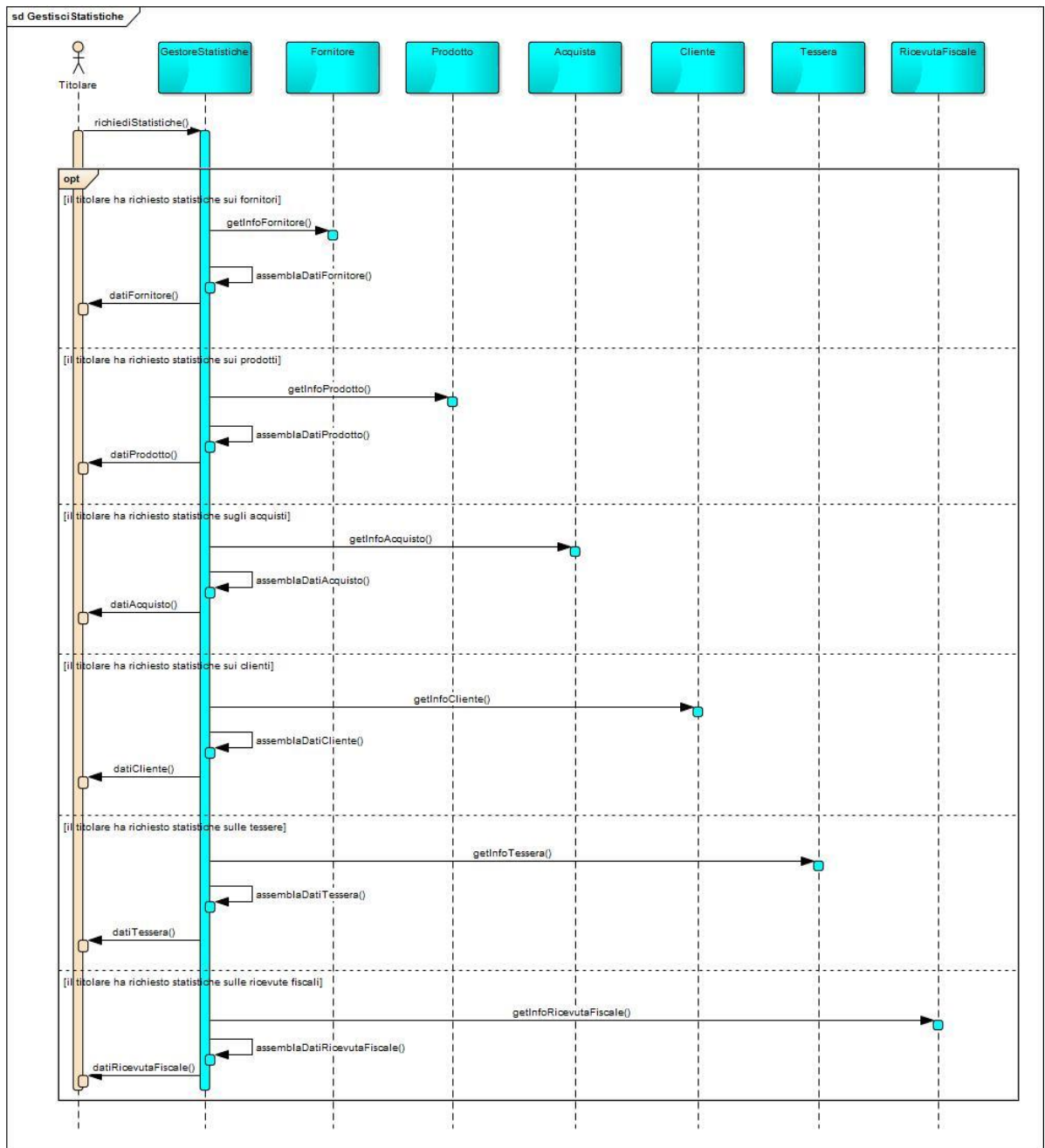


Dal diagramma di sequenza sopra si evince la necessità di aggiungere la classe GestoreBackup.

Inoltre, si evince la necessità di aggiungere i seguenti metodi:

- richiediBackupDati() alla classe GestoreBackup;
- getInfoFornitore() alla classe Fornitore;
- copiaDatiFornitore() alla classe GestoreBackup;
- getInfoCliente() alla classe Cliente;
- copiaDatiCliente() alla classe GestoreBackup;
- getInfoProdotto() alla classe Prodotto;
- copiaDatiProdotto() alla classe GestoreBackup;
- getInfoAcquisto() alla classe Acquista;
- copiaDatiAcquisto() alla classe GestoreBackup;
- getInfoTessera() alla classe Tessera;
- copiaDatiTessera() alla classe GestoreBackup;
- getInfoRicevutaFiscale() alla classe RicevutaFiscale;
- copiaDatiRicevutaFiscale() alla classe GestoreBackup.

➤ Diagramma di sequenza per il caso d'uso GestisciStatistiche



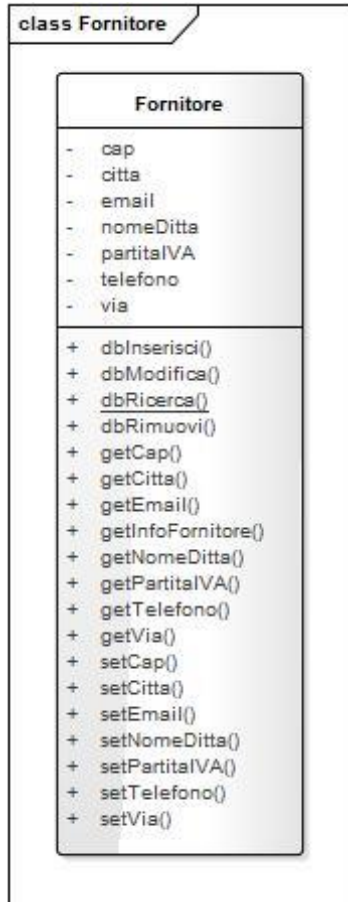
Dal diagramma di sequenza sopra si evince la necessità di aggiungere la classe GestoreStatistiche.

Inoltre, si evince la necessità di aggiungere i seguenti metodi:

- richiediStatistiche() alla classe GestoreStatistiche;
- getInfoFornitore() alla classe Fornitore;
- assemblaDatiFornitore() alla classe GestoreStatistiche;
- getInfoCliente() alla classe Cliente;
- assemblaDatiCliente() alla classe GestoreStatistiche;
- getInfoProdotto() alla classe Prodotto;
- assemblaDatiProdotto() alla classe GestoreStatistiche;
- getInfoAcquisto() alla classe Acquista;
- assemblaDatiAcquisto() alla classe GestoreStatistiche;
- getInfoTessera() alla classe Tessera;
- assemblaDatiTessera() alla classe GestoreStatistiche;
- getInfoRicevutaFiscale() alla classe RicevutaFiscale;
- assemblaDatiRicevutaFiscale() alla classe GestoreStatistiche.

CLASSI RAFFINATE

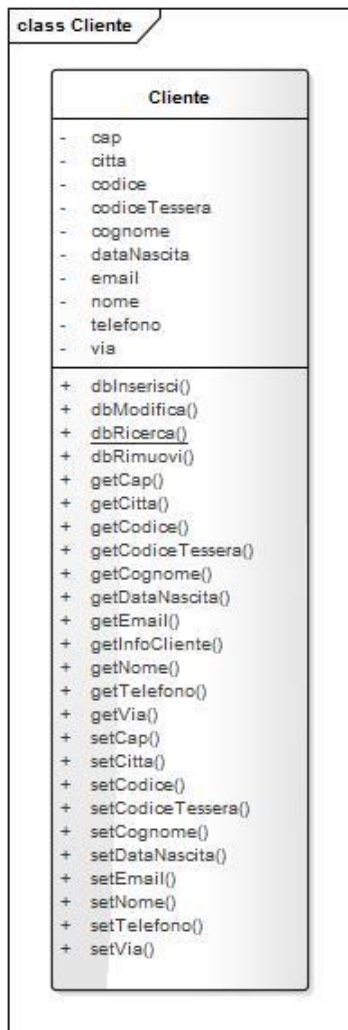
➤ CLASSE FORNITORE



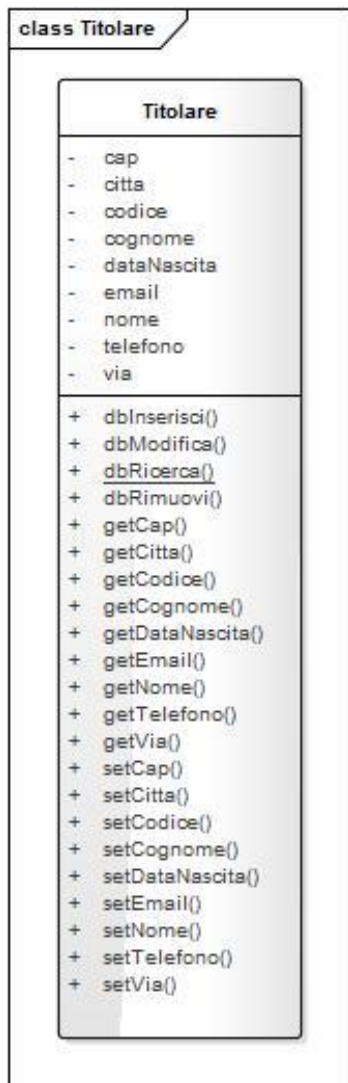
➤ CLASSE PRODOTTO



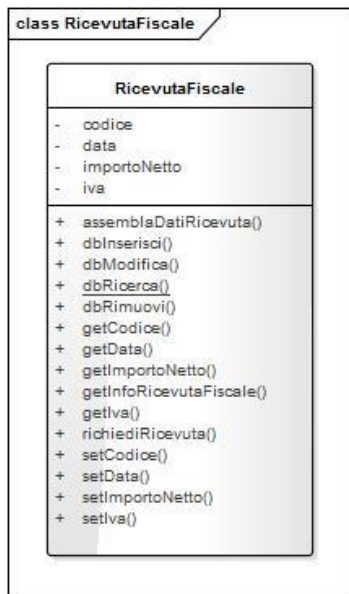
➤ CLASSE CLIENTE



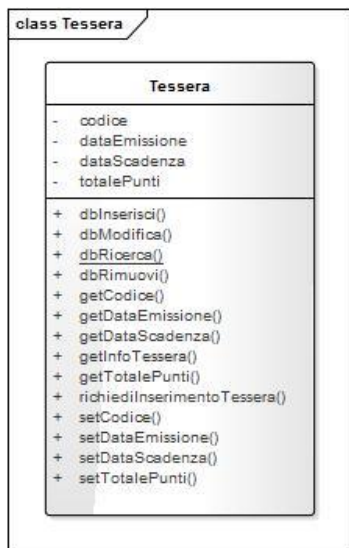
➤ CLASSE TITOLARE



➤ CLASSE RICEVUTAFISCALE



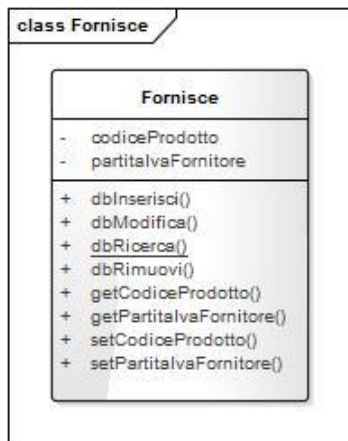
➤ CLASSE TESSERA



➤ CLASSE CONTRATTA



➤ CLASSE FORNISCE



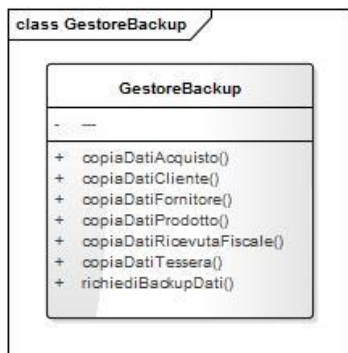
➤ CLASSE RILASCIA



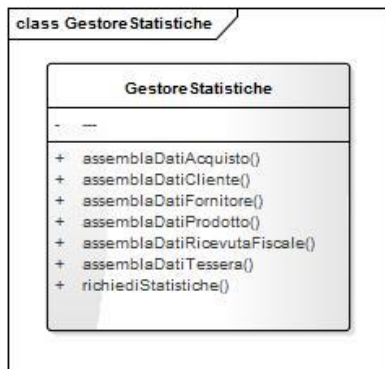
➤ CLASSE ACQUISTA



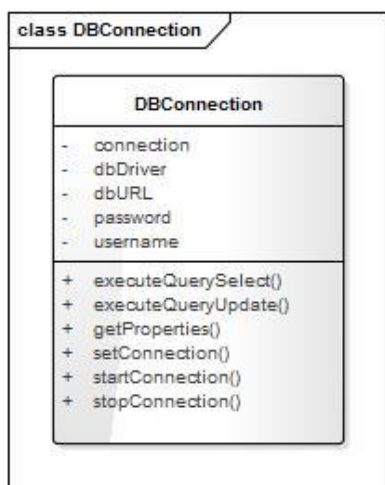
➤ CLASSE GESTOREBACKUP



➤ CLASSE GESTORESTATISTICHE



➤ CLASSE DBCONNECTION



INDICE

❖ Descrizione della realtà	pag. 1
❖ Requisiti Funzionali	pag. 2
❖ Requisiti Non Funzionali	pag. 3
❖ Diagrammi dei Casi d'Uso	pag. 4
❖ Lista dei Casi d'Uso	pag. 6
❖ Descrizione dei Casi d'Uso	pag. 7
❖ Progettazione Concettuale	
1. Diagramma E-R	pag. 24
2. Dizionario delle Entità	pag. 25
3. Dizionario delle Relazioni	pag. 26
4. Regole	pag. 27
❖ Progettazione Logica	
1. Ristrutturazione del Diagramma E-R	pag. 28
2. Diagramma E-R ristrutturato	pag. 30
3. Traduzione verso il Modello Relazionale	pag. 31
❖ Progettazione Fisica	
1. Scelta degli Indici	pag. 32
2. Stima delle Dimensioni	pag. 32
❖ Progettazione delle Componenti Applicative	
1. Diagrammi delle Classi	pag. 37
❖ Progettazione delle Funzionalità e Raffinamento delle Classi	
1. Diagrammi di Sequenza	pag. 45
2. Classi Raffinate	pag. 54