



Basi di Dati
Progetto A.A. 2022/2023

Azienda di riparazione dispositivi elettronici

0279707

Gianluca Ronzello

Indice

1. Descrizione del Minimondo.....	2
2. Analisi dei Requisiti	3
3. Progettazione concettuale.....	6
4. Progettazione logica	13
5. Progettazione fisica	25
Appendice: Implementazione	Errore. Il segnalibro non è definito.

1. Descrizione del Minimondo

L'azienda FixIt S.r.l. si occupa di riparazione di dispositivi elettronici. In particolare, ripara cellulari e computer portatili. I clienti dell'azienda portano in sede dispositivi guasti o rotti e richiedono la sostituzione di componenti elettroniche per dare nuova vita ai loro dispositivi. Per quanto riguarda i cellulari, l'azienda è in grado di sostituire: schermi rotti, connettori USB non funzionanti, batterie, schede madri. Rispetto ai portatili, l'azienda sostituisce tastiere, schermi, trackpad, batterie, ventole, schede madri, moduli RAM. L'azienda tratta differenti marche di prodotti e, per ogni marca, differenti modelli. Per ciascun modello è noto quali sono le componenti che sono necessarie ad una riparazione. La gestione di magazzino tiene traccia della disponibilità di ciascuna componente, che per gestione interna è identificata da un codice alfanumerico e riporta una descrizione testuale della componente stessa. Per questo motivo, è in grado di interrogare l'applicazione per sapere qual è la giacenza di magazzino di ogni componente. Per ogni componente viene tenuto traccia del fornitore. Ogni fornitore è identificato dal nome, dalla partita IVA, da un indirizzo e da un numero arbitrario di recapiti (telefono, email, eccetera). Se il report sulla giacenza di magazzino indica una scarsità di prodotti, la gestione di magazzino può effettuare un nuovo ordine di componenti. In ciascun ordine viene indicato il numero di pezzi che si intende acquistare dal fornitore. Quando un cliente porta un dispositivo in riparazione, la segreteria registra l'intervento necessario. Tale intervento è corredato da una breve descrizione del malfunzionamento. Il personale tecnico dell'azienda, a valle dell'ispezione del dispositivo, indica quali componenti sono necessarie per la riparazione. Per calcolare il costo dell'intervento, ogni componente è associata ad un prezzo. Il costo di manodopera è calcolato nei termini del 10% del costo delle componenti necessarie alla riparazione. Il personale di segreteria, utilizzando le informazioni fornite dai tecnici, può calcolare il costo dell'intervento. A valle della riconsegna del prodotto al cliente, la segreteria può marcare una scheda di intervento come conclusa, ad indicare che il pagamento dell'intervento è stato completato dal cliente.

2. Analisi dei Requisiti

Identificazione dei termini ambigui e correzioni possibili

Linea	Termine	Nuovo termine	Motivo correzione
1	In particolare	Esclusivamente	Maggiore precisione a seguito della conversazione con il cliente
8	Prodotto	Dispositivo elettronico	Sinonimo
14	Fornitore	fornitori	A seguito della conversazione con il cliente si è stabilito che uno stesso componente può essere acquistato da più fornitori e ognuno ha un proprio prezzo di vendita
23	riparazione	intervento	sinonimo

Specificazione disambiguata

L'azienda FixIt S.r.l. si occupa di riparazione di dispositivi elettronici. Ripara esclusivamente cellulari e computer portatili. I clienti dell'azienda portano in sede dispositivi guasti o rotti e richiedono la sostituzione di componenti elettroniche per dare nuova vita ai loro dispositivi.

Per quanto riguarda i cellulari, l'azienda è in grado di sostituire: schermi rotti, connettori USB non funzionanti, batterie, schede madri.

Rispetto ai portatili, l'azienda sostituisce tastiere, schermi, trackpad, batterie, ventole, schede madri, moduli RAM.

L'azienda tratta differenti marche di dispositivi e, per ogni marca, differenti modelli. Per ciascun modello è noto quali sono le componenti che sono necessarie ad una riparazione.

Il componente è identificato da un codice alfanumerico e ne rappresentiamo la disponibilità nel magazzino, una descrizione testuale, i fornitori e il prezzo.

Ogni fornitore ha un proprio prezzo di vendita per un componente.

Per il fornitore rappresentiamo nome, partita IVA, indirizzo, recapiti telefonici e e-mail.

La gestione di magazzino verifica la giacenza in magazzino di un componente, se c'è scarsità di prodotti si può effettuare un nuovo ordine dei componenti.

Per l'ordine rappresentiamo la quantità di componenti e il fornitore.

Si tiene traccia di un ordine per 30 anni.

La segreteria registra l'intervento necessario a un dispositivo e una descrizione del

malfunzionamento.

Il personale tecnico a valle dell'ispezione indica le componenti necessarie per l'intervento.

La segreteria calcola il costo dell'intervento.

Il costo è dato dalla somma del costo delle componenti e il costo di manodopera.

Il costo di manodopera è calcolato nei termini del 10% del costo delle componenti necessarie alla riparazione

La segreteria può marcare una scheda di intervento come conclusa, ad indicare che il pagamento dell'intervento è stato completato dal cliente.

Si tiene traccia di un intervento per 30 anni.

Glossario dei Termini

Termine	Descrizione	Sinonimi	Collegamenti
Dispositivo	Dispositivo elettronico dei clienti che deve essere riparato	Dispositivo elettronico	Modello, intervento
Componente	Pezzo necessario alla riparazione di un dispositivo	Componente elettronico	Fornitore, Ordine, Modello, intervento
Modello	Modello di un dispositivo elettronico		Dispositivo, Componente
Segreteria	Personale addetto all'interazione con i clienti		Intervento
Fornitore	Azienda da cui FixIt S.r.l. acquista i componenti		Componente, ordine
Gestione di magazzino	Personale che si occupa della gestione del magazzino		Componente, ordine
Ordine	Ordine delle componenti effettuato da FixIt S.r.l.		Fornitore, componente
Personale tecnico	Personale che si occupa della riparazione dei dispositivi		Intervento, componente, modello
Intervento	Rappresenta la riparazione di un dispositivo		Modello, componente, segreteria, personale tecnico

Raggruppamento dei requisiti in insiemi omogenei

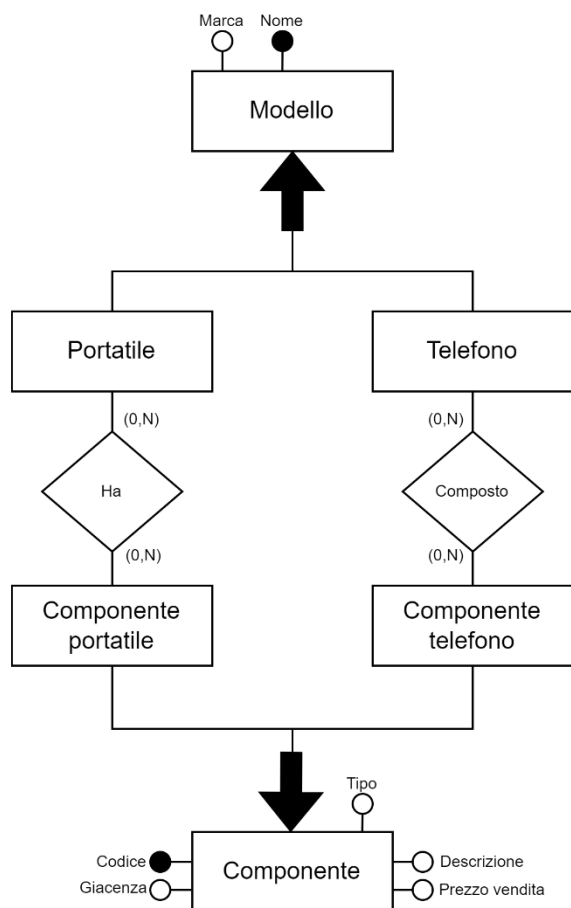
Frasi di carattere generale

L'azienda FixIt S.r.l. si occupa di riparazione di dispositivi elettronici. I clienti dell'azienda portano in sede dispositivi guasti o rotti e richiedono la sostituzione di componenti elettroniche per dare

nuova vita ai loro dispositivi.
Frase relative a dispositivo
Ripara esclusivamente cellulari e computer portatili. L'azienda tratta differenti marche di dispositivi e, per ogni marca, differenti modelli.
Frase relative a modello
L'azienda tratta differenti marche di dispositivi e, per ogni marca, differenti modelli. Per ciascun modello è noto quali sono le componenti che sono necessarie ad una riparazione. Per quanto riguarda i cellulari, l'azienda è in grado di sostituire: [...] Rispetto ai portatili, l'azienda sostituisce [...]
Frase relative a componente
Per il modello rappresentiamo la marca e le componenti necessarie alla riparazione. Il componente è identificato da un codice alfanumerico e ne rappresentiamo la disponibilità nel magazzino, una descrizione testuale, i fornitori e il prezzo. Ogni fornitore ha un proprio prezzo di vendita per un componente. Per quanto riguarda i cellulari, l'azienda è in grado di sostituire: schermi rotti, connettori USB non funzionanti, batterie, schede madri. Rispetto ai portatili, l'azienda sostituisce tastiere, schermi, trackpad, batterie, ventole, schede madri, moduli RAM.
Frase relative a fornitore
Per il fornitore rappresentiamo nome, partita IVA, indirizzo, recapiti telefonici e e-mail. Si tiene traccia di un ordine per 30 anni.
Frase relative a gestione di magazzino
La gestione di magazzino verifica la giacenza in magazzino di un componente, se c'è scarsità di prodotti si può effettuare un nuovo ordine dei componenti.
Frase relative a ordine
Se c'è scarsità di prodotti si può effettuare un nuovo ordine dei componenti. Per l'ordine rappresentiamo la quantità di componenti e il fornitore.
Frase relative a segreteria
La segreteria registra l'intervento necessario a un dispositivo. La segreteria calcola il costo dell'intervento, (il costo è dato dalla somma del costo delle componenti e il costo di manodopera.) La segreteria può marcare una scheda di intervento come conclusa, ad indicare che il pagamento dell'intervento è stato completato dal cliente.
Frase relative a intervento
Per l'intervento rappresentiamo una descrizione del malfunzionamento. Il personale tecnico a valle dell'ispezione indica le componenti necessarie per l'intervento(riparazione) La segreteria calcola il costo dell'intervento, (il costo è dato dalla somma del costo delle componenti e il costo di manodopera.) Il costo di manodopera è calcolato nei termini del 10% del costo delle componenti necessarie alla riparazione La segreteria può marcare una scheda di intervento come conclusa, ad indicare che il pagamento dell'intervento è stato completato dal cliente. Si tiene traccia di un intervento per 30 anni.
Frase relative a personale tecnico
Il personale tecnico a valle dell'ispezione indica le componenti necessarie per l'intervento(riparazione)

3. Progettazione concettuale

Costruzione dello schema E-R



Per distinguere il dispositivo salvato dall'azienda rispetto a quello fisico portato dal cliente viene creata l'entità modello, la quale ha la sua marca ed è identificato dal suo nome che è univoco.

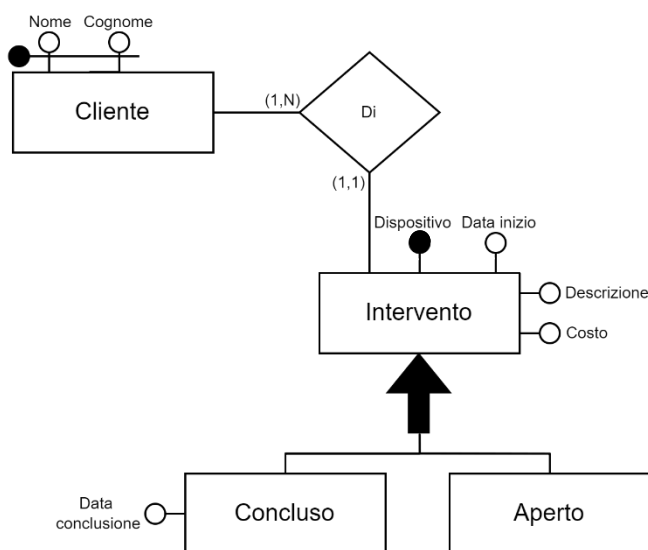
Per rappresentare che l'azienda si occupa esclusivamente di telefoni e portatili viene utilizzata una generalizzazione totale.

Stesso discorso viene fatto per le componenti di quale abbiamo la lista completa sia per portatili che telefoni.

Inoltre, da specifica il componente ha un codice alfanumerico per identificarlo, e con attributo tipo si intende che pezzo è (esempio schermo, tastiera, batteria, ecc.)

Le relazioni sono con cardinalità minima a 0 poiché può essere aggiunto un modello che ha tutte componenti nuove o può essere aggiunto un componente che non appartiene ancora a nessun modello.

La cardinalità massima è messa a N poiché un componente può appartenere a diversi modelli e un modello a diverse componenti.



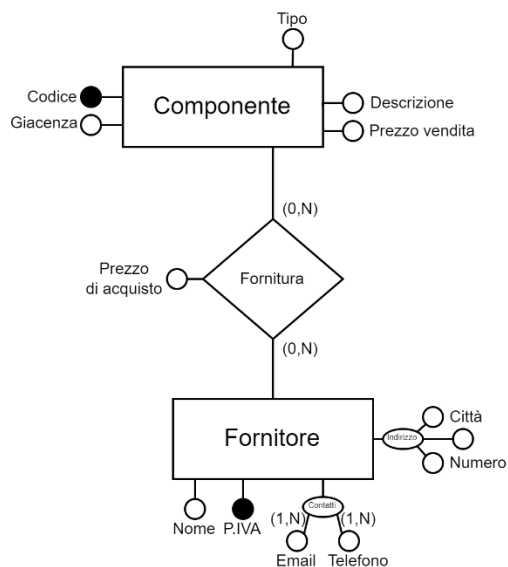
Per identificare il dispositivo fisico del cliente viene utilizzato un attributo “dispositivo”, ed è utilizzato come identificatore univoco dell’intervento.

È presente un attributo descrizione che rappresenta la descrizione del malfunzionamento.

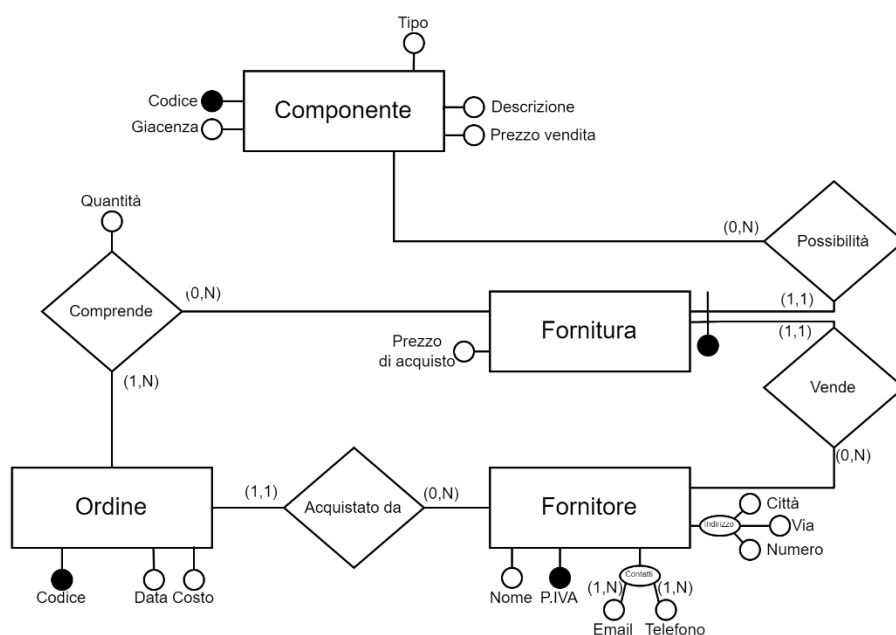
Per rappresentare che la segreteria può marcare una scheda intervento come conclusa viene usato un pattern di “Storicizzazione di entità”.

Per sapere di chi è l’intervento viene creata l’entità cliente, e siccome nelle specifiche non vi è una particolare richiesta per l’acquisizione dei dati del cliente viene preso solo il nome e il cognome.

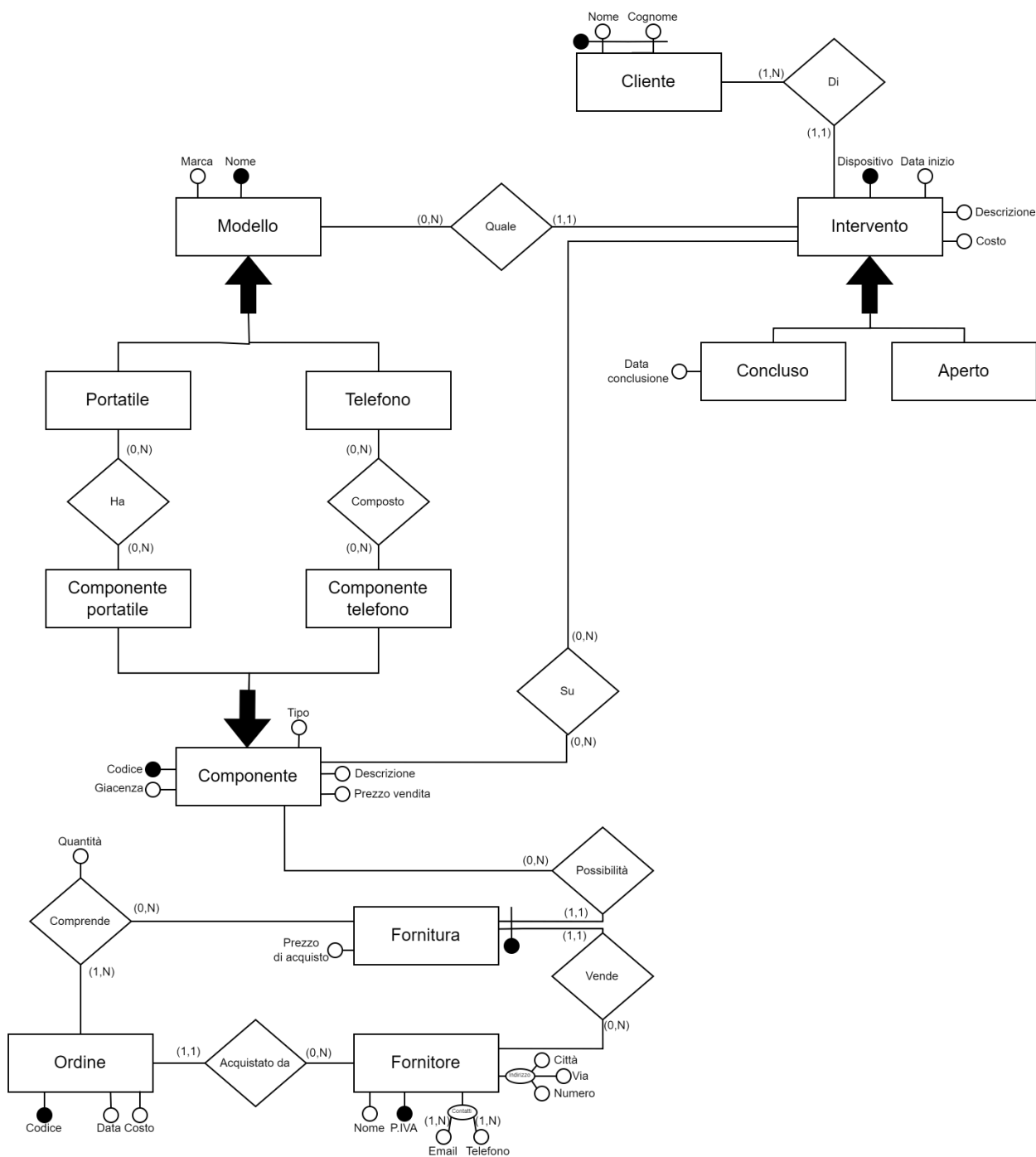
La relazione “Di”, nella direzione che va da “Cliente” a “Intervento”, ha una cardinalità minima di 1 poiché un cliente diventa tale solo dopo aver portato un dispositivo a riparare, e una cardinalità massima N poiché può fare varie riparazioni. Nella direzione opposta ha cardinalità 1:1 poiché un intervento è di un solo cliente.



La relazione “Fornitura” ha cardinalità (0,N) in entrambi i sensi poiché un fornitore può fornire varie componenti e a seguito dell’interazione con il cliente è stato stabilito che un componente può essere fornito da più fornitori, e la possibilità di prezzi diversi è stata espressa con l’attributo “prezzo d’acquisto”.



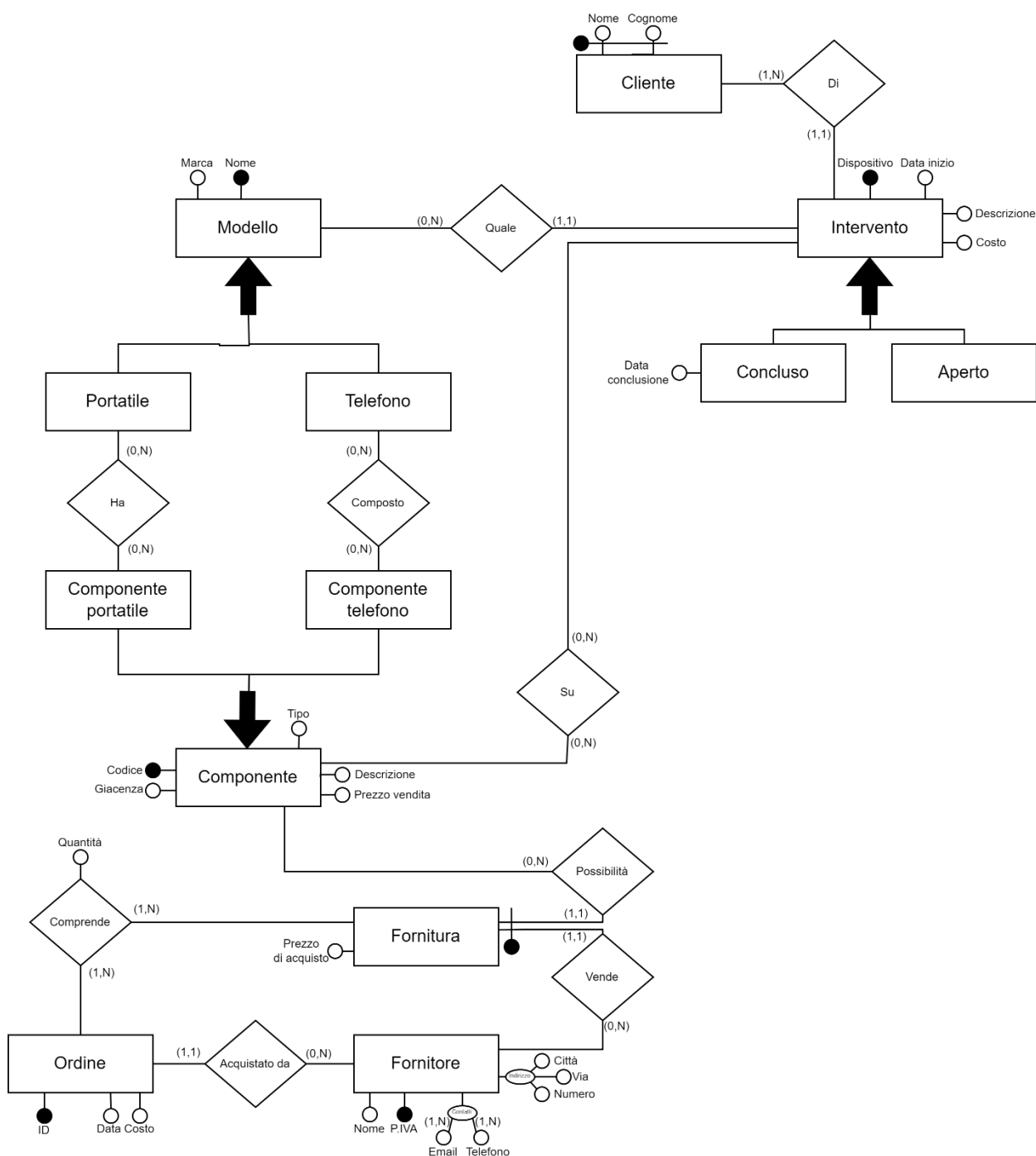
Può essere fatto un ordine di più componenti e ognuna con la propria quantità, ma da un unico fornitore. Eventualmente possono essere fatti anche più ordini a uno stesso fornitore nella stessa giornata. Si è deciso di reificare la relazione fornitura.



La relazione “Quale” ha cardinalità (0,N) poiché può essere che nessun cliente ha portato un determinato modello oppure vari interventi sono stati fatti su un particolare modello, la cardinalità (1,1) poiché un intervento viene effettuato su un unico dispositivo che può essere di un solo modello. La relazione “Su” ha cardinalità (0,N) in entrambi i sensi poiché un componente può non essere stato utilizzato per nessun intervento o per diversi interventi, la minima 0 da intervento a componente

perché le eventuali componenti utilizzate vengono scelte in seguito dal personale tecnico e massima N perché per un intervento si possono utilizzare diverse componenti.

Integrazione finale



Si è deciso di cambiare l'identificatore di ordine da codice a ID per evitare confusione con codice di componente.

Regole aziendali

La giacenza di un componente non può essere negativa.

Il prezzo d'acquisto di una fornitura deve essere non negativo.

Il prezzo di un componente deve essere non negativo.

La data di conclusione di intervento deve essere maggiore o uguale della data di inizio.

Le componenti di un ordine devono essere tutte di uno stesso fornitore.

Il costo di un intervento si ottiene sommando tutti i prezzi di vendita delle componenti utilizzate, più il 10% del totale.

Il sistema mantiene i dati relativi a ordini, interventi per 30 anni.

Dizionario dei dati

Entità	Descrizione	Attributi	Identificatori
Cliente	Le informazioni su chi porta un dispositivo da riparare		Nome, cognome
Intervento	Le informazioni su una riparazione	Descrizione, costo, cliente, data inizio	Dispositivo
Concluso	Informazioni su un intervento già pagato	Descrizione, costo, cliente, data inizio, data conclusione	Dispositivo
Aperto	Informazioni su un intervento da pagare	Descrizione, costo, cliente, data inizio	Dispositivo
Modello	Un modello che l'azienda può riparare	Marca	Nome
Portatile	Tipo di portatile che l'azienda può riparare	Marca	Nome
Telefono	Tipo di telefono che l'azienda può riparare	Marca	Nome
Componente	Informazioni su un componente per riparare un dispositivo che può essere riparato dall'azienda	Giacenza, prezzo vendita, descrizione, tipo	Codice
Componente portatile	Informazioni su un componente di un portatile	Giacenza, prezzo vendita, descrizione, tipo	Codice
Componente telefono	Informazioni su un componente di un telefono	Giacenza, prezzo vendita, descrizione, tipo	Codice
Fornitore	Informazioni su un fornitore dell'azienda	Nome, e-mail, telefono, città, via, numero	P.IVA
Fornitura	Informazioni su un componente venduto	Prezzo d'acquisto	Codice P.IVA

	da uno specifico fornitore		
Ordine	Informazioni su un ordine effettuato dall'azienda	Data, costo, fornitore	ID

4. Progettazione logica

Volume dei dati

Il sistema mantiene i dati relativi a ordini, interventi per 30 anni.

Si stima che la media di componenti sostituiti in un intervento sia circa 2

Si stima che un ordine sia in media di 4 componenti

Si stima che in media un cliente faccia 20 interventi

Uno stesso componente può essere acquistato in media da 10 fornitori differenti

Concetto nello schema	Tipo ¹	Volume atteso
Cliente	E	100.000
Intervento	E	2.000.000
Di	R	2.000.000
Concluso	E	1.986.000
Aperto	E	14000
Modello	E	3000
Portatile	E	1500
Telefono	E	1500
Quale	R	2.000.000
Componente	E	4500
Componente portatile	E	3000
Componente telefono	E	1500
Ha	R	12000
Composto	R	6000
Su	R	4.000.000
Fornitore	E	100
Fornitura	E	50000
Vende	R	50000
Possibilità	R	50000
Ordine	E	220000
Acquistato da	R	220000
Comprende	R	880000

Tavola delle operazioni

¹ Indicare con E le entità, con R le relazioni

Cod.	Descrizione	Frequenza attesa
L1	Login	300/giorno
SE1	Registra intervento	200/giorno
SE2	Calcola costo	200/giorno
SE3	Registra conclusione	200/giorno
SE4	Visualizza interventi cliente	1/giorno
SE5	Visualizza intervento concluso	40/mese
SE6	Elimina intervento aperto	5/giorno
SE7	Aggiungi fornitore	10/anno
SE8	Elimina fornitore	1/anno
SM 1	Visualizza ordine	20/giorno
GM1	Visualizza componente magazzino	600/giorno
GM2	Registra ordine	20/giorno
GM3	Aggiungi componente	3/mese
GM4	Aggiungi modello	3/mese
GM5	Modifica quantità componente	600/giorno
GM6	Aggiungi fornitura	3/mese
GM7	Aggiorna prezzo fornitura	3/mese
GM8	Aggiorna prezzo componente	5/giorno
GM9	Visualizza fornitore	20/giorno
GM10	Elimina modello	3/mese
GM11	Elimina componente	3/mese
GM12	Elimina ordine	5/giorno
ST1	Visualizza intervento aperto	200/giorno
MT1	Visualizza modello	200/giorno
ST2	Visualizza interventi aperti	50/giorno
TE1	Registra componenti intervento	200/giorno
TE2	Elimina componenti intervento	10/giorno

Costo delle operazioni

Concetto: numero accessi | tipo di accessi | totale

Considerando che in nessuna delle operazioni si fa distinzione tra telefono e portatile (ne quando si parla di modello che di componente), per il successivo calcolo si è supposto che fossero uniti in quanto se separati avrebbero avuto lo stesso costo siccome si doveva vedere in entrambe le tabelle.

Le due relazioni 'Ha' e 'Composto' sono fuse in 'Ha'

SE1 – Registra intervento

Se intervento concluso e aperto fossero uniti

- Intervento: 1 S 4.000.000
- Cliente: 1 S 200.000
- Di: 1 S 200.000

- Quale: 1 S 4.000.000
- Modello: 1 L 3000

Costo totale 8.403.000

Accessi /giorno 1.680.600.000

Accorpendo l'entità cliente e intervento, e separando intervento concluso da aperto

- Aperto: 1 S 28.000
- Quale: 1 S 28.000
- Modello 1 L 3.000

Costo totale 59.000

Accessi/giorno 11.800.000

Risparmio 1.668.800.000 di accessi

SE2 – Calcola costo

Se intervento concluso e aperto fossero uniti

- Su: 2 L 8.000.000
- Componente: 2 L 9.000
- Intervento: 1 S 4.000.000

Costo 12.009.000

Accessi/giorno 2.401.800.000

Separando intervento concluso da aperto

Su_aperto è la nuova relazione che si verrebbe a crea per tenere collegato intervento aperto e componenti, per il volume si tiene presente la stima di 2 componenti per intervento

- Aperto: 1 S 28.000
- Su_aperto: 2 L 56.000
- Componente: 2 L 18.000

Costo 102.000

Accessi/giorno 20.400.000

Risparmio 2.381.400.000 di accessi

SE3 - Registra conclusione

Se intervento concluso e aperto fossero uniti

- Intervento: 1 S 4.000.000

Costo 4.000.000

Accessi/giorno 800.000.000

Separando intervento concluso e aperto

- Concluso; 1 S 3.972.000
- Quale_concluso: 1 S 3.972.000
- Su_concluso: 2 S 15.888.000
- Aperto: 1 S 28.000
- Su_aperto: 2 S 112.000
- Quale_aperto: 1 S 28.000
- Componente: 2 L 9.000
- Modello 1 L 3.000

Costo 24.012.000

Accessi/giorno circa 4.802.400.000

Peggioramento circa 4.000.000.000 accessi

SE4 Visualizza interventi cliente

- Intervento: 20 L 40.000.000
- Su: 40 L 80.000.000
- Quale: 20 L 40.000.000
- Modello: 20 L 60.000
- Componente: 40 L 180.000

Costo 160.240.000

Accessi/giorno 160.240.000

Se intervento aperto e concluso fossero separati sarebbe quasi uguale perché bisogna vedere sia tra quelli aperti che conclusi

SE5 Visualizza intervento concluso

Se intervento concluso e aperto fossero uniti

- Intervento: 1 L 2.000.000
- Di: 1 L 2.000.000
- Cliente: 1 L 100.000
- Su: 2 L 8.000.000
- Componente: 2 L 9.000
- Quale: 1 L 2.000.000

- Modello: 1 L 3.000

Costo 14.112.000

Accessi/mese 564.480.000

Se invece fossero separati ma uniti con cliente

- Concluso: 1 L 1.986.000
- Su_concluso: 2 L 7.944.000
- Quale_concluso: 1 L 1.986.000
- Componente: 2 L 9.000
- Modello: 1 L 3.000

Costo 11.928.000

Accessi/mese 477.120.000

Miglioramento circa 70.000.000 accessi/mese

SE6 Elimina intervento aperto

Se intervento aperto e concluso fossero uniti

- Intervento: 1 S 4.000.000
- Di: 1 S 4.000.000
- Cliente: 1 S 200.000
- Quale: 1 S 4.000.000
- Su: 2 S 16.000.000

Costo 28.400.000

Accessi/giorno 142.000.000

Se intervento aperto e concluso fossero separati ma uniti con cliente

- Aperto: 1 S 28.000
- Quale_aperto: 1 S 28.000
- Su_aperto: 2 S 112.000

Costo 168.000

Accessi/giorno 840.000

Miglioramento 141.160.000 accessi/giorno

SE7 Aggiungi fornitore

- Fornitore: 1 S 200

Costo 200

Accessi/giorno 2000 anno

SE8 Elimina fornitore

- Fornitore: 1 S 200
 - Acquistato da: 1 L 220.000
 - Vende: 500 S 70.000.000
 - Fornitura: 500 S 70.000.000
 - Possibilità: 500 S 70.000.000
- Costo 210.220.200
- Accessi/anno 210.220.200

SM1 Visualizza ordine

- Ordine: 1 L 220.000
 - Comprende: 4 L 3.520.000
 - Fornitura: 4 L 280.000
 - Componente: 4 L 18.000
 - Acquistato_da: 1 L 220.000
 - Fornitore: 1 L 100
- Costo 4.258.000
- Accessi/giorno 85.160.000

GM1 Visualizza componente magazzino

Si stima che il report venga fatto su una media di 5 componenti

- Componente: 5 L 22.500
- Costo 22.500
- Accessi/giorno 13.500.000

GM2 Registra ordine

Si stima che l'ordine venga fatto su una media di 4 componenti

- Ordine: 1 S 440.000
 - Acquistato da: 1 S 440.000
 - Comprende: 4 S 7.040.000
 - Fornitore: 1 L 100
 - Fornitura: 4 L 280.000
- Costo 8.200.100

Accessi/giorno 164.000.000 giorno

GM3 Aggiungi componente

- Componente: 1 S 9.000
Costo 9.000
Accessi/mese 27.000

GM4 Aggiungi modello

- Modello: 1 S 6000
- Ha: 6 S 216.000
Costo 222.000
Accessi/mese 666.000

GM5 Modifica quantità componente

- Componente: 1 S 9.000
Costo 9.000
Accessi/giorno 5.400.000

GM6 Aggiungi fornitura

Viene considerato che si effettua su 50 componenti

- Fornitura: 50 S 7.000.000
- Vende: 50 S 7.000.000
- Possibilità: 50 S 7.000.000
- Componente: 50 L 225.000
- Fornitore: 50 L 5000
Costo 21.230.000
Accessi/mese 63.690.000

GM7 Aggiorna prezzo fornitura

Viene effettuato in media su 50 componenti

- Fornitura: 50 S 7.000.000
Costo 7.000.000
Accessi/mese 21.000.000

GM8 Aggiorna prezzo componente

- Componente: 1 S 9.000
Costo 9.000
Accessi/giorno 45.000

GM9 Visualizza fornitore

- Fornitore: 1 L 100
Costo 100

Accessi/giorno 2.000

GM10 Elimina modello

- Modello: 1 S 6.000
- Ha: 6 S 216.000
- Quale: 1 L 2.000.000

Costo: 2.222.000

Accessi/mese 6.666.000

GM11 Elimina componente

- Componente: 1 S 9.000
- Ha: 1 L 18.000
- Fornitura: 3 S 420.000
- Possibilità: 3 S 420.000

Costo 867.000

Accessi mese 2.601.000

GM12 Elimina ordine

- Ordine: 1 S 440.000
- Comprende: 4 S 7.040.000
- Acquistato_da: 1 S 440.000

Costo 7.920.000

Accessi/giorno 39.600.000

ST1 Visualizza intervento aperto

Se intervento aperto e concluso fossero uniti

- Intervento: 1 L 2.000.000
- Di: 1 L 2.000.000
- Cliente: 1 L 100.000
- Quale: 1 L 2.000.000
- Modello: 1 L 3.000
- Su: 2 L 8.000.000
- Componente: 2 L 9.000

Costo 14.112.000

Accessi/giorno 2.822.400.000

Se intervento concluso e aperto fossero separati ma uniti a cliente

- Aperto: 1 L 14.000
- Quale: 1 L 14.000
- Modello: 1 L 3.000
- Componente: 2 L 9.000
- Su_aperto: 2 L 56.000

Costo 96.000

Accessi/giorno 19.200.000

Miglioramento circa 2.800.000.000 accessi/giorno

MT1 Visualizza modello

- Modello: 1 L 3.000
- Ha: 6 L 108.000
- Componente: 6 L 27.000

Costo 138.000

Accessi/giorno 27.600.000

ST2 Visualizza interventi aperti

(Tendenzialmente gli interventi aperti sono molto meno di 14.000 poiché sarebbe una settimana in cui viene accettato lavoro ma non svolto, ma ci si potrebbe arrivare in casi eccezionali)

Se intervento aperto e concluso fossero uniti

- Intervento: 200 L 400.000.000
- Quale: 200 L 400.000.000
- Su: 400 L 1.600.000.000
- Di: 200 L 400.000.000
- Cliente: 200 L 200.000
- Componente: 400 L 1.800.000
- Modello: 200 L 600.000

Costo circa 2.802.000.000

Accessi/giorno circa 100.000.000.000

Se intervento aperto e concluso fossero separati ma uniti a cliente

- Aperto: 200 L 2.800.000
- Quale_aperto: 200 L 2.800.000
- Su_aperto: 400 L 11.200.000
- Componente: 400 L 1.800.000

- Modello: 200 L 600.000

Costo 19.200.000

Accessi/giorno 960.000.000

Miglioramento 99.040.000.000 accessi/giorno

TE1 Registra componenti intervento

Si tiene presente la stima di 2 componenti per intervento

Se intervento concluso e aperto fossero uniti

- Intervento: 2 L 4.000.000
- Su: 2 S 16.000.000
- Componente: 2 L 9.000

Costo 20.009.000

Accessi/giorno 400.001.800

Separando intervento concluso da aperto ma unendoli a cliente

- Aperto: 2 L 28.000
- Su_aperto: 2 S 112.000
- Componente: 2 L 9.000

Costo 149.000

Accessi/giorno 29.800.000

Miglioramento circa 371.000.000 accessi

TE2 Elimina componente intervento

Se intervento concluso e aperto fossero uniti

- Su: 1 S 8.000.000

Costo 8.000.000

Accessi/giorno 80.000.000

Se intervento concluso e aperto fossero separati

- Su_aperto: 1 S 56.000

Costo 56.000

Accessi/giorno 560.000

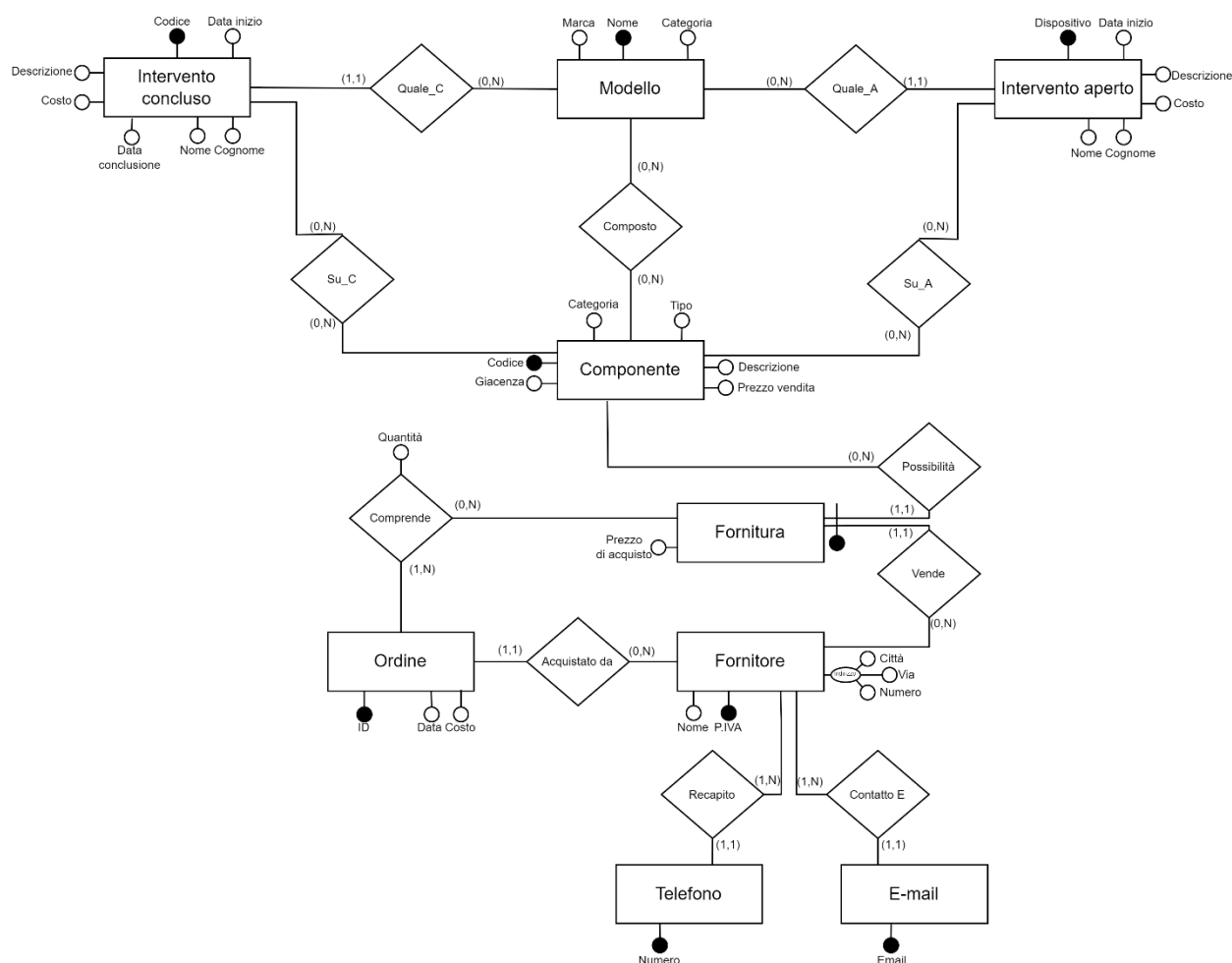
Miglioramento circa 79.500.000 accessi

Ristrutturazione dello schema E-R

L'attributo "costo" dell'entità "intervento" non è una ridondanza, anche se la prima volta è derivato dall'attributo "prezzo vendita" dell'entità "componente". Si tiene in considerazione che nel tempo il

prezzo di un componente può variare, mentre all'applicazione viene richiesto di mantenere i dati di un intervento per almeno 30 anni, il quale costo (eventualmente già pagato) rimane lo stesso.

Lo stesso discorso viene fatto per l'ordine.



In seguito al calcolo delle operazioni si è scelto di accorpare l'entità "intervento" nelle figlie.

Per intervento concluso l'identificatore viene cambiato da dispositivo a codice, poiché non essendoci più il dispositivo fisico da identificare quel valore ora serve solo a identificare l'intervento.

Invece per le entità "Modello" e "Componente" si è scelto di accorpare le figlie nel genitore poiché le operazioni non fanno distinzione tra le occorrenze e gli attributi delle varie entità.

Per distinguere se un'occorrenza apparteneva a "Telefono" o "Portatile" viene aggiunto l'attributo categoria.

Essendo telefono ed e-mail attributi multi-valore vengono eliminati.

Trasformazione di attributi e identificatori

Non è stata necessaria

Traduzione di entità e associazioni

Intervento aperto (Dispositivo, DataInizio, Descrizione, Costo*, Nome, Cognome, Modello)

Intervento concluso (Codice, DataInizio, DataConclusione, Descrizione, Costo, Nome, Cognome, Modello)

Su_A (Dispositivo, Componente)

Su_C (Intervento concluso, Componente)

Componente (Codice, Categoria, Tipo, Descrizione, PrezzoVendita, Giacenza)

Composto (Modello, Componente)

Fornitore (P.IVA, Nome, Città, Via, Numero)

Telefono (Numero, Fornitore)

E-mail (Email, Fornitore)

Fornitura (Fornitore, Componente, PrezzoD'Acquisto)

Ordine (ID, Data, Fornitore, Costo)

Comprende (Ordine, Fornitura.Componente, Fornitura.Fornitore, Quantità)

Normalizzazione del modello relazionale

Il modello non è in forma normale per la relazione 'Comprende' perché il fornitore si dovrebbe ricavare direttamente da ordine. Bisogna controllare che il fornitore di ordine e il fornitore di fornitura siano lo stesso.

5. Progettazione fisica

Utenti e privilegi

Si prevedono quattro ruoli, per implementare il Principle of Least Privilege:

- Login:
 - Grant in esecuzione sull'operazione L1
- Segreteria
 - Grant in esecuzione sulle operazioni SE1, SE2, SE3, SE4, SE5, SE6, SE7, SE8, ST1, ST2, SM1
- Magazzino
 - Grant in esecuzione sulle operazioni GM1, GM2, GM3, GM4, GM5, GM6, GM7, GM8, GM10, GM11, GM12, MT1, SM1
- Tecnico
 - Grant in esecuzione sull'operazione TE1, TE2, MT1, ST1, ST2

Per identificare gli utenti si introduce una tabella Utenti per mantenerne le credenziali.

Strutture di memorizzazione

Tabella <Intervento aperto>		
Colonna	Tipo di dato	Attributi ²
Dispositivo	Int	PK NN AI
DataInizio	DATE	NN
Descrizione	MEDIUMTEXT	NN
Costo	DECIMAL(10,2)	
Nome	VARCHAR(45)	NN
Cognome	VARCHAR(45)	NN
Modello_Nome	VARCHAR(45)	NN

Tabella <Intervento concluso>		
Colonna	Tipo di dato	Attributi ³
Codice	Int	PK NN
DataInizio	DATE	NN
Descrizione	MEDIUMTEXT	NN
Costo	DECIMAL(10,2)	NN
Nome	VARCHAR(45)	NN
Cognome	VARCHAR(45)	NN
DataConclusione	DATE	NN
Modello_Nome	VARCHAR(45)	NN

² PK = primary key, NN = not null, UQ = unique, UN = unsigned, AI = auto increment. È ovviamente possibile specificare più di un attributo per ciascuna colonna.

Tabella <Componente>		
Colonna	Tipo di dato	Attributi ⁴
Codice	VARCHAR(45)	PK NN
Giacenza	INT	NN
Tipo	ENUM	NN
Categoria	ENUM	NN
Descrizione	MEDIUMTEXT	NN
Prezzo Vendita	DECIMAL(10,2)	NN

Tabella <Componente_has_Fornitore >		
Colonna	Tipo di dato	Attributi ⁵
Componente_Codice	VARCHAR(45)	PK NN
Fornitore_P.IVA	VARCHAR(45)	PK NN
Prezzo d'acquisto	DECIMAL(10,2)	NN

Tabella <Ordine_has_Componente_has_Fornitore >		
Colonna	Tipo di dato	Attributi ⁶
Ordine_Codice	INT	PK NN
Componente_has_Fornitore_Fo rnitore_P.IVA	VARCHAR (45)	NN
Componente_has_Fornitore_Co mponente_Codice	VARCHAR (45)	PK NN
Quantità	INT	NN

Tabella <Email >		
Colonna	Tipo di dato	Attributi ⁷
Email	VARCHAR(45)	PK NN
Fornitore_P.IVA	VARCHAR(45)	PK NN

Tabella <Fornitore>		
Colonna	Tipo di dato	Attributi ⁸
P.IVA	VARCHAR(45)	PK NN

Città	VARCHAR(45)	NN
Via	VARCHAR(45)	NN
Numero	INT	NN
Nome	VARCHAR(45)	NN

Tabella <Intervento concluso_has componente >		
Colonna	Tipo di dato	Attributi ⁹
Intervento concluso_Codice	INT	PK NN
Componente_Codice	VARCHAR(45)	PK NN

Tabella <Intervento aperto_has componente >		
Colonna	Tipo di dato	Attributi ¹⁰
Intervento aperto_dispositivo	INT	PK NN
Componente_Codice	VARCHAR(45)	PK NN

Tabella < Modello >		
Colonna	Tipo di dato	Attributi ¹¹
Nome	VARCHAR(45)	PK NN
Marca	VARCHAR(45)	NN
Categoria	ENUM	NN

Tabella <Modello_has_Componente >		
Colonna	Tipo di dato	Attributi ¹²
Modello_Nome	VARCHAR(45)	PK NN
Componente_Codice	VARCHAR(45)	PK NN

Tabella <Ordine>		
Colonna	Tipo di dato	Attributi
Codice	INT	PK NN AI
Data	DATETIME(2)	NN
Fornitore_P.IVA	VARCHAR(45)	NN

Costo	DECIMAL(10,2)	
-------	---------------	--

Tabella <Telefono >		
Colonna	Tipo di dato	Attributi ¹³
Numero	VARCHAR(64)	PK NN
Fornitore_P.IVA	VARCHAR(45)	NN

Tabella <Utenti>		
Colonna	Tipo di dato	Attributi ¹⁴
Username	VARCHAR(45)	PK NN
Password	CHAR(32)	NN
Ruolo	ENUM	NN

Indici

Non sono stati utilizzati indici

Trigger

Per sviluppare le due regole aziendali

- 1) La giacenza di un componente non può essere negativa.
- 2) Il prezzo di un componente non può essere negativo.

Vengono utilizzati i seguenti trigger

```
CREATE DEFINER = CURRENT_USER TRIGGER `fixit`.`Componente_BEFORE_INSERT`
BEFORE INSERT ON `Componente` FOR EACH ROW
BEGIN
if NEW.`Prezzovendita` < 0 or NEW.`giacenza` < 0 then
signal sqlstate '45000';
end if;
END
```

```
CREATE DEFINER = CURRENT_USER TRIGGER `fixit`.`Componente_BEFORE_UPDATE`
BEFORE UPDATE ON `Componente` FOR EACH ROW
```

```
BEGIN
if NEW.giacenza<0 or NEW.prezzovendita < 0 then
signal sqlstate'45000';
end if;
END
```

3)Il prezzo d'acquisto di una fornitura non può essere negativo.

```
CREATE DEFINER = CURRENT_USER TRIGGER
`fixit`.`Componente_has_Fornitore_BEFORE_INSERT` BEFORE INSERT ON
`Componente_has_Fornitore` FOR EACH ROW
BEGIN
if NEW.`Prezzo d'acquisto` < 0 then
signal sqlstate '45000';
end if;
END
```

```
CREATE DEFINER = CURRENT_USER TRIGGER
`fixit`.`Componente_has_Fornitore_BEFORE_UPDATE` BEFORE UPDATE ON
`Componente_has_Fornitore` FOR EACH ROW
BEGIN
if NEW.`Prezzo d'acquisto` < 0 then
signal sqlstate '45000';
end if;
END
```

4)La data di inizio di intervento non può essere maggiore della data di conclusione.

```
CREATE DEFINER = CURRENT_USER TRIGGER `fixit`.`Intervento
concluso_BEFORE_INSERT` BEFORE INSERT ON `Intervento concluso` FOR EACH ROW
BEGIN
if NEW.DataInizio>NEW.DataConclusione then
signal sqlstate '45000';
```

```
end if;  
END
```

5) Per un intervento bisogna utilizzare componenti della stessa categoria del dispositivo che si sta riparando

```
CREATE DEFINER = CURRENT_USER TRIGGER `fixit`.`Intervento  
aperto_has_Componente_BEFORE_INSERT` BEFORE INSERT ON `Intervento  
aperto_has_Componente` FOR EACH ROW  
BEGIN  
  declare c1 varchar(45);  
  declare m1 varchar(45);  
  select c.categoria into c1  
  from `componente` as c  
  where c.`codice` = NEW.`componente_codice`;  
  select m.categoria into m1  
  from `Intervento_aperto` as i join modello as m on i.modello_nome=m.nome  
  where i.dispositivo = NEW.`Intervento_aperto_Dispositivo`;  
  if c1!=m1 then  
    signal sqlstate'45000';  
  end if;  
END
```

6) Le componenti di un ordine devono essere tutte di uno stesso fornitore.

```
CREATE DEFINER = CURRENT_USER TRIGGER  
`fixit`.`Ordine_has_Componente_has_Fornitore_BEFORE_INSERT` BEFORE INSERT ON  
`Ordine_has_Componente_has_Fornitore` FOR EACH ROW  
BEGIN  
  declare f varchar(45);  
  select `fornitore_P.iva` into f  
  from ordine  
  where codice=NEW.ordine_codice;  
  if f!=NEW.`Componente_has_Fornitore_Fornitore_P.IVA` then
```

```
signal sqlstate '45000';  
end if;  
END
```

7)Le componenti di un modello devono essere della stessa categoria del modello

```
CREATE DEFINER = CURRENT_USER TRIGGER  
`fixit`.`Modello_has_Componente_BEFORE_INSERT` BEFORE INSERT ON  
`Modello_has_Componente` FOR EACH ROW  
BEGIN  
declare c1 varchar(45);  
declare m1 varchar(45);  
select c.categoria into c1  
from `componente` as c  
where c.`codice` = NEW.`componente_codice`;  
select m.categoria into m1  
from modello as m  
where m.nome = NEW.`modello_nome`;  
if c1!=m1 then  
    signal sqlstate'45000';  
end if;  
END
```

Eventi

Si è implementato un evento per poter eliminare i dati relativi a ordini e interventi che hanno più di 30 anni. L' evento viene istanziato in fase di configurazione del sistema.

```
set global event_scheduler = on;  
create event if not exists `cleanup`  
on schedule  
every 30 year  
on completion preserve  
comment 'Remove old tuples'  
do  
delete from `intervento_concluso` where `dataconclusione` < (NOW() - interval 30 year);  
delete from `ordine` where `data` < (NOW() - interval 30 year);
```

Viste

Non sono state utilizzate viste

Stored Procedures e transazioni

```
CREATE PROCEDURE `aggiorna_prezzo_componente` (in comp varchar(45), in prezzo
decimal(10,2))
BEGIN
update componente
set prezzovendita = prezzo
where codice = comp;
select row_count();
END
```

```
CREATE PROCEDURE `aggiorna_prezzo_comp_fornitore` (in comp varchar(45),in iva
varchar(45),in prezzo decimal(10,2))
BEGIN
update `componente_has_fornitore`
set `Prezzo d'acquisto` = prezzo
where `componente_codice`= comp and `fornitore_p.iva`=iva;
select row_count();
END
```

```
CREATE PROCEDURE `aggiungi_componente` (in codice varchar(45), in giacenza int, in tipo
ENUM('Schermo', 'ConnettoreUSB', 'Batteria', 'SchedaMadre', 'Tastiera', 'Trackpad', 'Ventola',
'ModuloRAM'), in categoria ENUM('Telefono', 'Portatile'), in descrizione mediumtext, prezzo
decimal(2))
BEGIN
insert into `componente` (`Codice`,`Giacenza`,`tipo`,`categoria`,`descrizione`,`PrezzoVendita`)
values (codice,giacenza,tipo,categoria,descrizione,prezzo);
END
```



```
CREATE PROCEDURE `aggiungi_componente_fornitore` (in iva varchar(45),in comp
varchar(45),in prezzo decimal(10,2))
BEGIN
insert into `componente_has_fornitore` values (comp,iva,prezzo);
END
```

```
CREATE PROCEDURE `aggiungi_componente_ordine` (in cod_ordine int,in var_iva varchar(45),
cod_comp varchar(45), in quantita int)
BEGIN
insert into `Ordine_has_Componente_has_Fornitore` value (cod_ordine,cod_comp,var_iva,quantita);
END
```

```
CREATE PROCEDURE `aggiungi_componenteAmodello` (in modello varchar(45), in componente
varchar(45))
BEGIN
insert into `Modello_has_Componente` (`Modello_Nome`,`Componente_Codice`)
values (modello,componente);
END
```

```
CREATE PROCEDURE `aggiungi_email` (in iva varchar(45), in email varchar(45))
BEGIN
insert into `Email` (`Email`,`Fornitore_P.IVA`) values(email,iva);
END
```

```
CREATE PROCEDURE `aggiungi_fornitore` (in iva varchar(45),in citta varchar(45), in indirizzo
varchar(45), in numero int,in nome varchar(45))
BEGIN
insert into `Fornitore` (`p.iva`,`città`,`via`,`numero`,`nome`) values
(iva,citta,indirizzo,numero,nome);
```

END

```
CREATE PROCEDURE `aggiungi_modello` (in nome varchar(45), in marca varchar(45), in
categoria varchar(45))
BEGIN
insert into `Modello` (`nome`,`marca`,`categoria`) values (nome,marca,categoria);
END
```

```
CREATE PROCEDURE `aggiungi_telefono` (in iva varchar(45), in num varchar(45))
BEGIN
insert into `Telefono` (`Numero`,`Fornitore_P.IVA`) values(num,iva);
END
```

```
CREATE PROCEDURE `aggiungi_utente` (in var_user varchar(45), in var_pass char(32), in
var_role varchar(45))
BEGIN
insert into `utenti` (`Username`,`Password`,`Ruolo`)
values (var_user,md5(var_pass),var_role);
END
```

La seguente procedura viene utilizzata dalla segreteria per calcolare il costo di un intervento e implementa la regola aziendale secondo cui viene calcolato. In uscita fornisce il costo. Il livello di isolamento serializable serve poiché potrebbe essere aggiunto un altro componente nel mentre causando phantom reads

```
CREATE PROCEDURE `calcola_costo` (in intervento int, out o decimal(10,2))
BEGIN
declare p decimal(10,2);
declare exit handler for sqlexception
begin
rollback;
resignal;
end;
```

```
        set transaction isolation level serializable;
    set transaction read write;
    start transaction;
select sum(c1.`PrezzoVendita`) into p
from `intervento aperto_has_componente` as i1
join `componente` as c1
on c1.`codice`=i1.`Componente_Codice`
where i1.`Intervento aperto_Dispositivo` = intervento ;
set o = p*1.1;
update `intervento aperto`
set costo = o
where dispositivo = intervento;
commit;

END
```

Questa procedura calcola il costo di un ordine tramite il costo di una fornitura e la quantità in cui la si sta comprando. Il livello di isolamento è repeatable read perché il prezzo d'acquisto di una fornitura potrebbe cambiare, ma non serve il serializable perché non c'è la possibilità che venga modificato un ordine.

```
CREATE PROCEDURE `costo_ordine` (in cod_ordine int,out costo decimal(10,2))
```

```
BEGIN
```

```
declare exit handler for sqlexception
```

```
begin
```

```
    rollback;
```

```
    resignal;
```

```
end;
```

```
set transaction isolation level repeatable read;
```

```
set transaction read write;
```

```
start transaction;
```

```
select sum(c.`quantità`*f.`Prezzo d'acquisto`) into costo
```

```
from `Ordine_has_Componente_has_Fornitore` as c join `componente_has_fornitore` as f on  
c.`Componente_has_Fornitore_Componente_Codice`=f.`Componente_Codice`  
join `Ordine` as o on f.`Fornitore_P.IVA` = o.`Fornitore_P.IVA`  
where c.`Ordine_Codice` = cod_ordine and o.`Codice`=cod_ordine ;
```

```
update `ordine` set `ordine`.costo = costo where `ordine`.codice = cod_ordine;  
commit;  
END
```

```
CREATE PROCEDURE `elimina_componente` (in comp varchar(45))  
BEGIN  
delete from componente  
where codice= comp;  
select row_count();  
END
```

```
CREATE PROCEDURE `elimina_componente_intervento` (in var_intervento int,in  
var_componente varchar(45))  
BEGIN  
delete from `intervento_aperto_has_componente`  
where `Intervento_aperto_Dispositivo` =var_intervento and `Componente_Codice`=var_componente;  
select row_count();  
  
END
```

```
CREATE PROCEDURE `elimina_fornitore` (in iva varchar(45))  
BEGIN  
delete from fornitore where `P.IVA`=iva;  
select row_count();  
END
```

```
CREATE PROCEDURE `elimina_intervento_aperto` (in cod_intervento int)
BEGIN
delete from `intervento_aperto`
where Dispositivo = cod_intervento;
select row_count();
END
```

```
CREATE PROCEDURE `elimina_modello` (in var_nome varchar(45))
BEGIN
delete from modello
where nome = var_nome;
select row_count();
END
```

```
CREATE PROCEDURE `elimina_ordine` (in cod int)
BEGIN
delete from ordine
where codice=cod;
select row_count();
END
```

```
CREATE PROCEDURE `login` (in var_username varchar(45), in var_pass varchar(45), out var_role
INT)
BEGIN
    declare var_user_role ENUM('Tecnico', 'Magazino', 'Segreteria');

select `ruolo` from `Utenti`
    where `username` = var_username
    and `password` = md5(var_pass)
into var_user_role;
```

```
-- See the corresponding enum in the client
    if var_user_role = 'Tecnico' then
        set var_role = 1;
    elseif var_user_role = 'Segreteria' then
        set var_role = 2;
    elseif var_user_role = 'Magazzino' then
        set var_role = 3;
    else
        set var_role = 4;
    end if;
END
```

Questa procedura modifica la quantità di pezzi di un componente presente in magazzino.

Il livello repeatable read serve perché c'è un trigger che controlla che la quantità non scenda mai sotto lo zero.

```
CREATE PROCEDURE `modifica_quantita` (in var_componente varchar(45), in quant int)
BEGIN
declare exit handler for sqlexception
begin
    rollback;
    resignal;
end;

set transaction isolation level repeatable read;
set transaction read write;
start transaction;
update `componente`
set giacenza = giacenza + quant
where codice = var_componente;
select row_count();
commit;
END
```

```
CREATE PROCEDURE `registra_componenti_intervento` (in intervento int, in componente
varchar(45))
BEGIN
insert into `Intervento aperto_has_Componente` (`Intervento
aperto_Dispositivo`,`Componente_Codice`)
values(intervento,componente);
END
```

Questa procedura per registrare una conclusione copia tutti i dati dell'intervento aperto comprese le componenti utilizzate e aggiunge la data di conclusione. Poi deve eliminare l'intervento aperto, le componenti utilizzate nella tabella 'Intervento aperto_has_componente' verranno eliminate poiché messo in modalità on cascade.

```
CREATE PROCEDURE `registra_conclusione` (in var_disp int, in conclusione date)
BEGIN
declare cod int;
declare dini date;
declare descr mediumtext;
declare cost decimal(10,2);
declare nom varchar(45);
declare cogn varchar(45);
declare model varchar(45);
declare exit handler for sqlexception
begin
rollback;
resignal;
end;

set transaction isolation level serializable;
set transaction read write;
start transaction;
select * into cod,dini,descr,cost,nom,cogn,model
from `intervento aperto`
where Dispositivo=var_disp;
```

```
insert into `Intervento concluso`
(`codice`,`DataInizio`,`Descrizione`,`costo`,`Nome`,`Cognome`,`Modello_Nome`,`dataconclusione`)
values (cod,dini,descr,cost,nom,cogn,model,conclusione);
insert into `Intervento concluso_has_Componente` (`Intervento concluso_codice`,
`Componente_Codice`)
select *
from `Intervento aperto_has_Componente`
where `Intervento aperto_Dispositivo`=var_disp;
delete from `intervento aperto` where Dispositivo=var_disp;
commit;
END
```

```
CREATE PROCEDURE `registra_intervento` (in var_data date, in descrizione mediumtext, in
nome varchar(45),in cognome varchar(45), modello varchar(45), out cod_dispositivo int)
BEGIN
    insert into `Intervento aperto` (`DataInizio`,`Descrizione`,`Nome`,`Cognome`,`Modello_Nome`)
    values (var_data,descrizione,nome,cognome,modello);
    set cod_dispositivo = last_insert_id();
END
```

```
CREATE PROCEDURE `registra_ordine` (in var_fornitore varchar(45), in var_dataora date,out
cod_ordine int)
BEGIN
insert into `ordine` (`Data`,`Fornitore_P.IVA`) values (var_dataora,var_fornitore);
set cod_ordine = last_insert_id();
END
```

Il livello di isolamento è posto repeatable read perché può cambiare la giacenza ed è importante leggere quella aggiornata.

```
CREATE PROCEDURE `visualizza_componente` (in cod varchar(45))
BEGIN
declare exit handler for sqlexception
```



```
begin
    rollback;
    resignal;
end;

set transaction isolation level repeatable read;
set transaction read only;
start transaction;
select *
from componente
where codice = cod;
commit;
END
```

Questa procedura mostra tutti i dati di un fornitore così come i suoi contatti per poi poter fare un ordine.

```
CREATE PROCEDURE `visualizza_fornitore` (in var_iva varchar(45))
```

```
BEGIN
```

```
declare exit handler for sqlexception
```

```
begin
```

```
    rollback;
```

```
    resignal;
```

```
end;
```

```
set transaction isolation level read committed;
```

```
set transaction read only;
```

```
start transaction;
```

```
select *
```

```
from fornitore
```

```
where `P.IVA`=var_iva;
```

```
select numero
```

```
from telefono
```

```
where `Fornitore_P.IVA` = var_iva;
```

```
select email
```

```
from email
where `Fornitore_P.IVA` = var_iva;
commit;
END
```

Il livello di isolamento repeatable read poiché il costo potrebbe essere aggiornato, indicando che l'intervento è già stato preso in carico da un tecnico.

```
CREATE PROCEDURE `visualizza_interventi_aperti` ()
BEGIN
declare exit handler for sqlexception
begin
rollback;
resignal;
end;

set transaction isolation level repeatable read;
set transaction read only;
start transaction;
select *
from `intervento_aperto`;
commit;
END
```

Il livello di isolamento repeatable read poiché il costo potrebbe essere aggiornato, indicando che l'intervento è già stato preso in carico da un tecnico.

```
CREATE PROCEDURE `visualizza_interventi_cliente` (in var_nome varchar(45), in var_cognome
varchar(45))
BEGIN
declare exit handler for sqlexception
begin
rollback;
resignal;
end;
```

```
set transaction isolation level repeatable read;
set transaction read only;
start transaction;
select *
from `Intervento aperto`
where `Intervento aperto`.nome= var_nome and `Intervento aperto`.cognome = var_cognome;
select *
from `intervento concluso`
where nome= var_nome and cognome=var_cognome;
commit;
END
```

Il livello di isolamento serializable perché si potrebbero aggiungere delle componenti al intervallo aperto causando phantom reads.

```
CREATE PROCEDURE `visualizza_intervento_aperto` (in var_dispositivo int)
BEGIN
declare exit handler for sqlexception
begin
rollback;
resignal;
end;

set transaction isolation level serializable;
set transaction read only;
start transaction;
select *
from `Intervento Aperto`
where `Intervento Aperto`.dispositivo = var_dispositivo;
select c.codice,c.tipo
from `Intervento Aperto_Has_Componente` as i join `componente` as c on
i.`Componente_Codice`=c.`codice`
where i.`Intervento aperto_Dispositivo` = var_dispositivo;
commit;
```

END

Il livello è read uncommtd poiché non c'è possibilità che nulla di intervento concluso venga modificato

```
CREATE PROCEDURE `visualizza_intervento_concluso` (in cod_intervento int)
```

```
BEGIN
```

```
declare exit handler for sqlexception
```

```
begin
```

```
rollback;
```

```
resignal;
```

```
end;
```

```
set transaction isolation level read uncommitted;
```

```
set transaction read only;
```

```
start transaction;
```

```
select *
```

```
from `intervento concluso`
```

```
where codice=cod_intervento;
```

```
select c.codice,c.tipo
```

```
from `Intervento concluso_Has_Componente` as i join `componente` as c on
```

```
i.`Componente_Codice`=c.`codice`
```

```
where i.`Intervento concluso_codice` = cod_intervento;
```

```
commit;
```

```
END
```

```
CREATE PROCEDURE `visualizza_modello` (in model varchar(45))
```

```
BEGIN
```

```
declare exit handler for sqlexception
```

```
begin
```

```
rollback;
```

```
resignal;
```

```
end;
```

```
set transaction isolation level read committed;
set transaction read only;
start transaction;
select *
from modello
where nome = model;
select *
from componente as c join `Modello_has_Componente` as c1 on c.codice = c1.Componente_codice
where c1.`Modello_Nome` = model;
commit;
END
```

Il livello di isolamento è read uncommitted perché non c'è possibilità che i dettagli di un ordine vengano modificati

```
CREATE PROCEDURE `visualizza_ordine` (in cod int)
```

```
BEGIN
```

```
declare exit handler for sqlexception
```

```
begin
```

```
rollback;
```

```
resignal;
```

```
end;
```

```
set transaction isolation level read uncommitted;
```

```
set transaction read only;
```

```
start transaction;
```

```
select *
```

```
from ordine
```

```
where codice = cod;
```

```
select Componente_has_Fornitore_Componente_Codice,quantità
```

```
from Ordine_has_Componente_has_Fornitore
```

```
where Ordine_Codice = cod;
```

```
commit;
```

```
END
```

