**UNIVERSITÀ DEGLI STUDI DI BOLOGNA**

FACOLTÀ DI INGEGNERIA

Corso di Laurea in Ingegneria Informatica

Ingegneria del Software L-A – Prof. Giuseppe Bellavia

**Progetto di Ingegneria del Software L-A**

**“ViaggiateSicuri S.R.L.”**

Realizzato da:

*Raffaele Luca Iannario*

*Sara Sabioni*

*Enrico Baioni*

**Anno Accademico 2008 - 2009**

INDICE

DESCRIZIONE DEL PROBLEMA 4

DOCUMENTO DEI REQUISITI 4

ANALISI DEI REQUISITI 5

DOCUMENTO DEI REQUISITI EVIDENZIATO 6

GLOSSARIO 7

CASI D’USO 11

SCENARI 16

MODELLO STATICO 19

MODELLO DINAMICO 21

PROGETTAZIONE 22

# DESCRIZIONE DEL PROBLEMA

## DOCUMENTO DEI REQUISITI

La ViaggiateSicuri S.R.L. è un’azienda che si occupa della vendita al dettaglio di pneumatici, cerchi e vari tipi di accessori per vetture stradali.

Il sistema deve occuparsi della gestione del magazzino per tali prodotti, delle vendite e dell’anagrafica dei clienti. I prodotti sono divisi in categorie; ogni categoria ha un nome ed è raggruppabile in altre categorie.

Ogni prodotto è caratterizzato da un codice, una descrizione, il prezzo d'acquisto, il prezzo di vendita e la giacenza; si può depositare un prodotto in uno o più magazzini. Si prevede, inoltre, la possibilità di gestire l'anagrafica dei prodotti.

Il sistema di autenticazione prevede tre tipi di utenti: l'utente guest, l'operatore e l'amministratore. L'utente guest può solamente controllare lo stato delle giacenze per i vari prodotti. Il login come operatore permette di iniziare una nuova vendita, di effettuare un preventivo e di registrare l’arrivo di nuova merce. L'amministratore può: gestire gli amministratori, gestire gli operatori, gestire i magazzini, gestire le categorie. Inoltre, l'amministratore, deve poter stampare un promemoria d'acquisto per gli ordini da effettuare ai fornitori.

Il sistema deve tenere aggiornata la giacenza di ogni prodotto, registrare l’arrivo di nuova merce e avvisare l'amministratore, al termine di una vendita, quando la giacenza di un prodotto è inferiore ad una certa soglia.

Al momento della vendita si registrano i movimenti dei prodotti, la data, il cliente; ad ogni prodotto è possibile applicare un tasso di sconto; ad ogni vendita è associato l'operatore che l'ha effettuata. Come documento di vendita, i clienti possono scegliere tra la fattura e lo scontrino fiscale. Le modalità di pagamento previste sono i contanti e la carta di credito.

Al cliente viene offerta la possibilità di registrarsi in modo da poter recuperare i suoi dati ad ogni sua visita successiva. Ogni cliente può essere associato a una o più vetture delle quali vengono registrati modello e targa. Ad ognuno di loro viene rilasciata una WheelCard per tener traccia di eventuali bonus spesa. Il sistema deve poter notificare al cliente (ad esempio tramite sms), in una data scadenza stabilita dall'operatore all'atto della vendita, l'invito ad effettuare dei controlli per verificare lo stato del prodotto venduto.

## ANALISI DEI REQUISITI

*ANALISI DI DOMINIO*: il sistema richiesto rientra nel dominio dei sistemi di vendita dei prodotti e di gestione del magazzino.

Il sistema si deve occupare, registrando le vendite, di aggiornare le giacenze di magazzino dei prodotti.

L'utente guest viene utilizzato dall'azienda per dare la possibilità a persone al di fuori del negozio, come ad esempio un ipotetico operaio del magazzino, di poter controllare la giacenza di un determinato prodotto.

L'utente incaricato di effettuare una vendita è l'operatore: compone la vendita inserendo il cliente (se si tratta di un cliente registrato), la data, i prodotti venduti eventualmente applicando uno sconto per ogni prodotto; termina la vendita incassando l'importo totale del pagamento. Il cliente può decidere al momento dell'acquisto se non è ancora registrato, di lasciare i suoi dati al negozio. Al momento della registrazione gli viene rilasciata la Wheelcard che accumula un bonus spesa a seconda di quanti acquisti effettua e attraverso la quale l'operatore, al momento della vendita, recupera i dati del cliente che la possiede. Il cliente può anche scegliere di non registrarsi.

Il documento di vendita che gli può essere rilasciato in questo caso è uno scontrino fiscale; ma oltre a questo tipo può essere anche rilasciata una fattura nel caso il cliente registrato sia un'azienda. Il cliente può anche richiedere all'operatore un preventivo per l'acquisto di un prodotto che verrà salvato all'interno del sistema e avrà una data di scadenza. I clienti che sono registrati e che hanno acquistato prodotti che dovrebbero essere sottoposti a revisioni, possono usufruire di un servizio di promemoria che alla data specificata dall'operatore al momento dell'acquisto, invia un messaggio sul cellulare o una e-mail.

L'amministratore si occupa di tutta la gestione del negozio e degli operatori. Inoltre riceve gli avvisi dei prodotti che devono essere ordinati, nel momento in cui la disponibilità rimanente in magazzino si trova al di sotto di una determinata soglia. In questo modo può stampare la lista completa dei prodotti da ordinare.

## DOCUMENTO DEI REQUISITI EVIDENZIATO

La ViaggiateSicuri S.R.L. è un’azienda che si occupa della vendita al dettaglio di pneumatici, cerchi e vari tipi di accessori per vetture stradali.

Il sistema deve occuparsi della gestione del magazzino per tali prodotti, delle vendite e dell’anagrafica dei clienti. I prodotti sono divisi in categorie; ogni categoria ha un nome ed è raggruppabile in altre categorie.

Ogni prodotto è caratterizzato da un codice, una descrizione, il prezzo d'acquisto, il prezzo di vendita e la giacenza; si può depositare un prodotto in uno o più magazzini. Si prevede, inoltre, la possibilità di gestire l'anagrafica dei prodotti.

Il sistema di autenticazione prevede tre tipi di utenti: l'utente guest, l'operatore e l'amministratore. L'utente guest può solamente controllare lo stato delle giacenze per i vari prodotti. Il login come operatore permette di iniziare una nuova vendita, di effettuare un preventivo e di registrare l’arrivo di nuova merce. L'amministratore può: gestire gli amministratori, gestire gli operatori, gestire i magazzini, gestire le categorie. Inoltre, l'amministratore, deve poter stampare un promemoria d'acquisto per gli ordini da effettuare ai fornitori.

Il sistema deve tenere aggiornata la giacenza di ogni prodotto, registrare l’arrivo di nuova merce e avvisare l'amministratore, al termine di una vendita, quando la giacenza di un prodotto è inferiore ad una certa soglia.

Al momento della vendita si registrano i movimenti dei prodotti, la data, il cliente; ad ogni prodotto è possibile applicare un tasso di sconto; ad ogni vendita è associato l'operatore che l'ha effettuata. Come documento di vendita, i clienti possono scegliere tra la fattura e lo scontrino fiscale. Le modalità di pagamento previste sono i contanti e la carta di credito.

Al cliente viene offerta la possibilità di registrarsi in modo da poter recuperare i suoi dati ad ogni sua visita successiva. Ogni cliente può essere associato a una o più vetture delle quali vengono registrati modello e targa. Ad ognuno di loro viene rilasciata una WheelCard per tener traccia di eventuali bonus spesa. Il sistema deve poter notificare al cliente (ad esempio tramite sms), in una data scadenza stabilita dall'operatore all'atto della vendita, l'invito ad effettuare dei controlli per verificare lo stato del prodotto venduto.

## GLOSSARIO

* VENDITA: è l'atto con il quale il cliente ottiene il prodotto desiderato pagando un corrispettivo in denaro secondo le modalità di pagamento.

E' caratterizzata da:

* uno o più movimenti;
* data: la data in cui viene effettuata la vendita;
* cliente: l'intestatario della vendita.
* UTENTE: si intende la persona che utilizza il sistema. Gli utenti considerati sono tre: il guest, l'operatore e l'amministratore, suddivisi in questo modo a seconda delle funzioni specifiche che ognuno di loro può compiere.
* PRODOTTO: oggetto commercializzato dall'azienda, che può essere depositato presso uno o più magazzini.

E' caratterizzato da:

* codice;
* descrizione;
* giacenza;
* prezzo unitario;
* sconto.

Comprende nel dettaglio:

* Pneumatici;
* Cerchi;
* Accessori.

Sinonimi: merce.

* VETTURA STRADALE: veicolo di proprietà del cliente sul quale vengono installati i prodotti venduti.

E' caratterizzata da:

* Modello;
* Targa.
* MAGAZZINO: struttura in cui vengono conservati i prodotti in giacenza.

E' caratterizzato da:

* Nome;
* Indirizzo.
* CLIENTE: soggetto che acquista uno o più prodotti.

Può essere:

* Cliente Privato;
* Cliente Azienda: cliente in possesso di ragione sociale e partita iva.
* CODICE: identificativo univoco all'interno del sistema per ogni singolo prodotto.
* DESCRIZIONE: nota sintetica che riassume le caratteristiche del prodotto.
* PREZZO UNITARIO: prezzo finale di vendita di un singolo prodotto commercializzato dall'azienda.
* GIACENZA: quantità residua presente in magazzino di un determinato prodotto.
* MOVIMENTO: avviene al momento della vendita e coincide con l'aggiornamento della giacenza in magazzino del prodotto venduto.
* DOCUMENTO DI VENDITA: documento con il quale si attesta la vendita avvenuta.

Può essere di due tipi a seconda della richiesta del singolo cliente:

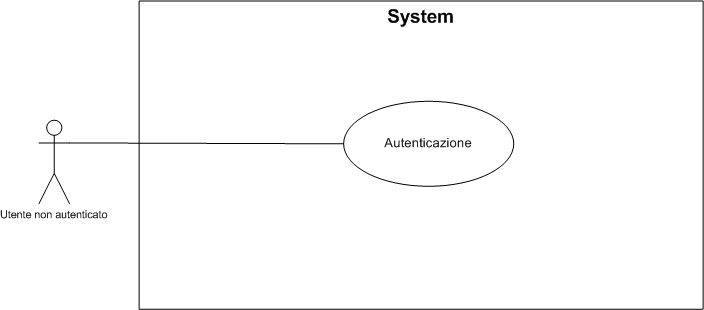
* Fattura: riporta la ragione sociale del cliente, la partita iva, la descrizione del prodotto venduto con il relativo prezzo e l'importo totale dovuto. Questo documento può essere emesso soltanto se il cliente è stato classificato come Azienda.
* Scontrino Fiscale: riporta direttamente il totale dovuto senza indicazione del cliente al quale viene effettuata la vendita.
* PREVENTIVO: è un documento che stima l'ammontare dei costi derivanti da una certa attività richiesta dal cliente, è caratterizzato da una data di scadenza.
* MODALITA' DI PAGAMENTO: modalità scelta dal cliente per pagare l'importo della vendita:

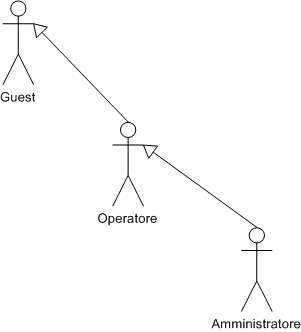
Può essere di due tipi:

* Contanti;
* Carta di credito.
* ACQUISTO: atto duale alla vendita con il quale il cliente entra in possesso del prodotto richiesto.
* ORDINE: definisce la quantità e il tipo di merce da richiedere al fornitore del prodotto stesso.
* DATI: generalità del cliente.
* WHEELCARD: rappresenta la "carta fedeltà" emessa dalla ditta al cliente al momento del primo acquisto nel quale sono contenuti i dati del cliente e il bonus spesa accumulato.
* BONUS SPESA: valore in denaro calcolato in base all'importo degli acquisti effettuati dal cliente.
* REGISTRAZIONE: si riferisce sia all'inserimento della data e della merce all'interno della vendita sia all’acquisizione dei dati del cliente da parte dell'operatore al momento in cui avviene il primo acquisto. La registrazione è a discrezione del cliente stesso che acconsente o meno a lasciare i suoi dati.
* NOTIFICA: si tratta nello specifico di un messaggio telefonico o di una e-mail spedito al cliente che è stato registrato all'interno del sistema, come promemoria di revisione per un prodotto acquistato.
* AVVISO: nel concreto non ha differenza con la notifica, soltanto l'avviso viene spedito all'amministratore nel momento in cui la giacenza di un determinato prodotto si trova al di sotto di una certa quantità specificata.
* RECUPERARE: si intende l'azione specifica con la quale l'operatore può recuperare i dati del cliente attraverso la sua Wheelcard.
* GESTIRE: con la gestione si intende l'insieme delle azioni che possono essere effettuate su un'anagrafica specifica. La gestione comprende l'inserimento, la modifica, la cancellazione e la ricerca di un elemento dell'anagrafica.
* AGGIORNARE: è l'azione con la quale il sistema, a vendita conclusa, oppure l'operatore o l'amministratore aggiorna la giacenza in magazzino dei prodotti venduti.
* STAMPARE: è l'azione con la quale l'amministratore del sistema può riassumere in un unico documento gli ordini da effettuare.
* TERMINE (della vendita): si riferisce all'azione che conclude la vendita stessa, si intende perciò il pagamento concluso.
* LOGIN: o autenticazione; è l'azione con la quale la persona che utilizza il sistema inserisce le proprie credenziali all'interno del sistema per farsi riconoscere e per essere abilitato ad effettuare le operazioni che deve compiere.

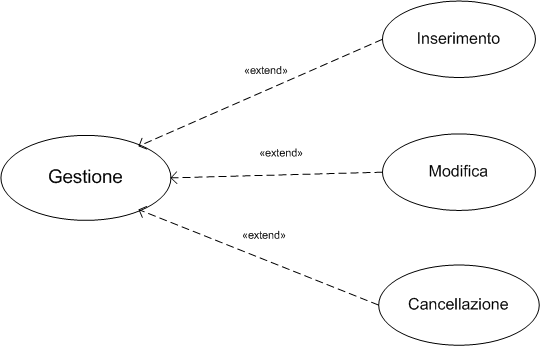
## CASI D’USO

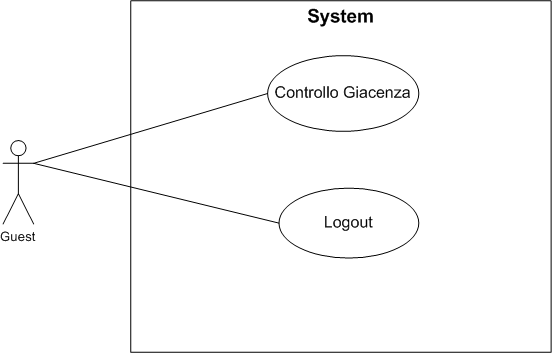
*Autenticazione:* permette all’utente di farsi riconoscere dal sistema attraverso le sue credenziali. Il sistema a sua volta, abilita l’utente ad effettuare operazioni in base al suo ruolo.

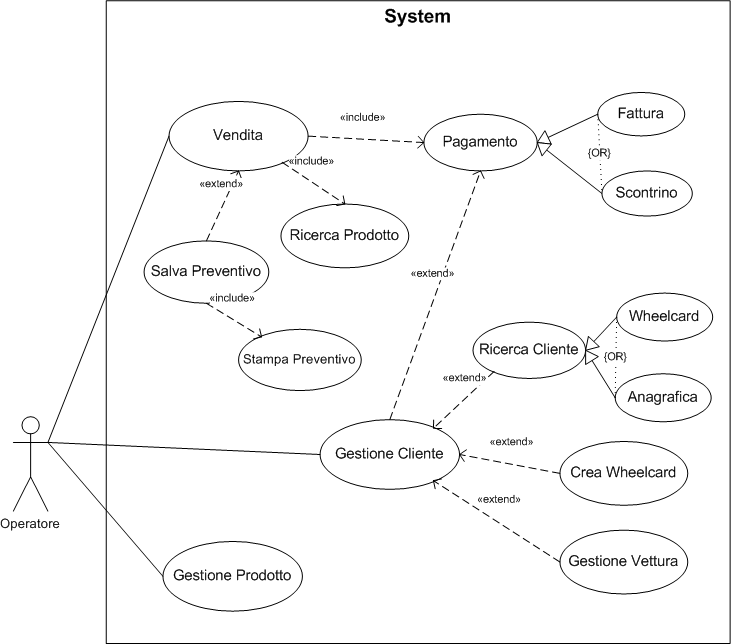


*Gerarchia degli attori:* i tre attori principali del sistema sono posti in gerarchia per fare in modo che *Operatore* erediti i casi d’uso di *Guest* e *Amministratore* erediti i casi d’uso di *Operatore*.

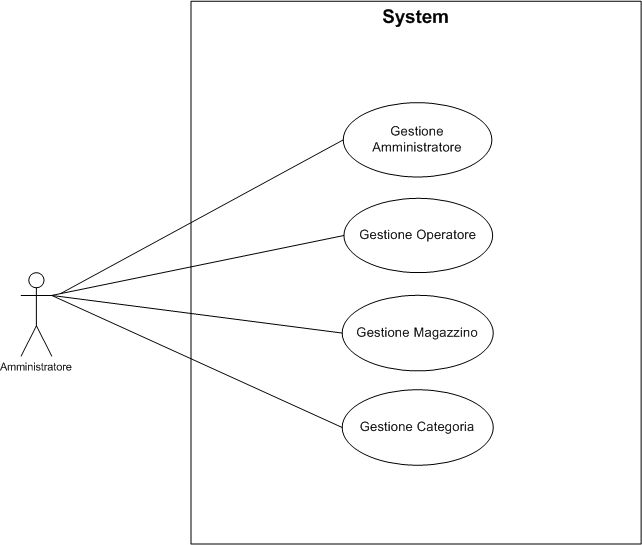
*Gestione*: caso d’uso generico che deve essere considerato ogni volta come *Inserimento*, *Modifica*, *Cancellazione*.

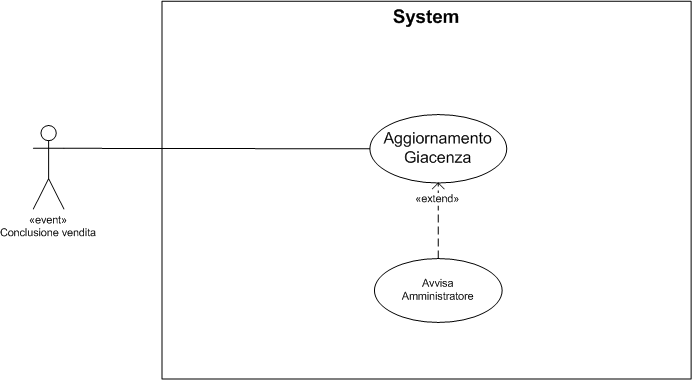


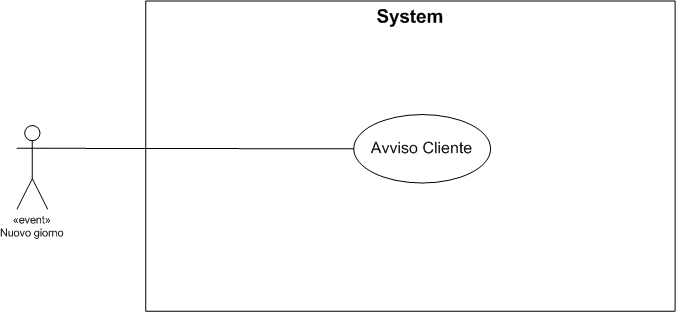
*Guest*: l’attore *Guest* può controllare la giacenza di un determinato prodotto nel magazzino ed effettuare eventualmente il *Logout* al termine della sessione di lavoro.

*Operatore:*  *Gestione prodotto* consente di gestire l’anagrafica dei prodotti. *Gestione cliente,* oltre alle classiche operazioni di inserimento, modifica e cancellazione di un cliente, permette di *Cercare un cliente* già registrato (tramite la *Wheelcard* oppure tramite *Anagrafica), Creare una Wheelcard* da assegnare ad un nuovo cliente e di *Gestire le vetture* appartenenti al cliente. Il caso d’uso *Vendita*, racchiude tutte le operazioni necessarie per iniziare e portare a termine una vendita. *Ricerca prodotto* permette di cercare e aggiungere un prodotto nella vendita corrente; il *Pagamento* permette di concludere la vendita, specificando il tipo di documento di vendita (fattura o scontrino). Durante il pagamento è possibile utilizzare *Gestione cliente*, ad esempio per aggiungere alla vendita i dettagli anagrafici del cliente (già precedentemente registrato) oppure per aggiungere un nuovo cliente, nel caso in cui sia la prima vendita che viene effettuata nei confronti di quel cliente. Dato che il sistema permette di salvare e stampare dei preventivi, è possibile non concludere la vendita con il *Pagamento*, ma di *Salvarla come preventivo* e stamparne una copia per il cliente.

*Amministratore: Gestione amministratore e operatore* permette di gestire, rispettivamente, i dati relativi agli amministratori e agli operatori. *Gestione magazzino* consente di trattare i dati dei magazzini appartenenti al negozio. Questo attore è l’unico che si occupa di decidere il modo di raggruppare i prodotti nelle varie categorie di appartenenza. *Gestione categoria* è il caso d’uso che gli permette di portare a termine questo compito.



*Aggiornamento giacenza:* quando viene conclusa una vendita, l’evento “Conclusione vendita” utilizza questo caso d’uso per Aggiornare la giacenza in magazzino dei prodotti venduti, ed eventualmente Avvisa l’amministratore se la giacenza si trova al di sotto della soglia stabilita da lui stesso.

*Avviso cliente:* questo caso d’uso viene utilizzato dall’evento “Nuovo giorno” attraverso il quale il sistema, ogni volta che viene avviato, verifica quali notifiche deve inviare ai clienti.

## SCENARI

*Scenario – Vendita*

*Descrizione*

* Lo scenario descrive l'operazione di vendita da parte di un operatore

*Attore*

* Operatore

*Pre-condizioni*

* Esiste almeno una categoria contenente almeno un prodotto

*Flusso principale*

1. L'operatore <<Ricerca uno o più prodotti>> da registrare nella vendita
2. L'operatore seleziona la quantità di ogni prodotto
3. Il sistema controlla che la giacenza di ogni prodotto sia sufficiente
4. L'operatore applica eventualmente un tasso di sconto ai prodotti
5. L'operatore sceglie le modalità di notifica al cliente
6. L'operatore conclude la vendita con il <<Pagamento>>

*Flusso alternativo*

3a. La giacenza di un prodotto non è sufficiente -> Il sistema avvisa l'operatore che la disponibilità non è sufficiente

6a. L’operatore <<Salva il preventivo>>

6b. L'operatore <<Stampa il preventivo>>

*Flusso alternativo secondario*6a. L'operatore annulla la vendita

*Scenario – Pagamento*

*Descrizione*

* Lo scenario descrive le operazioni per effettuare il pagamento di una vendita

*Attore*

* Operatore

*Pre-condizioni*

* Esiste almeno un prodotto registrato nella vendita

*Flusso principale*    1. L'operatore sceglie il documento di vendita (fattura o scontrino)

2. L'operatore <<Ricerca il cliente>>

*if ( scelta fattura )*   
 2.1 Il Sistema controlla che il cliente abbia una partita iva  
3. L'operatore conclude l'operazione di pagamento  
4. Il Sistema <<Aggiorna la giacenza>> dei prodotti venduti  
5. Il Sistema aggiorna il bonus spesa della Wheelcard  
6. Il Sistema notifica la corretta conclusione dell'operazione

*Flusso alternativo*

1a. *if ( scelto scontrino )*

1a.1 L'operatore può saltare al punto 3 ma non esegue il punto 5

2a. La ricerca del cliente non produce risultati -> L'operatore <<Inserisce un nuovo cliente>>

5a. Il cliente non ha con sé la Wheelcard -> non viene aggiornato il bonus spesa

6a. Il sistema notifica che si è verificato un problema

*Scenario - Ricerca Cliente*

*Descrizione*

* Lo scenario descrive le operazioni per effettuare la ricerca di un cliente

*Attore*

* Operatore

*Pre-condizioni*

* Esiste almeno un cliente registrato

*Flusso Principale*

1. L'operatore sceglie se cercare tramite Wheelcard o anagrafica

2. If ( Wheelcard )

2.1 Il sistema legge la Wheelcard dall'apposito lettore

If ( Anagrafica )  
 2.1 L'operatore inserisce nome e cognome del cliente  
3. Il sistema restituisce il risultato della ricerca del cliente

*Scenario - Ricerca Prodotto*

*Descrizione*

* Lo scenario descrive le operazioni per effettuare la ricerca di un prodotto da aggiungere alla vendita

*Attore*

* Operatore

*Pre-condizioni*

* Esiste almeno un prodotto
* L'operatore ha iniziato una vendita

*Flusso Principale*

1. L'operatore sceglie una o più categorie  
2. L'operatore inserisce i parametri di ricerca del prodotto  
3. Il Sistema restituisce il risultato della ricerca del prodotto  
4. L'operatore sceglie uno o più prodotti da aggiungere alla vendita  
5. L'operatore aggiunge alla vendita i prodotti scelti

*Flusso Alternativo*

3a. Il risultato della ricerca è vuoto -> l'operatore <<Ricerca Prodotto>>

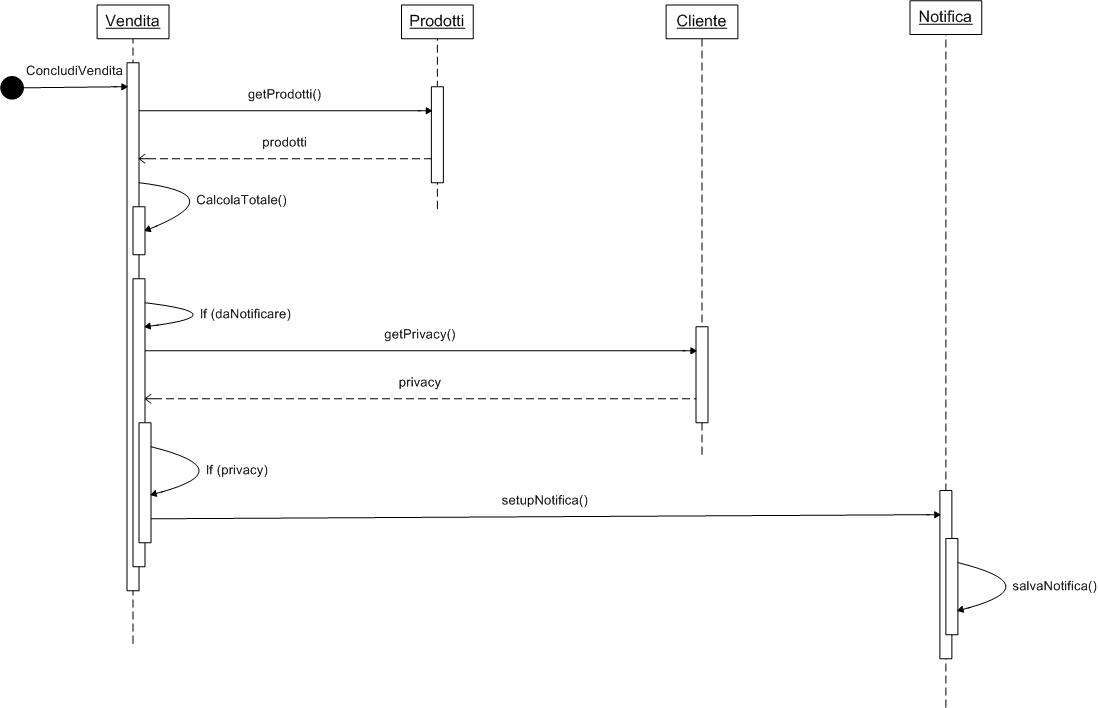
# ModelloDeiDati.pngMODELLO STATICO

Diagramma delle classi

Negozio modella i dati del punto vendita in cui viene utilizzato il sistema. È associato con gli Utenti che hanno accesso al sistema (abbiamo deciso che per ipotesi esistano sempre almeno due utenti predefiniti, che sono il Guest e l’Amministratore). Inoltre è associato tramite composizione alla classe Magazzino che modella il luogo in cui vengono depositati i prodotti acquistati dall’azienda stessa; è previsto il caso in cui il negozio possieda anche più di un magazzino. Il Magazzino a sua volta è associato tramite aggregazione a tutti i prodotti che sono conservati al suo interno; lo stesso articolo può trovarsi in più di un magazzino. I prodotti sono raggruppati in categorie e quindi ogni prodotto è associato ad una Categoria. La classe Categoria possiede una composizione ad anello su sé stessa perché ogni categoria può avere sottocategorie.

La classe Utente è estesa dalle sottoclassi Guest, Operatore e Amministratore che modellano i tre tipi di utenti che hanno accesso al sistema e che hanno la possibilità di svolgere compiti differenti. Come da specifiche l’Operatore è associato alle vendite che effettua e ogni vendita è associata ad un solo operatore. La classe Preventivo estende Vendita in quanto l’operatore ha anche la possibilità di salvarla come preventivo (invece di concluderla con il pagamento) e convertirla successivamente di nuovo in una vendita nel caso il cliente decida di accettarlo. La classe Vendita è inoltre associata con tutti gli elementi che la compongono ossia il Prodotto e il Cliente; la classe Prodotto descrive il dettaglio degli articoli venduti invece la classe Cliente associa alla vendita il cliente registrato che ha effettuato l’acquisto. In questo modo c’è la possibilità di aggiornare il saldo punti della Wheelcard del cliente (giustificando l’associazione con la classe Wheelcard) oltre a semplificare l’inserimento dei dati del cliente nel caso in cui sia un ClienteAzienda e il documento di vendita sia una fattura. La classe Cliente è estesa infatti dalla tipologia del cliente: ClientePrivato o ClienteAzienda. Ad ogni cliente sono associate per aggregazione le vetture di sua proprietà sulle quali vengono svolti i servizi richiesti. Infine la classe Vendita può essere associata ad una o più notifiche da inviare al cliente in una data stabilita dall’operatore durante la conclusione della vendita.

# MODELLO DINAMICO

****Diagramma di sequenza

Il diagramma di sequenza realizzato illustra la sequenza delle azioni che si susseguono al momento della conclusione di una nuova vendita.

Nel momento in cui viene richiamata l’azione rappresentata, l’istanza della classe *Vendita* esiste già; a questo punto vengono richiamati i *Prodotti* aggiunti alla vendita affinché se ne possano reperire i loro prezzi. Una volta ricevuto il valore di ritorno, *Prodotti* non viene più utilizzata e Vendita, attraverso l’azione *CalcolaTotale*, calcola l’importo finale della vendita creata, considerando anche tutti i dati che ha a disposizione al suo interno. Successivamente verifica se ci sono *Notifiche* da effettuare; se questo è vero, allora richiede alla classe *Cliente* il contenuto del campo *Privacy*, e dopo aver verificato che questo sia settato su vero (in quanto il campo rappresenta l’autorizzazione del cliente a ricevere notifiche) imposta la notifica (*setupNotifica*) da inviare. Infine, *salva la notifica* con tutti i dati impostati.

# PROGETTAZIONE

La classe *Negozio* è quella principale del progetto ed è implementata attraverso il pattern *singleton* per impedire istanziamenti multipli e rendere alcuni riferimenti sempre disponibili alle varie classi dell'applicazione. Si è deciso di scegliere questa classe in quanto la nostra applicazione si basa su installazioni locali e si occupa della gestione di un solo negozio.

*Negozio* contiene, oltre al classifico riferimento alla propria istanza, anche riferimenti a:

-*Magazzini*, la classe che detiene una collezione di oggetti di tipo Magazzino, al fine di rendere disponibili tale collezione a tutta l'applicazione

-Utente corrente, ovvero l'utente che sta usufruendo dell'applicazione in questo momento (non sono previsti accessi distinti contemporanei) ottenuto attraverso *UtenteFactory*.

-*Clienti*, la classe incaricata di contenere tutti gli oggetti di tipo Cliente, sempre per poter rendere visibili questi oggetti a tutta l'applicazione.

*UtenteFactory*

Questa classe, come si deduce dal nome, è la factory degli utenti la quale, all'atto del login, ha la responsabilità di restituire l'istanza della classe corretta tra quelle presenti nella gerarchia degli utenti (*Guest*, *Operatore*, *Amministratore*) attraverso il proprio metodo statico GetUtente().

*Utente*

La classe *Utente* è una classe astratta e rende la generalizzazione dei vari utenti del sistema contenendo al proprio interno gli attributi in comune alle varie sottoclassi (al momento tutti). Si è deciso di non far collassare verso l'alto la gerarchia in un singolo attributo "Ruolo", per evitare di dover effettuare controlli e quindi cascate di IF o istruzioni di SWITCH. La verifica del permesso di eseguire una operazione è affidata a un sistema di double dispatching che decide sia in base all'operazione da eseguire e sia in base alla classe dell'utente corrente.

*Guest, Operatore, Amministratore*

Sono semplicemente classi concrete della classe astratta Utente.

*Prodotti*

Classe contenitore. E' dotata anche di un metodo per la ricerca di un prodotto.

*Prodotto*

la classe modella il termine Prodotto del glossario. Ogni prodotto ha un proprio identificativo. E' possibile impostare la categoria di appartenenza del prodotto. E' stata dotata dei metodi di controllo giacenza, il quale verifica che la giacenza non sia sotto una certa soglia e di un metodo "AggiornaGiacenza" che viene invocato alla conclusione di ogni vendita per ogni prodotto presente nella vendita stessa.

*Clienti*

Contiene l'insieme dei clienti registrati all'interno del sistema.

*Cliente*

È la modellazione del cliente generico. E' una classe astratta e contiene le proprietà in comune a qualsiasi cliente. Da notare la proprietà "privacy", un flag di controllo per impedire l'invio di notifiche a clienti che non ne hanno dato esplicito consenso. Contiene un riferimento alla classe Vetture, in quanto ogni cliente può avere o richiedere interventi su più macchine e un riferimento alla classe *WheelCard*, in quanto, essendo un cliente registrato può essere identificato anche tramite WheelCard.

*ClientePrivato*

è un cliente privato, ovvero non possiede una partita iva.

*ClienteAzienda*

è un cliente in possesso di una ragione sociale e di una partita iva.

La gerarchia è stata realizzata per impedire di stampare una fattura ad un cliente che non ha una partita iva.

*Notifiche*

Rappresenta l'insieme delle notifiche che devono essere inviate.

*Notifica*

Modella le comunicazioni da inviare e come tutte i tipi di reminder è caratterizzata da una data di invio, un destinatario e da un enumerativo che ne indica il canale di invio. Il metodo AccadeOggi() verifica se la data corrente è la stessa della data di invio precedentemente impostata. In caso positivo avviene l'invio della notifica stessa e viene settato opportunamente il flag daNotificare per evitare che la stessa notifica sia inviata più volte.

*Vendite*

La classe rappresenta un archivio di tutte le vendite effettuate.

*Vendita*

Si è deciso di non distinguere vendita da vendita conclusa in quanto la seconda sul piano concreto non possiede nulla di più della prima (la distinzione infatti si basa sul pagamento: a pagamento effettuato la vendita viene considerata conclusa (e aggiunta alla collezione Vendite) ma il pagamento è un'azione estranea all'applicazione). La classe inoltre possiede una relazione d'uso con la classe *Clienti* perché al momento della vendita devono essere eventualmente ricercati e inseriti i dati del cliente registrato; eventualmente perché è possibile vendere un prodotto anche senza aver identificato il cliente. Possiede una relazione di uso con la classe Notifiche, che permette al momento della vendita di registrare le notifiche necessarie; un collegamento con la classe Prodotti per poter ricercare e inserire i prodotti destinati alla vendita.

Ha i metodi "SalvaComePreventivo()" che permette di trasformare la vendita in un preventivo con scadenza, il metodo "CalcolaTotale()" e il metodo "ConcludiVendita()" che marchia la vendita come effettuata e aggiorna la giacenza dei prodotti venduti.

*Preventivo*

caratterizzata dalla proprietà validità che ne indica il limite temporale di validità.

Ogni istanza della classe Preventivo, può tornare a essere una vendita a tutti gli effetti tramite il metodo EvolviInVendita().

*Vetture*

La classe contiene tutte le vetture registrate nel sistema associate al cliente che le possiede.

*Vettura*

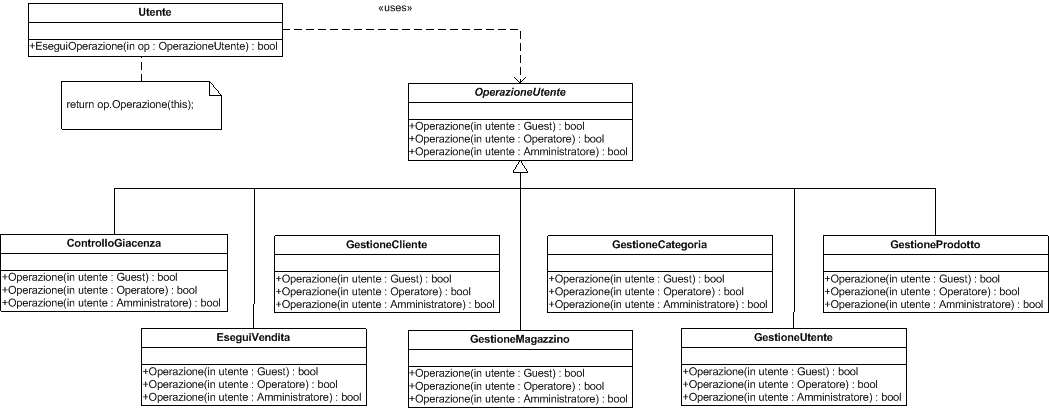
Semplice modellazione di una vettura generica, identificata da targa e modello.

*Cliente*

La classe è collegata alla classe *Wheelcard* per permettere sia la ricerca dei dati tramite Wheelcard sia l'aggiornamento del bonus accumulato dal singolo cliente nel momento in cui è soggetto di una nuova vendita. Inoltre è collegata alla classe Vetture per permettere l'associazione tra il cliente e l'elenco delle vetture che gli appartengono.

*Double dispatcher*

Per evitare verifiche continue sul tipo di utente e sui suoi permessi di eseguire l'operazione richiesta si è deciso di realizzare questo sistema:



La classe Utente ha un metodo EseguiOperazione(), già definito, che accetta come parametri oggetti che derivano dalla classe astratta OperazioniUtente caratterizzata dall'overloading del metodo Operazione() che accetta o un oggetto di tipo Guest o di tipo Operatore o di tipo Amministratore. Si è optato per una classe astratta per una politica di default denied, in modo da dover ridefinire solo i metodi per gli utenti che possono effettivamente svolgere l'operazione. La classe è resa astratta dal metodo che accetta come parametro un Amministratore.

In questa parte di progettazione, come richiesto dalle specifiche, non sono stati affrontati problemi (e relative soluzioni) che riguardano la persistenza dei dati.