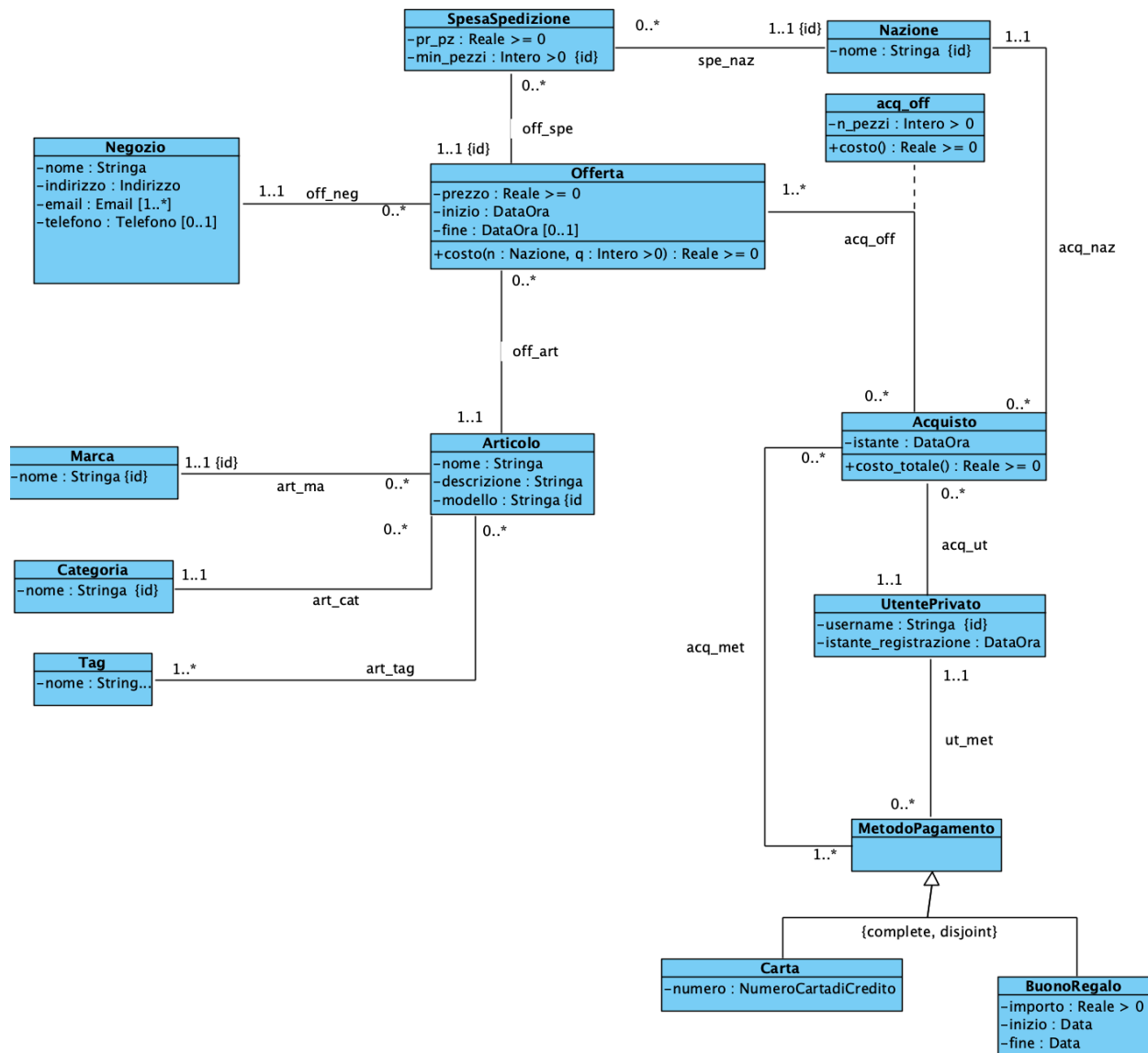


Progetto AmaZZon

Analisi Concettuale

Diagramma UML concettuale delle classi



Specifica delle classi

Specifica della classe Offerta

[V.Offerta.inizio.prima_di_fine]

ALL o, i, f inizio(o, i) and fine(o, f) => i <= f

Specifica della classe [Acquisto](#)

Non può esistere un acquisto di un'offerta verso una certa nazione se l'offerta non definisce una spesa di spedizione verso tale nazione.

[V.Acquisto.Nazione.esiste_spedizione]

ALL a, n, o, p

Acquisto(a) and acq_naz(a, n) and acq_off(a, o) and n_pezzi(a, o, p) =>

EXISTS ss, n_min

SpesaSpedizione(ss) and off_spe(o, ss) and spe_naz(ss, n) and

min_pezzi(ss, n_min) and n_min <= p

[V.Acquisto.Offerta.non_scaduta]

ALL a, o, i, in

acq_off(a, o) and istante(a, i) and inizio(o, in)

=> i >= in and (ALL f fine(o, f) => f >= i)

[V.Acquisto.MetodoPagamento.di_utente_acquirente]

ALL a, u, m

Acquisto(a) and acq_ut(a, u) and acq_met(a, m)

=> ut_met(u, m)

Operazioni di classe

costo_totale(): Reale >= 0

Pre: nessuna

Post: L'operazione non modifica il livello estensionale.

Viene restituito *result*, così definita

P = {(off, pr_off) | acq_off(this, off)}

and pr_off = (this, off).costo()

and costo(this, off, pr_off)

// formalmente più corretto

result = SUM p

(o, p) ∈ P

Specifica della classe [acq_off](#)

costo(): Reale ≥ 0

Pre: nessuna

Post:

L'operazione non modifica il livello estensionale.

Viene restituito *result*, così definita

sia this = (a: Acquisto, o: Offerta)

Sia n tale che Nazione(n) and acq_naz(a, n)

Sia p tale che n_pezzi(a, o, p)

Sia $S = \{(ss, n_min) \mid$

SpesaSpedizione(ss) and off_spe(o, ss) and spe_naz(ss, n) and

And min_pezzi(ss, n_min) and n_min \leq p}

(S ha un solo elemento per via del vincolo di identificazione nel diagramma

Sia (ss^*, n^*) l'unico elemento di argmax_n

$(ss, n) \in S$

Sia (ss^*, n^*) l'unico elemento di

$\{ (ss, n) \mid (ss, n) \in S \text{ and}$

NOT EXISTS ss', n'

$(ss', n') \in S \text{ and}$

$(ss', n') \neq (ss, n)$

and $n' > n \}$

Sia pr_sped tale che pr_pz(ss^* , pr)

Sia pr_off tale che prezzo(o, pr_off)

Result = $p * (pr_sped + pr_off)$

n_vendite_periodo(a: Articolo, inizio: Data, fine: Data): Intero >= 0

n_vendite_mese_corrente(a: Articolo): Intero >= 0

Sia m tale che mese(adesso, m)

Sia imc tale che mese(imc, m) and giorno(imc, 1)

Result = n_vendite_periodo(a, imc, oggi)

n_vendite_mese_scorso(a: Articolo): Intero >= 0

Sia m tale che mese(adesso, m)

Sia imp tale che mese(imp, m-1) and giorno(imp, 1)

Sia g il numero di giorni nel mese m

Sia fmp tale che mese(fmp, m-1) and giorno(fmp, g)

Result = n_vendite_periodo(a, imp, fmp)

Specifiche dello Use-Case Statistiche

paesi_più_acquisti(inizio: Data, fine: Data): Nazione [1..]*

*dato un periodo di tempo, il paese dal quale sono
provenuti più acquisti nel periodo dato*

Pre: EXISTS acq, i Acquisto(acq) and istante(acq, i) and inizio <= i <= fine

Post: non vengono modificati i dati

Viene restituito result così definita:

$$A = \{(naz, num_acq) \mid$$
$$\text{Nazione}(naz) \text{ and}$$
$$num_acq = \mid \{ acq \mid$$
$$\text{Acquisto}(acq) \text{ and } acq_naz(acq, naz) \text{ and}$$
$$\text{EXISTS } i \text{ and } istante(acq, i) \text{ and } inizio \leq i \leq fine\} \mid$$
$$\}$$
$$\text{Result} = \{ naz \mid \text{EXISTS } n_a$$
$$(naz, n_a) \in \text{argmax } n_a'$$
$$(naz', n_a') \in A\}$$

O, equivalentemente,

$$\text{Result} = \{naz \mid \text{EXISTS } n_a$$
$$(naz, n_a) \in A$$
$$\text{and NOT EXISTS } naz', n_a' (naz', n_a') \in A$$
$$(naz, n_a) \neq (naz', n_a')$$
$$\text{and } n_a' > n_a \}$$

Specifica dello use-case Vetrina

oggetti_filtro(c: Categoria, T: Tag [1..*]): Offerta [0..*]

Pre: nessuna

Post:

Result = { a, p | Exists i, f, a
 Offerta(o) and inizio(o, i) and adesso >= i
 and (ALL f fine(f, o) => f > adesso)
 and off_art(o, a) and art_cat(a, c)
 (ALL t t ∈ T => rt_tag(a, t))
}

Schema Relazionale

```
create table negozio (  
    id serial primary key  
    ...  
)  
  
create table marca (  
    nome varchar primary key  
)  
  
create table nazione (  
    nome varchar primary key  
)  
create table articolo (  
    modello varchar not null,  
    marca varchar not null,  
    primary key (modello, marca),  
    foreign key (marca) references marca(nome)  
)  
create table offerta (  
    id serial primary key,  
    inizio timestamp not null,  
    fine timestamp,  
    check (fine is null or fine > inizio),  
    negozio integer not null,  
    foreign key (negozio) references negozio(id),  
    modello varchar not null,  
    marca varchar not null,  
    foreign key (modello, marca)  
        references articolo(modello, marca)  
)  
  
create domain RealGEZ as real  
    check (value >= 0);  
  
create domain IntGZ as real  
    check (value > 0);  
  
create table spesaspedizione(  
    offerta integer not null,
```

```

foreign key (offerta)
    references offerta(id),
nazione varchar not null,
foreign key (nazione)
    references nazione(nome),
pr_pz RealGEZ not null,
min_pezzi IntGZ not null,
primary key (offerta, nazione, min_pezzi)

);

create table utenteprivato(
    username varchar primary key,
    istante_reg timestamp not null

);

create table acquisto(
    id serial primary key,
    utente varchar not null,
    foreign key (utente)
        references utenteprivato(username),

    v. incl. (id) occorre in acq_off(acquisto)

)

create table acq_off (
    offerta integer not null,
    acquisto integer not null,
    n_pezzi IntGZ not null,
    primary key (offerta, acquisto),
    foreign key (offerta)
        references offerta(id),
    foreign key (acquisto)
        references acquisto(id),

);

```


costo(acq_id: integer, off_id: integer): RealGEZ

Viene restituito l'unico valore nella colonna *costo* della tabella risultante dalla seguente query SQL

```
WITH SS as (  
    Select ss.pr_pz as prezzo_sped_per_pezzo  
    From spesespedizione ss, acquisto a, offerta o, acq_off, ao  
    where a.id = :acq_id  
        and o.id = :off_id  
        and ss.offerta = o.id  
        and ss.nazione = a.nazione  
        and ao.offerta = o.id and ao.acquisto = a.id  
        and ss.min_pezzi <= ao.n_pezzi  
        and ss.min_pezzi >= ALL(Select ss.min_pezzi  
                                from spesespedizione ss, acquisto a, offerta o,  
                                acq_off, ao  
                                where a.id = :acq_id  
                                    and o.id = :off_id  
                                    and ss.offerta = o.id  
                                    and ss.nazione = a.nazione  
                                    and ao.offerta = o.id  
                                    and ao.acquisto = a.id  
                                    and ss.min_pezzi <= ao.n_pezzi)  
    ),  
Select acq_off.n_pezzi *( o.prezzo + ss.prezzo_sped_per_pezzo) as  
costo  
From acquisto a, offerta o, acq_off ao, SS  
where a.id = :acq_id  
    and o.id = :off_id  
    and ao.offerta = o.id  
    and ao.acquisto = a.id;
```

costo_totale(acq_id: integer): RealGEZ

Viene restituito il valore dell'attributo *costo_totale*
dell'unica ennupla risultante dalla seguente query SQL

```
Select sum(costo(acq_id, ao.offerta)) as costo_totale
From acq_off ao
Where ao.acquisto = :acq_id;
```