

RISTORBOOK

CLIENTE

- NUM
- E-MAIL

PROMOZIONE

- SCONTO
- MAX_PERSONE
- D_INIZIO
- D_FINE

RISTORANTE

- NOME
- P_IVA
- INDIRIZZO
- CITTÀ
- TIPO_CUCINA

CHIUSURA

- DA
- A
- ATTIVA

ACCETTATA

- 1ST_TIMESTAMP
- ESITO: { USATA, NON_USATA, IN_ATTESA }

RIFIUTATA

- 1ST_TIMESTAMP

NOGATA

- 1ST_TIMESTAMP

ANNULLATA

- 1ST_TIMESTAMP

CLIENTE

- NOME: STR
- EMAIL: STR

1..1

0..*

Dist

PROPRIETARIO

- CF: STR

1..1

1..*

PRENOTAZIONE

- GIORNO: DATE
- ORA: TIME
- N_PERSONE: INT > 0

0..*

1..1

RISTORANTE

- NOME: STR
- P_IVA: STR
- INDIRIZZO: RECORD

0..*

1..1

CITTÀ

- NOME: STR

1..*

1..*

TIPO_CUCINA

- NOME: STR

PROMOZIONE

- SCONTO: [1..100]
- GIORNI: {L, MA, ...}
- DATA_IN: DATE
- DATA_FIN: DATE
- MAX_PERSONE: INT > 0
- ORA_IN: TIME
- ORA_FIN: TIME

0..1

CHIUSURA

- DATA_IN: DATE
- DATA_FIN: DATE
- ORA_IN: TIME
- ORA_FIN: TIME
- ATTIVA: BOOL

BASE -> PENDENTE

↑ < D ↓

RIF.

ACC.

USAPROD

↑ < D ↓

USATA

NO

ANNULLATA

VINCOLI

- TEMPORALI

FINE > INIZIO

$\forall x, IN, FIN \text{ CHIUSURA}(x) \wedge DATA_INIZIO(x, IN) \wedge DATA_FINE(x, FIN) \rightarrow IN < FIN$

$\text{ORA_IN} \quad \text{ORA_FIN}$

PROMOZIONE(x)

$\text{ORA_IN} \quad \text{ORA_FIN}$

TEMPO_PREN_PREN

$\forall p, pr, d, IN, FIN, o, oIN, oFIN$

$PRENOTAZIONE(p) \wedge USA_PROM(p, pr) \wedge GIORNO(p, d) \wedge ORA(p, o)$

$\wedge DATA_IN(pr, IN) \wedge DATA_FIN(pr, FIN) \wedge ORA_IN(pr, oIN) \wedge ORA_FIN(pr, oFIN)$

$\rightarrow d > IN \wedge d < FIN \wedge o > oIN \wedge o < oFIN$

IST_SOTTOPREN < PREN_DATA_ORA

$\forall x, d, o, IST \text{ PRENOTAZIONE}(x) \wedge GIORNO(x, d) \wedge ORA(x, o) \wedge RIFIUTATA(x) \wedge IST(x, IST)$

$\rightarrow IST_DATA < d \vee (IST_DATA = d \wedge IST_ORA < o)$

UGUALE PER ALTRE SOTTOCLASSI

GIORNO_COMPRESO_PREN_PREN

$\forall p, pr, G, G' \text{ PRENOTAZIONE}(p) \wedge USA_PROM(p, pr) \wedge GIORNO(p, G) \rightarrow GIORNO(G) \leq G'$
 $\wedge GIORNO(pr, G')$

ACC_USATA_O_NO

ORA(x, d)

$\forall x, d, o \text{ PRENOTAZIONE}(x) \wedge ACCETTATA(x) \wedge GIORNO(x, d) \wedge (NOW_GIORNO > GIORNO \vee (NOW_ORA > o)) \rightarrow ESITO(x, USATA) \vee ESITO(x, NON_USATA)$

DISS_PREN_RIST

$\forall c, p, p', r, d, d', o, o' \text{ CLIENTE}(c) \wedge PRENOTAZIONE(p, p') \wedge PR(p, r) \wedge DATA(p, d) \wedge ORA(p, o)$
 $\wedge PR(p', r) \wedge DATA(p', d') \wedge ORA(p', o')$
 $\wedge p \neq p' \rightarrow d \neq d' \vee (d = d' \wedge o \neq o')$

~~NUM_PERSONE (p: PROMOZIONE, d: DATA) : INT ≥ 0~~

~~PRE:~~

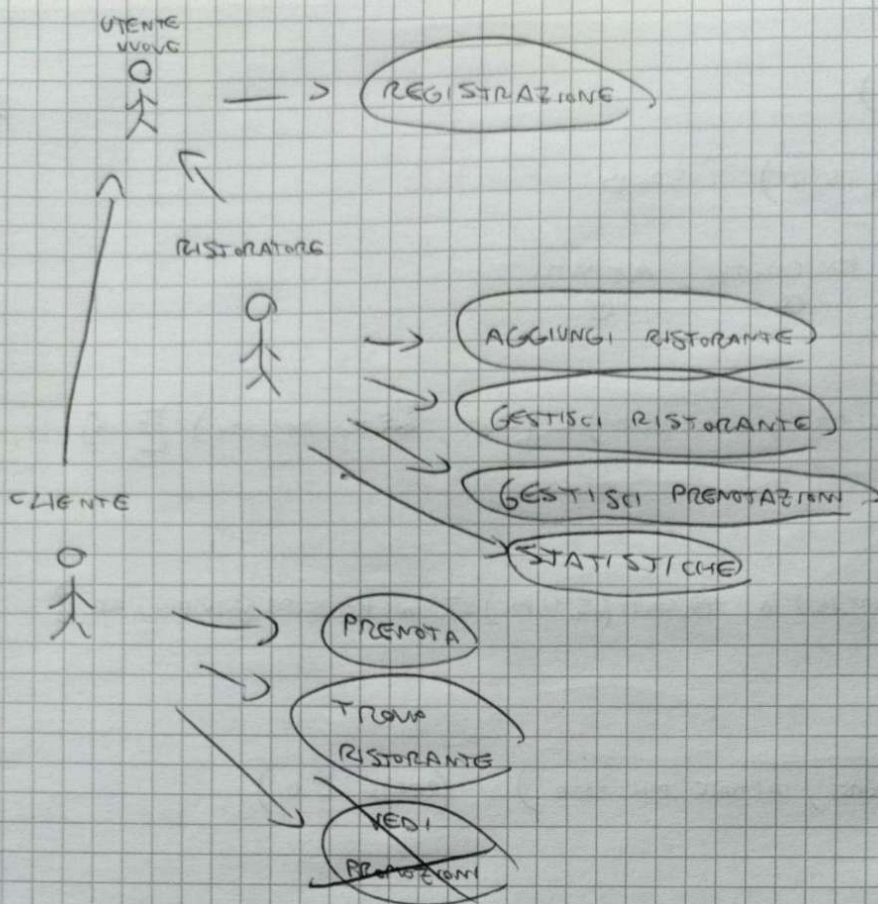
~~POST:~~

~~$PERS \vdash \{ (N) \mid USA_PROM(pr, p) \wedge N_PER(pr, N) \}$~~

~~RETURN $\sum_{(N) \in PERS} N$~~

$\wedge MAX(p, N)$

$\forall p, pr, N \text{ PRENOTAZIONE}(p) \wedge USA_PROM(p, pr) \wedge PER(p, N) \wedge MAX(p, N) \rightarrow NUM_PERSONE(p) \leq N$



REGISTRA_CLIENTE (N: ~~REAL~~ STRING, C: STRING, MAIL: E-MAIL) : CLIENTE

~~~~~ RISTORATORE (N: STRING, C: STRING, CF: CF) : RISTORATORE

~~CREA\_RISTORANTE (N: STRING, P: P.IVA, IND: INDIRIZZO)~~

CREA\_CHIUSURA (DIN: DATE, DEIN: DATE, ON: TIME, OFIN: TIME) : CHIUSURA

ANNULLA\_CHIUSURA (C: CHIUSURA)

ACCETTA\_PRENOT (P: PRENOTAZIONE)

RIFIUTA\_PRENOT (~~~~~)

USATA\_PRENOT (~~~~~)

~~NON\_~~~~~ (~~~~~)~~

MEDIA\_PRENOT\_GIORNO (R: RISTORANTE, IN: DATE, FIN: DATE) T (P: PROMOZIONE, REAL ≥ 0) [0..\*]

PRENOTA (R: RISTORANTE, G: DATE, O: TEMPO, N: INT > 0) : PRENOTAZIONE

ANNULLA (P: PRENOTAZIONE)

TROVA (C: CITTA, O: DATE, S: [1..100], T: TIPO\_CUCINA [1..\*], N.PER: INT > 0) : (R: RIST~) [0..\*]



MEDIA-PRODOTTO-GIORNO (R: RISTORANTE, INIZ: DATA, FINE: DATA)

PRE: IN < FIN

POST:

PROM =  $\sum (P) \mid$  PROMOTING(P)  $\wedge$  RP(R,P)  $\wedge$  DATA-IN(P,IN)  $\wedge$  DATA-FIN(P,FIN)  
 $\wedge$  INIZ < IN  $\wedge$  FINE > FIN  $\}$

RETURN  $\sum (P,N) \mid P \in \text{PROM} \wedge N \leq \text{MEDIA-GIORNO}(P) \}$

MEDIA-GIORNO (P: PROMOZIONE); REAL  $\geq 0$

PRE: —

POST:

GIORNI = (P.DATA-FIN - P.DATA-IN)

~~tot =  $\sum (N) \mid$  USA-PROM(PRE,P)  $\wedge$  N.PER(PRE,N)  $\}$~~

~~tot =  $\sum (N) \mid$  USA-PROM(PRE,P)  $\wedge$  N.PER(PRE,N)  $\}$~~

RETURN  $\frac{\sum_{\text{NETOT}} N}{\text{GIORNI}}$

trova

PRE: —

POST:

RES =  $\sum (R) \mid$  RISTORANTE(R)  $\wedge$  SODC(R,C)  $\wedge$  R.PROM(R,P)  $\wedge$  SCONT(P,SC)  $\wedge$   
 $S \geq SC \wedge (NUM-PERSONE(P,D) \leq$

RES =  $\sum (R) \mid$  RIST(R)  $\wedge$  CITTÀ(R,C)  $\wedge$  TC(R,T)  $\wedge$  T  $\in$  ~~ALL-TIPI(R)~~  $\wedge$  ( $\exists P$  ~~USA-PROM(R,P)~~  $\wedge$  ( $NUM-PERSONE(P) + N$ )  $\leq N$ )  $\wedge$  MAX(P,N)  $\wedge$  DATA-FIN(P,FIN)  $\wedge$  FIN > 0  $\}$

ALL-TIPI (R: RISTORANTE); (T: TIPO-CUCINA) [4..\*]

PRE:

POST:

RETURN  $\sum (T) \mid \text{TIPO}(R,T) \}$







## PRENOTAZIONE

| ID  | UTENTE | RISTORANTE | ISTANTE   | ESITO | DATA | ORA  | <del>PROMOZIONE*</del> | N_PER |
|-----|--------|------------|-----------|-------|------|------|------------------------|-------|
| SRL | INT    | INT        | TIMESTAMP | EST   | DATE | TIME | <del>INT</del>         | INT64 |

CHECK : (ESITO = 'OK' V ESITO = 'REF' ~) AND ISTANCE IS NOT NULL

(ESITO = 'IN ATTESA' AND ISTANCE IS NULL)

FK : UTENTE REF UTENTE(ID)

RISTORANTE REF RISTORANTE(ID)

~~PROMOZIONE~~ ~ PROMOZIONE(ID)

USA - PROM

PRENOTAZIONE | PROMOZIONE

INT | INT

UNIQUE (PRENOTAZIONE)

FK : PRENOT REF PRENOT (ID)

PROM ~ PROM (ID)

## PROMOZIONE

| ID  | RISTORANTE | D_IN | D_FIN | O_IN | O_FIN | MAX   |
|-----|------------|------|-------|------|-------|-------|
| SRL | INT        | DATE | DATE  | TIME | TIME  | INT64 |

CHECK (D\_IN < D\_FIN)

(O\_IN < O\_FIN)

FK : RISTORANTE REF RISTORANTE (ID)

## RISTORANTE

| ID  | NOME   | P_IVA  | INDIRIZZO | PROPRIETARIO | CITTA' |
|-----|--------|--------|-----------|--------------|--------|
| SRL | STRING | STRING | INDIR     | INT          | STRING |

### TIPO\_CUCINA

NOME  
STR

FA - CUCINA

RISTORANTE

INT

TIPS

STR

FK : RISTORANTE REF RISTORANTE (ID)

TIPO REF TIPO\_CUCINA (NOME)

## CHIUSURA

| ID  | RISTORANTE | D_IN | D_FIN | O_IN | O_FIN | ATTIVA |
|-----|------------|------|-------|------|-------|--------|
| SRL | INT        | DATE | DATE  | TIME | TIME  | Bool   |

CHECK (D\_IN < D\_FIN)

(O\_IN < O\_FIN)

FK : RISTORANTE REF RISTORANTE (ID)



T\_TEMPO\_PROM - PROM - PROM

INSERT / UPDATE ON USA\_PROM

PROM := (SELECT \* FROM PROMOZIONE WHERE NEW.PROMOZIONE = PROMOZIONE.ID)

~~PROMOT := (SELECT \* FROM PRENOTAZIONE WHERE NEW.PROMOTAZIONE = PRENOTAZIONE.ID)~~

PROMOT := (SELECT \* FROM PRENOTAZIONE WHERE NEW.PROMOTAZIONE = PRENOTAZIONE.ID)

~~OK~~ := EXISTS (SELECT \* FROM PROM P, PROMOT PR WHERE PR.DATA ≥ P.D\_IN AND PR.DATA ≤ P.D\_FIN AND PR.ORA ≥ P.ORA\_IN AND PR.ORA ≤ P.ORA\_FIN)

IF OK

COMMIT

T\_MAX\_PERSONE\_PROMOZIONE

INSERT / UPDATE ON USA\_PROM

~~MAX := (SELECT MAX FROM PROMOZIONE WHERE NEW.PROMOZIONE = PROMOZIONE.ID)~~

MAX := (SELECT MAX FROM PROMOZIONE WHERE NEW.PROMOZIONE = PROMOZIONE.ID)

NUM\_PER := (SELECT N\_PER N FROM PRENOTAZIONE WHERE NEW.PRENOTAZIONE = PRENOTAZIONE.ID)

TOTALI := ~~NUM\_PERSONE~~ NUM\_PERSONE (NEW.PROMOZIONE, DATA, DATA)

DATA := (SELECT DATA FROM PRENOTAZIONE WHERE PRENOTAZIONE.ID = NEW.PRENOTAZIONE)

IF TOTALI + NUM\_PER.N ≤ MAX.MAX

COMMIT

T\_NO\_PROMOT\_CHIUSURA

INSERT / UPDATE ON PRENOTAZIONE

~~CHiusura~~ <sup>EXISTS</sup> (SELECT \* FROM CHIUSURA WHERE NEW.RISTORANTE = ~~CHIUSURA~~ CHIUSURA.RISTORANTE  
AND CHIUSURA.D\_IN ≤ NEW.DATA AND CHIUSURA.D\_FIN ≥ NEW.DATA AND  
CHIUSURA.ORA\_IN ≤ NEW.ORA)

IF ERROR

RAISE

ELSE

COMMIT



T-DISS\_PRENOT

INSERT / UPDATE ON PRENOTAZIONE

GRUPPO & EXISTS ( SELECT \* FROM PRENOTAZIONE WHERE P.DATA = NEW.DATA AND P.RA = NEW.RA  
AND P.RISTORANTE = NEW.RISTORANTE )

IF ERROR

ERROR

~~GRUPPO~~ T-NEW-RISTORANTE-PROG

INSERT / UPDATE ON RISTORANTE

OK & EXISTS ( SELECT \* FROM ~~PROG~~ <sup>UTENTE</sup> WHERE ~~PROG~~ <sup>NEW</sup> UTENTE.ID = ~~PROG~~ <sup>NEW</sup> PROG  
AND UTENTE.ISPROP = TRUE )

MEDIA\_PRENOT\_GIORNO (R: INT64, IN: DATE, FIN: DATE) : (B: STRING, REAL64) [0..\*]

IF IN >= FIN

ERROR

~~PROG~~ = ( SELECT \* FROM PROMOZIONI WHERE RISTORANTE = R AND D.INIZIO < IN AND  
DATA\_FINE >= FIN )

MEDIA(ID)

RES = ( SELECT NAME, ~~AVG~~ ~~(MEDIA(ID))~~ FROM PROM ~~GROUP BY NAME~~ )

RETURN RES

MEDIA ( P : ~~INT~~ ) : REAL64

SUM

RES = ( SELECT ~~SUM~~ ~~(PR.N-PER)~~ FROM PROMOZIONE PR, USA\_PROM U, PRENOTAZIONE  
PR WHERE PR.ID = P AND U.PROMOZIONE = P AND U.~~PRENOTAZIONE~~ <sup>PRENOTAZIONE</sup> = PR.ID )

( RETURN ( RES.SUM / GIORNI ) )

GIORNI = EXT.CALCOLA\_GIORNI ( P )



TROVA (C: STRING, D: DATE, S: ~~INT~~ <sup>INT</sup> [0..100], T: STR [1..8], ~~INT~~ N: INT > 0)

RES = SELECT R.NOME FROM RISTORANTE<sup>R</sup>, PROMOZIONE<sup>P</sup>, <sup>PA\_CUINA F</sup> WHERE R.CITTA = C AND  
AND R.ID > F.RISTORANTE AND P.RISTORANTE = R.ID AND P.DATA IN <= D AND P.D  
AND P.SCONTO >= S AND F.TIPO IN T AND (P.MAX >= NUM\_PER (P.ID) + N)

NUM\_PER (P: INT) : INTGZ

~~RETURN~~ RES = SELECT SUM(<sup>PRE.NUM\_PER</sup> ~~NUM\_PER~~) FROM USA\_PROM U, PROMOZIONE PR, PRENOTAZIONE PRE  
WHERE ~~PRE~~ U.PRENOTAZIONE = PRE.ID AND U.PROMOZIONE = PR.ID AND  
PR.ID = P

RETURN RES.SUM