



Specifica Use-case RegistrazioneRistorante

```
nuovo_ristorante(
    nome:Stringa,
    piva:PartitaIVA,
    indirizzo:Indirizzo,
    tipi_cucine: Stringa [1..*],
    cap: CAP,
    citta: Citta
): Ristorante

pre:
    (cap, citta):cap_citta

post: viene creato e restituito un oggetto resultRistorante con:
    - result.nome = nome
    - result.partitaIVA = piva
    - result.indirizzo = indirizzo
    - result.cucine = tipi_cucine

Viene creato il link (result, cap): rist_cap
```

99. Requisiti sulle funzionalità

99.2. I ristoratori devono poter

99.2.1. iscriversi per registrare i propri ristoranti

99.2.2. gestire una prenotazione:

99.2.2.3. contrassegnarla come consumata (se accettata e se è nel passato)

99.2.2.3. contrassegnarla come non utilizzata (se accettata e se è nel passato)

99.2.3. creare una promozione fornendo i dati di cui al req. 5.

99.2.4. aggiungere un periodo di validità ad una promozione

99.2.3. aggiungere ed eliminare un periodo di indisponibilità (v. req. 2.6)

99.3. Lo staff di RistoBook deve poter:

99.3.1 aggiungere una tipologia di cucina.

Specifica Use-case GestionePrenotazioni

```
accetta_prenotazione(p:Prenotazione)
pre:
    p.stato = pendente
    p.istante_app > istante corrente

post:
    p.stato = accettata

rifiuta_prenotazione(p:Prenotazione)
pre:
    p.stato = pendente
    p.istante_app > istante corrente

post:
    p.stato = rifiutata
```

Specifica Use-case GestioneCucina

```
gestioneCucina (
    R:ristorante
)
pre:
    -deve esistere un Ristorante
    -R.tipi_cucine deve essere sempre valorizzato
post:
    -viene valorizzato l'attributo R.tipi_cucine=Stringa
```

99. Requisiti sulle funzionalità

99.1. I clienti devono poter

99.1.1. effettuare prenotazioni presso un ristorante, indicando i dati di cui al req. 4

99.1.2. cancellare una prenotazione (se in anticipo e non rifiutata)

Specifica Use-case PrenotazioneUtente

```
prenotazione(
    c:Cliente
    d:dataora
    nc:Numero_di_coperti
    r:Ristorante):Prenotazione

pre:
    -puo' esiste una Prenotazione solo se: d >= istante corrente
    and
        nc > 0
    -puo' cancellare una prenotazione solo se: d>=istante corrente
    and
        Prenotazione.stato != rifiutata

post:
    -Crea e restituisci resultPrenotazione con :
        -result.n_coperti=nc
        -result.istante_app=d
    -crea (c,result):cliente_pren
```