Contratti

**Pre-condizione generale**: l’attore è identificato con un’istanza *org* di Organizzatore.

1.**VerificaStatoCucina**()

**Pre-condizioni: - - -**

**Post-condizioni: - - -**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. **CreaEvento**(nome: testo, cliente: testo, dataInizio: data, dataFine: data)

**Pre-condizioni: - - -**

**Post-condizioni**:

* viene creata un’istanza *ev* di Evento
* *ev*.nome = nome
* *ev*.cliente = cliente
* *ev*.dataInizio = dataInizio
* *ev*.dataFine = dataFine
* *ev*.stato = “programmato”
* viene creata l’associazione **gestisce** tra *org* e *ev*

2a.1 **ApreEvento**(e:Evento)

**Pre-condizioni: - - -**

**Post-condizioni: - - -**

2a.2 **TerminaEvento**(note?:testo, documentazione?:testo)

**Pre-condizioni:**

* è in corso la visualizzazione di un evento *e*
* è presente un’associazione **gestisce** tra *org* ed *e*

**Post-condizioni**:

* se *e.*stato è “in corso”
  + *e*.stato = “terminato”
  + [se note è specificato] *e*.dettagli.note = note
  + [se documentazione è specificato] *e*.dettagli.documentazione = documentazione

2a.2a.1 **AnnullaEvento**(note?:testo, documentazione?:testo)

**Pre-condizioni:**

* è in corso la visualizzazione di un evento *e*
* è presente un’associazione **gestisce** tra *org* ed *e*

**Post-condizioni**:

* se *e*.stato è “in corso”
  + e.stato = “annullato”
  + [se note è specificato] *e*.dettagli.note = note
  + [se documentazione è specificato] *e*.dettagli.documentazione = documentazione
  + Per ogni Servizio *s* che *e* **prevede**:
    - *s*.stato= annullato
    - Per ogni istanza di Compito di Servizio *cs* associata via **è organizzato in** a *s:*
      * l’associazione **è assegnato** tra *cs* e ogni Personale *p* viene eliminata
      * l’associazione **svolto durante** tra *cs* e ogni Turno *t* viene eliminata

2a.2b.1 **EliminaEvento**()

**Pre-condizioni:**

* è in corso la visualizzazione di un evento *e*
* è presente un’associazione **gestisce** tra *org* ed *e*

**Post-condizioni**:

* se *e*.stato è “programmato”:
  + Per ogni istanza di Servizio *s* che *e* **prevede**:
    - Per ogni istanza di Compito di Servizio *cs* associata via **è organizzato in** a *s:*
      * l’associazione **è assegnato** tra *cs* e ogni Personale *p* viene eliminata
      * l’associazione **svolto durante** tra *cs* e ogni Turno *t* viene eliminata
      * l’associazione **è organizzato in** tra *cs* e *s* viene eliminata
      * L’istanza *cs* viene eliminata
    - *s* viene eliminata
  + *e* viene eliminato

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. **AggiungeDettagli**(luogo?: testo, durata?: numero, numServizi?: numero, numPartecipanti?: numero, note?: testo, documentazione?: testo)

**Pre-condizioni:**

* è in corso la visualizzazione di un evento *e*

**Post-condizioni:**

* *e*.stato non è terminato o annullato:
  + [se luogo è specificato] *e*.luogo = luogo
  + [se durata è specificato] *e*.durata = durata
  + [se numServizi è specificato] *e*.numServizi = numServizi
  + [se numPartecipanti è specificato] *e*.numPartecipanti = numPartecipanti
  + [se note è specificato] *e*.note = note
  + [se documentazione è specificato] *e*.documentazione = documentazione

3a.1 **ModificaDettagli**(dataInizio?:data, dataFine?:data, luogo?: testo, durata?: numero, numServizi?: numero, numPartecipanti?: numero, note?: testo, documentazione?: testo)

**Pre-condizioni:**

* è in corso la visualizzazione di un evento *e*

**Post-condizioni:**

* *e*.stato non è terminato o annullato:
  + [se dataInizio è specificato] *e*.dataInizio = dataInizio
  + [se dataFine è specificato] *e*.dataFine = dataFine
  + [se luogo è specificato] *e*.luogo = luogo
  + [se durata è specificato] *e*.durata = durata
  + [se numServizi è specificato] *e*.numServizi = numServizi
  + [se numPartecipanti è specificato] *e*.numPartecipanti = numPartecipanti
  + [se note è specificato] *e*.note = note
  + [se documentazione è specificato] *e*.documentazione = documentazione

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. **SceglieChef**(c: Chef)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*

**Post-condizioni:**

* viene creata l’associazione **assegnato a** tra *ev* e lo Chef c

4a.1 **ModificaChef**(c: Chef)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* esiste un’associazione **assegnato a** tra lo Chef *c1* ed *ev*

**Post-condizioni:**

* l’associazione **assegnato a** tra *c1* ed *ev* viene cancellata
* viene creata l’associazione **assegnato a** tra *ev* e lo Chef c

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. **AggiungeServizio**(tipologia: testo, data: data,orario: numero)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* ev.stato è “in corso” o “programmato“

**Post-condizioni:**

* Viene creata una nuova istanza *s* Servizio con:
  + *s*.tipologia = tipologia
  + *s*.data = data
  + *s*.orario = orario
  + *s*.stato = *“*in attesa*”*
  + viene creata l’associazione **prevede** tra *s* ed *ev*

5a.1 **ModificaServizio**(servizio: Servizio, tipologia?: testo, orario?: numero,data?: data)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* *ev* **prevede** servizio
* *ev*.stato è “in corso” o “programmato”

**Post-condizioni:**

* Se servizio.stato non è confermato,annullato o terminato:
  + [se tipologia è specificato] servizio.tipologia = tipologia
  + [se orario è specificato] servizio.orario = orario
  + [se data è specificato] servizio.data = data

5b.1 **AnnullaServizio**(servizio: Servizio)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* *ev* **prevede** servizio
* *ev*.stato è “in corso”

**Post-condizioni:**

* se servizio.stato non è annullato o terminato:
  + Per ogni istanza di Compito di Servizio *cs*, l’associazione **è assegnato** tra *cs* e ogni Personale *p* viene eliminata
  + Per ogni istanza di Compito di Servizio *cs*, l’associazione **svolto durante** tra *cs* e ogni Turno *t* viene eliminata
  + servizio.stato = “annullato”

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6. **ConfermaMenu**(servizio: Servizio)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* *ev* prevede servizio

**Post-condizioni:**

* Se servizio.stato = “in attesa” e un Menù **in uso in** servizio:
  + servizio.stato = “in corso”
  + *ev*.stato = “in corso”

6a.1 **RigettaMenu**(servizio: Servizio)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* *ev* prevede servizio

**Post-condizioni: - - -**

6b.1 **ProponeModifica**(tipologia:testo, voce: Voce) //AGGIUNGERE MENU??

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*

**Post-condizioni:**

* Viene creata un'istanza *pm* di Proposta di Modifica
  + *pm*.tipologia = tipologia
  + viene creata l’associazione **riferita a** tra *pm* e voce

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

7. **AssegnaPersonaleServizio**(servizio: Servizio,personale: Personale)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* *ev* **prevede** servizio

**Post-condizioni:**

* Se personale **da disponibilità** in turno:
  + Viene creata l’associazione **scelto per** tra personale e servizio

7a.1 **RimuovePersonaleServizio**(servizio: Servizio,personale: Personale)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* *ev* **prevede** servizio
* personale **scelto per** servizio

**Post-condizioni:**

* L’associazione **scelto per** tra personale e servizio viene cancellata

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

7. **AssegnaCompitoServizio**(servizio: Servizio,turno: Turno, personale: Personale,attività: testo)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* *ev* **prevede** servizio

**Post-condizioni:**

* Se personale **scelto per** turno:
  + viene creata un’istanza *cs* di Compito di Servizio
  + *cs*.attività = attività
  + viene creata l’associazione **è assegnato** tra *cs* e personale
  + viene creata l’associazione**svolto durante** tra *cs* e turno
  + viene creata l’associazione **è organizzato in** tra *cs* e servizio
  + viene rimossa l’associazione **scelto per** tra personale e servizio

7a.1. **RimuoveCompitoServizio**(servizio: Servizio, compito: Compito)

**Pre-condizioni:**

* è in corso la definizione di un Evento *ev*
* *ev* **prevede** servizio
* servizio **è organizzato in** compito

**Post-condizioni:**

* Se servizio.stato non è annullato o terminato:
  + L’associazione **è organizzato in** tra compito e servizio viene eliminata
  + Per ogni Turno *t*, l’associazione **svolto durante** tra compito e *t* viene eliminata
  + Per ogni istanza di Personale *p*, l’associazione **è assegnato** tra *cs* e *p* viene eliminata