

Contratti per lo UC “Gestire i turni”

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

1. creaTurniCucina(Insieme L di Date, orario: Orario, sede: testo?, scadenza: Data?)

Pre-condizioni:

Post-condizioni:

- per ogni Data *l* in *L* :
Se non c'è nessun turno di cucina con la stessa Data *l*, nello stesso orario e sede
 - è stata creata un'istanza *t* di Turno di cucina
 - *t.data*=*l*
 - *t.orario*=orario
 - [se sede è specificata] *t.location*=sede
 - [se sede non è specificata] *t.location*=sedeDiDefault
 - [se scadenza è specificata] *t.scadenza*=scadenza
 - il Calendario generale **contiene** *t*
 - il Tabellone dei turni **contiene** *t*

1a.1. consultaEventi ()

Pre-condizioni:

Post-condizioni:

1a.2. creaTurniServizio (servizio:Servizio, lista *L* di Date, orario: Ora?, scadenza: Data?)

Pre-condizioni:

Post-condizioni:

- per ogni data *l* in *L*:
Se non c'è nessun Turno di servizio con la stessa Data *l*, nello stesso orario e location di servizio
 - è stata creata un'istanza *t* di Turno di servizio
 - *t.data*=*l*
 - [se orario è specificato] *t.orario*=orario
 - [se orario non è specificato] *t.orario*=servizio.orario
 - *t.location*=servizio.location
 - [se scadenza è specificata] *t.scadenza*=scadenza
 - il Calendario generale **contiene** *t*

1b.1. modificaTurni (Insieme T di turni, data?: Data, orario?: Orario, sede: testo?, scadenza?: Data, periodo?: Periodo)

Pre-condizioni:

Post-condizioni:

- per ogni Turno t in T :
Se non esiste un altro Turno dello stesso tipo di t con caratteristiche simili (orario,data, location) e se t **non** ha delle disponibilità
 - [se data è specificato] $t.data=l$
 - [se orario è specificato] $t.orario=orario$
 - [se sede è specificata] $t.location=sede$
 - [se scadenza è specificata] $t.scadenza=scadenza$
 - [se periodo è specificato] per tutti i turni **in relazione con** t la cui data è compresa in periodo:
 - [se data è specificato] $t.data=l$
 - [se orario è specificato] $t.orario=orario$
 - [se sede è specificata] $t.location=sede$
 - [se scadenza è specificata] $t.scadenza=scadenza$

1c.1. eliminaTurni (Insieme T di turni, periodo?: Periodo)

Pre-condizioni:

Post-condizioni:

- per ogni turno in T :
Se turno **non** ha delle disponibilità
 - turno è eliminato
 - [se periodo è specificato] sono stati eliminati tutti i turni **in relazione con** turno la cui data è compresa in periodo

2. raggruppaInBlocchi (Insieme T di turni di cucina, periodo:Periodo?)

Pre-condizioni:

Post-condizioni: Se T contiene almeno due turni

- viene creata un'istanza b di Blocco
- per ogni turno in T :
 - turno **appartiene a** b
- [se periodo è specificato] è stata creata una nuova istanza di Blocco per ogni ricorrenza dei turni t_i **in relazione con** l'insieme T di turni di cucina la cui data è compresa in periodo

2a.1. modificaRaggruppamento (b:Blocco, Insieme Te di Turni da eliminare?, Insieme Ta di Turni da aggiungere?, periodo?: Periodo)

Pre-condizioni:

Post-condizioni:

- [se Te **appartiene** a b] sono stati rimossi tutti i turni presenti in Te da b
 - [se periodo è specificato] per tutti i turni Tr **in relazione con** Te e compresi in periodo sono state rimosse le ricorrenze di Tr dal blocco di appartenenza
- [se Ta non appartiene a b] sono stati aggiunti tutti i turni presenti in Ta a b
 - [se periodo è specificato] per tutti i turni Tr **in relazione con** Ta e compresi in periodo sono state aggiunte le ricorrenze di Tr al blocco di appartenenza

2b.1. eliminaRaggruppamento (b:blocco, periodo?: Periodo)

Pre-condizioni:

Post-condizioni:

- per ogni turno **appartenente a** blocco :
 - turno non **appartiene a** blocco
 - [se periodo è specificato] tutti i turni **in relazione con** turno e compresi in periodo non **appartengono a** nessun Blocco

3. programmaTurni(Insieme T di turni, periodo: Periodo, ricorrenza:testo)

Pre-condizioni:

Post-condizioni:

- per ogni turno t in T:
 - per ogni Data d compresa in periodo, tale che d è t.data ricorrente secondo ricorrenza (calcolata come nella business rules "Ricorrenza"):
Se **non** esiste un altro Turno dello stesso tipo di t con le caratteristiche simili (orario, data, location)
 - viene creato un nuovo Turno *tn*
 - *tn*.orario=t.orario
 - *tn*.location=t.location
 - *tn*.data=d
 - *tn* è **in relazione** con t
 - il Calendario generale **contiene** *tn*
 - [se t è Turno di cucina] il Tabellone dei turni contiene *tn*

4. impostaBlocco(Insieme T di turni, scadenza: Data)

Pre-condizioni:

Post-condizioni:

- per ogni t appartenente a T :
 - $t.scadenza = scadenza$

4a.1 eliminaBlocco(Insieme T di turni)

Pre-condizioni:

Post-condizioni:

- per ogni turno t in T :
 - $t.scadenza_è\ infinita$

Contratti per lo UC “Indicare le proprie disponibilità”

Pre-condizione generale: l'attore è identificato con un'istanza *dip* di Dipendente

1. indicaDisponibilità(Insieme T di turni, periodo: Data?, ricorrenza: testo?)

Pre-condizioni:

Post-condizioni: Se ogni turno t appartenente a T ha scadenza indefinita, oppure ha scadenza maggiore della data odierna e se t è un turno di cucina e **appartiene a** un blocco *b*, allora tutti i turni *tb* appartenenti a *b* sono contenuti in I

- *dip* è disponibile in tutti i turni *t* appartenenti a I
- [se periodo e ricorrenza sono specificati] *dip* è disponibile in tutti i turni *t* secondo ricorrenza (calcolata come nella business rules "Ricorrenza") compresi in periodo

1a. eliminaDisponibilità(Insieme T di turni, periodo: Data?, ricorrenza: testo?)

Pre-condizioni:

Post-condizioni: Se ogni turno t appartenente a T ha scadenza indefinita, oppure ha scadenza maggiore della data odierna e se t è un turno di cucina e **appartiene a** un blocco *b*, allora tutti i turni *tb* appartenenti a *b* sono contenuti in I

- *dip* non è disponibile in nessuno dei turni t appartenenti a T, esteso per tutti i turni compresi nel periodo, secondo ricorrenza (calcolata come nella business rules "Ricorrenza")
- [se periodo e ricorrenza sono specificati] *dip* **non** è disponibile in tutti i turni *t* secondo ricorrenza (calcolata come nella business rules "Ricorrenza") compresi in periodo