

# Contratti per UC “Gestire i turni”

## 1. generaTabelloneTurni(servizio: Servizio, evento: Evento)

### Pre-condizioni

- Il servizio e l'evento devono essere validi e associati a un organizzatore.

### Post-condizioni

- Viene creata un'istanza di tabellone turni
- Il tabellone turni è associato al servizio e all'evento forniti
- Il tabellone turni viene salvato nel sistema
- Il sistema notifica della generazione del tabellone dei turni

## 1a. apriTabelloneTurni(servizio: Servizio, Evento: evento)

### Pre-condizioni

- L'evento è registrato nel sistema (ossia “in carico”)
- Esiste già un tabellone turni associato al servizio e all'evento

### Post-condizioni

- Il tabellone turni esistente viene caricato dal sistema

## 2. generaTurnoCucina(Durata: durata, Luogo: luogo?, Data: scadenza?, Data: dataSvolg?, Data: dataFine?)

### Pre-condizioni

- La cucina corrispondente al luogo (se specificato) non deve risultare già occupata nello stesso orario

### Post-condizioni

- Viene creata un'istanza di turno di cucina
- Il turno viene associato ai parametri forniti (durata, luogo se presente, scadenza, ecc.)
- [se è specificato una quantità] datasvolg = dataSvolg
- [se è specificato una quantità] datafine = dataFine

## **2a.1 selezionaTurno(turno: Turno)**

### **Pre-condizioni**

- Il turno da selezionare esiste nel sistema

### **Post-condizioni**

- Il turno indicato risulta selezionato per le successive operazioni di modifica

## **2a.2 modificaInfoTurno(Durata: durata?, Luogo: luogo?, Data: scadenza?, Data: dataSvolg?, Data: dataFine?)**

### **Pre-condizioni**

- Il turno selezionato non deve avere disponibilità già registrate (cioè non deve essere “prenotato”)
- La nuova cucina/luogo (se modificata/o) non deve essere già occupata/o
- La nuova scadenza (se fornita) non deve superare il tempo rimanente rispetto alla data corrente

### **Post-condizioni**

- [se è specificato una quantità] durata.durata = durata
- [se è specificato una quantità] data.datafine= dataFine
- [se è specificato una quantità] luogo.luogo = luogo
- [se è specificato una quantità] data.scadenza = scadenza
- [se è specificato una quantità] data.datasvolg = dataSvolg
- Il sistema notifica che le informazioni del turno sono state aggiornate

## **2a.2a.1 raggruppaTurniSelezionati(Lista<Turno>: turni)**

### **Pre-condizioni**

- I turni nella lista non devono essere già stati raggruppati precedentemente

### **Post-condizioni**

- I turni forniti vengono raggruppati logicamente in un'unica unità
- Viene creato e assegnato un identificativo di raggruppamento
- Il sistema notifica l'avvenuto raggruppamento dei turni

## **2a.2b.1 eliminaTurniSelezionati(Lista<Turno>: turni)**

### **Pre-condizioni**

- I turni da eliminare non devono avere disponibilità già registrate

### **Post-condizioni**

- Ogni turno selezionato nella lista viene eliminato dal sistema
- Il sistema notifica l'eliminazione dei turni dal sistema

## **2b.1 selezionaRaggruppamento(Raggruppamento: raggruppamento)**

### **Pre-condizioni**

- Il raggruppamento da selezionare esiste nel sistema

### **Post-condizioni**

- Il raggruppamento indicato risulta selezionato per le successive operazioni

## **2b.2 sciogliRaggruppamentiSelezionati(Lista<Raggruppamento>: raggruppamenti)**

### **Pre-condizioni**

- I raggruppamenti forniti non devono contenere turni con disponibilità già registrate

### **Post-condizioni**

- Ogni raggruppamento indicato viene “sciolto”
- I turni al loro interno tornano indipendenti
- Il sistema notifica l'avvenuto scioglimento dei raggruppamenti

### **3. generaTurnoServizio(Durata: durataPrep?, Durata: durataRig?, Luogo: luogo?, Data: scadenza?, Data: dataFine?)**

#### **Pre-condizioni**

- La scadenza (se specificata) non deve superare il tempo rimanente rispetto alla data corrente.

#### **Post-condizioni**

- Viene creata un'istanza di turno di servizio.
- [se è specificato una quantità] durata.durataprep = durataPrep
- [se è specificato una quantità] durata.duratarig= durataRig
- [se è specificato una quantità] luogo.luogo = luogo
- [se è specificato una quantità] data.scadenza = scadenza
- [se è specificato una quantità] data.datafine = dataFine
- Il sistema notifica la creazione del turno di servizio

### **3a.1 selezionaTurno(Turno: turno)**

#### **Pre-condizioni**

- Il turno da selezionare esiste nel sistema

#### **Post-condizioni**

- Il turno indicato viene selezionato per le operazioni di modifica

### **3a.2 modificaTurniSelezionati(Durata: durataPrep?, Durata: durataRig?, Luogo: luogo?, Data: scadenza?, Data: dataFine?)**

#### **Pre-condizioni**

- I turni selezionati non devono avere disponibilità già registrate
- La nuova scadenza (se specificata) non deve superare il tempo rimanente rispetto alla data corrente

#### **Post-condizioni**

- [se è specificato una quantità] durata.durataprep = durataPrep
- [se è specificato una quantità] durata.duratarig = durataRig
- [se è specificato una quantità] luogo.luogo = luogo
- [se è specificato una quantità] data.scadenza = scadenza
- [se è specificato una quantità] data.datafine = dataFine
- Il sistema notifica l'avvenuta modifica dei turni selezionati

### **3a.2a.1 eliminaTurniSelezionati(Lista<Turno>: turni)**

#### **Pre-condizioni**

- I turni da eliminare non devono avere disponibilità già registrate

#### **Post-condizioni**

- Ogni turno selezionato viene rimosso dal sistema
- Il sistema notifica l'eliminazione dei turni dal sistema