

Allegato Tecnico Gestione Eventi

Requisiti

UC dettagliato

Informazioni generali

Nome caso d'uso: Gestire gli eventi

Portata: Sistema

Livello: Obiettivo utente

Attore primario: Organizzatore

Parti Interessate: Chef, Cuoco, Personale di servizio

Pre-condizioni:

- L'attore deve essere autenticato come Organizzatore

Post-condizioni:

- La scheda dell'evento è registrata e consultabile

Scenario principale di successo

#	Attore	Sistema
1	Crea un nuovo evento	Registra il nuovo evento. Se l'evento è ricorrente, crea un numero di occorrenze, uguali all'originale, pari a quanto indicato.
2	Opzionalmente , aggiunge un nuovo servizio all'evento	Registra le informazioni del nuovo servizio. Se l'evento è ricorrente e l'utente lo desidera, vengono create copie del servizio per tutte le altre occorrenze.
	<i>Ripete il passo 2, operando su tutti i servizi desiderati</i>	
	<i>Prosegue al passo 3, oppure termina il caso d'uso</i>	
3	Opzionalmente , assegna uno chef all'evento	Registra l'assegnazione dello chef
4	Opzionalmente , prenota un membro del personale (cuoco oppure personale di servizio)	Registra la prenotazione
	<i>Ripete il passo 4, operando su tutto il personale desiderato</i>	
5	Opzionalmente , approva un menù scelto dallo chef	Registra l'approvazione del menù Se tutti i menù per l'evento sono stati approvati, aggiorna lo stato dell'evento ad "in corso"

	<i>Ripete il passo 5 operando su tutti i menù desiderati.</i>	
6	Opzionalmente , assegna un ruolo ad un membro del personale di servizio	Registra l'assegnazione ed aggiorna il tabellone dei turni
	<i>Ripete il passo 6, operando su tutto il personale desiderato</i>	
	<i>Ripete a partire dal passo 4 e fino alla selezione di tutto il personale desiderato</i>	
	<i>Ritorna al passo 1 oppure termina il caso d'uso</i>	

Estensione 1a

#	Attore	Sistema
1a.1	Modifica le informazioni di un evento esistente	Registra le modifiche all'evento. Se l'evento è ricorrente e l'utente lo desidera, registra le modifiche in tutte le occorrenze.

Eccezione 1a

#	Attore	Sistema
1a.1	Modifica le informazioni di un evento esistente	L'evento è nello stato "in corso" e le informazioni che si vogliono modificare non possono subire variazioni.
	<i>Termina il caso d'uso</i>	

Estensione 1b

#	Attore	Sistema
1b.1	Elimina un evento esistente	Cancella l'evento e aggiorna la tabella dei turni. Se l'evento è ricorrente e l'utente lo desidera, vengono cancellate anche le altre occorrenze.

Eccezione 1b

#	Attore	Sistema
1b.1	Elimina un evento esistente	L'evento è nello stato "in corso"
	<i>Termina il caso d'uso</i>	

Estensione 1c

#	Attore	Sistema
1c.1	Annulla un evento esistente	Segna l'evento come "annullato". Se l'evento è ricorrente e l'utente lo desidera, vengono annullate anche le altre occorrenze.

Estensione 1d

#	Attore	Sistema
1d.1	Chiusura evento	Modifica lo stato dell'evento in "terminato". Registra le note nell'evento. Se l'evento è ricorrente e l'utente lo desidera, le note vengono registrate per tutte le occorrenze.

Estensione 1e

#	Attore	Sistema
1e.1	Caricamento informazioni evento esistente	Carica le informazioni relative all'evento indicato

Eccezione 2a

#	Attore	Sistema
2a.1	Opzionalmente , aggiunge un nuovo servizio all'evento	L'evento associato al servizio è "in corso"

Estensione 2a

#	Attore	Sistema
2a.1	Modifica un servizio tra quelli associato all'evento	Registra le modifiche al servizio. Se l'evento è ricorrente e l'utente lo desidera, vengono modificate anche le copie del servizio associate alle altre occorrenze.

Eccezione 2b

#	Attore	Sistema
2b.1	Modifica un servizio <i>tra quelli associato all'evento</i>	L'evento associato al servizio è "in corso"

Estensione 2b

#	Attore	Sistema
---	--------	---------

2b.1	Elimina un servizio tra quelli associato all'evento	Registra l'eliminazione del servizio. Se l'evento è ricorrente e l'utente lo desidera, vengono eliminate anche le copie del servizio associate alle altre occorrenze.
-------------	---	--

Eccezione 3a

#	Attore	Sistema
3a.1	<i>Opzionalmente</i> , assegna uno chef all'evento	Lo chef scelto è già impegnato in un altro evento nella stessa data ed ora
	<i>Ritorno al passo 3</i>	

Eccezione 4a

#	Attore	Sistema
4a.1	<i>Opzionalmente</i> , prenota un membro del personale	Il membro scelto lavora ad un altro evento

Eccezione 4b

#	Attore	Sistema
4b.1	<i>Opzionalmente</i> , prenota un membro del personale	L'evento è in corso

Estensione 4a

#	Attore	Sistema
4a.1	Libera un membro del personale prenotato	Se necessario, rimuove il compito assegnato al membro del personale. Cancella la prenotazione.

Eccezione 4c

#	Attore	Sistema
4c.1	Libera un membro del personale prenotato	L'evento è in corso

Eccezione 4d

#	Attore	Sistema
---	--------	---------

4d.1	Libera un membro del personale prenotato	Il membro non è stato prenotato per uno dei servizi dell'evento corrente
-------------	--	--

Estensione 5a

#	Attore	Sistema
5a.1	Propone una modifica ad un menù scelto dallo chef	Registra la proposta di modifica

Eccezione 5a

#	Attore	Sistema
5a.1	Propone una modifica ad menù scelto dallo chef	Il menù è associato ad un evento in corso
	<i>Termina il caso d'uso</i>	

Eccezione 5b

#	Attore	Sistema
5b.1	<i>Opzionalmente</i> , approva un menù scelto dallo chef	Il menù è associato ad un evento in corso
	<i>Termina il caso d'uso</i>	

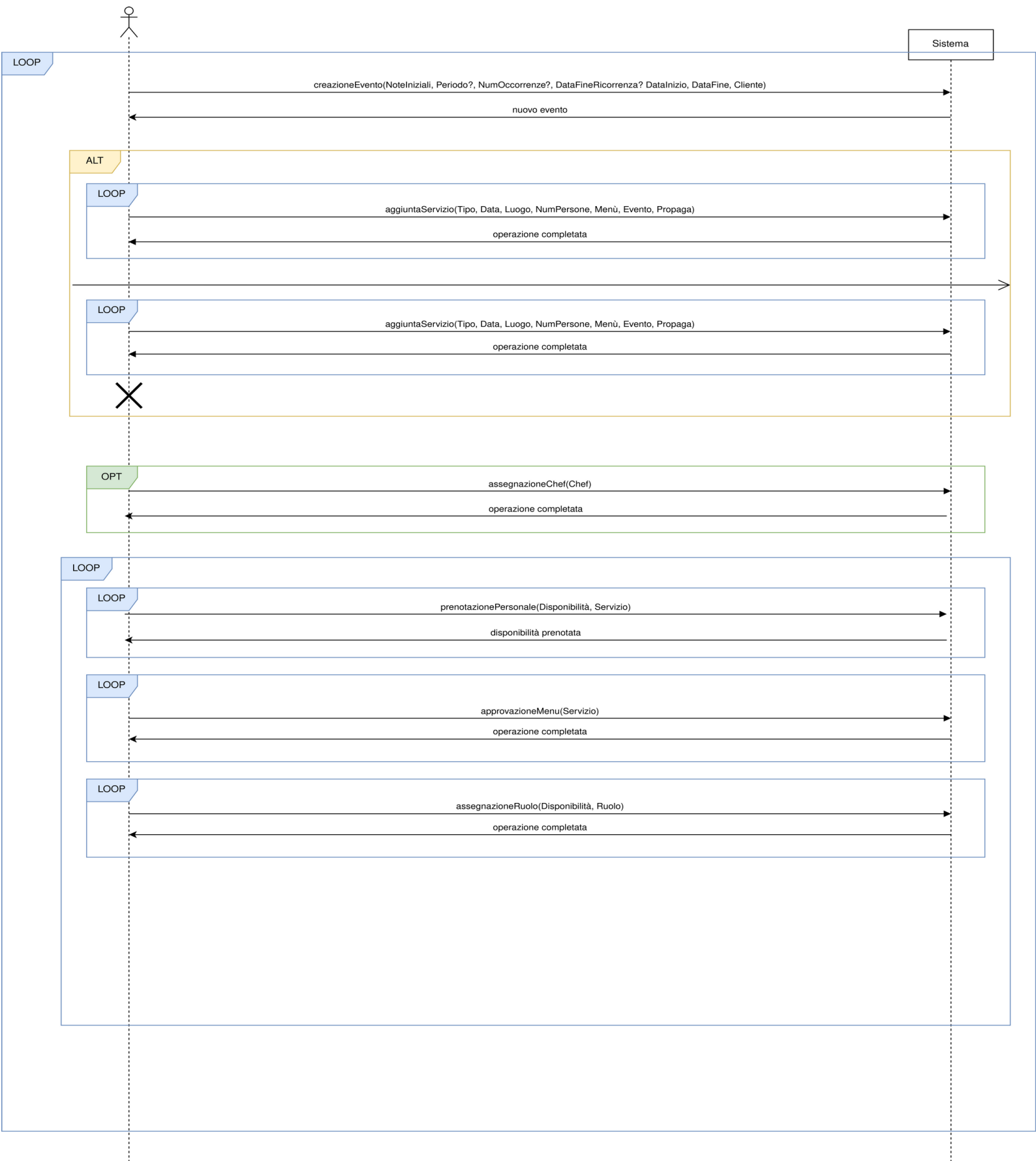
Eccezione 6a

#	Attore	Sistema
6a.1	<i>Opzionalmente</i> , assegna un ruolo ad un membro del personale di servizio	Il membro del personale indicato non è stato prenotato per un servizio dell'evento corrente

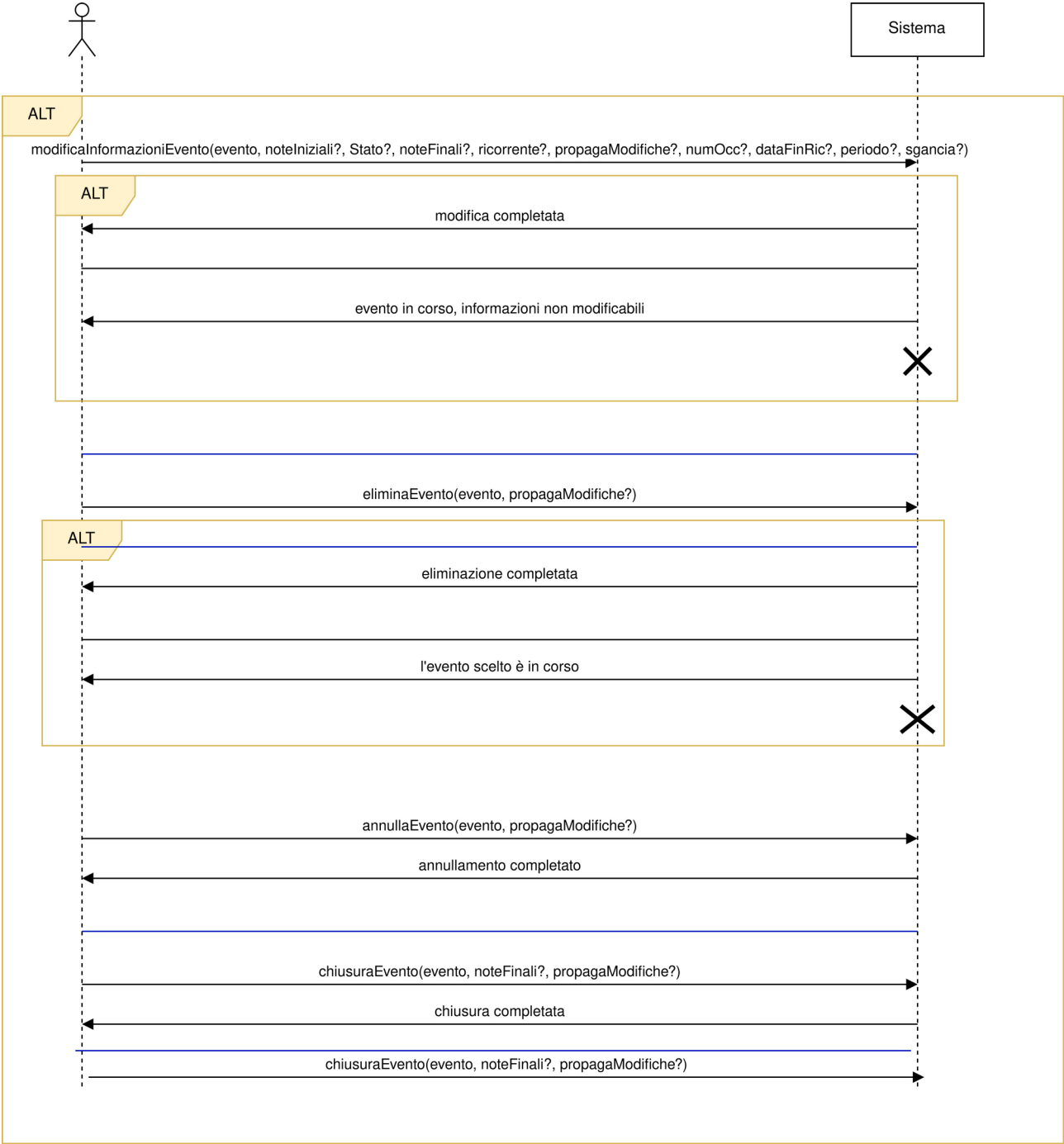
Modello di dominio



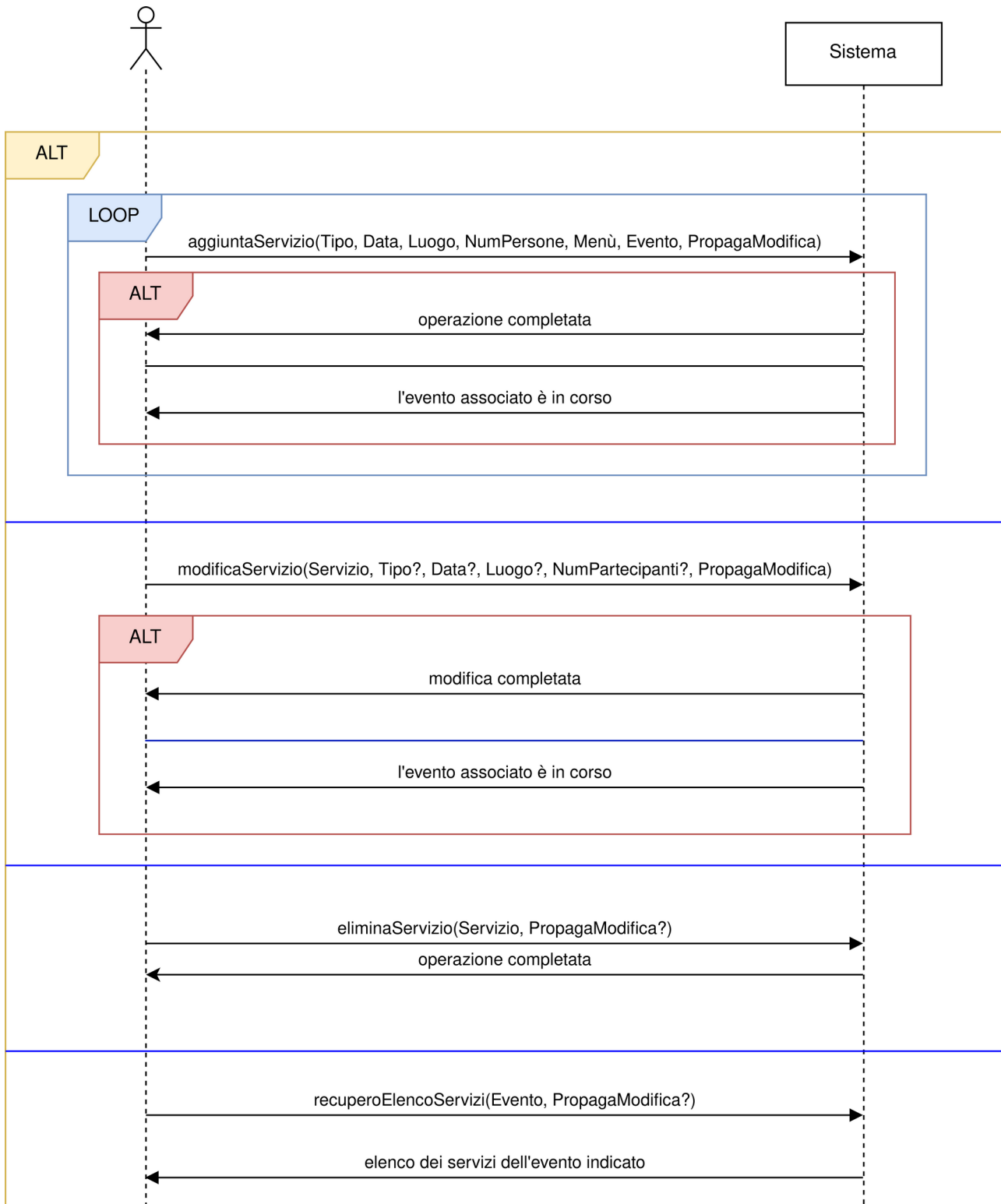
Scenario principale



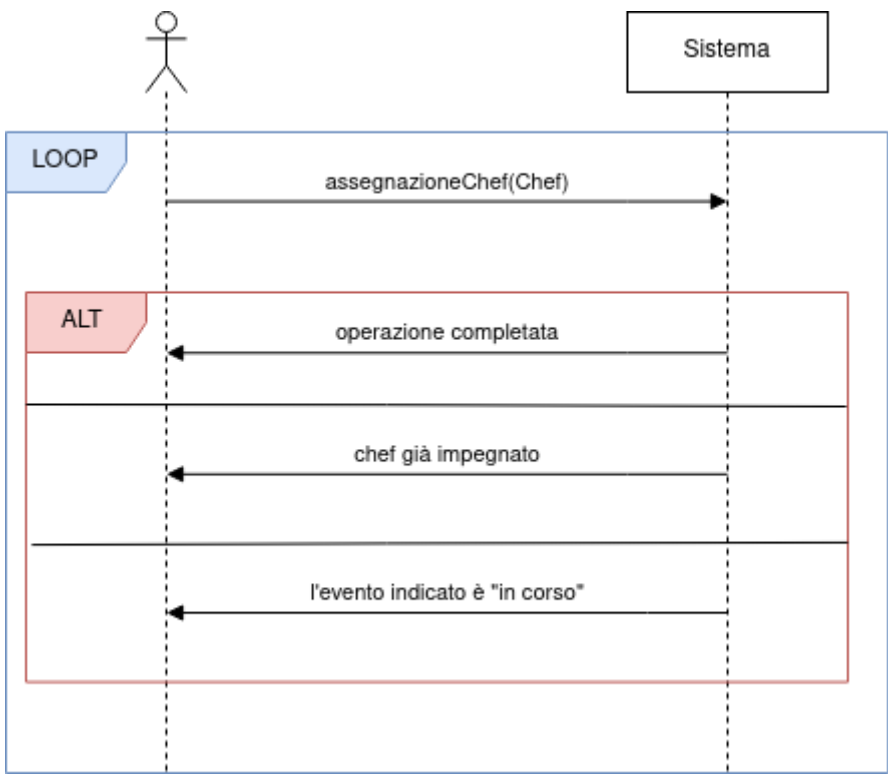
Estensioni/Eccezioni passo 1



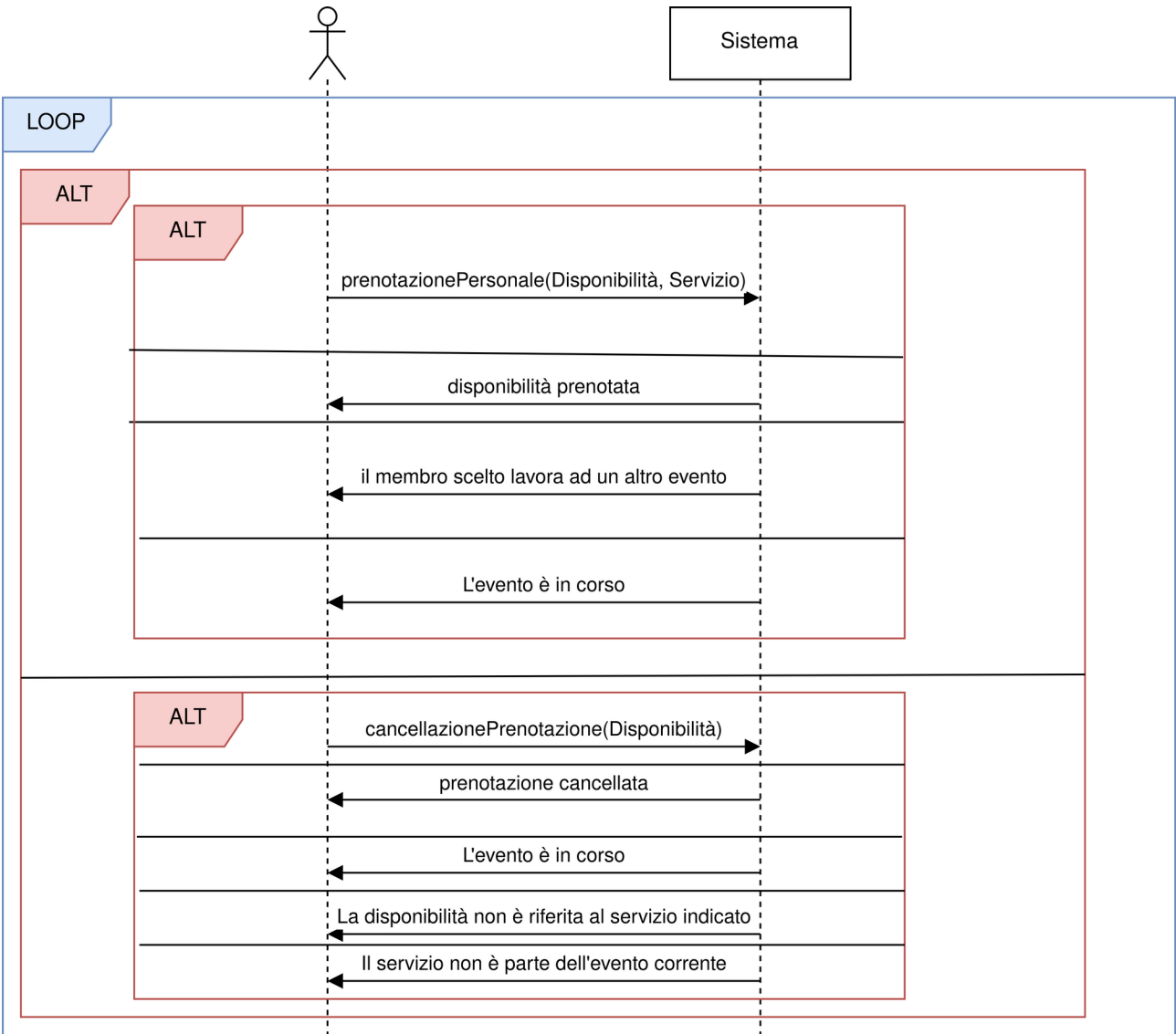
Estensioni/Eccezioni passo 2



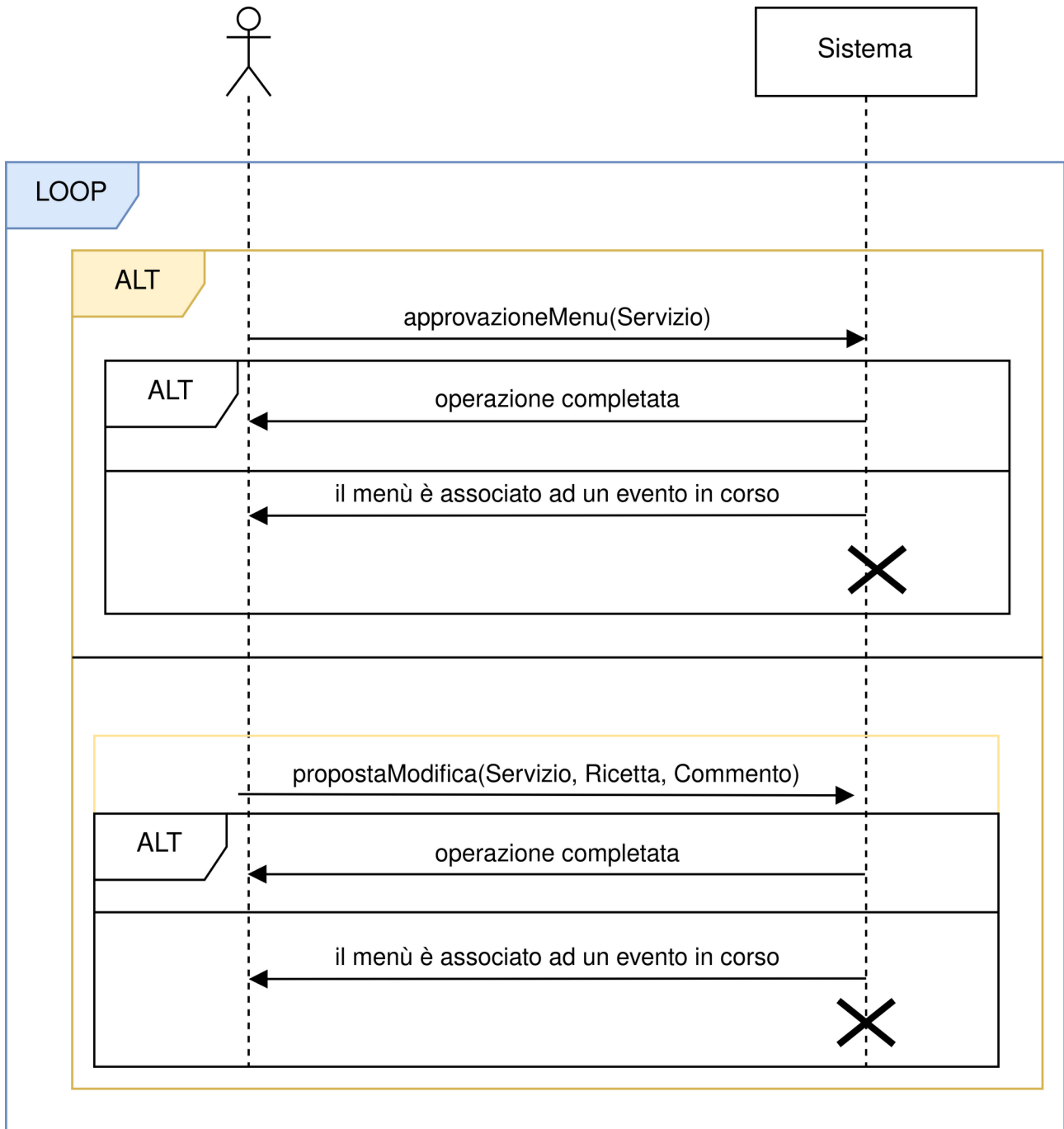
Estensioni/Eccezioni passo 3



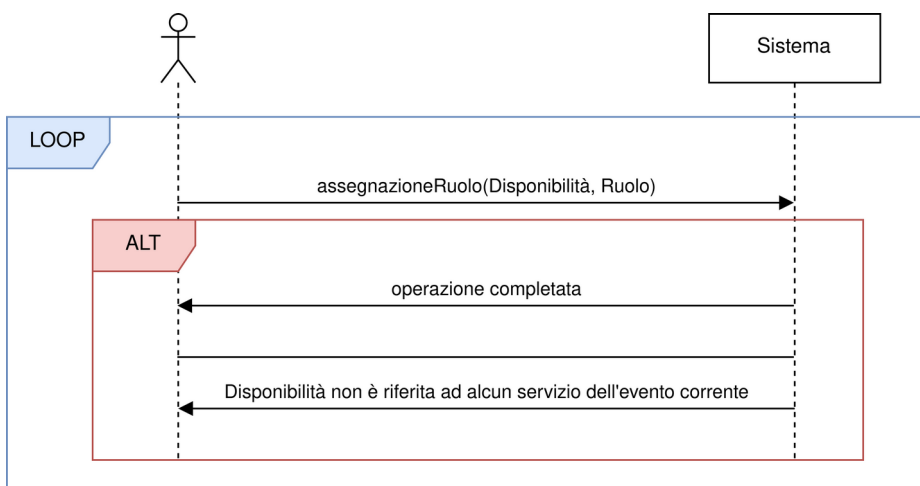
Estensioni/Eccezioni passo 4



Estensioni/Eccezioni passo 5



Estensioni/Eccezione 6



Regole di business

- Evento.Stato può essere in corso se e solo se per ogni Servizio srv tale che Evento prevede srv, srv.approvato = sì
- Evento.DataInizio è minore di t
- Evento.Stato può valere soltanto
 - "Creazione": evento di cui stiamo svolgendo la parte "amministrativa" (nomina del personale, scelta dei menù, ecc...)
 - "In Corso": evento che si sta svolgendo. Sono consentite modifiche limitate.
 - "Terminato"
 - "Annullato"
- Menù.Approvato = sì se e solo se tutte le proposte di modifica associate sono accettate
- Per un EventoCapofila ev, ev.NumeroOccorrenze è uguale al numero di Occorrenze o tali che ev composto da o
- Per un EventoCapofila ev, ev.DataFineRicorrenza è uguale alla più grande tra le date di fine di Occorrenze o tali che ev composto da o
- Per un Servizio srv, srv.Approvato = sì se e solo se per ogni Proposta di Modifica pm è specificato l'attributo accettata
- Una Proposta di Modifica pm tale che pm menù di srv, può essere riferita soltanto a Ricette referenziate da Voci appartenenti al Menù in uso in srv
- Un menù è in uso in un Evento se e solo se esso prevede un Servizio srv tale che menù in uso in srv
- Uno Chef può possedere soltanto Fogli Riepilogativi contenenti compiti di Eventi assegnatigli.
- Un Foglio Riepilogativo può essere composto soltanto da Compiti che consistono in Mansioni di Cucina che sono Ricette referenziate
- da Voci del Menù in uso nel Servizio associato al foglio, oppure Ingredienti di tali Ricette
- Un Compito può essere svolto soltanto da un Membro del Personale che ha inserito una Disponibilità nel Servizio a cui fa riferimento il Foglio Riepilogativo che contiene il Compito.
- Se per un Evento e, e.Stato = "In Corso"
- Un Compito può essere svolto soltanto da un Cuoco per cui esiste un Disponibilità d verso il Servizio associato al Compito
- in cui d.Prenotato = Sì
- Se l'attributo "Da Preparare" di un compito vale "No", allora questo non è associato ad alcun Cuoco e Turno

- Un Compito non può essere prosecuzione di se stesso
- Se un Compito c' è prosecuzione di un altro Compito c, allora c' deve essere svolto durante un Turno successivo a quello a cui è associato c'
- Un Preparazione non può essere Ingrediente di se stessa
- Solo una tra gli attributi NumeroOccorrenze e DataFineRicorrenza di Evento Capofila può essere valorizzato
- Disponibilità.Ruolo è valorizzato solo se Disponibilità.Prenotato = Sì, Disponibilità è associata ad un servizio ed il Membro del Personale associato alla Disponibilità è un Membro del Personale di Servizio
- La DataFine di un Evento Capofila è pari a quella dell'ultima delle sue ricorrenze

Contratti delle operazioni

Passo 1

1. creazioneEvento(noteIniziali: Testo, cl: Cliente, dataInizio: Testo, dataFine: Testo, periodo?: Testo, numOccorrenze?: Numero, dataFineRic?: Testo)

Pre-condizioni: Nessuna

Post-condizioni:

- È stato creato un nuovo Evento Capofila ev
- ev.NoteIniziali = noteIniziali
- ev.Stato = “Creazione”
- ev.DataInizio = dataInizio
- ev.DataFine = dataFine
- or **organizza** ev
- cl **comissiona** ev
- [Se è specificato periodo]
 - o Ev.periodo = periodo
 - o [se è specificato numOccorrenze]
 - Ev.NumeroOccorrenze = numOccorrenze
 - Sono state create numOccorrenze Occorrenze oc, per ciascuna delle quali vale
 - ev è **composto da** oc
 - oc.NoteIniziali = ev.noteIniziali
 - oc.Stato = “Creazione”
 - Se oc è l'i-esima occorrenza, oc.DataInizio = ev.DataInizio + (periodo * I) (*)
 - Se oc è l'i-esima occorrenza, oc.DataFine = ev.DataFine + (periodo * I)
 - cl **comissiona** ev
 - or **organizza** ev
- ❖ [Se è specificato dataFineRic]
 - Ev.DataFineRicorrenza = dataFineRic
 - Per ogni data della forma dataInizio + (periodo * k) nell'intervallo [dataInizio, dataFine], è stata creata un'Occorrenza oc, tale che
 - ev è **composto da** oc

- oc.Note Iniziali = ev.NoteIniziali
- oc.Stato = “Creazione”
- oc.DataInizio = ev.DataInizio + (periodo * k)
- oc.DataFine = ev. DataFine + (periodo * k)
- cl **comissiona** ev
- or **organizza** ev

1a. modificaEvento(ev: Evento, noteIniziali?: Testo, noteFinali?: Testo, cl?: Cliente, dataInizio?: Testo, dataFine?: Testo, periodo?: Testo, numOccorrenze?: Numero, dataFineRic?: Testo, propMod?: Sì/No, sgancia?: Sì/No)

Pre-condizioni:

- or **organizza** ev

Post-condizioni:

- [Se è specificato noteIniziali]
 - ev.NoteIniziali = noteIniziali
- [Se è specificato noteFinali ed ev.Stato = “Terminato”]
 - ev.noteFinali = noteFinali
- [Se ev è un’Occorrenza]
 - [Se sgancia è specificato e sgancia = “Sì”]
 - [Se ev’ è l’evento capofila **composto da** ev]
 - [Se ev’.DataFineRicorrenza è specificato]
 - ev’.DataFineRicorrenza = massima data di fine tra quelle delle ricorrenze di cui è **composto** ev’
 - [Se ev’.NumeroOccorrenze]
 - ev’.NumeroOccorrenze = ev’.NumeroOccorrenze – 1
 - È stato creato un Evento Capofila con le stesse informazioni di ev
 - ev è stato eliminato
- [Se è specificato cl]
 - [Se esiste un Cliente cl’ tale che cl’ **commissiona** ev]
 - l’associazione tra cl’ ed ev è stata eliminata
 - cl **commissiona** ev
- [Se è specificato dataInizio e ev.Stato == “Creazione”]
 - Per ogni servizio srv **previsto da** ev
 - srv.Data è stata modifica in modo che tra di essa ed ev.DataInizio sussista la stessa proporzionalità presente rispetto al vecchio valore di ev.DataInizio
 - le date delle occorrenze sono state adeguate, in modo da rispettare il periodo, secondo le stesse regole seguite in fase di creazione
 - il personale impegnato in altri eventi durante le nuove date è stato liberato
 - [Se esiste un Evento ev’ tale che ev **assegnato** ad uno Chef c ed ev’ **assegnato** a c]
 - [Se ev’.DataInizio >= ev.DataInizio ed ev’.DataFine <= ev.DataFine]
 - L’associazione tra c ed ev è stata eliminata
 - Per ogni servizio srv **previsto da** ev
 - [Se m è il Menù **in uso in** srv]
 - l’associazione tra m ed srv è stata eliminata
- [Se è specificato dataFine ed ev.Stato == “Creazione”]
 - le date delle occorrenze sono state adeguate, in modo da rispettare il periodo, secondo le stesse regole seguite in fase di creazione
 - il personale impegnato in altri eventi durante le nuove date è stato liberato

- $srv.Data$ è stata modificata in modo che tra di essa ed $ev.DataInizio$ sussista la stessa proporzionalità presente rispetto al vecchio valore di $ev.DataInizio$
- [Se esiste un Evento ev' tale che ev **assegnato ad** uno Chef c ed ev' **assegnato a** c]
 - [$ev'.DataInizio \geq ev.DataInizio$ ed $ev'.DataFine \leq ev.DataFine$]
 - L'associazione tra c ed ev è stata eliminata
- Per ogni servizio srv **previsto da** ev
 - [m è il Menù **in uso in** srv]
 - l'associazione tra m ed srv è stata eliminata
- [Se ev è un Evento Capofila]
 - [Se periodo è specificato]
 - [Se numeroOccorrenze è specificato]
 - [numeroOccorrenze > $ev.NumeroOccorrenze$]
 - Sono state create numeroOccorrenze - $ev.NumeroOccorrenze$ nuove occorrenze oc con le stesse caratteristiche di ev e sono state associate ad ev
 - [$ev.NumeroOccorrenze > \underline{numeroOccorrenze}$]
 - Sono state mantenute solo le prime numeroOccorrenze occorrenze di cui è **composto** ev
 - $ev.NumeroOccorrenze = \underline{numeroOccorrenze}$
 - [Se dataFineRic è specificato]
 - [dataFineRic > $ev.DataFineRicorrenza$]
 - Sono state create nuove occorrenze, a distanza $ev.Periodo$ l'una dall'altra, fino a raggiungere dataFineRic e sono state associate ad ev
 - [altrimenti]
 - Sono state eliminate tutte le occorrenze che compongono ev con data di inizio maggiore di dataFineRic
 - $ev.DataFineRicorrenza = \underline{dataFineRic}$
 - [Se ev è un'Occorrenza]
 - [Se propMod == Sì]
 - Se ev' è l'Evento Capofila tale che ev' **composto da** ev
 - Per ogni Occorrenza Oc di ev'
 - oc è stata modificata come ev

1b. eliminazioneEvento(ev : Evento, propMod: Sì/No)

Pre-condizioni:

- or **organizza** ev
- ev.stato = "Creazione"

Post-condizioni:

- Per ogni Servizio srv tale che ev **prevede** srv
 - eliminaServizio è stato applicato ad srv
- [Se il Menù m è **in uso in** ev]
 - È stata eliminata l'associazione tra m ed ev
- [Se ev è **assegnato a** Chef c]
 - È stata eliminata l'associazione tra ev e c
- [Se propMod = "Sì"]
 - [Se ev.Periodo è specificato]
 - Per ogni Occorrenza oc tale che ev **composto da** oc
 - eliminazioneEvento(oc , propMod)
 - [altrimenti]
 - Se ev' è **composto da** ev , per ogni Occorrenza oc tale che ev' **composto da** oc

- eliminazioneEvento(oc, propMod)
- È stato eliminato l'evento ev

1c. annullaEvento(ev: Evento, propMod: Sì/No)

Pre-condizioni:

- or Organizza ev

Post-condizioni:

- ev.Stato = "Annullato"
- Per ogni servizio srv tale che ev **prevede** srv
 - Per ogni Disponibilità disp **in** srv
 - disp.Prenotato = "No"
 - disp.Ruolo = " "
 - l'associazione tra disp ed srv è stata eliminata
- [Se propMod = Sì]
 - [ev è un Evento Capofila]
 - Per ogni Occorrenza oc tale che ev **composto da** oc
 - è stata applicata annullaEvento(oc, propMod)
 - [altrimenti]
 - Se ev' è composto da ev, per ogni Occorrenza oc tale che ev' **composto da** oc
 - è stata applicata annullaEvento(oc, propMod)

1d. chiusuraEvento(ev: Evento, noteFin?: Testo, propMod: Sì/No)

Pre-condizioni:

- or **organizza** ev

Post-condizioni:

- ev.Stato = "Terminato"
- [Se è specificato noteFin] ev.NoteFinali = noteFin
- [Se propMod = "Sì"]
 - [Se ev è un EventoCapofila]
 - Per ogni Occorrenza oc da cui è **composto** ev
 - chiusuraEvento(oc, noteFine, propMod)
 - [altrimenti]
 - Se ev' è **composto da** ev, per ogni Occorrenza oc tale che ev' **composto da** oc
 - chiusuraEvento(oc, noteFine, propMod)

1e. caricamentoEvento(ev: Evento)

Pre-condizioni:

- or **organizza** ev

Post-condizioni: nessuna

Passo 2

2. aggiuntaServizio(tipo: Testo, data: Testo, luogo: Testo, numPersone: Testo, menu: Menù, propaga: Sì/No)

Pre-condizioni:

- È in corso la modifica di un evento e
- e.Stato = “Creazione”
- e.DataInizio <= data && e.DataFine >= data

Post-condizioni:

- È stato creato un nuovo Servizio srv
- Srv.Tipo = tipo
- Srv.Data = data
- Srv.Luogo = luogo
- Srv.NumeroPersone = numPersone
- menu In uso in Srv
- e **Prevede** Srv
- Srv.approvato = No
- [Se e.Periodo è specificato e propaga = Sì]
 - o Per ogni Occorrenza oc tale che e **composto da** oc
 - È stato creato un nuovo Servizio srv'
 - srv'.Tipo = tipo
 - Se oc è l'i-esima occorrenza
 - [oc.DataInizio <= data && oc.DataFine >= data]
 - o srv'.Data = data + (e.Periodo * i)
 - srv'.Luogo = luogo
 - srv'.NumeroPersone = numPersone
 - menu In uso in srv'
 - oc **Prevede** srv'
 - srv'.Approvato = “No”

2a. modificaServizio(srv: Servizio, tipo?: Testo, luogo?: Testo, numPartecipanti?: Testo, propagaModifica: Sì/No)

Pre-condizioni:

- È in corso la modifica di un evento ev
- ev prevede srv
- ev.Stato == “Creazione”

Post-condizioni:

- [Se è specificato tipo]
 - o srv.Tipo = tipo
- [Se è specificato luogo]
 - o srv.Luogo = luogo
- [Se è specificato numPartecipanti]
 - o srv.NumeroPersone = numPartecipanti
- [Se propagaModifica == “Sì”]
 - o [se e è un Evento Capofila]
 - Per ogni Occorrenza oc tale che e **composto da** oc
 - modificaServizio è stata applicata, con gli stessi parametri, al servizio **previsto da** oc equivalente ad srv
 - o [altrimenti]
 - Se ev è l'Evento Capofila tale che ev **composto da** e
 - Per ogni Occorrenza oc tale che ev **composto da** oc

- modificaServizio è stata applicata, con gli stessi parametri, al servizio **previsto da** oc equivalente ad srv

2b. eliminaServizio(srv: Servizio, propMod: Sì/No)

Pre-condizioni:

- ev **prevede** srv
- ev.Stato = “Creazione”

Post-condizioni:

- [Se m è il Menù **in uso in** srv]
 - l’associazione tra m ed srv è stata eliminata
- L’associazione tra srv ed ev è stata eliminata
- Per ogni proposta di modifica m tale che m **menù di** srv
 - l’associazione tra m ed srv è stata eliminata
 - l’associazione tra m e Ricetta è stata eliminata
 - m è stata eliminata
- [Se f è il foglio riepilogativo che **contiene compiti** di srv]
 - l’associazione tra f ed srv è stata eliminata
 - Per ogni compito c tale che f **composto da** c
 - l’associazione tra c e turno è stata eliminata
 - l’associazione tra c e Cuoco è stata eliminata
 - [Se c’ è **prosecuzione di** c]
 - c’ è stato eliminato, così come ogni altro compito sua prosecuzione, diretta od indiretta
 - c è stato eliminato
 - l’associazione tra Chef ed f è stata eliminata
 - f è stato eliminato
- Per ogni Disponibilità d **in** srv
 - l’associazione tra d ed srv è stata eliminata
 - d.Prenotato = “No”
 - d.Ruolo = “”
- [Se propMod = “Sì”]
 - [Se ev è un EventoCapofila]
 - Per ogni Occorrenza oc tale che ev **composto da** oc
 - eliminaServizio è stata applicata anche al servizio di oc avente le stesse caratteristiche
 - [altrimenti]
 - Se ev’ è l’Evento Capofila tale che ev’ **composto da** ev]
 - Per ogni Occorrenza oc tale che ev’ **composto da** oc
 - eliminaServizio è stata applicata anche al servizio di oc avente le stesse caratteristiche
 - ev.DataFine = massima data tra quelle dei servizi associati
 - ev.DataFine = minima data tra quelle dei servizi associati

Passo 3

3. assegnazioneChef(chef: Chef)

Pre-condizioni:

- è in corso la modifica di un Evento *ev*
- *ev*.Stato == “Creazione”
- Non esiste un Evento *ev'*, tale che
 - o *Ev'* **assegnato a** *chef*
 - o *Ev'*.DataInizio >= *ev*.DataInizio ed *ev'*.DataFine <= *ev*.DataFine

Post-condizioni:

- *ev* **assegnato a** *chef*

Passo 4

4. prenotazionePersonale(*disp*: Disponibilità, *srv*: Servizio)

Pre-condizioni:

- *ev*.Stato == “Creazione”
- *disp*.Prenotato == “No”
- è in corso la modifica di un evento *ev*
- *ev* **prevede** *srv*

Post-condizioni:

- *disp*.Prenotato = “Sì”
- Disp **in** *srv*
- or **prenotazione** *disp*

4a. cancellazionePrenotazione(*disp*: Disponibilità)

Pre-condizioni

- è in corso la modifica di un evento *ev*
- Or **prenotazione** *disp*
- *disp*.prenotato == “Sì”
- *ev*.stato == “Creazione”

Post-condizioni:

- *disp*.prenotato = “No”
- *disp*.ruolo = “”
- *disp* non è più associata a *srv*

Passo 5

5. approvazioneMenu(*srv*: Servizio)

Pre-condizioni:

- *Ev* **prevede** *srv*
- *ev*.Stato == “Creazione”

Post-condizioni:

- *srv*.approvato = “Sì”

- ev.stato = “In corso”

5a. propostaModifica(srv: Servizio, ric: Ricetta, cm: Commento)

Pre-condizioni:

- è in corso la modifica di un evento ev
- ev **prevede** srv
- ev.stato == “Creazione”

Post-condizioni:

- È stata creata una PropostaModifica pm
- Pm.commento = cm
- Pm.accettata = “No”
- Pm **menù di** srv
- Pm **referita a** ric

Passo 6

6. assegnazioneRuolo(disp: Disponibilità, ruolo: Testo)

Pre-condizioni:

- Or **prenotazione** disp
- Se mp **inserimento** disp, allora mp è un Membro del personale di servizio
- è in corso la modifica di un evento ev

Post-condizioni:

- disp.Ruolo = ruolo

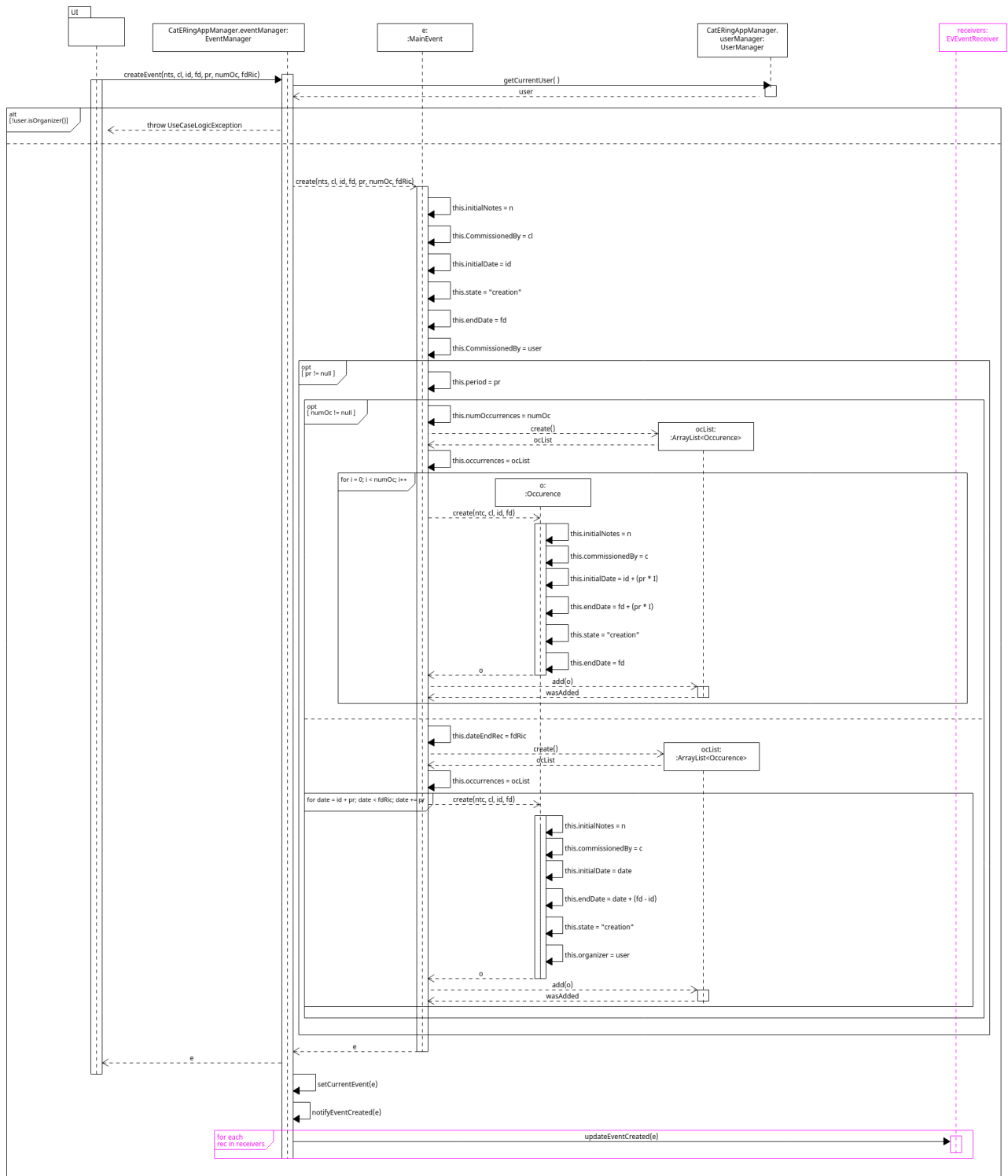
Note:

- Per definire la semantica delle creazione dell'evento ricorrente, si è assunto che il periodo sia in giorni. È impreciso, perchè potrebbe non esserlo, ma rende l'idea di cosa dovrebbe succedere

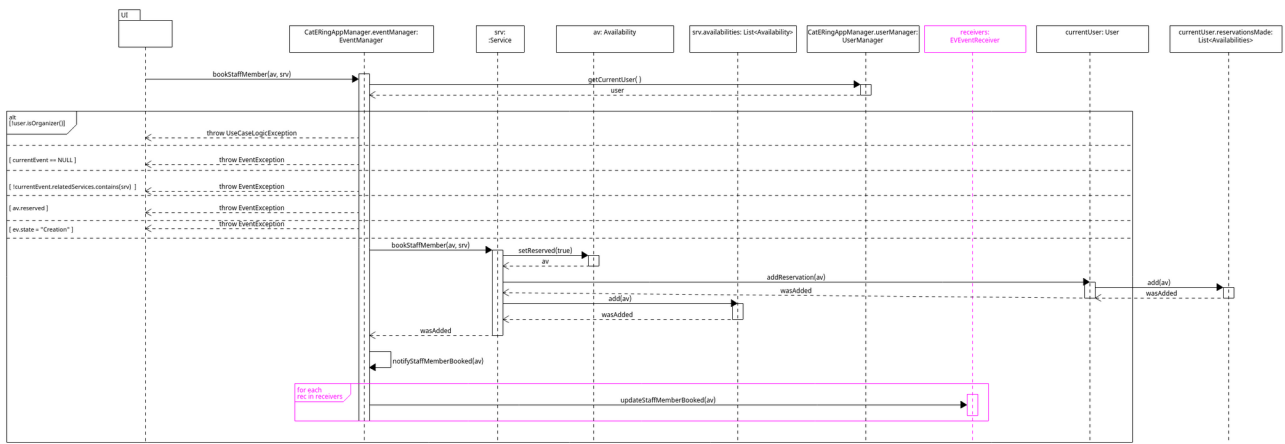
Progettazione:

Design Class Diagram

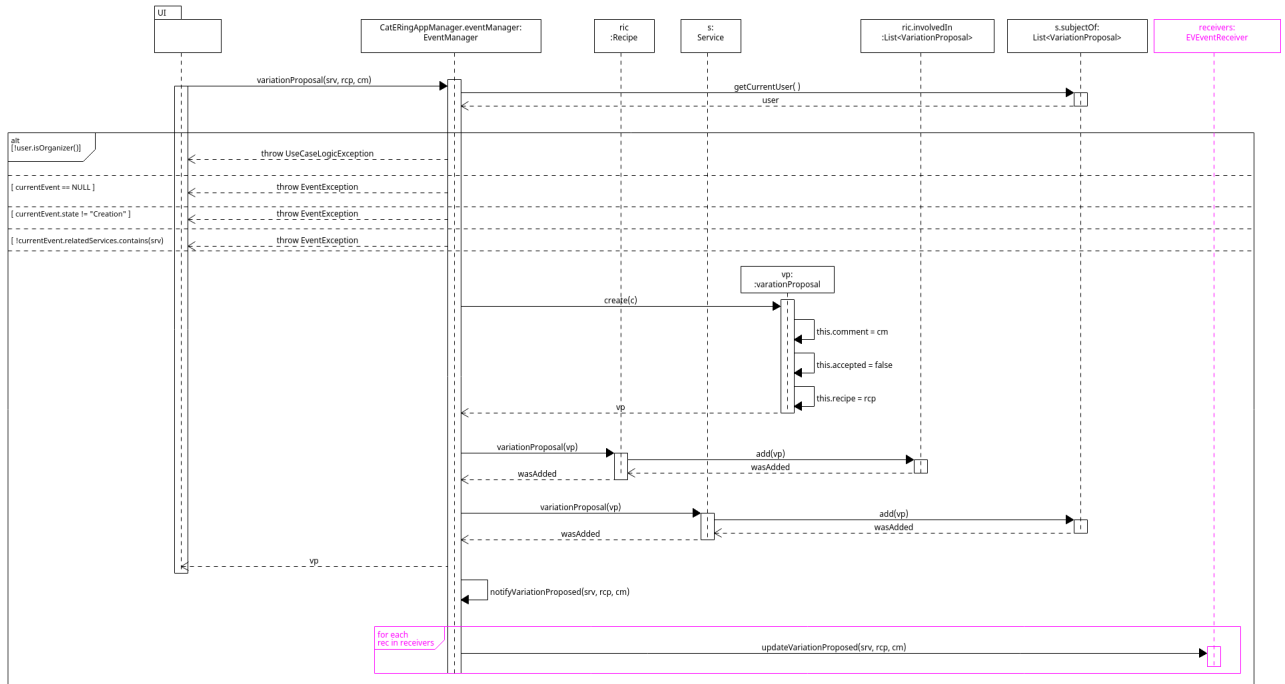
Operazione 1



Operazione 2



Operazione 5, estensione b



Operazione 6

