

Nome : Mickael

Cognome : Ordine

Matricola : 926927

Progetto S.A.S 22/23

Gestione degli Eventi

Informazioni generali

Nome caso d'uso: Gestione degli Eventi

Portata: sistema

Livello: Obbiettivo Utente

Attore primario: Organizzatore

Parti Interessate: Organizzatore, Chef e dipendenti

Pre-condizioni: l'attore deve essere autenticato come organizzatore

Garanzie di successo o post-condizioni: L'evento viene organizzato assieme ai suoi servizi, viene scelto lo chef, i cuochi e il personale.

Scenario principale di successo

#	Attore	Sistema
1	Verifica dello stato dei turni relativi ai giorni dell'evento	Viene fornito il calendario
2	Creazione evento specificando la data d'inizio, il nome, il luogo, durata e il numero di partecipanti	Registra l'inizio della creazione di un nuovo evento con il nome specificato e tutti i dettagli
3	Opzionalmente effettua la compilazione dei servizi associati all'evento	Aggiunge i servizi all'evento
	<i>Ripete il passo 3 finchè non è soddisfatto</i>	
4	Opzionalmente si sceglie lo chef	Assegna uno chef allo specifico evento
	<i>Se soddisfatto continua al passo 5, altrimenti torna al passo 3</i>	
5	Opzionalmente si Approva il Menu associato al servizio	Viene approvato il menu al relativo servizio
	<i>Se soddisfatto prosegue al passo 6, altrimenti torna al passo 3</i>	
6	Compilazione del personale	Il personale viene assegnato all'evento attraverso i turni
	<i>Ripete il passo 6 finchè non è soddisfatto</i>	
7	Opzionalmente viene Aggiunta una nota sul servizio o evento	Verrà aggiunta una nota
	<i>Se non è soddisfatto torna al passo 3 altrimenti Termina il caso d'uso</i>	

Estensione 2a

#	Attore	Sistema
2.a	Eliminazione dell'evento	L'evento viene eliminato

Estensione 2b

#	Attore	Sistema
2.b.1	annullamento dell'evento	L'evento viene segnato come annullato

Estensione 2c

#	Attore	Sistema
2.c.1	Termine dell'evento	L'evento viene segnato come terminato

Estensione 2d

#	Attore	Sistema
2.d.1	Apertura di un evento esistente	L'evento selezionato viene aperto

Estensione 2e

#	Attore	Sistema
2.e.1	modifica dell'evento	L'evento viene modificato coi nuovi parametri

Estensione 3a

#	Attore	Sistema
3.a.1	Modifica di un servizio	Il servizio viene modificato

Estensione 3b

#	Attore	Sistema
3.b.1	eliminazione di un servizio	Il servizio viene eliminato, liberando i turni associati

Estensione 5a

#	Attore	Sistema
5.a.1	Proposta di modifica del Menu	Registra presso il menù del servizio eventuali modifiche

Estensione 6a

#	Attore	Sistema
6.a.1	Modifica Assegnamento del personale	L'incarico viene modificato

Estensione 6b

#	Attore	Sistema
6.b.1	Elimina Assegnamento del personale	L'incarico viene eliminato

Estensione 7a

#	Attore	Sistema
7.a.1	Modifica nota	La nota viene modificata

Estensione 7b

#	Attore	Sistema
7.b.1	Eliminazione nota	La nota viene eliminata

Eccezione 2

#	Attore	Sistema
2.a.1	Creazione evento	La data d'inizio specificata è già passata
	<i>Termina il caso d'uso</i>	

Eccezione 2a

#	Attore	Sistema
2.a.1.a	Eliminazione di un evento	L'evento è già terminato
	<i>Termina il caso d'uso</i>	

Eccezione 2a

#	Attore	Sistema
2.a.1.b	Eliminazione di un evento	L'evento è in corso
	<i>Termina il caso d'uso</i>	

Eccezione 2a

#	Attore	Sistema
2.a.1.c	Eliminazione di un evento	L'organizzatore non gestisce l'evento
	<i>Termina il caso d'uso</i>	

Eccezione 2b

#	Attore	Sistema
2.b.1.a	Annullamento di un evento	L'evento è già terminato/annullato/eliminato
	<i>Termina il caso d'uso</i>	

Eccezione 2c

#	Attore	Sistema
2.c.1.a	Terminazione di un evento	L'evento è già terminato/annullato/eliminato
	<i>Termina il caso d'uso</i>	

Eccezione 2d

#	Attore	Sistema
2.d.1.a	Apertura di un evento esistente	L'organizzatore non gestisce l'evento indicato
	<i>Termina il caso d'uso</i>	

Eccezione 2e.1

#	Attore	Sistema
2.e.1.a	modifica dell'evento cambiando il la data d'inizio	l'evento è già stato confermato quindi i giorni non sono più modificabili

Eccezione 2e.1

#	Attore	Sistema
2.e.1.b	modifica dell'evento cambiando il numero dei partecipanti	l'evento è già stato confermato e viene modificato il numero di partecipanti viene imposta una penale se il nuovo numero di partecipanti ha sfiorato del 30% rispetto al valore di partenza

Eccezione 2e.1

#	Attore	Sistema
2.e.1.c	modifica dell'evento cambiando il la data d'inizio	La nuova data di inizio è precedente a quella attuale

Eccezione 3

#	Attore	Sistema
3.a.1	Creazione di un servizio	La data di inizio deve essere maggiore di quella attuale

Eccezione 3

#	Attore	Sistema
3.a.2	Creazione di un servizio	La data di inizio deve essere maggiore di quella dell'evento

Eccezione 3a

#	Attore	Sistema
3.a.1.a	Modifica un servizio	Il servizio è già terminato

Eccezione 3a

#	Attore	Sistema
3.a.1.b	Modifica un servizio in cui si inserisce un nuovo orario	La data inserita è già passata

Eccezione 3b

#	Attore	Sistema
3.b.1.a	Elimina un servizio	Il servizio è già terminato

Eccezione 5

#	Attore	Sistema
5.a.1	Approvazione Menu	Il servizio è già terminato

Eccezione 5.a

#	Attore	Sistema
5.a.1.a	Proposta modifica Menu	Il servizio è già terminato

Eccezione 6

#	Attore	Sistema
6.a.1	Assegnamento del personale	Il turno è già terminato

Eccezione 6

#	Attore	Sistema
6.a.2	Assegnamento del personale	Il personale non è disponibile in quel turno

Eccezione 6

#	Attore	Sistema
6.a.3	Assegnamento del personale	Il servizio è già segnato come terminato

Eccezione 6a.1

#	Attore	Sistema
6.a.1.a	Modifica del personale	Il turno è già terminato

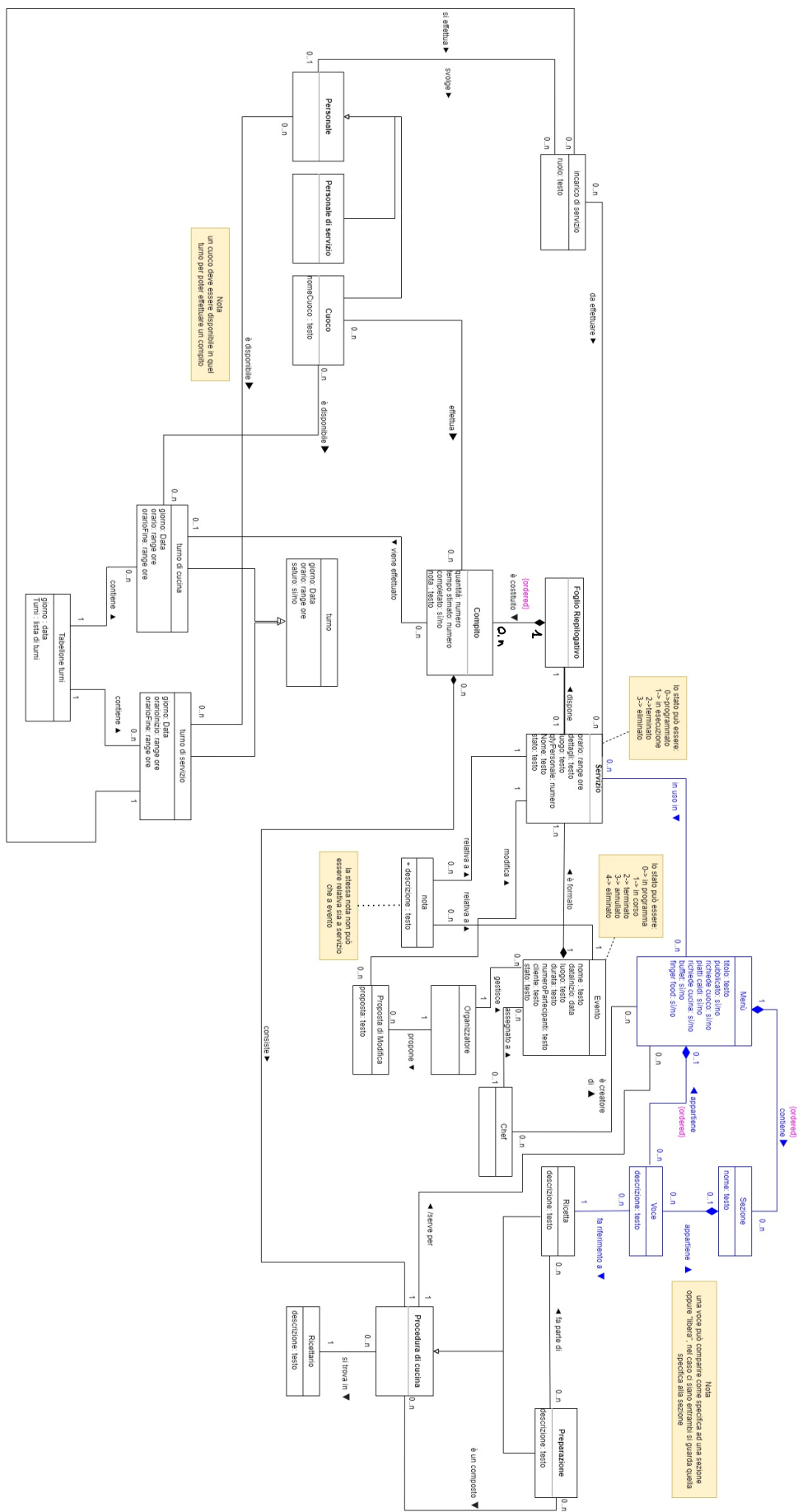
Eccezione 6a.1

#	Attore	Sistema
6.a.1.b	Modifica del personale	Il personale non è disponibile in quel turno

Eccezione 6b

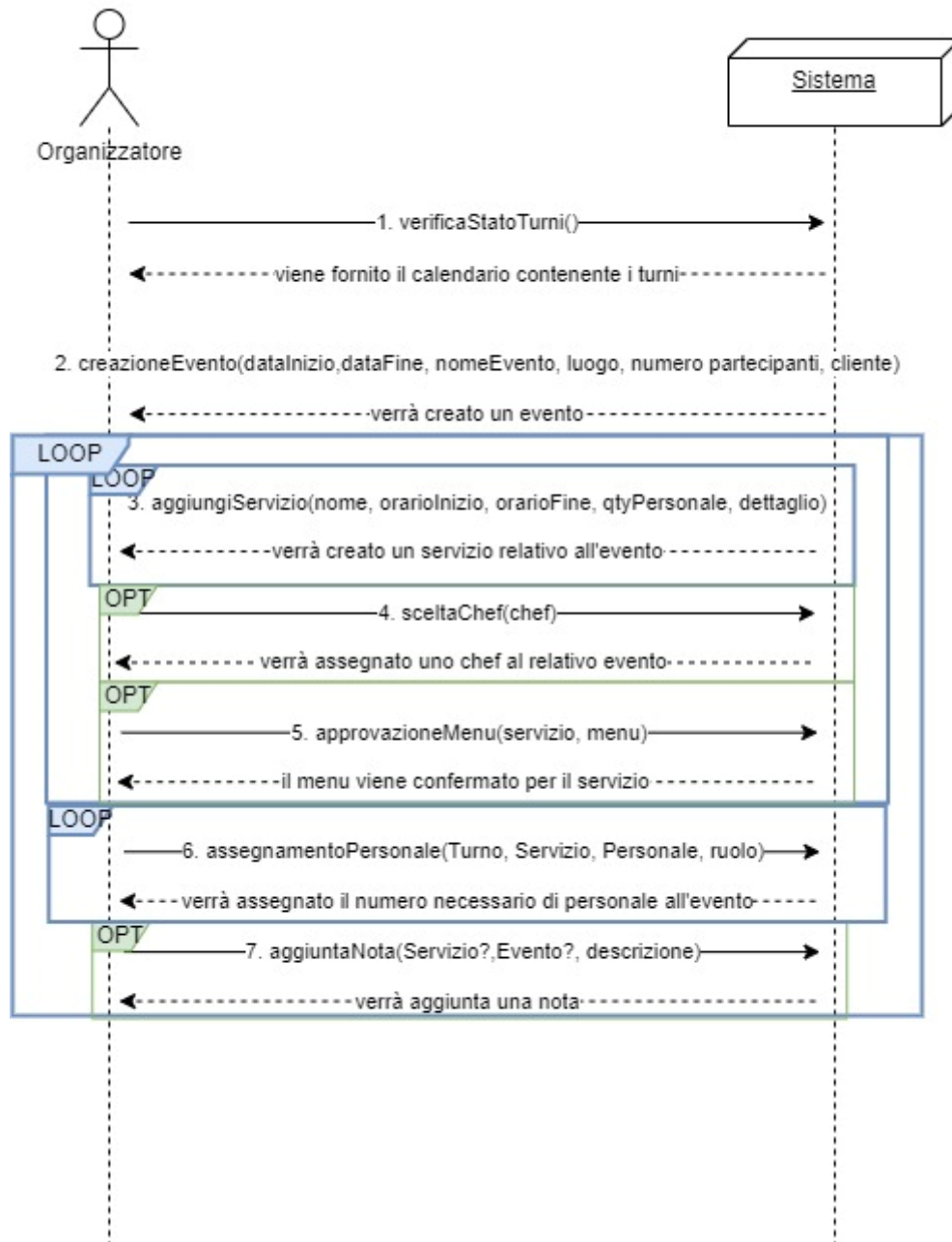
#	Attore	Sistema
6.b.1.a	Eliminazione del personale	Il turno relativo all'incarico è già terminato

Modello di dominio

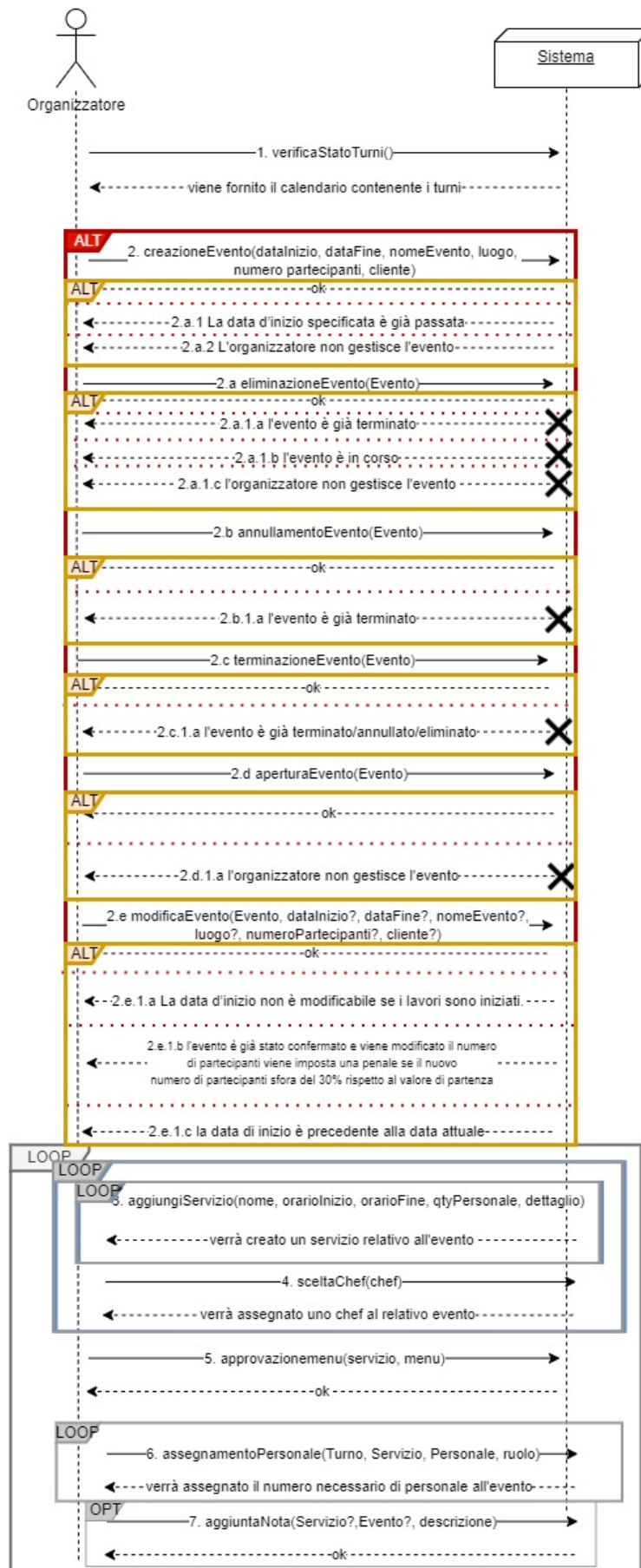


Diagrammi di sequenza di sistema per l'UC Gestire Eventi

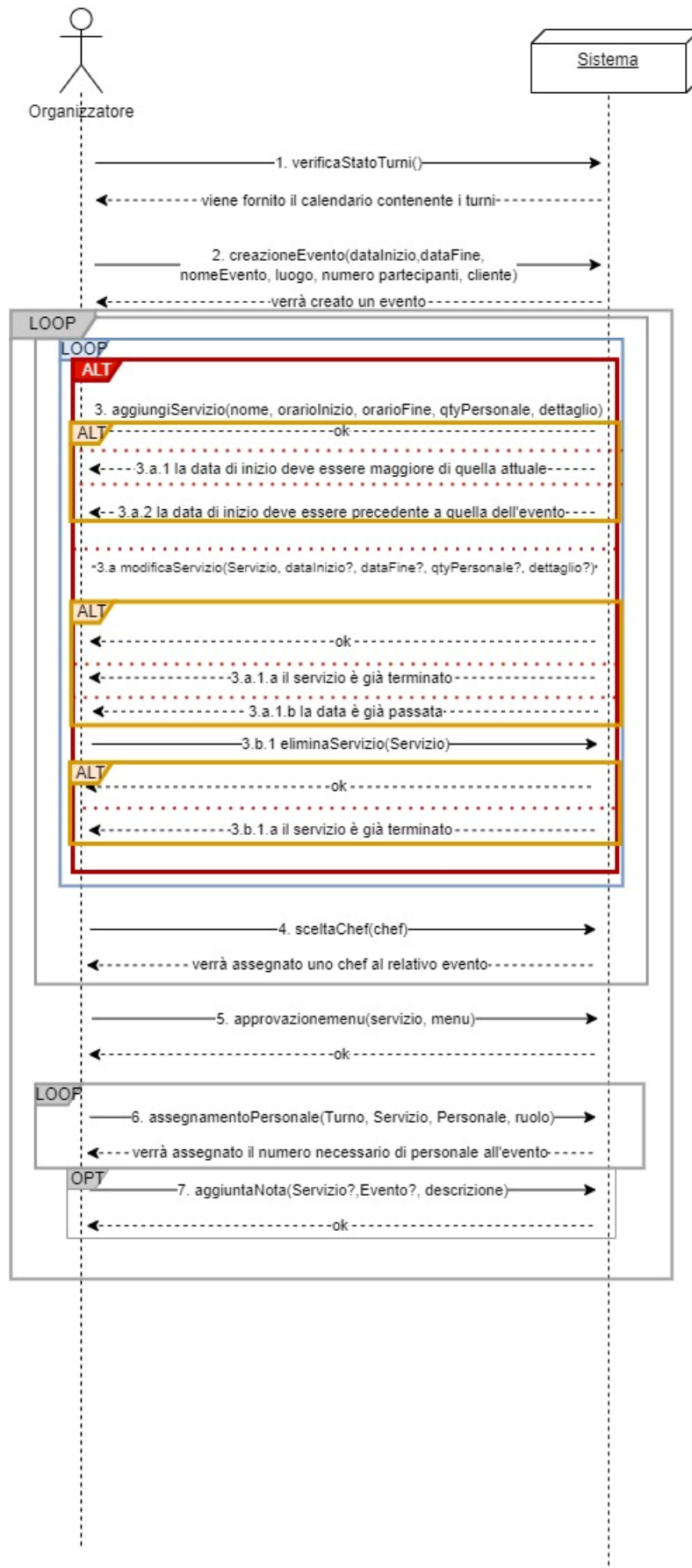
scenario principale di successo



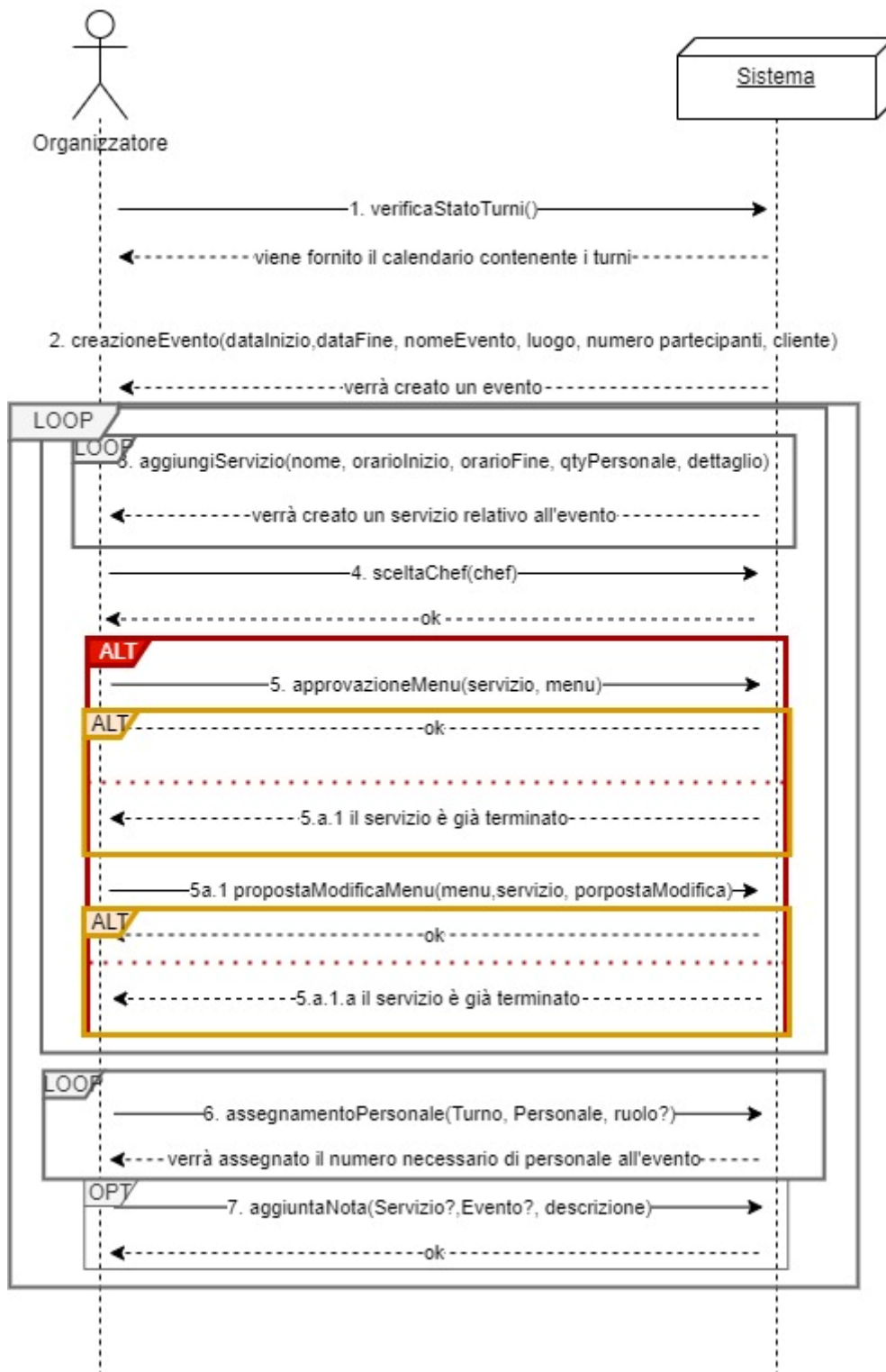
estensione passo 2 (2a,2b,2c,2d,2e)
con relative eccezioni (2a1,2a1,2a1a,2a1b,2a1c,2b1a,2c1a,2d1a,2e1a,2e1b,2e1c)



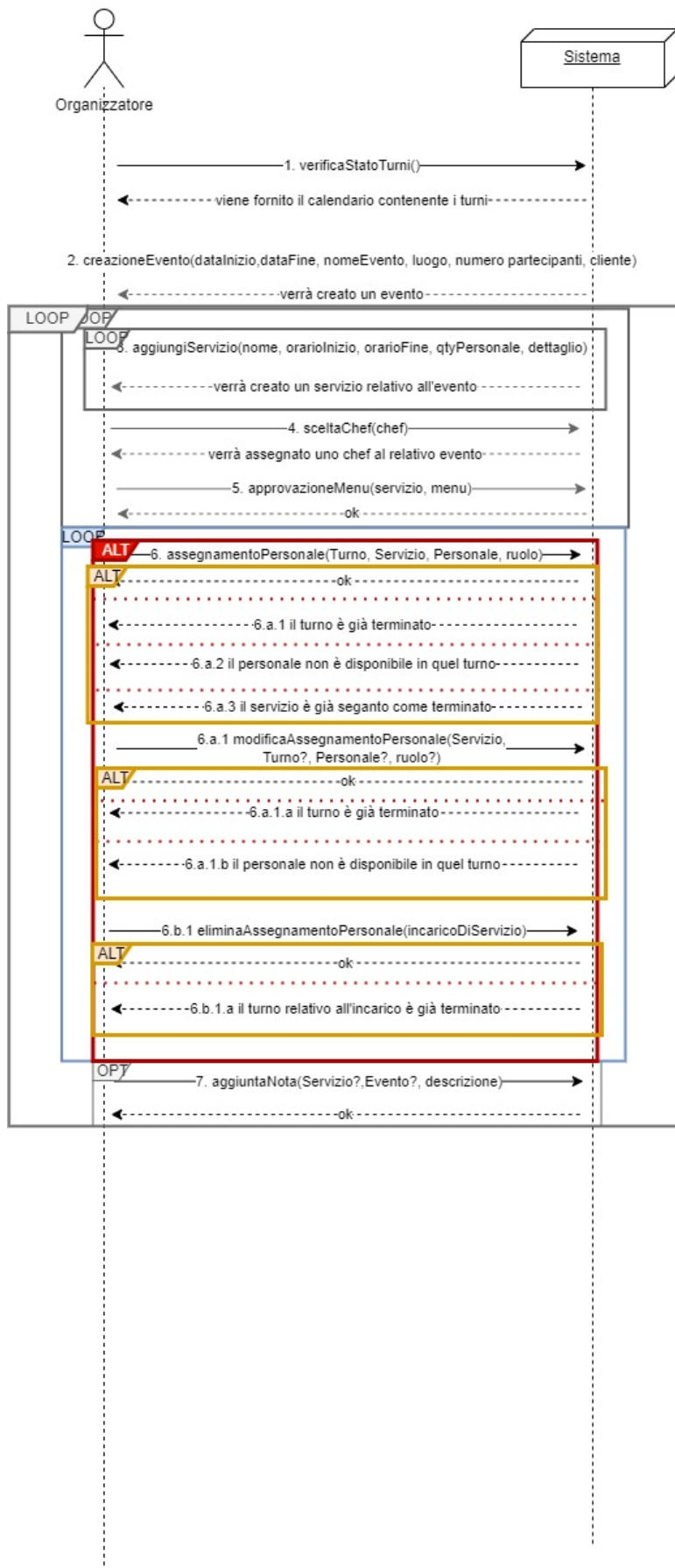
estensione passo 3 (3a,3b) con relative eccezioni (3a1,3a2,3a1a,3a1b,3b1a)



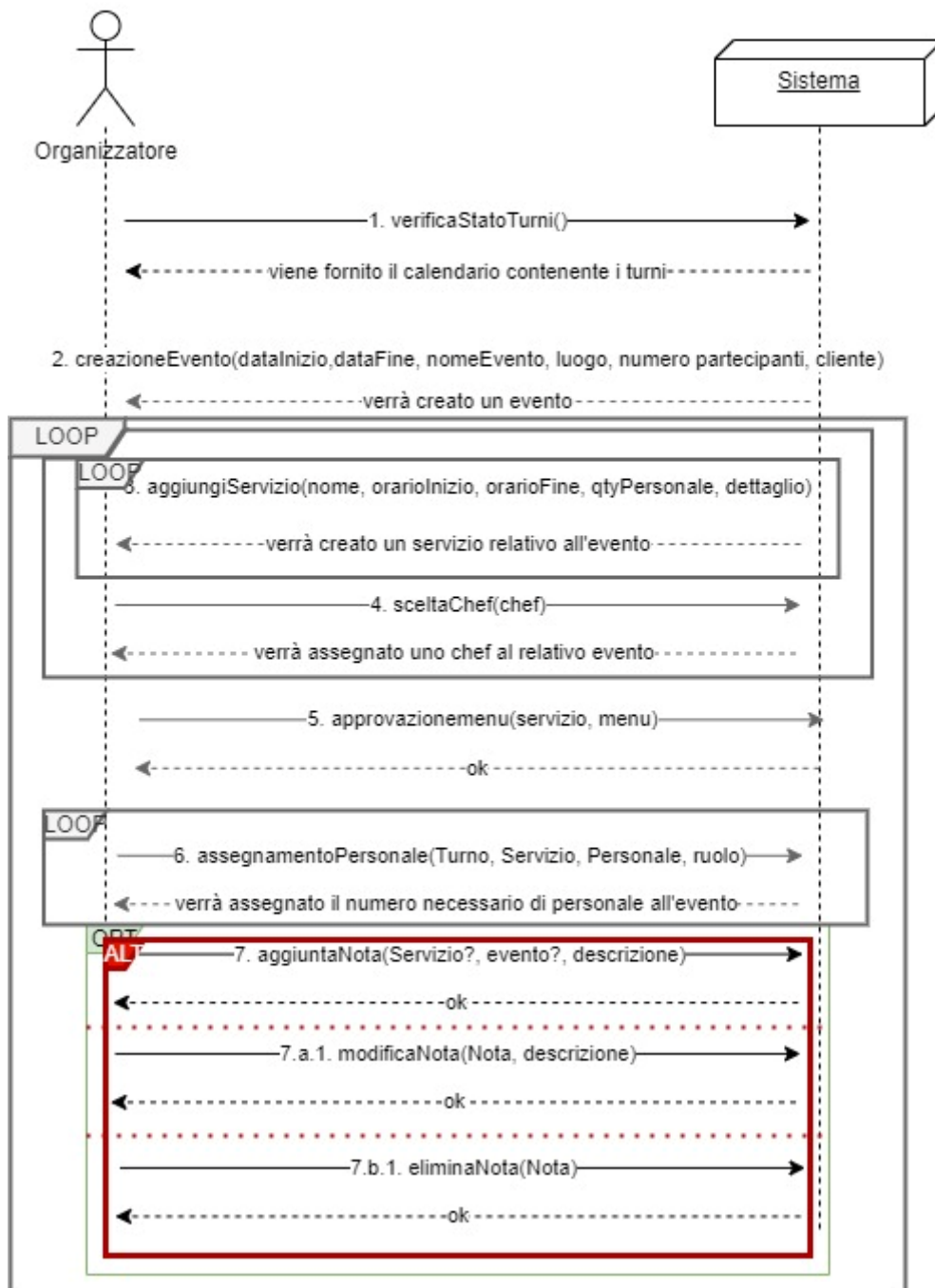
estensioni del passo 5 (5a) con relative eccezioni 5a,5a1a



estensioni del passo 6 (6a,6b) con relative eccezioni (6a1,6a2,6a3,6a1a,6a1b)



estensioni del passo 7 (7a,7b)



Contratti per lo UC “Gestire gli Eventi”

- Pre-condizioni generali: l'attore deve essere identificato con un'istanza *org* di organizzatore

1. VerificaStatoTurni()

Pre-condizioni: -

Post-condizioni: -

2. creazioneEvento(dataInizio: testo, dataFine: testo, nomeEvento: testo, luogo: testo, numeroPartecipanti: intero, cliente: testo)

Pre-condizioni:

- la data specificata non è precedente a quella in cui sta creando l'evento *evento*
- l'organizzatore *org* **gestisce** l'evento

Post-condizioni:

- viene creata un istanza evento *evento*
- *evento*.stato = “in programma”
- *evento*.nome = nomeEvento
- *evento*.luogo = luogo
- *evento*.dataInizio = dataInizio
- *evento*.dataFine = dataFine
- *evento*.numeroPartecipanti = numeroPartecipanti
- *evento*.cliente = cliente

2.a eliminazioneEvento(Evento: evento)

Pre-condizioni:

- l'evento non è in stato terminato o in corso
- l'organizzatore **gestisce** l'evento

Post-condizioni:

- evento.stato = eliminato
- per ogni servizio *s* che **formava** l'evento
 - tutti i compiti *c* che **costituivano** il foglioripetilogativo *f di s* sono state eliminate
 - tutte le istanze di incaricodiservizio *incServizio* da **effettuare** in *s* sono state eliminate
- L'associazione **formava** tra *s* e evento è stata eliminata
- il *personale* e lo *chef* sono tronati disponibili in quei turni
- è stata eliminata l'istanza evento specificata

2.b annullamentoEvento(Evento: evento)

Pre-condizioni:

- l'evento non è in stato terminato

Post-condizioni:

- evento.stato = annullato
- per ogni servizio *s* che **formava** l'evento
 - tutti i compiti *c* che **costituivano** il foglioripetilogativo *f di s* sono state eliminate
 - tutte le istanze di incaricodiservizio *incServizio* da **effettuare** in *s* sono state eliminate
 - L'associazione **formava** tra *s* e evento è stata eliminata
- il *personale* e lo *chef* sono tronati disponibili in quei turni
- è stata eliminata l'istanza evento specificata

2.c TerminazioneEvento(Evento: evento)

Pre-condizioni:

- l'evento non è in stato terminato/annullato/eliminato

Post-condizioni:

- evento.stato = terminato

2.c aperturaEvento(Evento: evento)

Pre-condizioni:

- l'organizzatore *org* gestisce l'evento

Post-condizioni: -

2.c modificaEvento(Evento: evento, dataInizio?: testo, dataFine?: testo, nomeEvento?: testo, luogo?: testo, numeroPartecipanti?: intero, Cliente?: testo)

Pre-condizioni:

- [se inserita la data di inizio] allora evento.stato != in esecuzione
- [se inserita data di inizio] allora dataInizio > evento.dataInizio
- [se inserita dataFine] allora dataFine > evento.dataInizio
- [se inserito un numero di partecipanti] se evento.stato == confermato, allora il numero di partecipanti nuovi sono deve sfiorare del 30% rispetto al numero di partenza, altrimenti mora.

Post-condizioni:

- [se viene inserita la dataInizio] evento.dataInizio = dataInizio
- [se viene inserita la dataFine] evento. dataFine = durata
- [se viene inserita il nomeEvento] evento. nomeEvento = nomeEvento
- [se viene inserita il luogo] evento. luogo = luogo
- [se viene inserita il numeroPartecipanti] evento.numeroPartecipanti = numeroPartecipanti
- [se viene inserita il cliente] evento. cliente = cliente

3. aggiungiServizio(nome: testo ,orarioInizio: orario, orarioFine: orario, qtyPersonale: qtyPersonale, dettagli: testo)

Pre-condizioni:

- È in corso la definizione dell'evento *evento*
- L'orario di inizio deve essere maggiore di quella *dell'evento*
- L'orario di inizio deve essere maggiore di quella attuale

Post-condizioni:

- viene creata un istanza di servizio *s*
- *s.orario* = orario
- *s.qtyPersonale* = qtyPersonale
- *s.dettaglio* = dettaglio
- *s.stato* = programmato
- *s forma evento*

3.a modificaServizio(Servizio: servizio, orarioInizio?: testo, orarioFine?: testo, qtyPersonale?: qtyPersonale, dettaglio?: testo)

Pre-condizioni:

- è in corso la definizione di un *evento formato* da servizio s
- il servizio non deve essere già segnato come terminato
- [se inserito l'orarioInizio] l'orarioInizio non deve essere già passato
- [se inserito l'orarioFine] l'orarioFine non deve essere precedente a quello di inizio

Post-condizioni:

- [se inserito l'orarioInizio] *s. orarioInizio* = orarioInizio
- [se inserito l'orarioFine] *s. orarioFine* = orarioFine
- [se inserito il qtyPersonale] *s. qtyPersonale* = qtyPersonale
- [se inserito dettaglio] *s.dettaglio* = dettaglio

3.b eliminaServizio(Servizio: servizio)

Pre-condizioni:

- il servizio non deve essere già segnato come terminato
- è in corso la definizione di un *evento formato* da servizio s

Post-condizioni:

- l'associazione **forma** tra servizio e *evento* è stata eliminata
- per ogni incarico di servizio *inc* da **svolgere** nel servizio è stata tolta l'associazione
- ogni compito *c* che ne **costituisce** il foglio riepilogativo *fp* del servizio, è stata eliminata l'associazione

4. sceltaChef(Chef: chef)

Pre-condizioni:

- è in corso la definizione di un *evento*

Post-condizioni:

- lo chef è **stato assegnato** al relativo *evento*

5. approvazioneMenu(Servizio: servizio, Menu: menu)

Pre-condizioni:

- è in corso la definizione del servizio di cui è **formato** *l'evento*
- il servizio.stato non è terminato o annullato

Post-condizioni:

- *l'evento*.stato = "in corso"
- il menu è diventato **in uso** per il servizio

5.a propostaModificaMenu(menu: menu, servizio: servizio, propostaModifica: propostaDiModifica)

Pre-condizioni:

- è in corso la definizione del servizio di cui è **formato** *l'evento*
- menu è **in uso** nel servizio
- il servizio non è in stato annullato o terminato

Post-condizioni:

- *l'evento*.stato = "in corso"
- proposta di modifica **modifica** il Menu **in uso** nel servizio

6. assegnamentoPersonale(Turno: turno, Servizio: servizio, Personale: personale, ruolo: ruolo)

Pre-condizioni:

- è in corso la definizione del servizio di cui è **formato** *l'evento*
- il turno specificato non deve già essere terminato
- il turno specificato non deve essere saturo
- il servizio specificato non deve già essere terminato
- il personale si è reso **disponibile** in quel turno

Post-condizioni:

- è stata creata un'istanza di *incaricoDiServizio* *incServizio*
- *l'incServizio* è da **effettuare** nel servizio
- *l'incServizio* si **effettua** nel turno specificato
- il personale specificato **svolge** *l'incServizio*

- `incServizio.ruolo = ruolo`

6.a modificaAssegnamentoPersonale(Servizio: Servizio, Turno?: turno, Personale?: personale, ruolo?: ruolo)

Pre-condizioni:

- è in corso la definizione del servizio di cui è **formato** *l'evento*
- [se specificato un turno] il turno specificato non deve già essere terminato
- [se specificato turno] il turno non deve essere saturo

Post-condizioni:

- È stato modificato l'incaricoDiServizio *incServizio* con i nuovi valori
- [se specificato il ruolo] `incServizio.ruolo = ruolo`
- [se specificato il turno] *l'incServizio* si **effettua** nel turno
- [se specificato il personale] *l'incServizio* viene **svolto** dal personale

6.b eliminaAssegnamentoPersonale(IncaricoDiServizio: incaricoDiServizio)

Pre-condizioni:

- è in corso la definizione del servizio *servizio* in cui è **da effettuare** l'incaricoDiServizio
- *Servizio.stato* != terminato

Post-condizioni:

- È stata eliminata l'associazione **svolge** tra personale *personale* e incarico di servizio
- È stata eliminata l'associazione da **effettuare** tra incarico di servizio e *servizio*
- È stato eliminato l'incarico di servizio

7. aggiungiNota (Servizio?: servizio, Evento?: evento, Descrizione: testo)

pre-condizioni:

- è in corso la visualizzazione di un servizio oppure è in corso la visualizzazione di un evento

post-condizioni:

- [se specificato il servizio] è stata creata un istanza Nota *N* **relativa al** Servizio
- [se specificato l'evento] è stata creata un istanza Nota *N* **relativa all'** Evento

7.a modificaNota(Nota: Nota, Descrizione: testo)

pre-condizioni:

- è in corso la visualizzazione di un *servizio* oppure è in corso la visualizzazione di un *evento*

post-condizioni:

- [se la Nota **era relativa a servizio**] è stata modificata l'istanza Nota **relativa al** *Servizio*

- [se la Nota **era relativa a evento**] è stata modificata l'istanza Nota **relativa a evento**

7.b eliminaNota(Servizio: servizio, Descrizione: testo)

pre-condizioni:

- è in corso la visualizzazione di un *servizio* oppure è in corso la visualizzazione di un *evento*

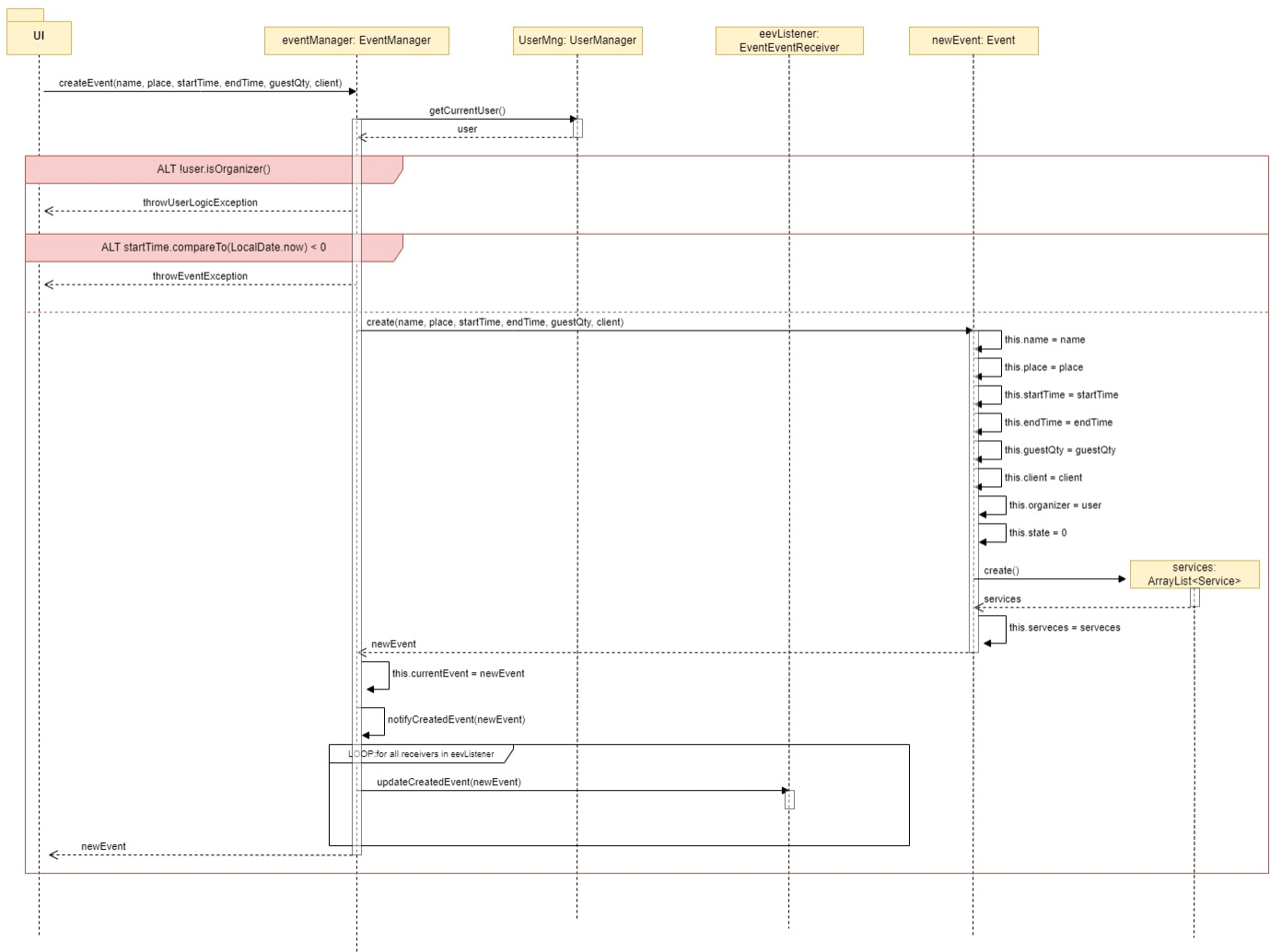
post-condizioni:

- [se la Nota **era relativa a servizio**] è stata eliminata l'istanza Nota **relativa al Servizio**
- [se la Nota **era relativa a evento**] è stata eliminata l'istanza Nota **relativa a evento**

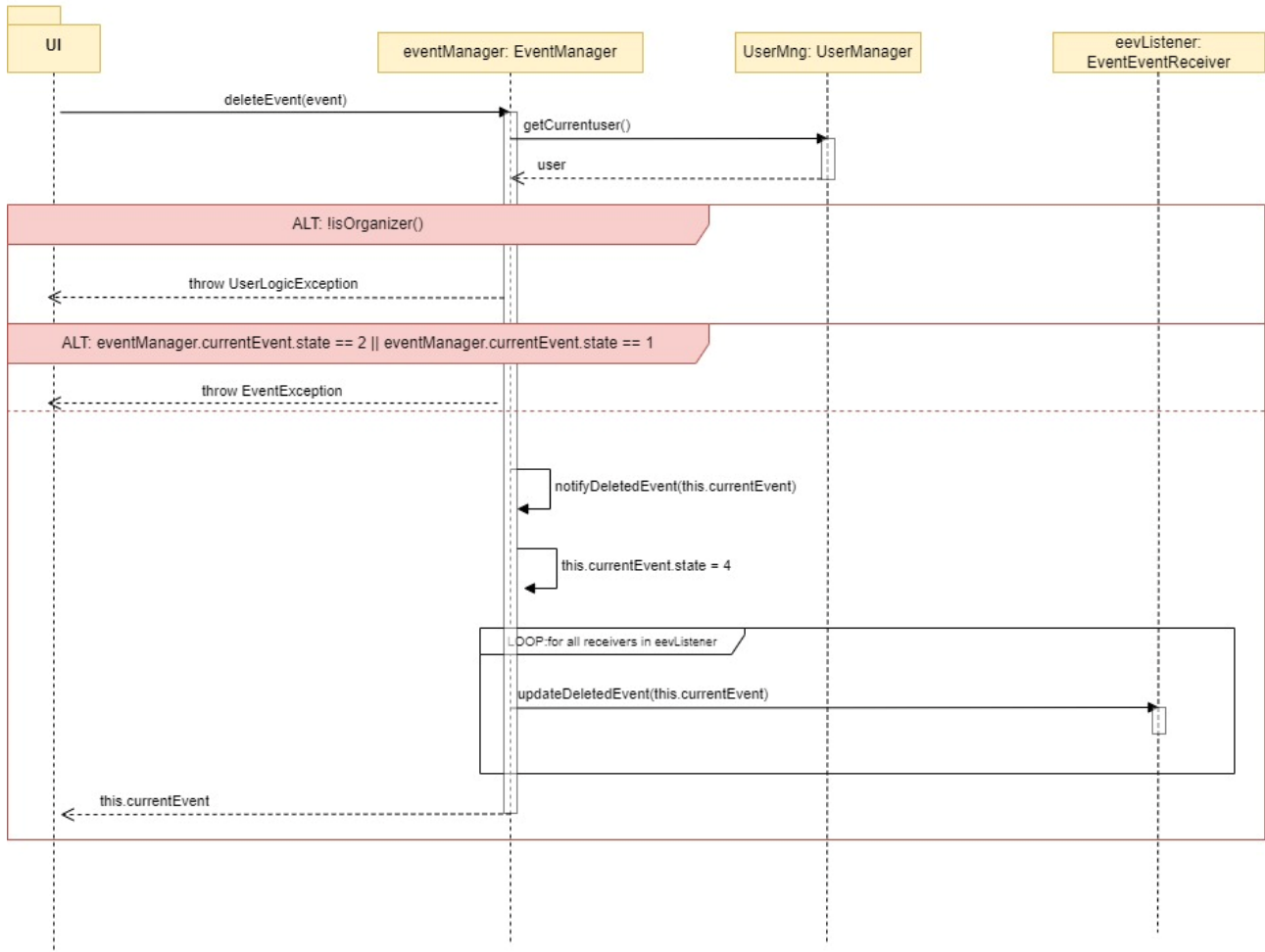
DCD

DSD

2. creazioneEvento(dataInizio : testo, nomeEvento: testo, luogo: testo, durata: testo, numeroPartecipanti: intero, cliente: testo)



2.a eliminazioneEvento(Evento: evento)



3. aggiungiServizio(nome: testo ,orarioInizio: orario, orarioFine: orario, qtyPersonale: qtyPersonale, dettagli: testo)

