

Sviluppo delle Applicazioni Software Laboratorio

a.a 2019/2020

Andrea Maggio

Gestire eventi

informazioni generali

Nome caso d'uso: gestire eventi

Portata: Sistema

Livello: obbiettivo utente

Attore primario: organizzatore

Parti Interessate: organizzatore, chef

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

Garanzie di successo o post-condizioni: l'evento è registrato e disponibile

Scenario principale di successo

#	Attore	Sistema
1	Crea nuovo evento ed eventuali ricorrenze	Crea nuovo evento ed eventuali ricorrenze
2	Annota informazioni evento ed eventuali ricorrenze	Registra informazioni evento, eventualmente anche per le ricorrenze
3	Designa lo chef	Registra chef in evento ed eventualmente nelle ricorrenze
4	Opzionalmente aggiunge servizio	Registra servizio in evento ed eventualmente nelle ricorrenze
5	Opzionalmente controlla informazioni sull'evento	Fornisce informazioni sull'evento
6	Opzionalmente controlla informazioni sul personale	Fornisce informazioni sul personale evento
7	Opzionalmente controlla informazioni del menù	Fornisce informazioni sul menù
	Se necessario ripete da punto 4	
8	Opzionalmente annota note di fine evento	Registra note aggiuntive evento ed eventualmente nelle ricorrenze

Estensione 1a

#	Attore	Sistema
1a.1	Copio evento registrato	Fornisce una copia di un evento

		registrato
--	--	------------

Estensione 1b

#	Attore	Sistema
1b.1	Visualizza evento registrato	Fornisce evento registrato

Estensione 1c

#	Attore	Sistema
1c.1	Elimina evento ed eventualmente le sue estensioni	Elimina evento ed eventualmente anche le ricorrenze
	Termina caso d'uso	

Estensione 4a

#	Attore	Sistema
4a.1	Elimina servizio	Elimina un servizio in evento, ed eventualmente anche nelle sue ricorrenze

Estensione 4b

#	Attore	Sistema
4b.1	Modifica tipologia servizio	Modifica tipologia in evento, ed eventualmente anche nelle sue ricorrenze

Estensione 4c

#	Attore	Sistema
4c.1	Modifica degli invitati	Registra modica il numero di invitati in evento ed eventualmente anche nelle sue ricorrenze

Estensione 4d

#	Attore	Sistema
4d.1	Mostra informazioni servizio	Mostra informazioni su un servizio

Estensione 5a

#	Attore	Sistema
5a.1	Modifica della data	Registra nuova data per evento ed eventualmente per le sue ricorrenze

Estensione 5b

#	Attore	Sistema
5b.1	Modifica della location	Registra nuova location per evento e eventualmente anche per le sue ricorrenze

Estensione 5c

#	Attore	Sistema
5c.1	Modifica del tema	Registra modifica del tema dell'evento ed eventualmente anche per le sue ricorrenze

Estensione 5d

#	Attore	Sistema
5d.1	Aggiungi annotazioni	Registra annotazioni evento, ed eventualmente anche per le sue ricorrenze

Estensione 5e

#	Attore	Sistema
5de.1	Modifica ricorrenza	Registra modifica della ricorrenza per evento e per le sue ricorrenze

Estensione 6a

#	Attore	Sistema
6a.1	Aggiungi personale ad un servizio	Registra nuovo personale per servizio in evento, ed eventualmente anche nelle sue ricorrenze

Estensione 6b

#	Attore	Sistema
6b.1	Rimuovi personale ad un servizio	Cancella personale per il servizio in evento, ed eventualmente anche per le sue ricorrenze

Estensione 7a

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

7a.1	Aggiungi menù a un servizio	Associa un menù ad un servizio in evento, ed eventualmente anche per le sue ricorrenze
-------------	-----------------------------	--

Estensione 7b

#	Attore	Sistema
7b.1	Elimina menù di un servizio	Elimina associazione tra un menù e un servizio in evento, ed eventualmente anche nelle sue ricorrenze

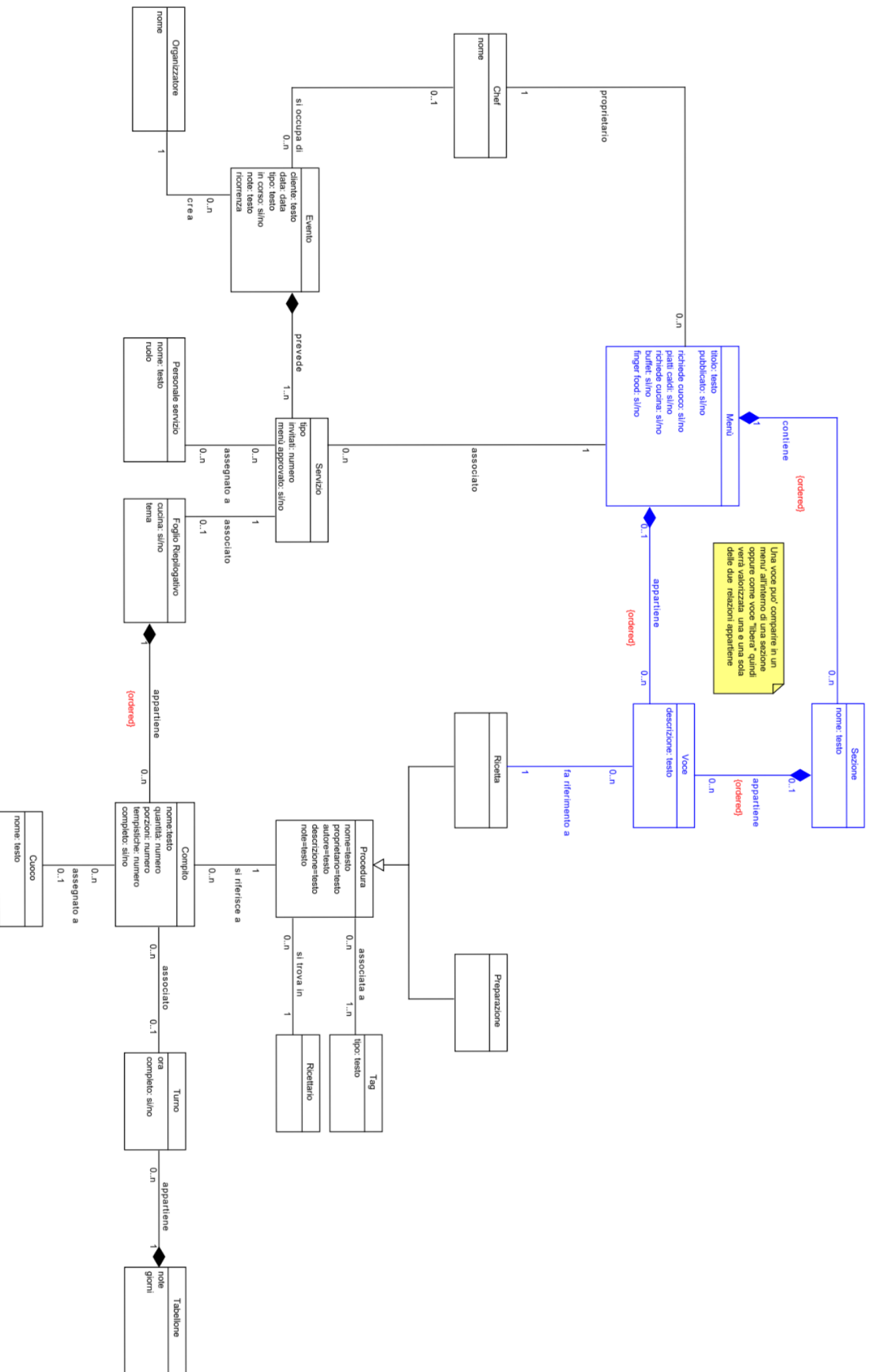
Estensione 7c

#	Attore	Sistema
7c.1	Approva menù	Segna il menù come approvato in evento, ed eventualmente anche nelle sue ricorrenze

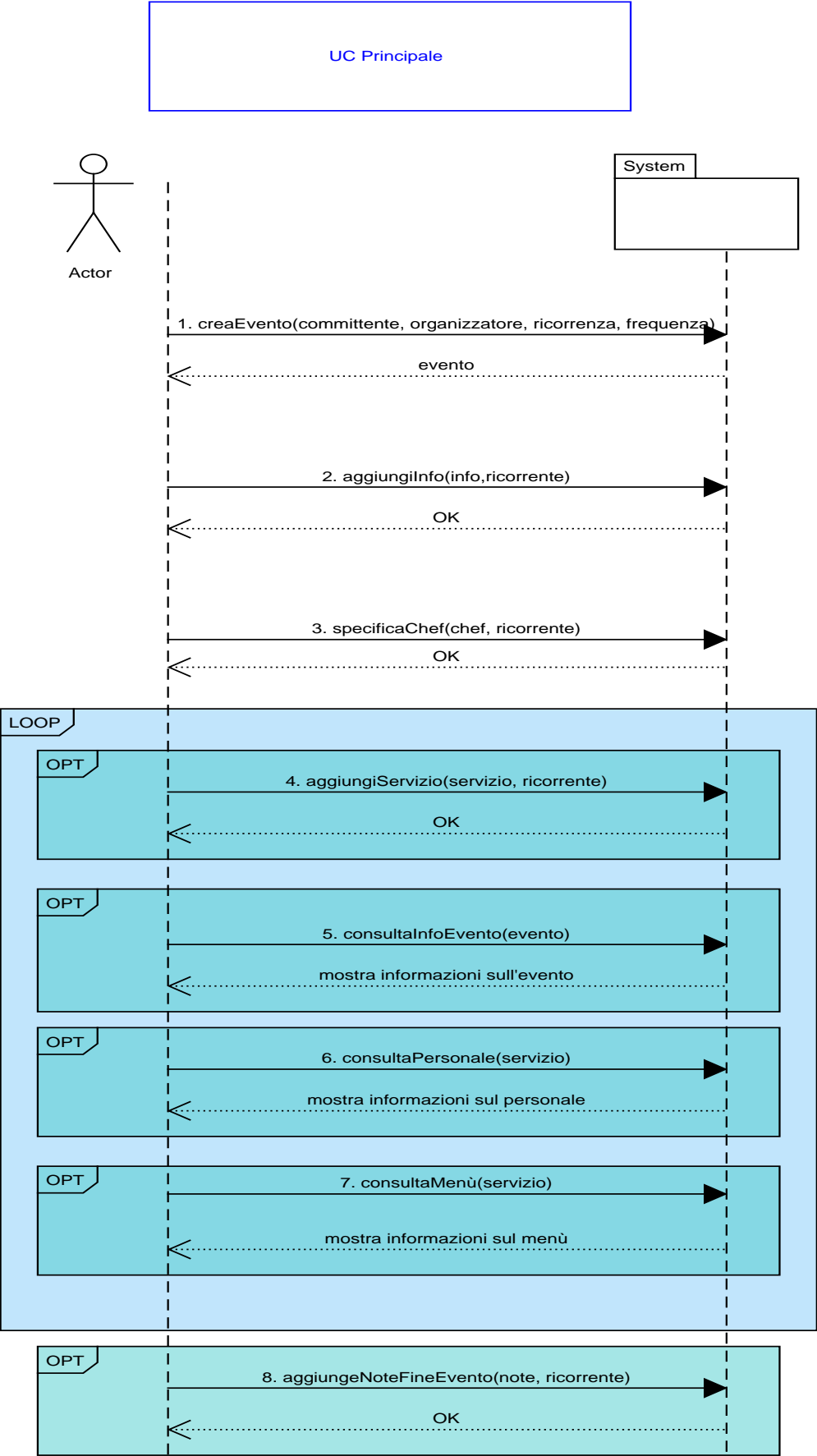
Estensione 8a

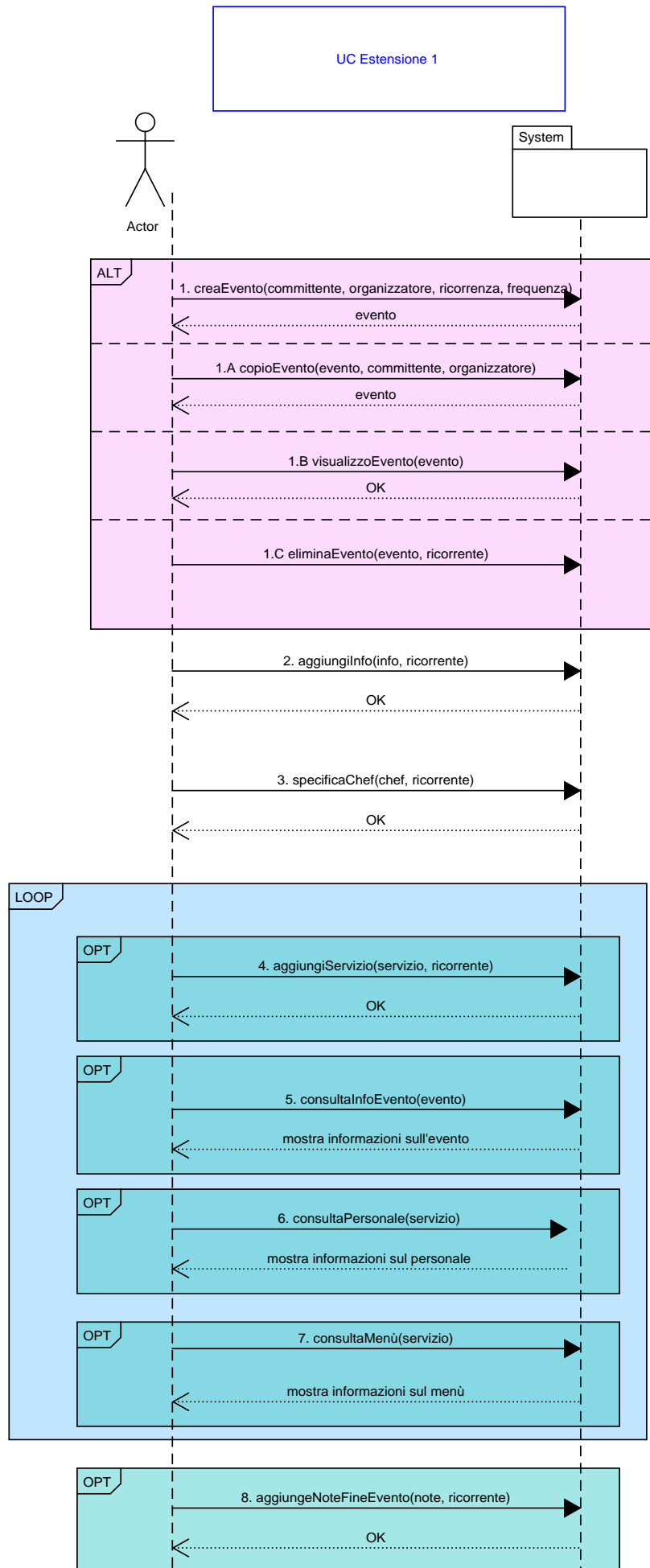
#	Attore	Sistema
8a.1	Annulla evento	Registra un evento come annullato, ed eventualmente anche le sue ricorrenze

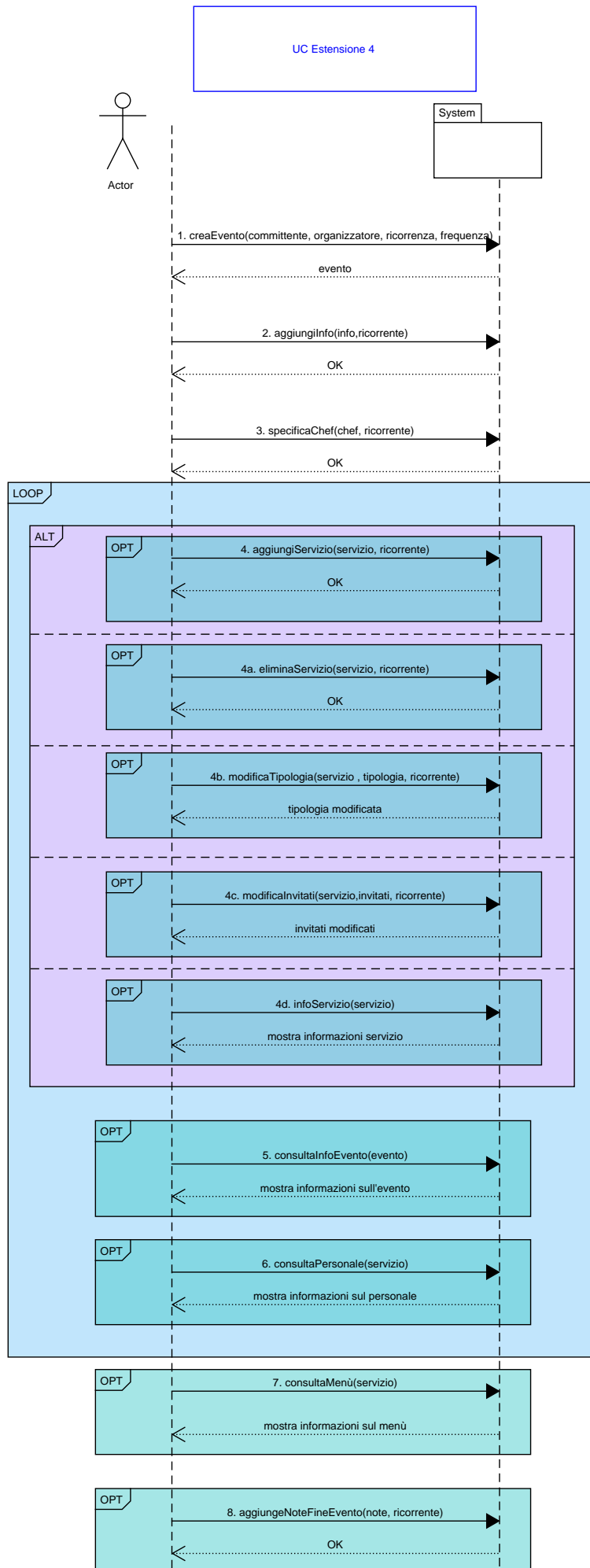
Modello di dominio



Diagrammi di sequenza







UC Estensione 5



Actor

System



1. creaEvento(committente, organizzatore, ricorrenza, frequenza)

evento

2. aggiungiInfo(info,ricorrente)

OK

3. specificaChef(chef, ricorrente)

OK

LOOP

OPT

4. aggiungiServizio(servizio, ricorrente)

OK

ALT

OPT

5. 5. consultaInfoEvento(evento)

mostra informazioni sull'evento

OPT

5A. modificaData(data, ricorrente)

OK

OPT

5B. modificaLocation(location)

OK

OPT

5C.modificaTema(tema, ricorrente)

OK

OPT

5D. aggiungiAnnotazioni(annotazioni, ricorrente)

OK

OPT

5E. modificaRicorrenza(frequenza, ricorrenza, padre)

OK

OPT

6. consultaPersonale(servizio)

mostra informazioni sul personale

OPT

7. consultaMenù(servizio)

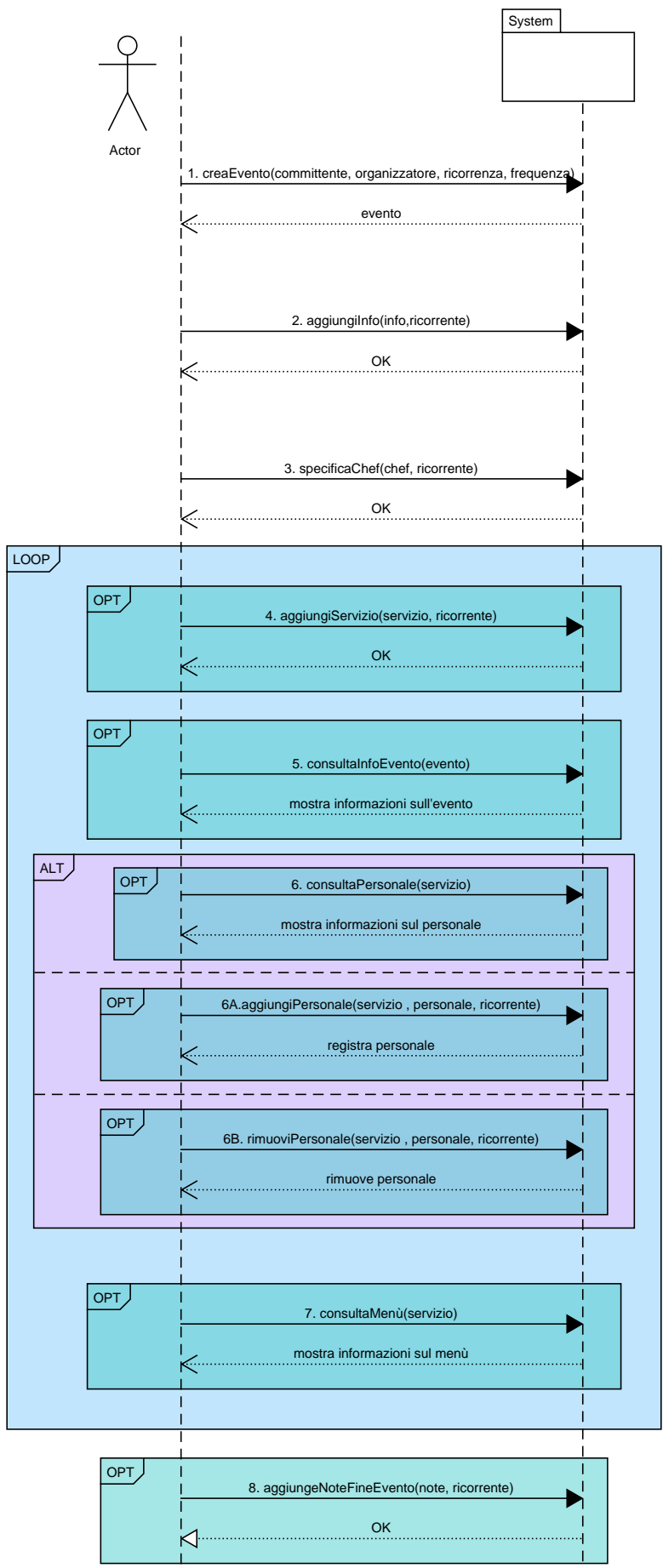
mostra informazioni sul menù

OPT

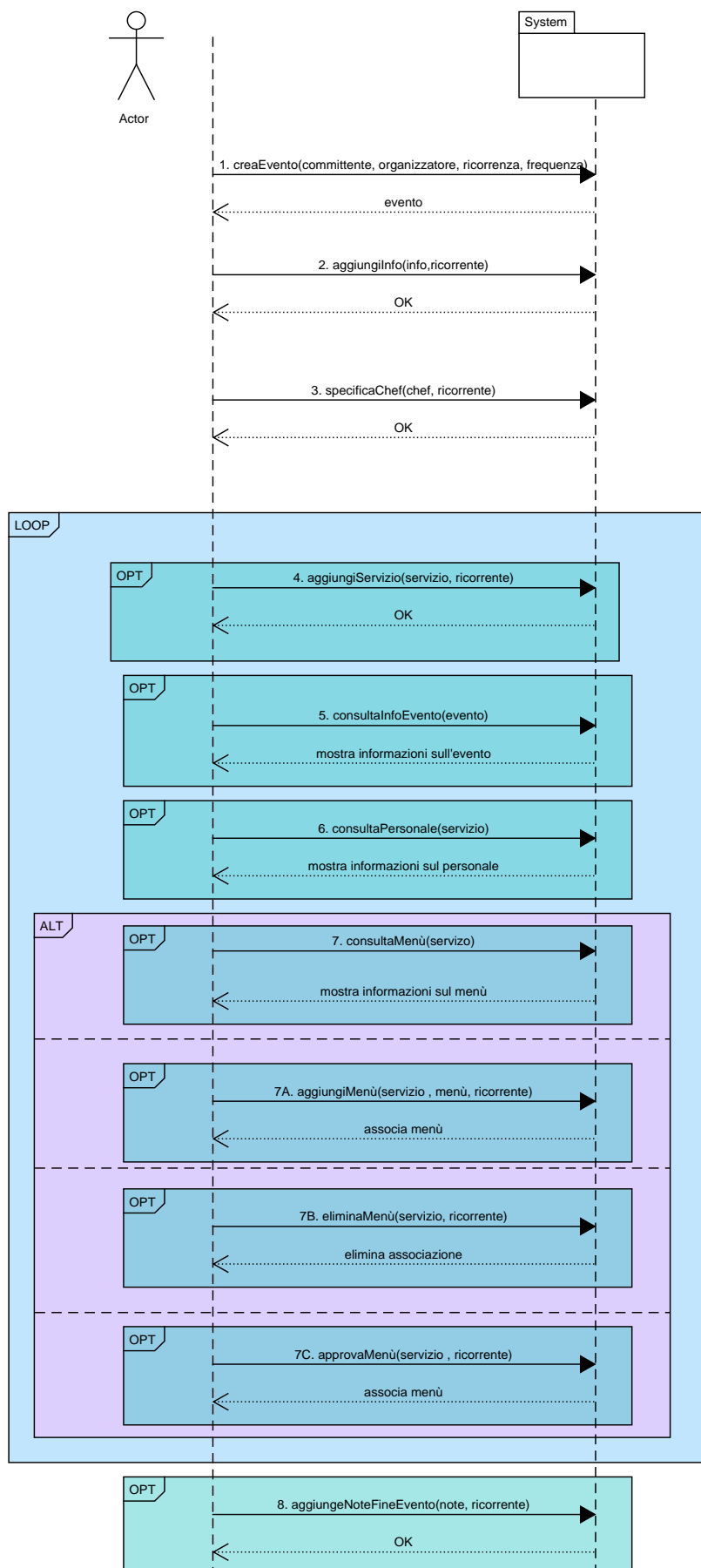
8. aggiungeNoteFineEvento(note, ricorrente)

OK

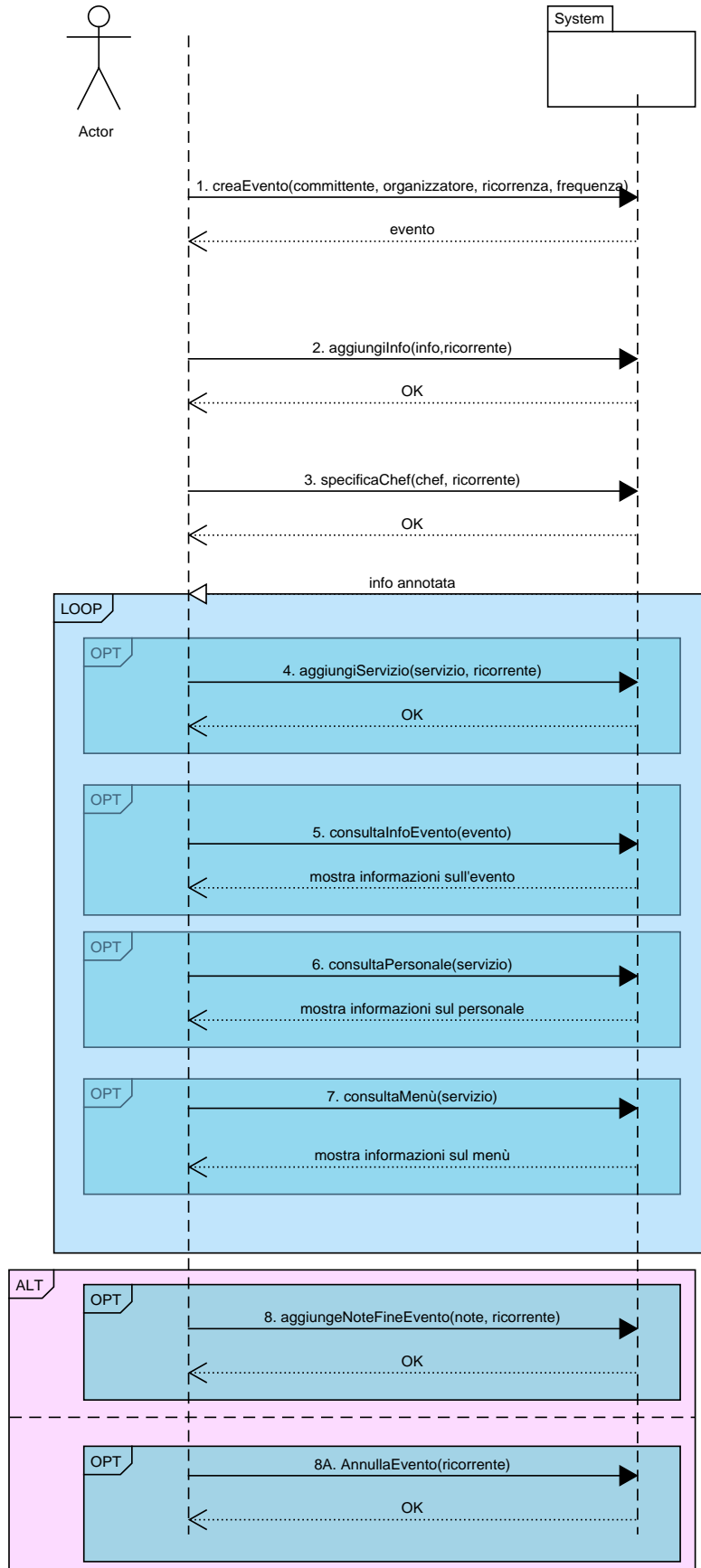
UC estensione 6



UC estensione 7



UC estensione 8



Contratti di “Gestire eventi”

Pre-condizione generale: l'utente è autenticato come Organizzatore

1. ***creaEvento(committente: testo, organizzatore: Organizzatore, ricorrenza: testo, frequenza: numero)***

pre-condizione:

- è in corso la definizione di un Evento *evento*

post-condizione:

- è stata creata un'istanza *evento* di Evento
- è stata creata un'istanza *servizio* di Servizio
- è stata creata un'associazione di tipo *evento* **prevede servizio**
- è stata creata un'istanza *ricorrenza* di Ricorrenza
- *ricorrenza.ricorrenza*=*ricorrenza*
- *ricorrenza.frequenza*=*frequenza*
- *evento.ricorrenza*=*ricorrenza*
- *this.committente* = *committente*
- è stata creata un'associazione di tipo organizzatore **crea evento**
- *evento.inCorso* =no
- *evento* viene aggiunto alla lista degli eventi associati
- se *frequenza*>0

crea un evento copia per ogni ricorrenza e l'aggiunge alla lista degli eventi ricorrenti

1a. ***copioEvento(evento: Evento, committente: testo, organizzatore: Organizzatore)***

pre-condizione: /

post-condizione:

- è stata creata un'istanza *ev* di Evento
- *ev.committente* = *committente*
- *ev.organizzatore* = *organizzatore*

- ev.inCorso = true
- [se specificate] ev.luogo = *evento*.luogo
- [se specificate] ev.data = *evento*.data
- [se specificate] ev.tema = *evento*.tema
- [se specificato] ev.infoTipo = *evento*. InfoTipo
- ev.ricorrenza=*evento*.ricorrenza
- per ogni Servizio s che ha una relazione evento **prevede** s
viene creata una copia di sc di s e viene creata una relazione ev **prevede** sc
- per ogni e di Evento associato ad evento, viene creata una copia e
aggiunta alla lista degli eventi ricorrenti di ev

1b. visualizzoEvento(evento: Evento)

Interrogazione al sistema: non ha post-condizioni.

Nemmeno pre-condizioni, potrebbe essere fatta in qualsiasi momento.

1c. eliminaEvento(evento:Evento, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento
- esiste un'istanza evento di Evento

post-condizione:

- sono state eliminate tutte le istanze di servizio per cui esiste un'associazione evento prevede servizio
- è stata eliminata l'istanza relativa all'evento in corso
- se ricorrente=si
 - viene eliminato ogni ev di Evento nella lista degli eventi ricorrenti di evento

2. aggiungiInfo(luogo?: testo, data?: data, tema?: testo,infoTipo?:testo, ricorrente; si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- [se specificato] *evento*.luogo = luogo

- [se specificata] *evento.data* = data
- [se specificato] *evento.tema* = tema
- [se specificato] *evento.infoTipo* = infoTipo

-se *ricorrente*=si

vengono modificati tutti gli ev di Evento nella lista degli eventi ricorrenti di evento con i parametri specificati

3. **specificaChef(chef: Chef, ricorrente:si/no)**

pre-condizione:

- è in corso la definizione di un Evento *evento*

post-condizione:

- è stata creata l'associazione chef si occupa di evento

-se *ricorrente*=si

viene creata un'associazione del tipo chef si occupa di ev per ogni ev di Evento nella lista degli eventi ricorrenti di evento

4. **aggiungiServizio(servizio: Servizio, ricorrente: si/no)**

pre-condizione:

- è in corso la definizione di un Evento *evento*

post-condizione:

- è stata creata un'associazione evento ***prevede*** servizio

- se *ricorrente*=si

per ogni ev di Evento nella lista degli eventi ricorrenti di evento

-è stata creata un'istanza sv di Servizio

-è stata creata un'associazione ev prevede sv

4.a **eliminaServizio(servizio: Servizio)**

pre-condizione:

- è in corso la definizione di un Evento *evento*
- esiste un'associazione evento prevede servizio

post-condizione:

- è stata eliminata l'associazione evento ***prevede*** servizio

- se *ricorrente*=si

per ogni ev di Evento nella lista degli eventi ricorrenti di evento
per ogni sv di Servizio corrispondente a servizio è stata eliminata
l'associazione ev **prevede** sv

4b. modificaTipologia(servizio: Servizio, tipologia: testo, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- servizio.infoTipo = infoTipo

- se ricorrente=si

per ogni sv di Servizio corrispondente a servizio di evento di ogni
ev di Evento nella lista degli eventi ricorrenti di evento

sv.infoTipo=infoTipo

4c. modificaInvitati(servizio: Servizio, invitati: numero, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- servizio.invitati=invitati

- se ricorrente=si

per ogni sv di Servizio corrispondente a servizio di evento di ogni
ev di Evento nella lista degli eventi ricorrenti di evento

sv.invitati=invitati

4d. infoServizio(servizio: Servizio)

Interrogazione al sistema: non ha post-condizioni.

Nemmeno pre-condizioni, potrebbe essere fatta in qualsiasi momento

5. consultaInfoEvento(evento: Evento)

Interrogazione al sistema: non ha post-condizioni.

Nemmeno pre-condizioni, potrebbe essere fatta in qualsiasi momento.

5a. modificaData(data: numero, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- evento.data=data

-se ricorrente=si

*per ogni ev di Evento appartenente alle liste degli eventi ricorrenti di
evento ev.data=data*

5b. modificaLocation(luogo: testo, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- evento.luogo=luogo

-se ricorrente=si

*per ogni ev di Evento appartenente alla liste degli eventi ricorrenti di
evento ev.luogo=luogo*

5c. modificaTema(tema: testo, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- evento.tema=tema

-se ricorrente=si

*per ogni ev di Evento appartenente alle liste degli eventi ricorrenti di
evento ev.tema=tema*

5d. aggiungiAnnotazioni(annotazioni: testo, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- evento.note = note
- se ricorrente=si

per ogni ev di Evento appartenente alle liste degli eventi ricorrenti di evento ev.note=note

5.e modificaRicorrenza(ricorrenza: testo, frequenza: numero, padre: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- evento.tipo=tipo
- evento.frequenza=frequenza
- se padre=si

per ogni ev di Evento appartenente alle liste degli eventi ricorrenti di evento ev.tipo=tipo ev.frequenza=frequenza

6. consultoPersonale(servizio: Servizio)

Interrogazione al sistema: non ha post-condizioni.

Nemmeno pre-condizioni, potrebbe essere fatta in qualsiasi momento.

6a. aggiungiPersonale(servizio:Servizio, personale: Personale, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- è stata creata un'associazione personale **assegnato a servizio**
- se ricorrente=si

*viene creata un'associazione del tipo personale **assegnato a sv**
per ogni sv di Servizio corrispondente a servizio di evento di ogni ev di Evento nella lista degli eventi ricorrenti di evento*

6b. rimuoviPersonale(servizio: Servizio, personale: Personale, ricorrente=si/no)

pre-condizione:

- è in corso la definizione di un Evento evento
- esiste un'associazione personale assegnato a servizio

post-condizione:

- viene rimossa l'associazione personale **assegnato a** servizio
- se ricorrente=si

viene rimossa l'associazione del tipo personale assegnato a sv per ogni sv di Servizio corrispondente a servizio di evento di ogni ev di Evento nella lista degli eventi ricorrenti di evento

7. consultaMenù(servizio: Servizio)

Interrogazione al sistema: non ha post-condizioni.

Nemmeno pre-condizioni, potrebbe essere fatta in qualsiasi momento.

7a. aggiungiMenù(servizio: Servizio, menù: Menù, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento

post-condizione:

- è stata creata un'associazione servizio **associato a** menù
- se ricorrente=si

viene creata un'associazione del tipo sv **associato a** menù per ogni sv di Servizio corrispondente a servizio di evento

7b. eliminaMenù(servizio: Servizio)

pre-condizione:

- è in corso la definizione di un Evento evento
- esiste un'istanza menù di Menù tale per cui esiste un'associazione menu associato a servizio

post-condizione:

- è stata eliminata l'associazione servizio **associato a** menù

- se ricorrente=si

viene eliminata l'associazione del tipo sv associato a menù per ogni sv di Servizio corrispondente a servizio di evento di ogni ev di Evento nella lista degli eventi ricorrenti di evento

7b. approvaMenù(servizio: Servizio, ricorrente: si/no)

pre-condizione:

- è in corso la definizione di un Evento evento
- esiste un'istanza menù di Menù tale per cui esiste un'associazione menu associato a servizio

post-condizione:

- menù.approvato = si
- evento.inCorso=si
- se ricorrente=si

per ogni sv di Servizio corrispondente a servizio di evento di ogni ev di Evento nella lista degli eventi ricorrenti di evento

sv.approvato = si, ev.inCorso=si

8. aggiungiNoteFineEvento(note: testo, ricorrente: si/no)

pre-condizione:

- si è concluso l'evento o è stato annullato

post-condizione:

- evento.note = note
- se ricorrente=si

per ogni ev di Evento appartenente alla liste degli eventi ricorrenti di evento ev.note = note

8a. annullaEvento()

pre-condizione:

- è in corso la definizione di un Evento *evento*

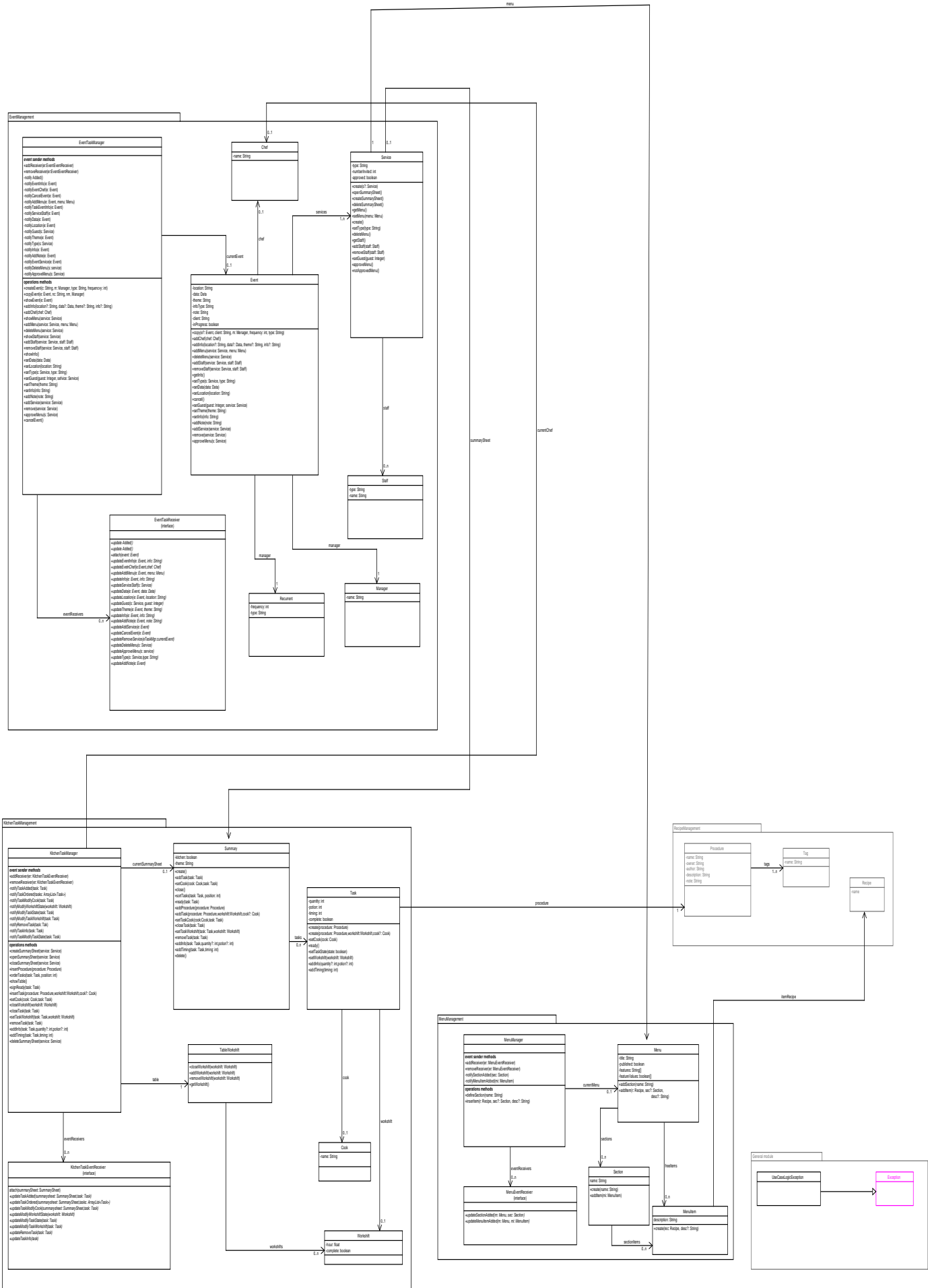
post-condizione:

- evento.annullato = si
- evento.inCorso = no
- se ricorrente=si

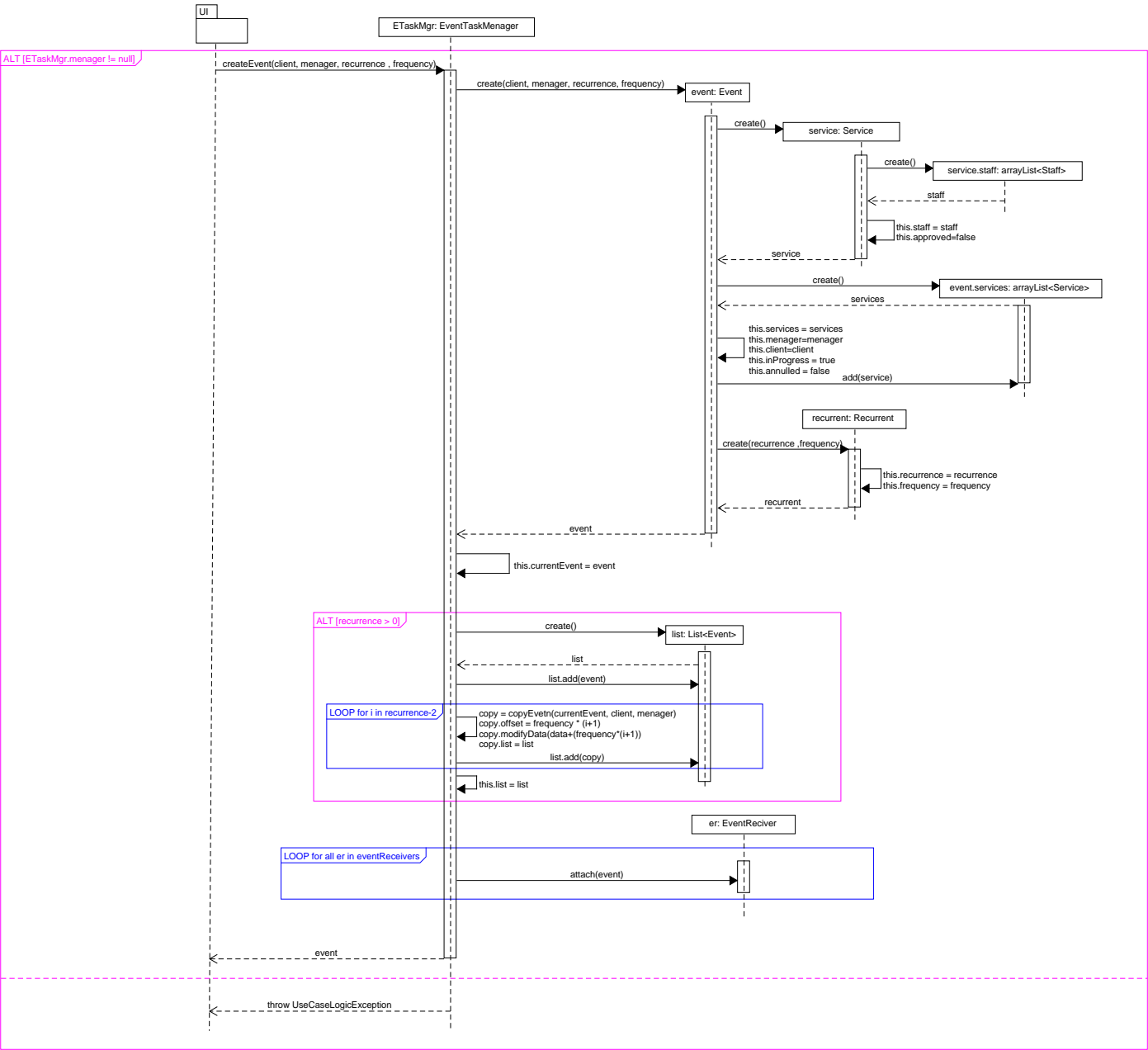
*per ogni ev di Evento appartenente alla liste degli eventi ricorrenti di
evento ev.annullato=si, ev.inCorso=no*

Diagramma delle classi

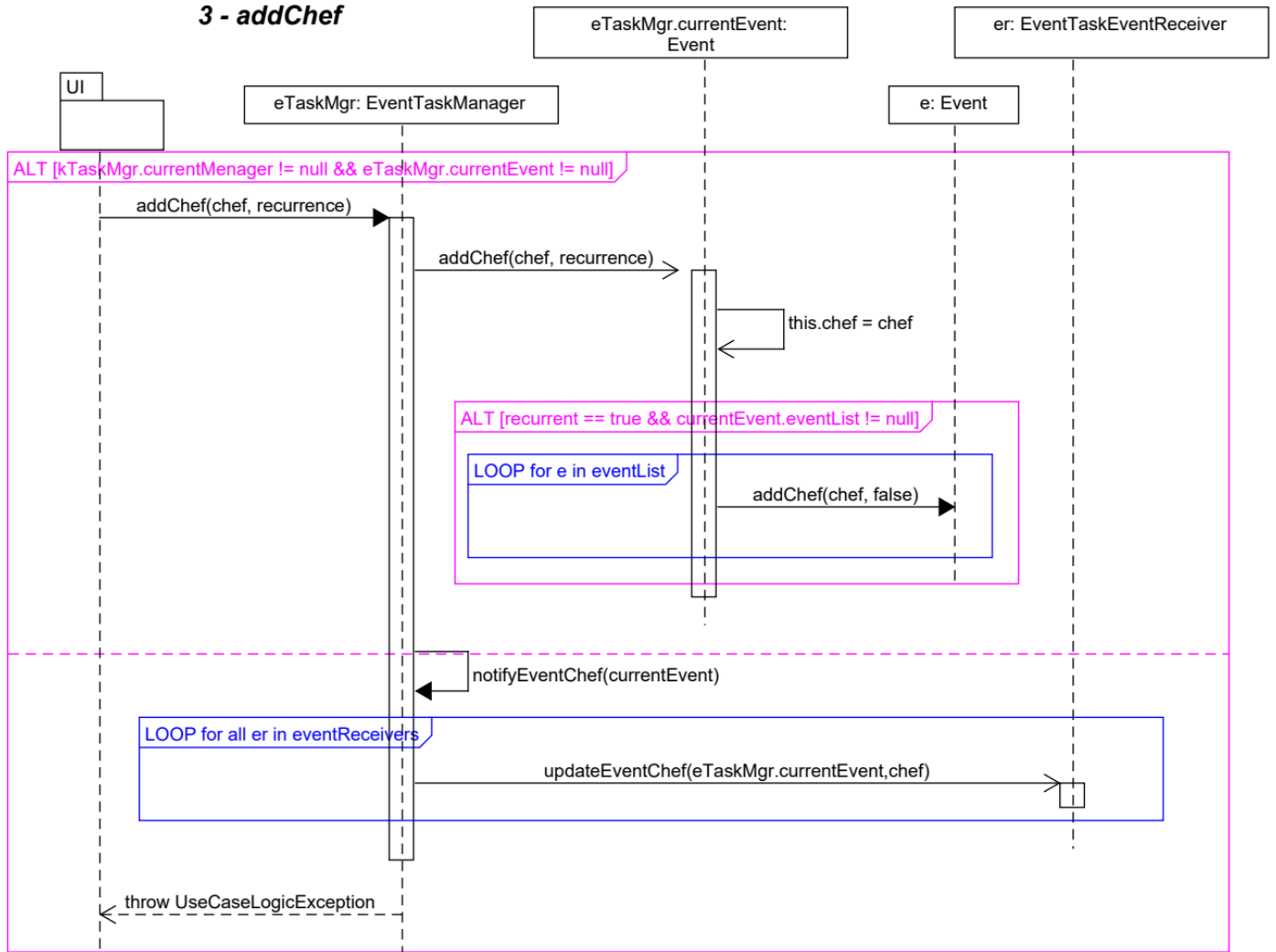
Design Class Diagram (DCD)



1 - createEvent



3 - addChef



er: EventTaskEventReceiver

4 - addService

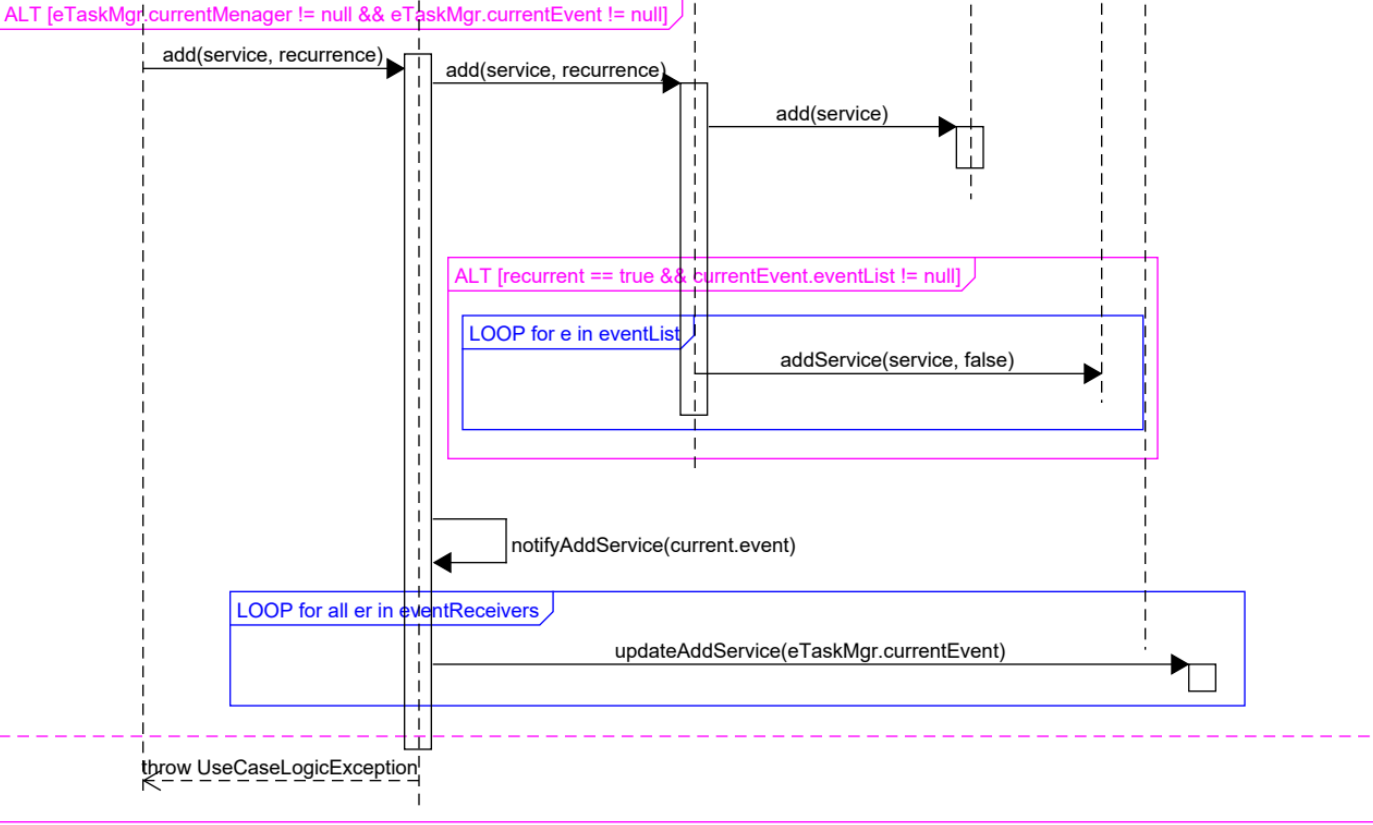
UI

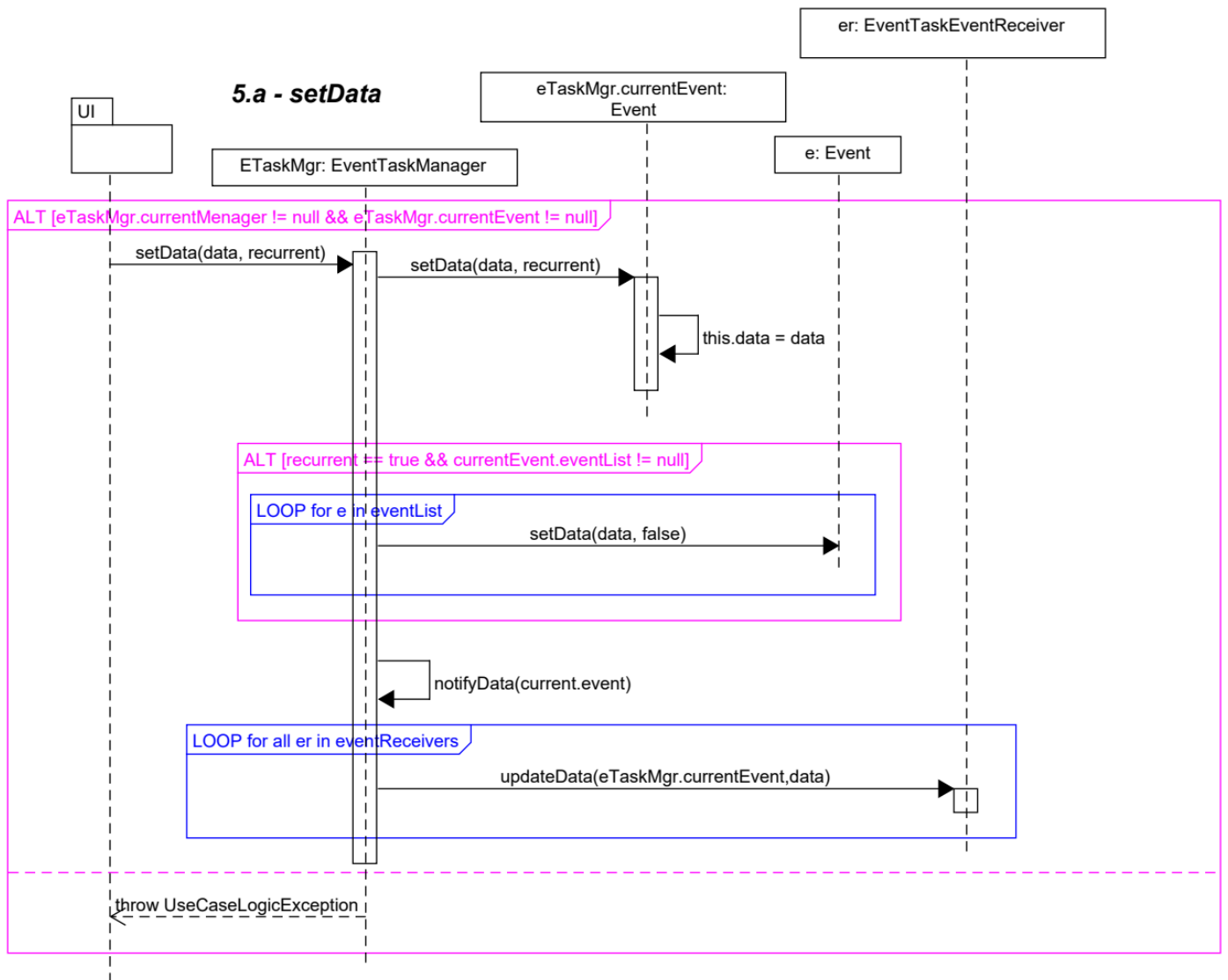
eTaskMgr.currentEvent:
Event

e: Event

ETaskMgr: EventTaskManager

event.services: arrayList<Service>





8.a - cancelEvent

