

Allegato Tecnico UC - Gestire Eventi

Mocanu Alexandru

Matr. 813322

Rondinelli Andrea

Matr. 834560

Warth Jakob

Matr. 835080

Analisi dei requisiti

Gestire Eventi

Informazioni generali

Nome caso d'uso: Gestire eventi
Portata:
Livello:
Attore primario: Organizzatore
Parti Interessate: Chef, cuoco, personale
Pre-condizioni: L'attore dev'essere autenticato come Organizzatore
Garanzie di successo o post-condizioni:

Scenario principale di successo

#	Attore	Sistema
1	Predisporre un nuovo evento	Inserisce un nuovo evento
2	Compila una scheda con data, luogo e numero partecipanti	Salva la scheda dell'evento
3	Assegna uno Chef	Inserisce uno Chef per l'evento
4	Assegna il personale	Inserisce il personale per l'evento
	Se desidera torna al passo 2, altrimenti prosegue	
5	Opzionalmente scrive delle note sull'evento al termine dello stesso	Inserisce un documento dell'evento
	Termina il caso d'uso	

Estensione 1a

#	Attore	Sistema
1a.1	Seleziona un evento esistente per modificarlo	Visualizza l'evento preesistente
	<i>Quando soddisfatto, vai al passo 2</i>	

Eccezione 1a.1a

#	Attore	Sistema
1a.1a.1	Sceglie un evento esistente per modificarlo	L'evento è terminato e non può essere modificato.
	<i>Termina il caso d'uso</i>	

Eccezione 1a.1b

#	Attore	Sistema
1a.1b.1	Sceglie un evento esistente per modificarlo	L'evento non è di proprietà dell'organizzatore e non può essere modificato.
	<i>Termina il caso d'uso</i>	

Estensione 1b

#	Attore	Sistema
1b.1	Seleziona un evento esistente per eliminarlo	Elimina l'evento preesistente
	<i>Termina il caso d'uso</i>	

Eccezione 1b.1a

#	Attore	Sistema
1b.1a.1	Seleziona un evento esistente per eliminarlo.	L'evento non è di proprietà pertanto non si può proseguire.
	<i>Termina il caso d'uso</i>	

Estensione 2a

#	Attore	Sistema
2a.1	Modifica le informazioni dell'evento	Aggiorna i dati dell'evento
	<i>Quando soddisfatto, vai al passo 3</i>	

Estensione 3a

#	Attore	Sistema
3a.1	Modifica lo chef	Aggiorna lo chef per l'evento
	<i>Quando soddisfatto, vai al passo 4</i>	

Eccezione 3.1a

#	Attore	Sistema
3.1a.1	Assegna uno chef	Lo chef non è disponibile.
	<i>Assegna un nuovo chef disponibile. Va al passo 4 quando finito</i>	

Estensione 4a

#	Attore	Sistema
4a.1	Aggiungi personale per l'evento	Aggiorna il personale per l'evento.
	<i>Quando soddisfatto prosegui.</i>	

Eccezione 4a.1a

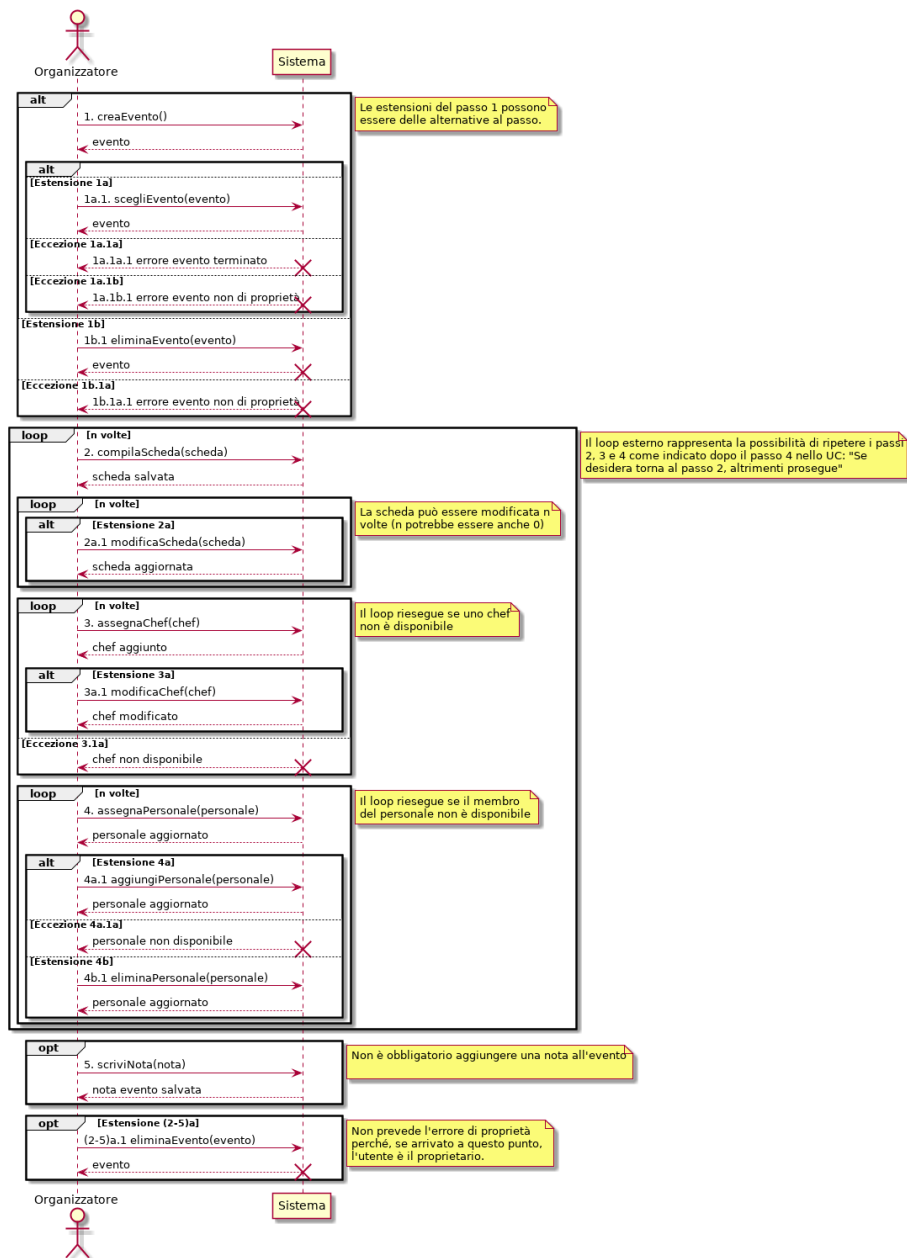
#	Attore	Sistema
4a.1a.1	Assegna il personale	Membri del personale non disponibili
	<i>Assegna un nuovo staff disponibile. Va al passo 5 quando finito</i>	

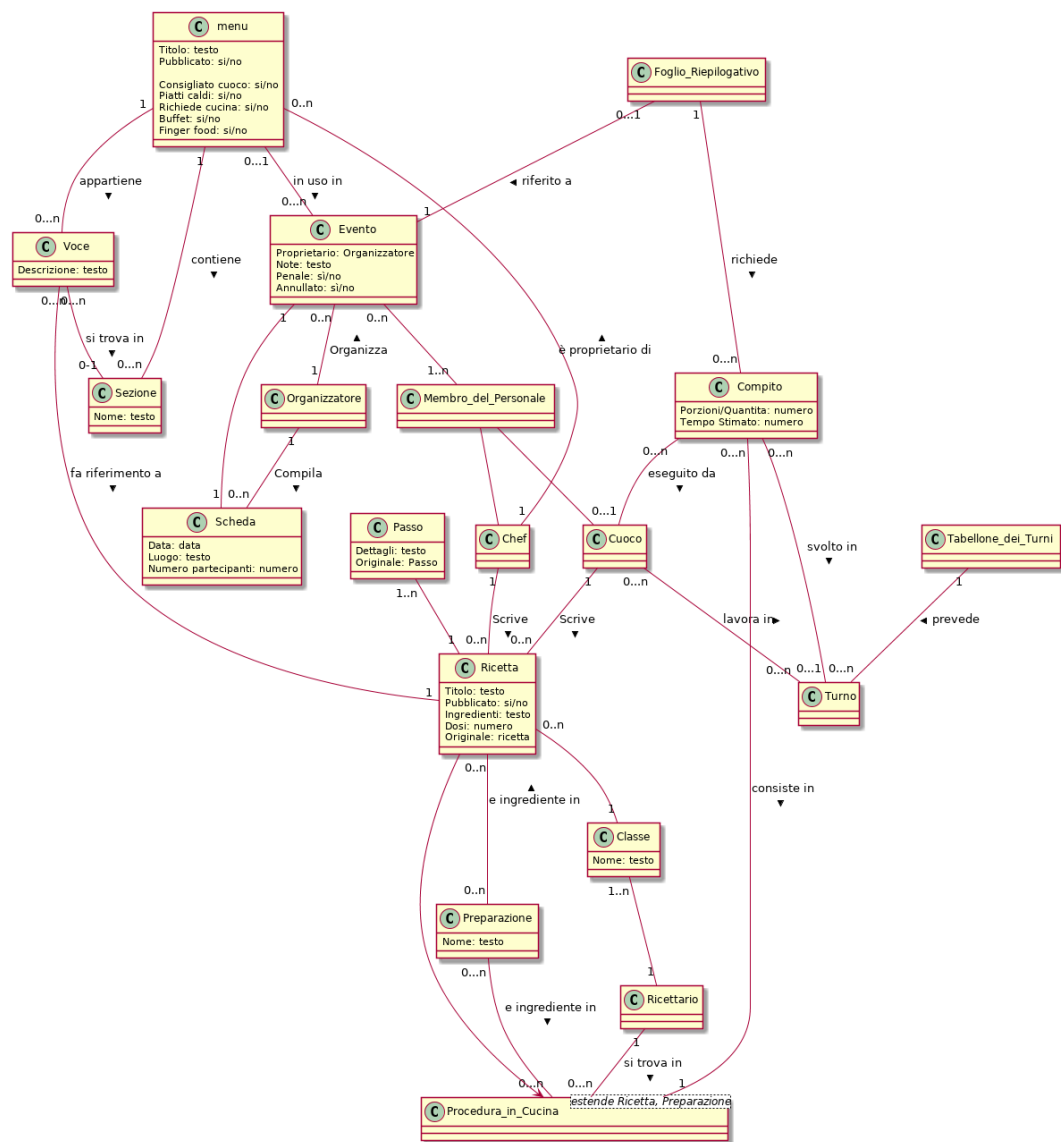
Estensione 4b

#	Attore	Sistema
4b.1	Elimina personale per l'evento	Aggiorna il personale per l'evento.
	<i>Quando soddisfatto prosegui.</i>	

Estensione (2-5)a

#	Attore	Sistema
(2-5)a.1	Rimuove l'evento	Elimina l'evento
	<i>Termina il caso d'uso</i>	





Contratti

Pre-condizione generale

- L'utente dev'essere registrato come una istanza **org** di Organizzatore

1. creaEvento()

Pre-condizione

- Pre-condizione generale.

Post-condizione

- è stata creata un'istanza **e** di *Evento*
- *org* è **proprietario** di **e**

1a. scegliEventoPreesistente()

Pre-condizione

- Pre-condizione generale.
- *org* è **proprietario** di **e**

Post-condizione

- viene restituita una istanza **e** di un *Evento* preesistente

1b. eliminaEvento(evento: e)

Pre-condizione

- L'evento *e* esiste

Post-condizione

- L'evento *e* è annullato

2. compilaScheda(evento: Evento, data: data, luogo: testo, numero_partecipanti?: numero)

Pre-condizione

- è in corso la creazione di un *Evento* *e*

Post-condizione

- è stata creata un'istanza *s* di *Scheda*
- *s.data* = *data*
- *s.luogo* = *luogo*
- *s.numero_partecipanti* = *numero_partecipanti*
- *e* contiene *s*

2a. modificaScheda(scheda: Scheda, data?: data, luogo: testo, numero_partecipanti?: numero)

Pre-condizione

- è in corso la creazione di un *Evento* *e*

Post-condizione

- una istanza *s* di *Scheda* è modificata
- (se presente) *s.data* = *data*
- (se presente) *s.luogo* = *luogo*
- (se presente) *s.numero_partecipanti* = *numero_partecipanti*

3. assegnaChef(evento: Evento, chef: Chef)

Pre-condizione

- è in corso la creazione di un *Evento* **e**

Post-condizione

- è stata creata un'istanza **c** di *Chef*
- **c** è associato all'evento **e**

3a. modificaChef(c: Chef, chef_new: Chef)

Pre-condizione

- lo chef **c** esiste
- è in corso la creazione di un *Evento* **e**

Post-condizione

- lo chef **chef_new** è associato all'evento

4. assegnaPersonale()

Pre-condizione

- è in corso la creazione di un *Evento* **e**

Post-condizione

- è stata creata una istanza **m** di *Membro_del_Personale*
- **m** è associato all'evento

4a. aggiungiPersonale(membro: Membro_del_Personale)

Pre-condizione

- è in corso la creazione di un *Evento* **e**
- esiste una istanza **m** di *Membro_del_Personale*

Post-condizione

- il membro del personale è aggiunto all'evento

4b. eliminaPersonale(membro: *Membro_del_Personale*)

Pre-condizione

- è in corso la creazione di un *Evento* e
- membro è parte del personale dell'evento

Post-condizione

- membro è eliminato dal personale dell'evento

5. scriviNote(evento: *Evento*, nota: testo)

Pre-condizione

- l'evento è terminato

Post-condizione

- evento.note = nota

(2-5)a. eliminaEvento(evento: *Evento*, penale: si/no)

Pre-condizione

- è in corso la creazione di un *Evento* e

Post-condizione

- e annullato = sì
- e.penale = penale
- la compilazione è interrotta

Allegato Tecnico UC - Gestire Eventi

Mocanu Alexandru

Matr. 813322

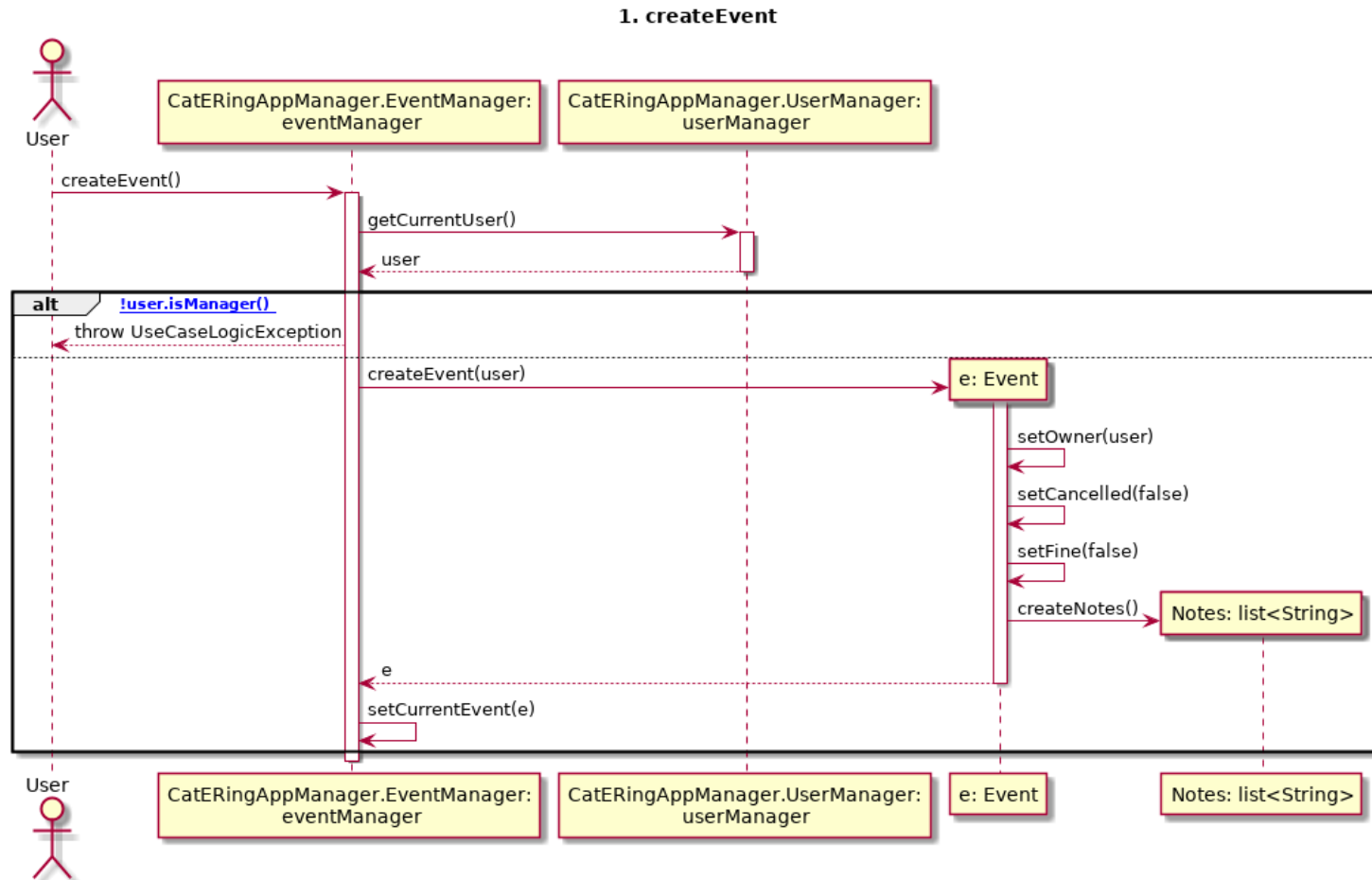
Rondinelli Andrea

Matr. 834560

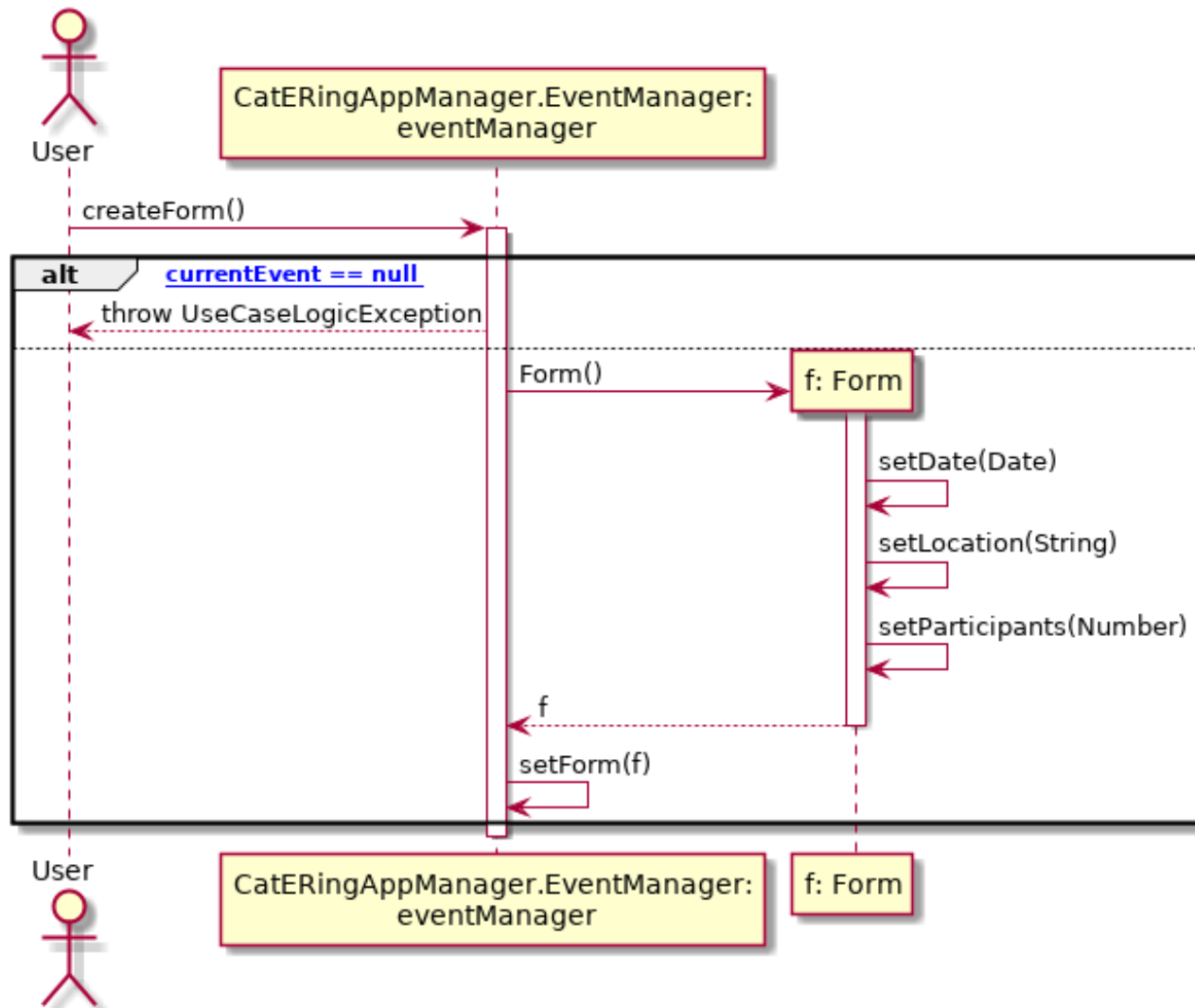
Warth Jakob

Matr. 835080

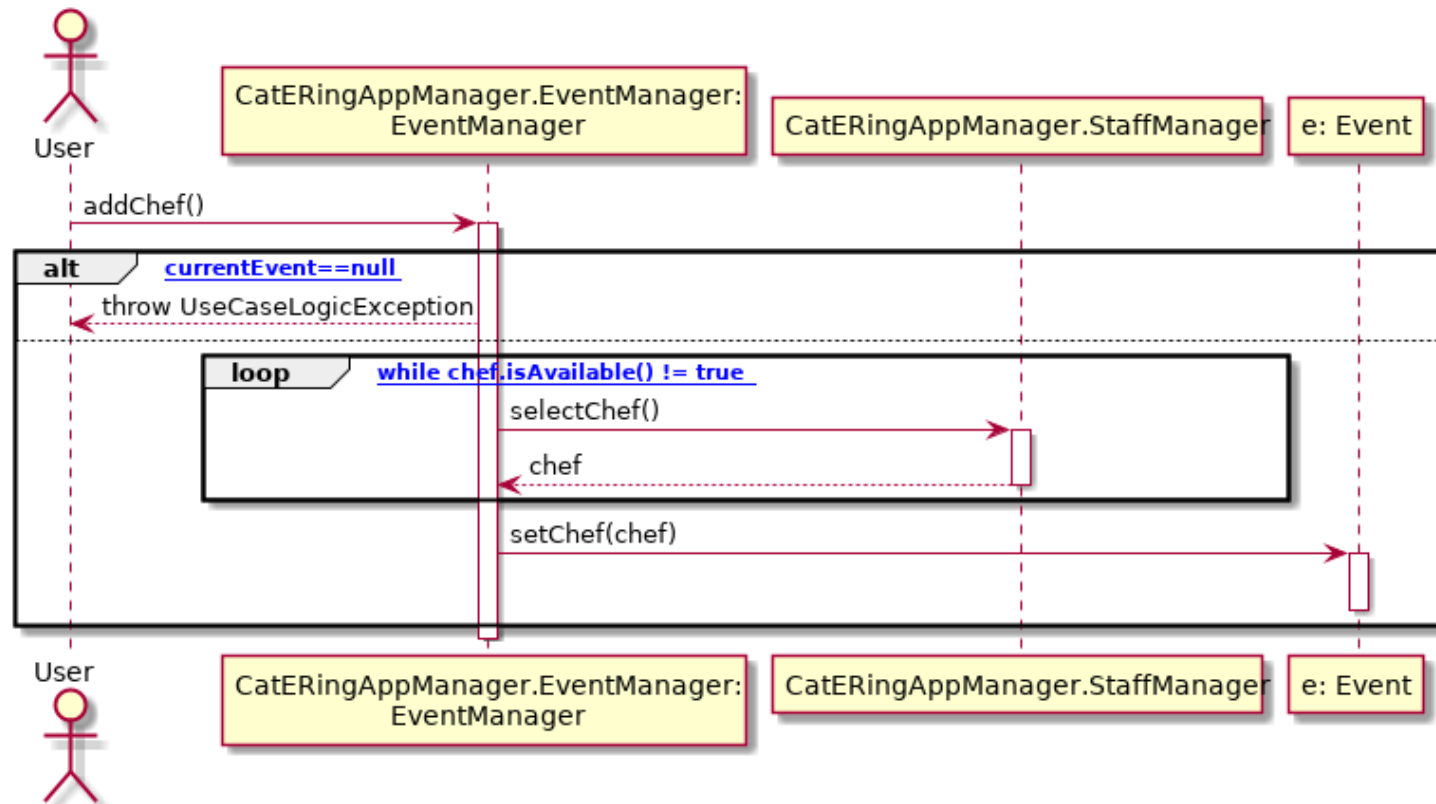
Progettazione



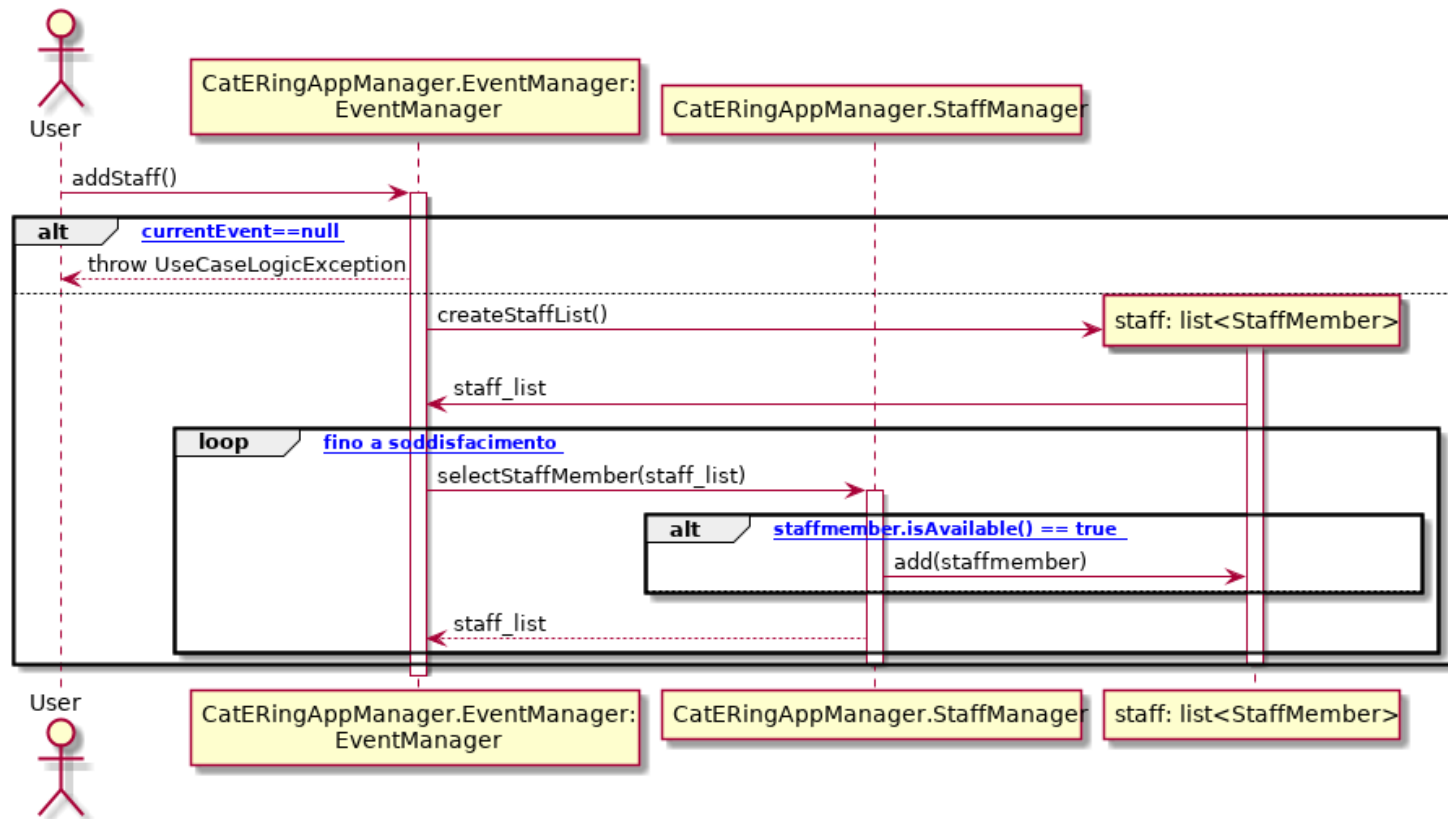
2. defineForm



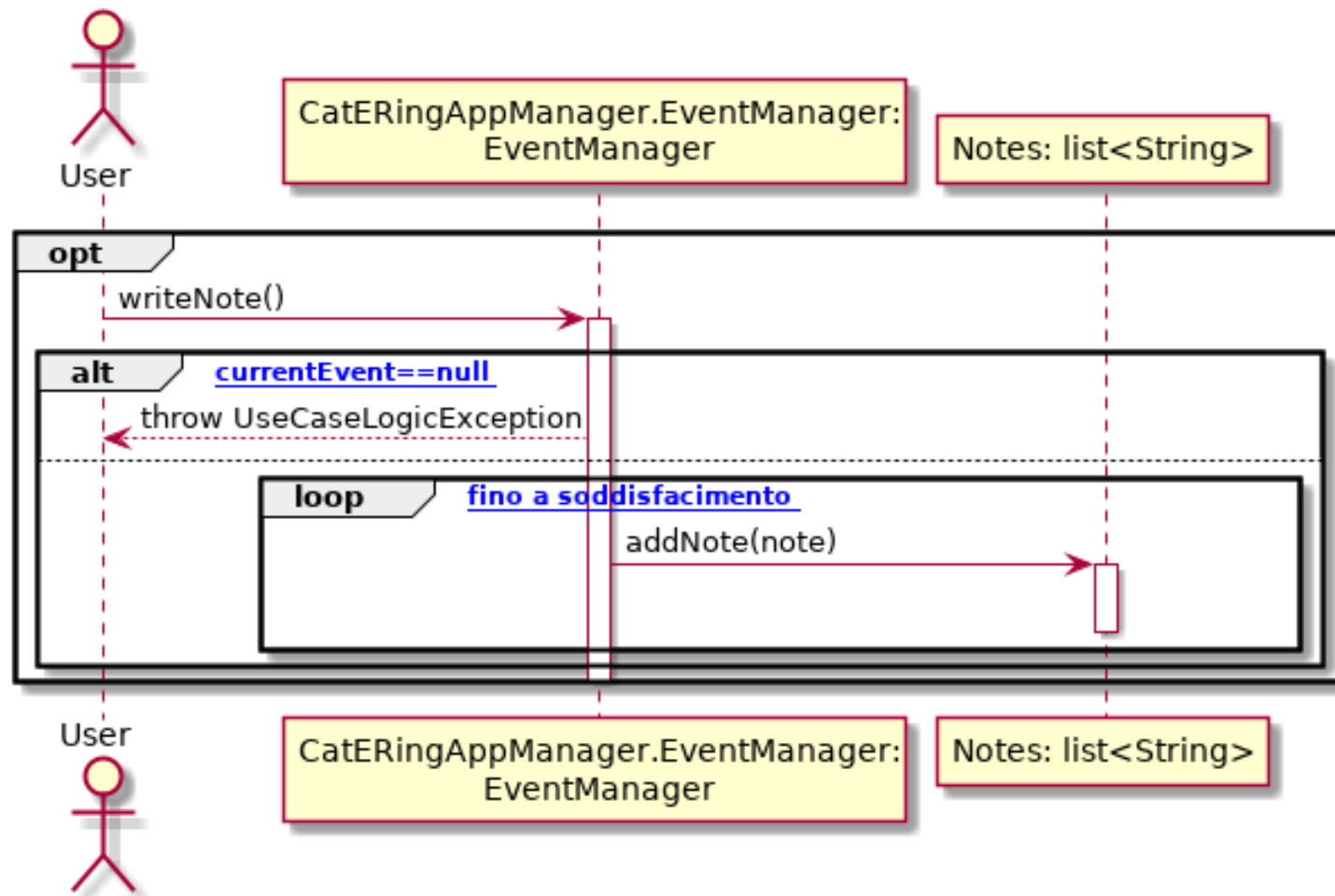
3. assignChef



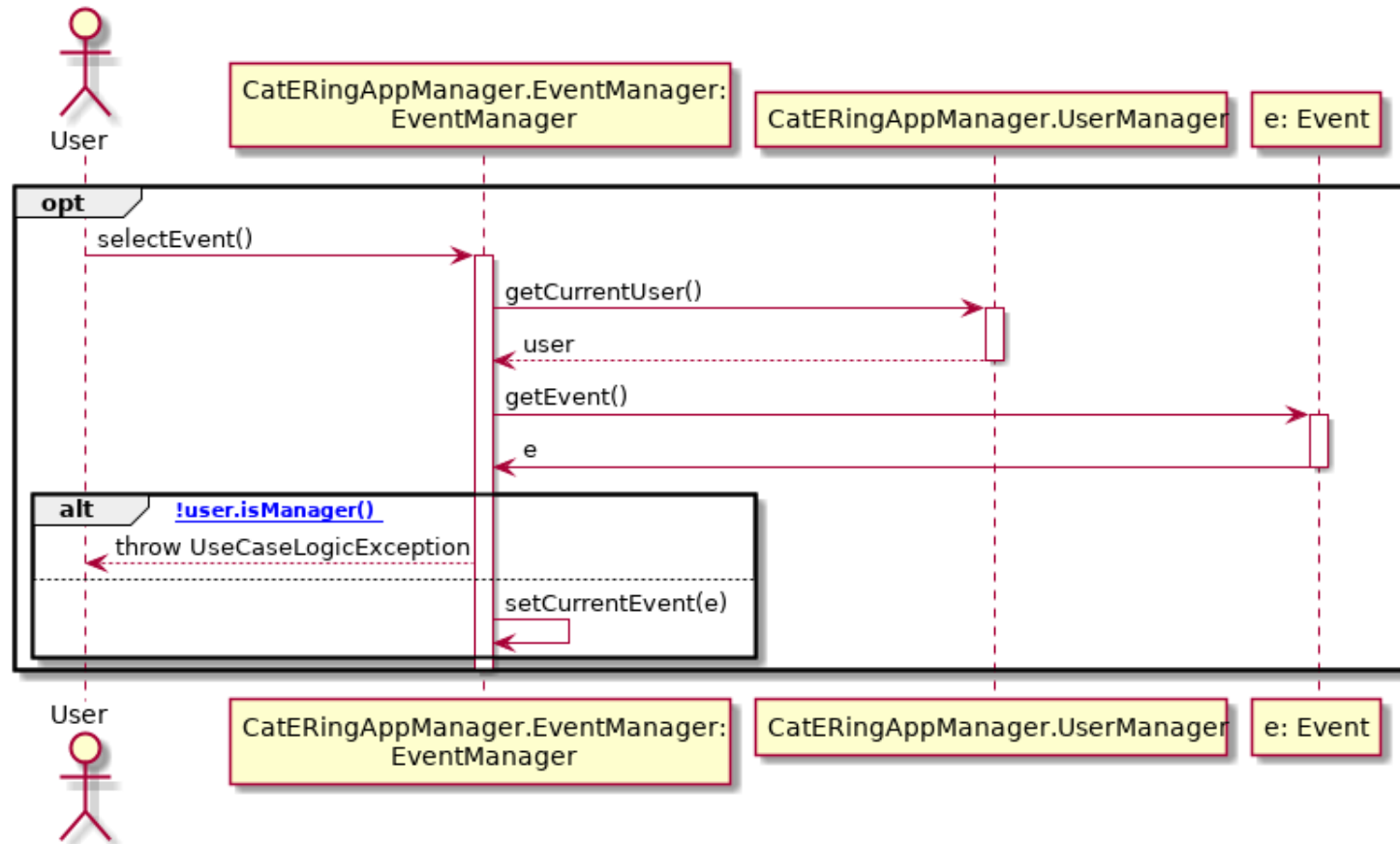
4. assignStaff



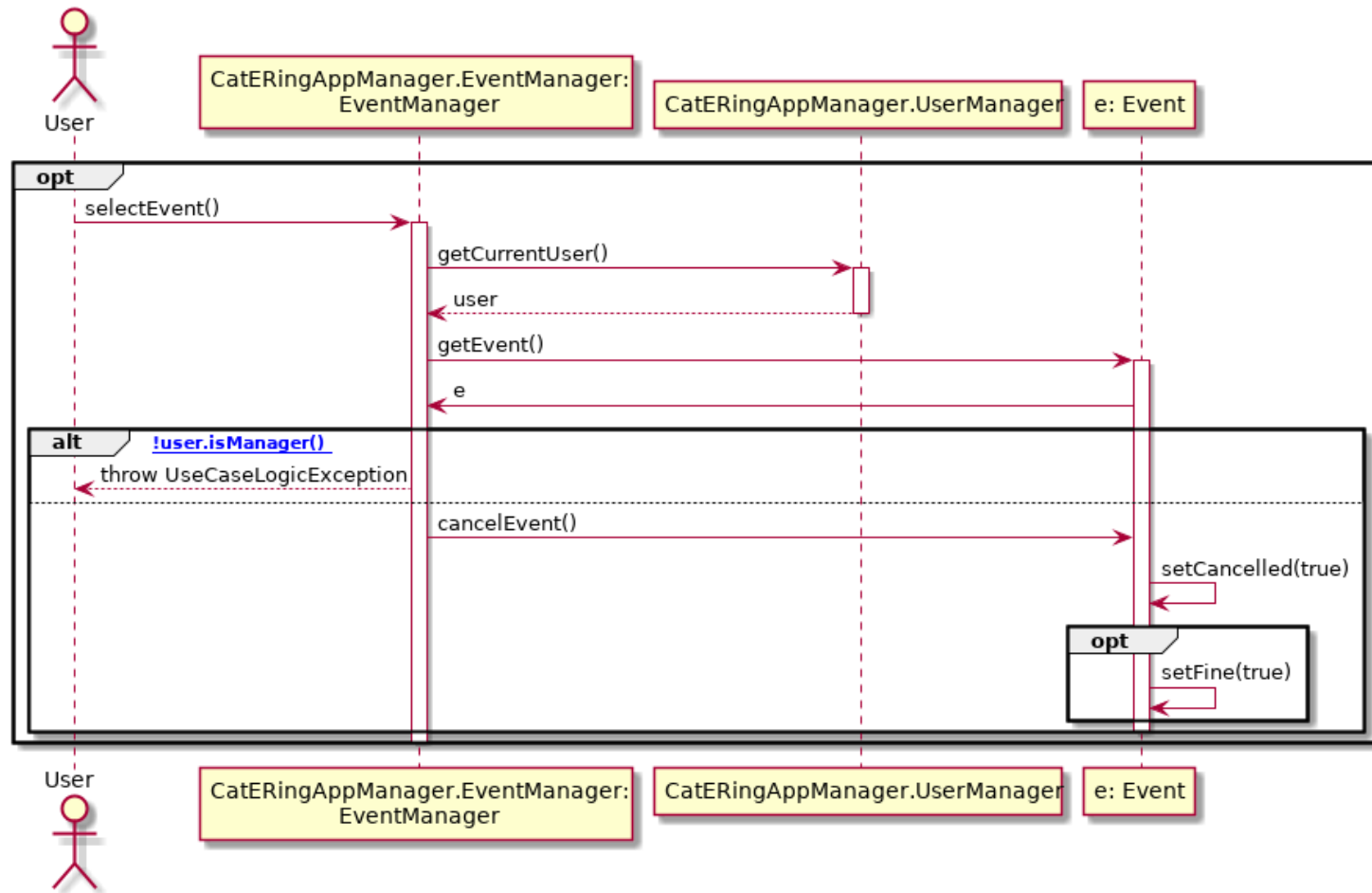
5. writeNote



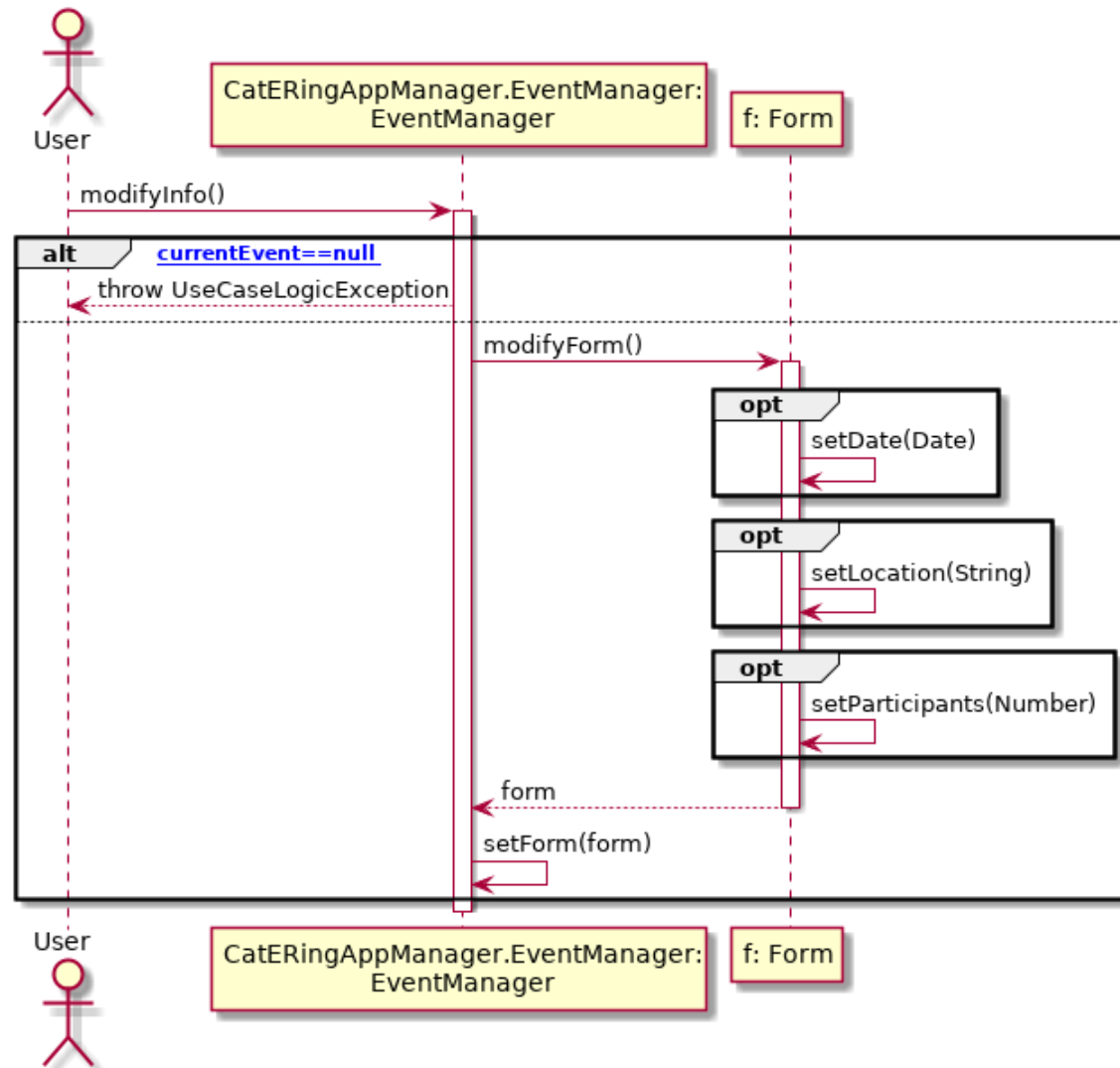
1a. selectEvent



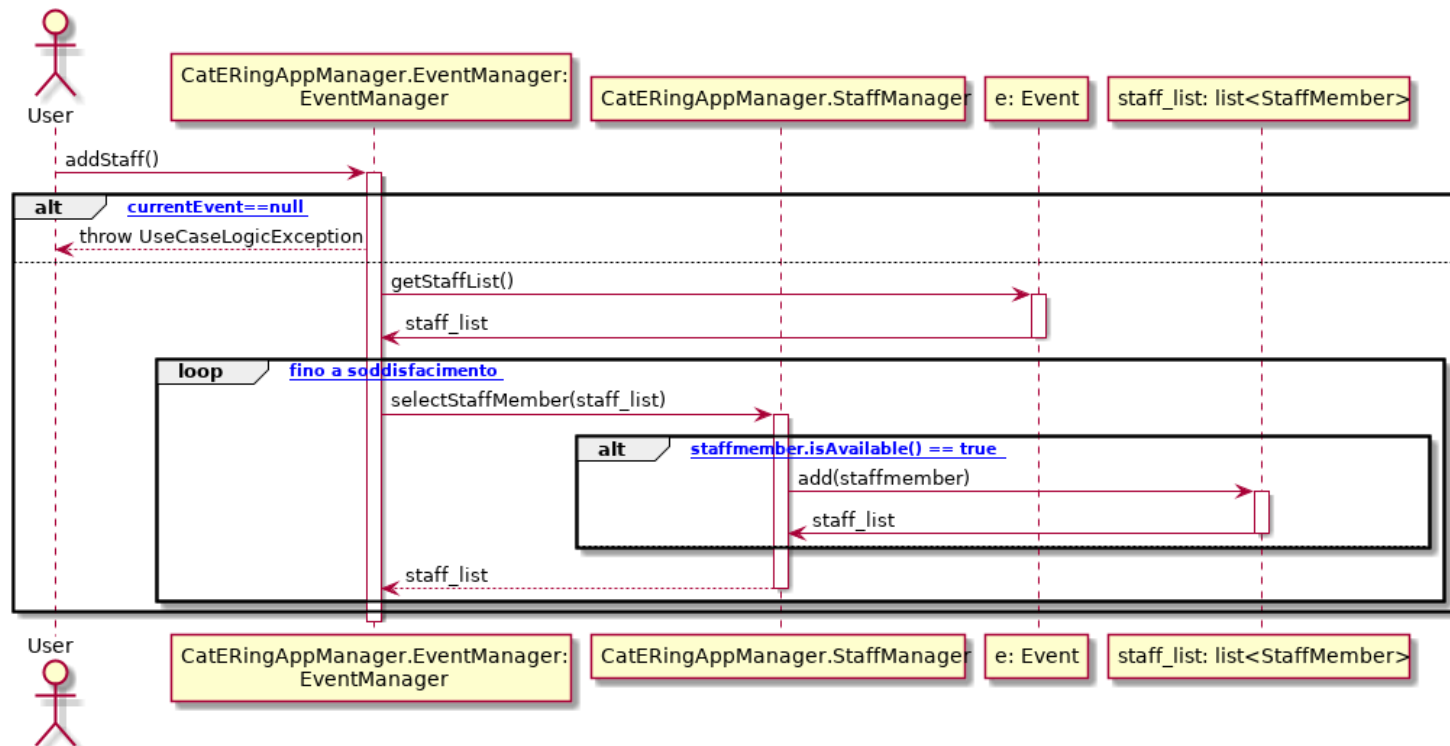
1b. removeEvent



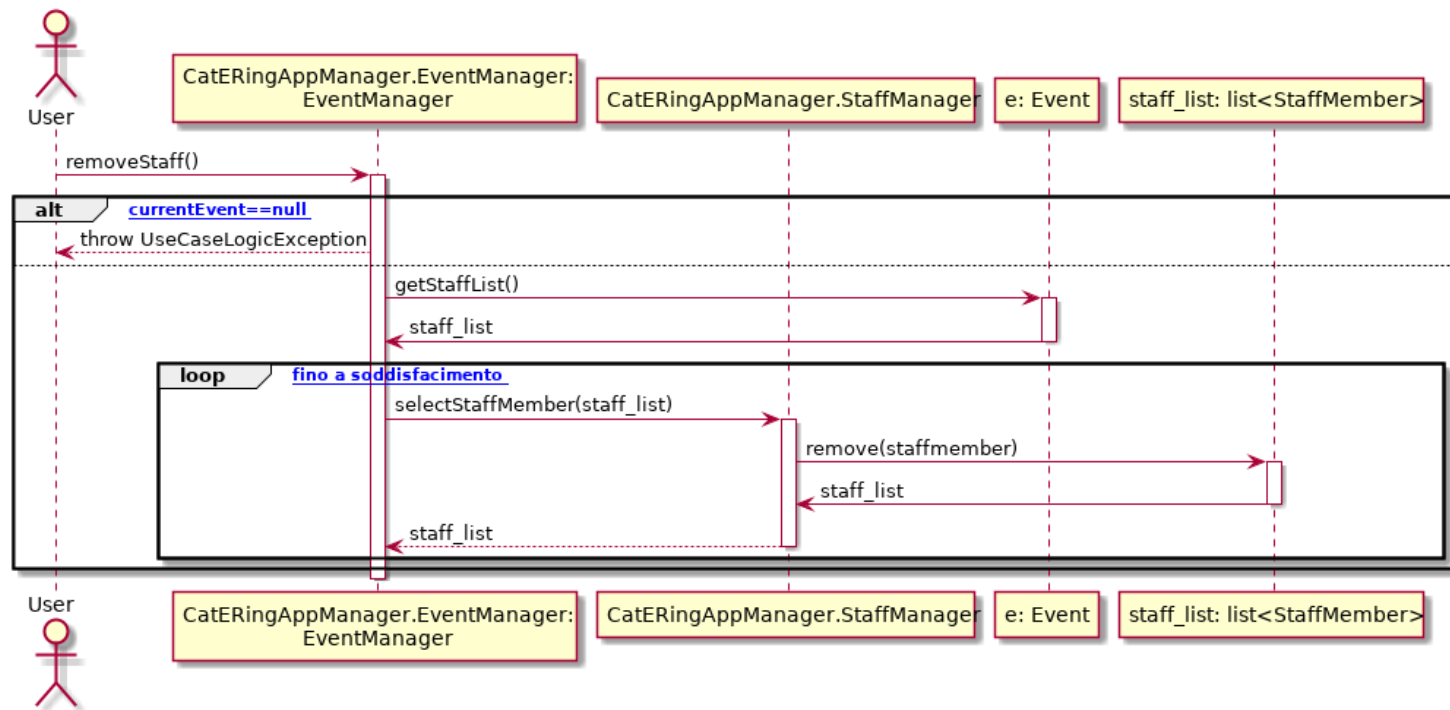
2a. modifyInfos



4a. addStaff



4b. removeStaff



(2-5)a. removeEvent

