1a.2 cancelEvent

UI

eventMgr: EventManager

kTaskReceiver: KitcheTaskEventReceiv

eventMgr.currentTemplateEvent: TemplateEvent

e: Event

s: Service

s.sheet: SummmarySheet

task: Task

job.shift: KitchenShift

cancelEvent(event, option)

alt [eventMgr.currentTemplateEvent.hasEvent(event)]

throw UseCaseLogicException

alt [event.getState() != "active"]

throw EventException

cancelEvent(event, option)

create()

cancelledEvents: ArrayList<Event>

cancelledEvents

for e in events

alt [(options == "EveryEvent") || (options == "CurrentAndNext" && e.startDate > event.startDate) || e == event]

alt [e.state == "active"]

cancelEvent()

this.state = "canceled"

for s in this.services

alt [s == service]

cancelService()

this.state = "canceled"

deleteAllCookingJobs()

for task in s.sheet.tasks

deleteAllCookingJobs()

for job in task.jobs

freeTime(job.estimatedTime)

this.jobs = null

this.serviceJobs = null

this.serviceJobs = null

add(e)

cancelledEvents

for cancelledEvent in cancelledEvents

notifyEventCancelled(cancelledEvent)

for kTaskReceiver in kTaskReceivers

updateEventCancelled(cancelledEvent)