



**UNIVERSITÀ  
DI TRENTO**

Dipartimento di Ingegneria e Scienza dell'Informazione

---

**Progetto:**

**Animati**



**Titolo del documento:**

**???**

**Gruppo:**

**T51**

Indice

Diagramma delle Classi ..... 3

OCL ..... ?

## **Scopo del documento**

[...]

## Diagramma delle classi

[...]

## OCL

Nel presente capitolo vengono descritti i vincoli OCL delle classi.

### Cronometro

**stato : Enum**

**Invarianti:**

- stato assume i valori "reset", "run", "pause"

Metodo	Precondizioni	Postcondizioni
start()	<b>stato</b> deve essere in "reset"	<b>stato</b> assume valore "run"
stop()	<b>stato</b> deve essere in "run"	<b>stato</b> assume valore "pause"
riprendi()	<b>stato</b> deve essere in "pause"	<b>stato</b> assume valore "run"
ripristina()	<b>stato</b> deve essere in "pause"	<b>stato</b> assume valore "reset"
parziale()	<b>stato</b> deve essere in "run"	<ul style="list-style-type: none"> <li>• <b>stato</b> rimane al valore "run"</li> <li>• aggiunto tempo a lista <b>parziali</b></li> </ul>

### Segna punti

Metodo	Precondizioni	Postcondizioni
aggiungiContatore(nome : String)	<ul style="list-style-type: none"> <li>• il nome non può eccedere i 99 caratteri</li> <li>• il numero di squadre può essere al massimo 99</li> <li>• nel caso l'utente non abbia inserito il nome di tutte o alcune squadre, alle squadre senza nome verrà assegnato di default il nome "Squadra i", dove i è un numero che va da 1 al numero di squadre senza nome</li> </ul>	l'attributo contatori viene impostato con i nomi scelti e i punti inizialmente tutti a 0
incrementa(nome : String)	il contatore di ogni squadra può assumere valore massimo 500	il contatore della squadra col nome scelto viene incrementato di 1
decrementa(nome : String)	il contatore di ogni squadra può assumere valore minimo -500	il contatore della squadra col nome scelto viene decrementato di 1

### Timer

**stato : Enum**

**Invarianti**

- stato assume i valori "reset", "run", "pause"

Metodo	Precondizioni	Postcondizioni
start()	<b>stato</b> deve essere in "reset"	<b>stato</b> assume valore "run"
riprendi()	<b>stato</b> deve essere in "pause"	<b>stato</b> assume valore "run"
stop()	<b>stato</b> deve essere in "run"	<b>stato</b> assume valore "pause"
annulla()	<b>stato</b> deve essere in "pause"	<b>stato</b> assume valore "reset"

**Dado**

Metodo	Precondizioni	Postcondizioni
estrai() : Object		
cambiaModalità(riestrazione : bool)		
aggiungi(oggetto : Object)		

**Fischietto**

Metodo	Precondizioni	Postcondizioni
start()	il fischietto deve essere premuto	il suono viene riprodotto
stop()	il fischietto non deve essere premuto	nessun suono viene riprodotto
scegliSuono()		l'attributo suono viene impostato con il suono scelto

**Creazione squadre**

**metodo** : Enum

**Invarianti:**

- metodo assume i valori "Round robin", "Random", "Fill first" e "Balanced"

Metodo	Precondizioni	Postcondizioni
inserisciNumSquadre(numero : int)	numeroSquadre può essere al massimo 99	informazioni viene incrementato di 1
inserisciNumComponenti(numero : int)	numeroComponenti può essere al massimo 99	informazioni viene incrementato di 1
inserisciNumPartecipanti(numero : int)	numeroPartecipanti può essere al massimo 9801	informazioni viene incrementato di 1
scegliMetodo(metodo : Enum)		l'attributo metodo viene impostato con quello scelto

inserisciNome(nome : String)	la lunghezza dei nomi non deve eccedere i 99 caratteri	l'attributo nomi viene impostato con i nomi scelti
estrai()	<ul style="list-style-type: none"> <li>• informazioni deve essere uguale a 2 o 3</li> <li>• nel caso i tre valori: numeroSquadre, numeroPartecipanti e numeroComponenti non siano compatibili, verranno solamente considerati numeroPartecipanti e numeroSquadre.</li> </ul>	<ul style="list-style-type: none"> <li>• se metodo assume il valore "Random" l'ordine delle squadre assegnate sarà completamente casuale</li> <li>• se metodo assume il valore "Round robin" l'assegnamento delle squadre sarà sequenziale</li> <li>• se il metodo assume il valore "Fill first" l'assegnamento avverrà per completamento delle squadre, ovvero riempiendo i posti di ogni squadra prima di procedere con l'assegnamento per la prossima</li> <li>• se il metodo assume il valore "Balanced" tutte le squadre dovranno avere lo stesso numero di partecipanti prima di procedere con gli assegnamenti</li> <li>• qualsiasi sia il metodo scelto, ogni volta che un partecipante viene assegnato ad una squadra il contatore di quella squadra viene incrementato di 1</li> </ul>

## Utente

**ruolo : Enum**

**Invarianti:**

- ruolo assume i valori "Amministratore" e "Base"

Metodo	Precondizioni	Postcondizioni
cambiaRuolo(utente, promotore, nuovoRuolo)	<ul style="list-style-type: none"> <li>• un amministratore può essere declassato a utente comune unicamente dall'amministratore che lo ha promosso</li> <li>• se l'attributo ruolo dell'utente ha il valore "Amministratore", il suo ruolo può essere</li> </ul>	l'attributo ruolo dell'utente assume il valore di nuovoRuolo

	cambiato solo se promotore è diverso dall'attributo promossoDa dell'utente <ul style="list-style-type: none"> <li>il promotore deve avere ruolo "Amministratore"</li> </ul>	
--	--	--

## Autenticazione

Metodo	Precondizioni	Postcondizioni
verificaOAuth()		
generaToken()		
ottieniUtente() : Utente		
logout()		

## Catalogo

Metodo	Precondizioni	Postcondizioni
aggiornaCatalogo() : Attività[0...N]		il catalogo viene aggiornato
filtra(cerca : String, etichette : Etichette[0...N]) : Attività[0...N]	<ul style="list-style-type: none"> <li>nella barra di ricerca del titolo non si possono inserire più di 20 caratteri</li> <li>i due valori della durata media sono compresi tra 0 e 999, sono interi e il primo è minore del secondo</li> <li>il numero di partecipanti non può superare 99</li> </ul>	il catalogo viene filtrato secondo le etichette previste
creaAttività(attività : Attività)	<ul style="list-style-type: none"> <li>la descrizione non può superare i 2000 caratteri</li> <li>i due valori della durata media sono compresi tra 0 e 999, sono interi e il primo è minore del secondo</li> <li>il numero di partecipanti non può superare 99</li> <li>il titolo non può superare i 20 caratteri di lunghezza</li> </ul>	viene aggiunta una nuova attività al catalogo

## Lista di attività

Metodo	Precondizioni	Postcondizioni
creaLista() : listaAttività	<ul style="list-style-type: none"> <li>il nome della lista di attività non può superare i 20 caratteri di lunghezza</li> <li>un utente non può creare più di 99 liste di attività</li> </ul>	viene creata una nuova lista di attività



	<ul style="list-style-type: none"> <li>il nome della lista di attività non può essere uguale al nome di un'altra lista già presente</li> <li>il nome della lista di attività non può essere nessuno</li> </ul>	
aggiungiAttività(attività : Attività)	il numero di attività in una lista non può superare 9999	l'attività scelta viene aggiunta alla lista
esporta(formato : String) : File		la lista viene esportata in formato pdf o json
eliminaAttività(indice : int)		l'attività con l'indice scelto viene rimossa dalla lista

## Attività

Metodo	Precondizioni	Postcondizioni
modifica(attivitàModificata : Attività)		l'attività viene modificata come voluto

## Segnalazione

Metodo	Precondizioni	Postcondizioni
inviaSegnala-autore : Utente, attività : Attività, voto : String)	l'attributo messaggio non può superare i 500 caratteri di lunghezza	viene aggiunta una segnalazione per l'attività scelta

## Valutazione

Metodo	Precondizioni	Postcondizioni
inviaValutazione-autore : Utente, attività : Attività, voto : String)	il voto inserito deve essere un numero decimale compreso tra 0 e 5, con scarto di 0.5	<ul style="list-style-type: none"> <li>viene aggiunta la valutazione all'attività scelta</li> <li>cambia la media di voti dell'attività scelta</li> </ul>