

Specifica dei tipi di dato

Tipo posizione: tipo enumerativo:

- {ricercatore, professore associato, professore ordinario}

Specifica della classe Impegno

calcolo_ore_fino_a (d : Data, o: Ora) : Intero > 0

pre-condizioni:

- l'oggetto di invocazione ("this") è coinvolto in almeno una associazione fra "Docente" e "Impegno", con il valore "d" non antecedente al valore "inizio_giorno" o successivo al valore dell'attributo "fine_giorno" o "giorno_fine", ed il valore "o" non deve esser successivo al valore dell'attributo "fine_ora" o "ora_fine"

post-condizioni:

- l'operazione non modifica il livello estensionale (gli oggetti)
- il valore del risultato ("result") è definito come segue:
 - sia IG la data di partenza del valore dell'attributo "inizio_giorno"
 - siano IO le ore da sommare iniziali del valore dell'attributo "inizio_ora"
 - sia FG la data di fine del valore "d" non antecedente al valore di IG e non successiva al valore dell'attributo "fine_giorno" o "giorno_fine"
 - siano FO le ore del valore "o" non antecedente al valore di IG e non successiva al valore dell'attributo "fine_ora" o "ora_fine"
 - sia S il calcolo da sommare a partire dal valore IG fino al valore FG convertito da giorni in ore
- result = S + IO + FO ("il valore di S sommato al valore di IG ed al valore di FO")

Specifica della classe AssenzaConclusa

calcolo_ore() : Intero > 0

pre-condizioni:

- l'oggetto di invocazione ("this") è coinvolto in almeno una associazione fra "Docente" e "Impegno"
- il valore dell'attributo "fine_giorno" non deve esser antecedente al valore dell'attributo "inizio_giorno"

post-condizioni:

- l'operazione non modifica il livello estensionale (gli oggetti)
- il valore del risultato ("result") è definito come segue:
 - sia IG la data di partenza del valore dell'attributo "inizio_giorno"
 - siano IO le ore da sommare iniziali del valore dell'attributo "inizio_ora"
 - sia FG la data di fine del valore dell'attributo "fine_giorno" non antecedente a IG
 - siano FO le ore da sommare finali del valore dell'attributo "fine_ora"
 - sia S il calcolo da sommare a partire dal valore IG fino al valore FG convertito da giorni in ore
- result = S + IO + FO ("il valore di S sommato al valore di IG ed al valore di FO")

Specifica della classe MalattiaConclusa

calcolo_giorni() : Intero > 0

pre-condizioni:

- l'oggetto di invocazione ("this") è coinvolto in almeno una associazione fra "Docente" e "Impegno"
- il valore dell'attributo "fine_giorno" non deve esser antecedente al valore dell'attributo "inizio_giorno"

post-condizioni:

- l'operazione non modifica il livello estensionale (gli oggetti)
- il valore del risultato ("result") è definito come segue:
 - sia IG la data del valore dell'attributo "inizio_giorno" non successiva al valore dell'attributo "fine_giorno"
 - sia FG la data del valore dell'attributo "fine_giorno" non antecedente al valore dell'attributo "inizio_giorno"
 - sia S il calcolo da sommare a partire dal valore IG fino al valore FG
- result = IG + S + FG ("il valore di IG sommato al valore di S ed al valore di FG")

Specifica della classe ImpegnoConcluso

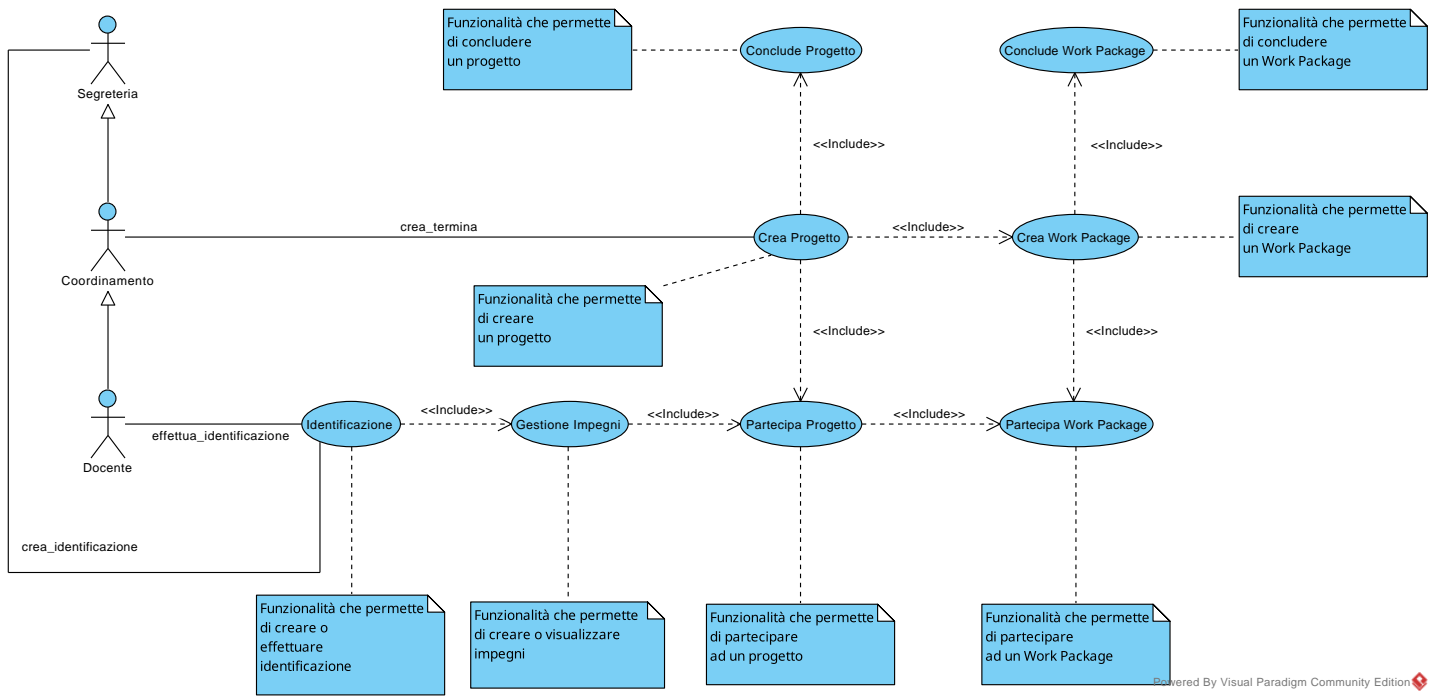
calcolo_ore() : Intero > 0

pre-condizioni:

- l'oggetto di invocazione ("this") è coinvolto in almeno una associazione fra "Docente" e "Impegno"
- il valore dell'attributo "giorno_fine" non deve esser antecedente al valore dell'attributo "inizio_giorno"

post-condizioni:

- l'operazione non modifica il livello estensionale (gli oggetti)
- il valore del risultato ("result") è definito come segue:
 - sia IG la data del valore dell'attributo "inizio_giorno" non successiva al valore dell'attributo "giorno_fine"
 - siano IO le ore da sommare iniziali del valore dell'attributo "inizio_ora"
 - sia FG la data del valore dell'attributo "giorno_fine" non antecedente al valore dell'attributo "inizio_giorno"
 - siano FO le ore da sommare finali del valore dell'attributo "ora_fine"
 - sia S il calcolo da sommare a partire dal valore IG fino al valore FG convertito da giorni in ore
- result = IO + S + FO ("il valore di IO sommato al valore di S ed al valore di FO")



Specifica dello use-case Identificazione

iscrizione(mat : Stringa, nome : Stringa, cognome : Stringa, data_nascita : Data, luogo_nascita : Città, pos_universitaria : {ricercatore, professore associato, professore ordinario}) : Docente

pre-condizioni:

- non esiste alcun oggetto di classe Docente con valore "mat" per l'attributo "matricola"

post-condizioni:

- viene creato e restituito un nuovo oggetto result: Docente con i valori "mat", "nome", "cognome", "data_nascita", "luogo_nascita", "pos_universitaria" per rispettivamente gli attributi: nome, cognome, data_nascita, luogo_nascita, matricola, pos_universitaria

Specifica dello use-case Gestione Impegni

gestisce(i: Impegno, AC: AssenzaConclusa, MC: MalattiaConclusa, IC: ImpegnoConcluso, t: {assenza, malattia, impegno}, di: Data, oi: Ora, df: Data, of: Ora, m: Stringa)

pre-condizioni:

- (i) non è (già) associato a "Docente"
- il valore "df" non è antecedente al valore "di"

post-condizioni:

- viene creato il link (attività_docente) di associazione "Docente", con valori "m", "di", "oi", per rispettivamente gli attributi: motivazione, inizio_giorno, inizio_ora
- il valore (t) se equivalente ad "assenza" ed il valore "df" non è antecedente al valore "di" viene creata e restituito un nuovo oggetto result: AssenzaConclusa con i valori "df", "of" per rispettivamente gli attributi: fine_giorno, fine_ora
- il valore (t) se equivalente a "malattia" ed il valore "df" non è antecedente al valore "di" viene creata e restituito un nuovo oggetto result: MalattiaConclusa con il valore "df" per rispettivamente l'attributo: fine_giorno
- il valore (t) se equivalente ad "impegno" ed il valore "df" non è antecedente al valore "di" viene creata e restituito un nuovo oggetto result: ImpegnoConcluso con i valori "df", "of" per rispettivamente gli attributi: giorno_fine, ora_fine

Specifica dello use-case Crea Progetto

creaP(p: Progetto, acr : Stringa)

pre-condizioni:

- non esiste alcun oggetto di classe Progetto con valore "acr" per l'attributo "acronimo"

post-condizioni:

- viene creato e restituito un nuovo oggetto result: Progetto con i valori "nome", "acr", "dat_in" per rispettivamente gli attributi: nome, acronimo, data_inizio

Specifica dello use-case Crea Work Package

creaWP(WP : WorkPackage, nome : Stringa)

pre-condizioni:

- non esiste alcun oggetto di classe WorkPackage con valore "nome" per l'attributo "nome" e deve esser coinvolto in almeno un link ("pro_work") dell'associazione Progetto

post-condizioni:

- viene creato e restituito un nuovo oggetto result: WorkPackage con i valori "nome", "dat_in" per rispettivamente gli attributi: nome, data_inizio

Specifica dello use-case Conclude Progetto

fineP(PC: ProgettoConcluso, d: Data)

pre-condizioni:

- il valore "d" deve esser successivo o equivalente al valore dell'attributo ("data_inizio")
- non esiste alcun oggetto di classe ProgettoConcluso con valore "d" per l'attributo "data_fine"

post-condizioni:

- viene creato e restituito un nuovo oggetto result: ProgettoConcluso con il valore "d", per rispettivamente l'attributo: data_fine

Specifica dello use-case Conclude Work Package

fineWP(WP: WorkPackageConcluso, d: Data)

pre-condizioni:

- il valore "d" deve esser successivo o equivalente al valore dell'attributo ("data_inizio")
- non esiste alcun oggetto di classe WorkPackageConcluso con valore "d" per l'attributo "data_fine"

post-condizioni:

- viene creato e restituito un nuovo oggetto result: WorkPackageConcluso con il valore "d", per rispettivamente l'attributo: data_fine