

R1. Tactual:

1. Claus: (Persona, nom); (Data, data); (Restaurant, nom); (Menú, nom)
2. Els menús de les reserves han de ser menús ofertats pel restaurant on s'ha fet la reserva.
3. La capacitat d'un restaurant ha de ser més gran que el sumatori dels assistents de les reserves de aquell restaurant per una data determinada.
4. Tots els atributs de tipus Integer i Float han de ser positius
5. El numAssistents d'una reserva de grup ha de ser més gran que 5
6. En una reserva ordinària el numAssistents ha de coincidir amb la quantitat de menús demanats informació derivada:
 - preResGrp: numAssistents de la reserva multiplicat per el valor del preu del menú demanat per la reserva de grup corresponent, menys el %DesocGrup del restaurant corresponent.
7. HaReservat: associa els restaurants on la persona ha fet reserves.
8. A més, en donar el següent contracte d'especificació, per l'establiment atReservaGrp:
 - Context Sistema: atReservaGrp(nomP: String, data: Date, nomR: String, nAss: Integer, nomMenu: String) pre persona-existia: la persona nomP existia.
 - pre restaurant-existia: el restaurant nomR existia.
 - pre menú-existia: el menú nomMenu existia.
 - post es crea una reserva de grup i es formen totes les associacions corresponents.

R2. Tactual:

1. Claus: (Persona, nom); (Data, data); (Restaurant, nom); (Menú, nom)
2. Els menús de les reserves han de ser menús ofertats pel restaurant on s'ha fet la reserva.
3. La capacitat d'un restaurant ha de ser més gran que el sumatori dels assistents de les reserves d'aquell restaurant per una data determinada.
4. Tots els atributs de tipus Integer i Float han de ser positius
5. El numAssistents d'una reserva de grup ha de ser més gran que 5
6. En una reserva ordinària el numAssistents ha de coincidir amb la quantitat de menús demanats informació derivada:
 - preResOrd: numAssistents de la reserva multiplicat per el valor del preu del menú demanat per la reserva de grup corresponent, menys el %DesocOrd del restaurant corresponent.
7. HaReservat: associa els restaurants on la persona ha fet reserves.
8. A més, en donar el següent contracte d'especificació, per l'establiment atReservaOrd:
 - Context Sistema: atReservaOrd(nomP: String, data: Date, nomR: String, nAss: Integer, nomMenu: String) pre persona-existia: la persona nomP existia.
 - pre restaurant-existia: el restaurant nomR existia.
 - pre menú-existia: el menú nomMenu existia.
 - post es crea una reserva de grup i es formen totes les associacions corresponents.

Context Sistema: atReservaGrp(nomP: String, data: Date, nomR: String, nAss: Integer, nomMenu: String)
accepta:
 [PersonasExistia] la persona nomP no existia.
 [RestaurantNoExistia] el restaurant nomR no existia.
 [MenuNoExistia]: el menú nomMenu no existia
post:
 es crea una reserva de grup identificada per nomP i data i es formen totes les associacions corresponents.

Context Sistema: atReservaOrd(nomP: String, data: Date, nomR: String, nAss: Integer, nomMenu: String)
accepta:
 [PersonasExistia] la persona nomP no existia.
 [RestaurantNoExistia] el restaurant nomR no existia.
 [MenuNoExistia]: el menú nomMenu no existia
post:
 es crea una reserva de grup identificada per nomP i data i es formen totes les associacions corresponents.

```

classDiagram
    class Jugador {
        -id: String
        -nom: String
    }
    class Guàrdia {
        -data: Date
    }
    class Partida {
        -data: Date
    }
    class Història {
        -titol: String
        -resum: String
    }
    class Personatge {
        -nom: String
        -edat: Int
        -puntsVida: Int
    }
    class KO {
        -tempsDesc: Int
    }
    class Enemic {
        -nom: String
        -lider: Bool
        -puntsDeVida: Natural
    }
    class TerribilitatPntg {
        -valor: Int
    }
    class Habitatz {
        -nom: String
        -descripció: String
    }
    class Humà {
        -inteligència: Int
    }
    class Monstre {
    }
    class Raça {
        -nom: String
        -força: Int
    }

    Jugador "1" -- "0..1" Guàrdia : Crea
    Jugador "1" -- "0..1" Partida : Organitza
    Jugador "1" -- "0..1" Història : titol, resum
    Guàrdia "1" -- "0..20" Partida : Organitza
    Partida "1" -- "1" Història : titol, resum
    Personatge "1" -- "0..1" KO : tempsDesc
    Personatge "1" -- "0..1" Enemic : nom, lider, puntsDeVida
    Personatge "1" -- "0..1" TerribilitatPntg : valor
    Personatge "1" -- "0..1" Habitatz : nom, descripció
    Enemic "0..1" -- "0..1" Humà : tipus
    Enemic "0..1" -- "0..1" Monstre : tipus
    Enemic "0..1" -- "0..1" Raça : tipus
    Raça "1" -- "0..1" Monstre : tipus
    
```

Restriccions textuale:

- 1- Class externes: {Jugador, id}, {Habitatz, nom}, {Història, titol}, {Raça, nom}, {Personatge, nom}
- 2- No hi pot haver dos Enemic amb el mateix nom dins de la mateixa Partida.
- 3- No hi pot haver dos Habitatz amb el mateix Personatge i la mateixa Habitatz.
- 4- No hi pot haver dues Partida amb la mateixa data, Guàrdia i Història.
- 5- HabitatzPers valor < 100

Operació: CrearPartidaBasica(idG: String, titolH: String, nomP: data, date, nomE: String, pv: Natural, humà? bool, intel: Natural, nomR: String)

Excepcions:

- GuàrdiaNoExisteix: El guàrdia amb id = idG no existeix.
- HistòriaNoExisteix: La història amb titol = titolH no existeix.
- PersonatgeNoExisteix: El personatge amb nom = nomP no existeix.
- RaçaNoExisteix: L'entitat és fals i la tipus de nom = nomR no existeix.
- PartidaNoExisteix: La partida de identificat per idG, titolH i data ja existeix.
- OPartida: El personatge nomP ja participa a 20 partides.
- MassaPoderia: El personatge nomP ja té 5 o més habitatz amb valors superiors o iguals a 90.
- MassejVegent: El guàrdia idG ja ha dirigit la història titolH un mínim de 10 vegades.

Post:

- Es crea una nova partida identificat per idG, nomH i data, relacionada amb els elements que correspon.
- Es crea un enemic de nom = nomE, lider = true i puntsVida = pv i s'associa a la partida.
- Si humà? és cert, l'enemic serà del subtipus Humà i tindrà un valor d'intel·ligència = intel.
- Si humà? és fals, l'enemic serà del subtipus Monstre i s'haurà de seleccionar amb la raça de nom = nomR.

Operació: ObtenirPersonatgesRestants (idG: String), Set(String)

Excepcions:

- GuàrdiaNoExisteix: No existeix el guàrdia amb id = idG.
- body: Retorna, del guàrdia idG, els noms dels personatges que ha creat tals que:
 - No estan KO i tenen una habitatz de nom "titol" amb un valor inferior o igual a 30.

Han participat a un mínim de 2 partides cadascun, tals que, per cada partida, es compleix almenys una de les condicions següents:

- o La suma de la intel·ligència dels enemics de tipus humà és superior a 500.
- o La suma de la força de la raça dels enemics de tipus monstre és superior a 300.

[illegible][illegible]