

PROP – Primera entrega v1.0

ADRIÀ CABEZA SANT'ANNA [adria.cabeza](#)

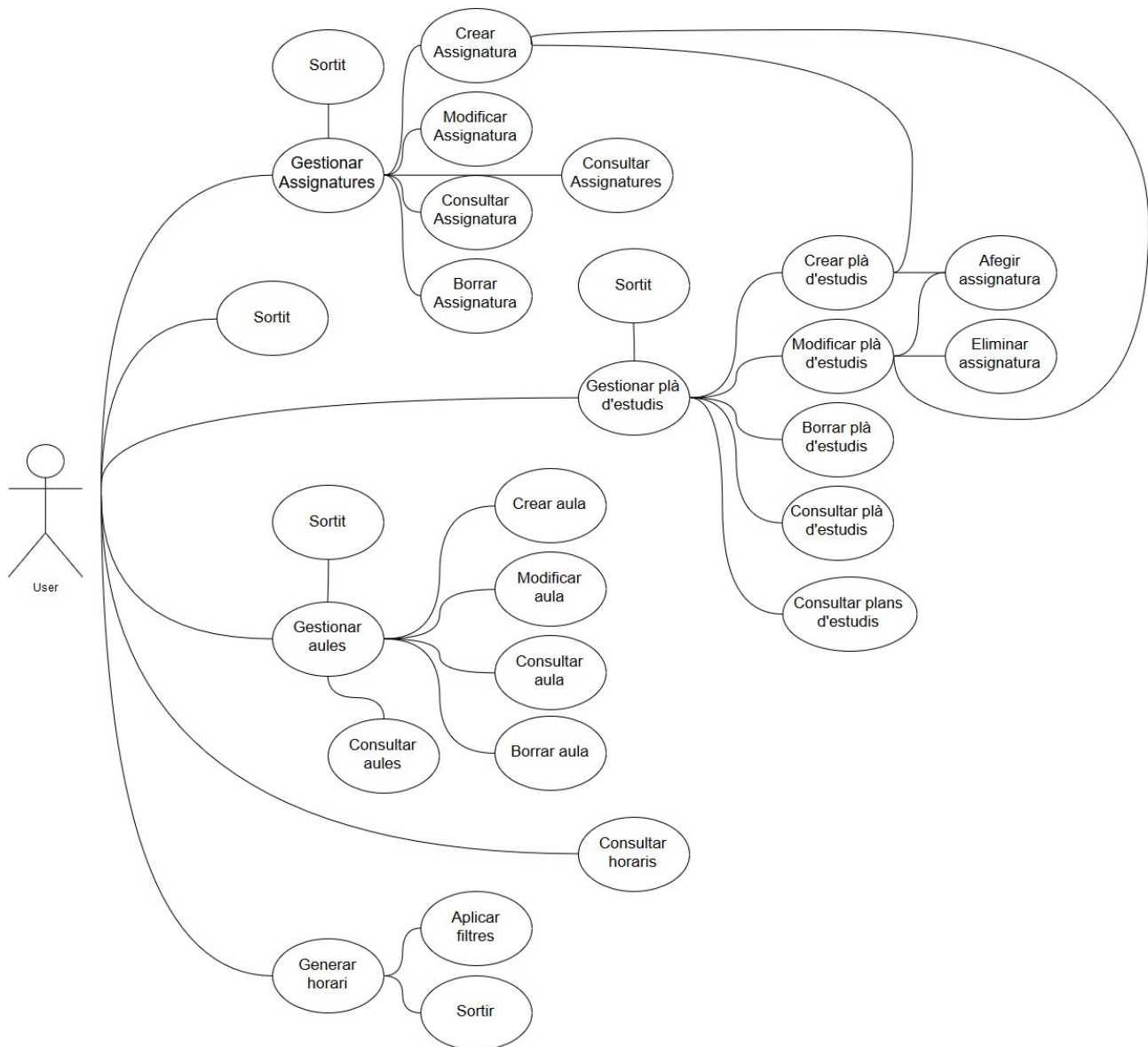
AINA GARCIA ESPRIU [aina.garcia.espriu](#)

ANTONI RAMBLA MOISET [antoni.rambla](#)

Table of Contents

| | |
|---|----|
| CASOS D'ÚS – Esquema..... | 1 |
| CASOS D'ÚS – Explicació..... | 2 |
| DIAGRAMA UML – Esquema..... | 5 |
| DIAGRAMA UML – Explicació de les classes..... | 6 |
| DESCRIPCIÓ DE LES ESTRUCTURES DE DADES..... | 9 |
| DESCRIPCIÓ DELS ALGORITMES..... | 10 |
| Backtracking cronològic..... | 10 |
| Pseudocodi del backtracking cronològic:..... | 11 |
| Forward Checking..... | 11 |
| Pseudocodi del forward checking:..... | 12 |

CASOS D'ÚS - Esquema



CASOS D'ÚS – Explicació

Gestionar grups

L'usuari visualitzarà tots els grups d'una assignatura i des d'allà podrà consultar-los, modificar-los o esborrar-los. A més també pot crear des d'allà un grup.

Crear grups

L'usuari proporcionarà la informació necessària per tal de crear grups per una assignatura en concret, com ara el número de grup, la capacitat, i la quantitat de subgrups. El sistema donarà d'alta aquests grups per l'assignatura en qüestió.

Modificar grups

L'usuari proporcionarà la informació necessària per tal de modificar els grups per una assignatura en concret, com ara els nous valors de número de grup, la capacitat, i la quantitat de subgrups, si algun valor no ha estat introduït es mantindrà el número anterior. El sistema donarà d'alta aquests grups per l'assignatura en qüestió.

Esborrar grups

Quan l'usuari borra grups per una assignatura el sistema en borra les seves instàncies.

Consultar grups

Permet consultar tota la informació dels grups i subgrups d'una assignatura en concret.

Gestionar Assignatura

L'usuari visualitzarà totes les assignatures, des d'allà podrà crear-ne una, consultar una assignatura en concret o totes, modificar una assignatura o esborrar-la.

Crear Assignatura

Quan l'usuari vulgui crear una assignatura nova, introduirà el seu nom i el quadrimestre al que pertany, adicionalment podrà afegir la informació sobre les sessions de teoria i laboratori o ho podrà fer més tard a través de modificar assignatura i accedir a la opció de crear nous grups i posar-la en un pla d'estudis vigent. El sistema dona d'alta aquesta assignatura.

PROP – Primera entrega

Modificar assignatura

Permet modificar la informació de les sessions de teoria i laboratori de l'assignatura. L'usuari introdueix la duració de les sessions, el número de sessions setmanals i el tipus d'aula requerit per el tipus de sessió que vol modificar, si no introdueix tots els parametres es carreguen els anteriors si n'hi havia. En cas de que no hi hagués registrada informació de laboratori o teoria, el sistema en dona d'alta una nova instancia.

Esborrar assignatura

Quan l'usuari vol esborrar una assignatura, n'introdueix el nom i el sistema esborra la seva instancia.

Consultar assignatura

Permet a l'usuari consultar la informació completa d'una assignatura, es a dir, el nom, quadrimestre, grups i subgrups i informació de les seves respectives sessions de laboratori i teoria. L'usuari introdueix el nom de l'assignatura i el sistema li mostra tota la informació en cas de que aquesta existeixi.

Consultar assignatures

Permet a l'usuari consultar una llista de totes les assignatures existents en el sistema. El sistema mostra tots els noms de les assignatures existents.

Gestionar pla d'estudis

Permet a l'usuari accedir a les funcionalitats de crear, modificar, borrar i consultar plans d'estudi.

Crear pla d'estudis

L'usuari podrà crear un nou pla d'estudis vigent introduint el nom de la titulació al que pertany i l'any en que entra en vigor. També podrà accedir a crear assignatures i afegir assignatures existents al pla si aixó ho desitja. El sistema dona d'alta una instancia del pla d'estudis i hi afegeix les assignatures que l'usuari li hagi assignat si n'hi ha.

Modificar pla d'estudis

L'usuari pot modificar les assignatures que formen part del pla d'estudis que vol modificar o marcar-lo com a obsolet. El sistema gestiona els canvis en les assignatures o marca el pla com a obsolet.

Borrar pla d'estudis

L'usuari pot introduir el nom del pla d'estudis que vol donar de baixa del sistema i el sistema l'esborrara en cas que aquest pla estigui anteriorment marcat com a obsolet. No podrà esborrar un pla d'estudis vigent.

Consultar pla d'estudis

L'usuari introduirà el nom del pla d'estudis que vol consultar i el sistema li mostrarà la informació d'aquest (nom, any i obsolet o no) i els noms de les assignatures que en formen part.

Consultar plans d'estudis

El sistema mostra a l'usuari una llista de noms, any d'entrada en vigor i vigencia de tots els plans.

PROP – Primera entrega

Gestionar Aules

Permet a l'usuari accedir a les funcionalitats de crear noves aules, modificar-les, consultar les existents o esborrar-ne.

Crear aula

Permet crear una aula, assignar-li el número o nom identificador i la seva capacitat.

Modificar aula

L'usuari pot introduir els valors de l'aula que vol modificar, es a dir la seva capacitat o el tipus d'aula. El sistema actualitza la informació de l'aula o deixa els valors que no s'hagin modificat tal i com estaven anteriorment.

Esborrar aula

L'usuari introdueix el nom de l'aula que vol esborrar i el sistema en dona de baixa la instància.

Consultar aula

L'usuari indica el nom de l'aula que vol consultar i el sistema li mostra la informació referent a aquesta, es a dir, el nom, la seva capacitat i el tipus d'aula que és.

Consultar aules

El sistema mostra una llista amb els noms de totes les aules donades d'alta, el seu tipus i la seva capacitat.

Aplica filtres per a la generació de nous horaris

L'usuari selecciona quins filtres i restriccions s'aplicaràn a l'hora de generar un horari i es permet accedir a la funcionalitat de generar horari. El sistema inicialitza aquests filtres i restriccions i intenta generar un horari si així se li ha indicat.

Generar Horari

El sistema intenta generar un horari amb les restriccions i filtres que hi hagi inicialitzats si n'hi ha.

Veure horari resultant

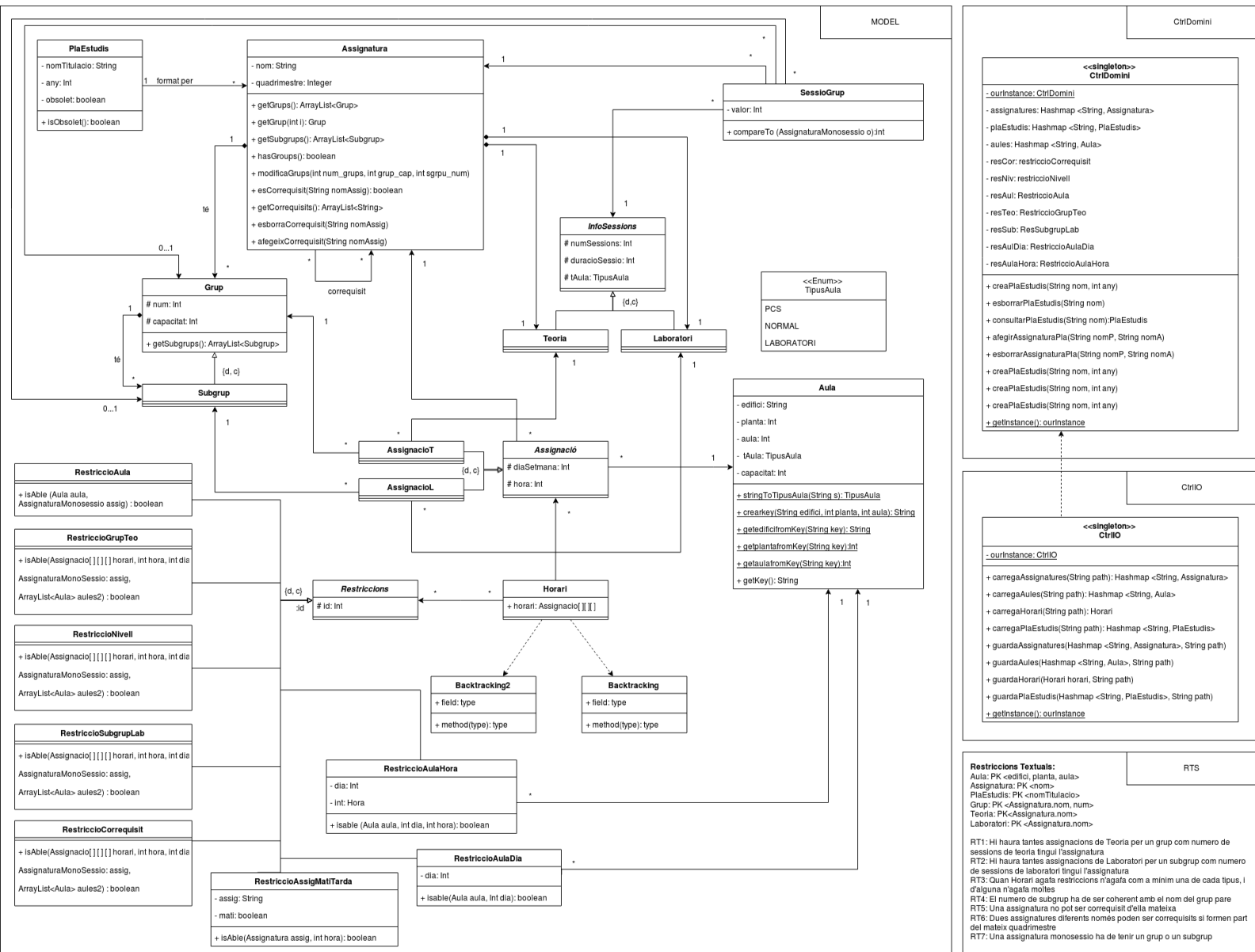
El sistema mostra a l'usuari visualitzar l'horari resultant si n'hi ha.

Sortir

Permet a l'usuari sortir del programa des de qualsevol punt anteriorment descrit.

PROP - Primera entrega

DIAGRAMA UML - Esquema



***Hi ha una versió al pendrive del UML en .png per veure'l millor**

DIAGRAMA UML – Explicació de les classes

PlaEstudis

Un plà d'estudis està identificat pel nom de la seva titulació. Té un any d'entrada en vigor i un valor de si està vigent o obsolet. Inclou també assignatures que només poden estar a un pla d'estudis a la vegada.

Assignatura

Una assignatura està identificada pel seu nom. Té un quadrimestre al qual en forma part i pot tenir altres assignatures que són correquisits seus, i que per tant, formen part del mateix quadrimestre. També pot tenir associada informació sobre els seus grups i subgrups i informació sobre com son les seves sessions de teoria i laboratori.

Grup

Un grup està identificat pel nom de l'assignatura al qual pertany i el número de grup. Té informació com la seva capacitat i pot tenir associats un número de subgrups. La creació d'un grup depen doncs, de tenir una assignatura del qual en forma part.

Subgrup

Un subgrup està identificat pel nom de l'assignatura al qual pertany i el número de subgrup. Té informació com la seva capacitat. El seu nom ha de ser coherent amb el nom del grup pare i la seva existència va lligada a l'existència del grup i de l'assignatura al qual pertany.

InfoSessions

InfoSessions fa referència a la informació de com seran les sessions de l'assignatura, és a dir, quantes n'hi haurà per setmana, de quantes hores seran les sessions i quin tipus d'aula requereixen per ser realitzades. Estan identificades per l'assignatura de la qual formen part i la seva creació requereix de l'existència de l'assignatura.

Teoria

Informació sobre les sessions de teoria, identificada pel nom de l'assignatura, fa referència a com seran les sessions de teoria concretament.

Laboratori

Informació sobre les sessions de laboratori, identificada pel nom de l'assignatura, fa referència a com seran les sessions de laboratori concretament.

Assignació

Una assignació fa referència a un slot d'un horari, és a dir, una aula i una assignatura en un dia i hora en concret.

AssignacioT

Una assignació de teoria inclou un grup i n'hi haurà tantes com numero de sessions *duració de les sessions hi hagi en la informació de sessions de teoria per cada subgrup. El tipus d'aula requerit ha de ser el mateix que l'assignat.

PROP – Primera entrega

Assignació

Una assignació de laboratori inclou un subgrup i n'hi haurà tantes com numero de sessions *duració de les sessions hi hagi en la informació de sessions de laboratori per cada subgrup. El tipus d'aula requerit ha de ser el mateix que l'assignat.

Aula

Una aula està identificada pel seu nom format pel nom de l'edifici, la planta i el número d'aula. Conté informació com la seva capacitat i atributs com ara la planta, que ens pot servir per exemple, per mirar si es apta per gent amb mobilitat reduïda i el tipus d'aula que ens ajuda a comparar-lo amb el requerit per dur a terme assignacions.

SessióGrup

Una sessió grup és una classe auxiliar d'assignatura que tracta per separat cada una de les sessions que aquesta té de teoria o de laboratori, per cada grup. Esta identificada per l'assignatura, el grup i un valor que simbolitza el número de sessió.

Horari

Un horari és un conjunt de «slots» en tres dimensions que fa referència a l'assignatura i grup que està donant classe per cada aula, dia i hora en concret.

Restriccions

Una restricció és una característica o prohibició aplicada a l'hora de generar horaris. Esta identificada cada tipus per un enter.

RestriccióAula

Aquesta restricció s'assegura que el tipus d'aula de la sessió que estem evaluant es del mateix tipus que el tipus que el de la aula al qual l'assignarem.

RestriccióGrupTeo

Aquesta restricció assegura que no es pugui fer classe de teoria si ja hi ha un laboratori d'un dels seus subgrups en marxa o un solapament de teories

RestriccióNivell

En aquesta restricció es comprova que no hi hagi solapaments de sessions d'assignatures que formen part del mateix nivell o quadrimestre.

RestriccióAssigMatiTarda

Restricció on es comprova que en una sessió d'una determinada assignatura és realitzi de matí o de tarda.

RestriccióSubgrupLab

Restricció on es comprova que en una sessió de laboratori d'un determinat subgrup no hi hagi solapaments.

RestriccióGrupTeo

Restricció on es comprova que en una sessió de teoria d'un determinat grup no hi hagi solapaments.

RestriccióCorrequisit

Restricció on es comprova que una sessió d'una assignatura d'un determinat grup no hi hagi conflictes amb correquisits.

PROP – Primera entrega

RestriccioAulaDia

Restricció on es comprova si es pot realitzar una assignació en una determinada aula en un determinat dia.

RestriccioAulaHora

Restricció on es comprova si es pot realitzar una assignació en una determinada aula en una determinada hora d'un determinat dia.

Backtracking

Classe on es crea un horari de forma bàsica aplicant restriccions.

Backtracking2

Classe on es crea un horari de forma més òptima aplicant restriccions. Per més detalls, veure descripció de les estructures de dades i algoritmes.

DESCRIPCIÓ DE LES ESTRUCTURES DE DADES

Les dues estructures principals que fem servir en el nostre projecte són les següents:

- La d'horari
- La de sessions i possibilitats

El nostre horari és un una matriu 3D d'assignacions on cada eix representa els dies, les hores i les aules que ja s'han assignat.

```
private Assignacio[][][] horari;
```

Per a guardar el domini de cada sessió de cada grup fem servir un HashMap on la clau és una sessió d'un grup que hem d'assignar i el valor és una llista d'aules possibles representades com a enters.

```
HashMap<SessioGrup, ArrayList<ArrayList<ArrayList<Integer>>>> possibilitats
```

La llista està feta de manera que pot no haver-hi res, de manera que no és una via possible o haver-hi una aula on cada vegada que interioritzés més per l'estructura de dades cada índex representaria el dia i l'hora que és possible.

DESCRIPCIÓ DELS ALGORITMES

El nostre projecte crea l'horari mitjançant un algorisme que s'estructura en dues parts principals:

- Backtracking cronològic
- Forward checking

Backtracking cronològic

En aquesta part explicarem el backtracking cronològic que és la primera part del nostre algorisme.

Com a paràmetres de l'algorisme tenim l'horari, en un inici buit, que anem emplenant a mesura que va avançant l'algorisme, l'estructura de dades esmentada anteriorment on hi ha guardades totes les sessions que s'han d'assignar i les seves possibles aules en determinats dies i hores, i un iterador que es es va movent per aquesta estructura de dades.

L'algorisme funciona de manera que itera pels dies, hores i aules possibles que pot tenir una sessió, si el gap de l'horari està buit, comprova les restriccions i si les compleix l'assigna.

Llavors propaga les possibilitats per l'estructura de dades on tenim guardades les possibilitats. Llavors podem passar dues coses:

- Aquesta propagació provoca que hi hagi alguna sessió sense possibilitats: Esborrem l'assignació efectuada anteriorment, recuperem les possibilitats que hi havien abans de propagar i continuem el bucle.
- Es fa una poda de domini per totes les sessions, Esborrem de l'estructura de dades les possibilitats de la sessió ja assignada i continuem amb una crida recursiva per a la següent sessió.

En el cas que fem una crida recursiva per a la següent sessió també tenim dues possibilitats:

- No hi hagi cap error i l'algorisme continuï fins que estigui tot ple
- Falli en algun lloc: en aquest cas recuperem les possibilitats que teníem abans de propagar, esborrem totes les assignacions que havíem fet i continuem el bucle.

Finalment si s'han realitzat tots els bucles, retornem que no s'ha pogut efectuar la sessió que estàvem comprovant. En el cas que sigui la primera sessió significarà que no es pot efectuar l'horari de cap manera, altrament, es farà backtracking.

Un cop l'iterador arribi a l'última sessió que s'ha d'assignar retorna true perquè vol dir que ha acabat de crear l'horari.

PROP – Primera entrega

Pseudocodi del backtracking cronològic:

```
creaHorari(iterador per sessions, horari, HashMap<Sessió, ArrayList<possibilitats>> pos)
if(iterador == última sessió ) return true "S'HA FET L'HORARI"
for(d -> dies_possibles)
    for(h->d.hores_possibles)
        for(a->h.aules_possibles){
            if(no està assignat aquest slot al horari){
                if(es compleixen les restriccions){
                    Assignar a l'horari a l'slot a,d,h durant la duració de la sessió
                    propagarPossibilitats()
                    if(propagarPossibilitats() fa que alguna sessió no tingui opcions)
                        Esborrar assignació feta a a,d,h durant la duració de la sessió
                        continuarBucle()
                }
            }
            else
                Esborrar de possibilitats la sessió assignada i la seva informació
                creaHorari(iterador+1, horari, pos)

            if(creaHorari() acaba per a totes les sessions)
                return true "S'HA FET L'HORARI"
            else {
                Recuperar la sessió i la seva informació, posar-la a possibilitats
                Esborrar assignació feta a a,d,h durant la duració de la sessió
                continuarBucle()
            }
        }
return false "NO ES POT FER AIXÍ"
```

Forward Checking

Tal i com hem comentat anteriorment en l'explicació del backtracking cronològic el nostre algorisme no itera envers a totes les possibles aules de la universitat, tots els dies i totes les hores; sinó que només ho fa a través de les que la sessió té com a possibles. Aquestes aules a hores a dies possibles s'inicialitzen en la primera poda de domini. Si aquesta primera ja ens crea una sessió sense possibilitats, sabem que l'horari no es pot efectuar. A més, després de cada assignació fa una poda de domini en el que es propaguen les possibilitats. Així a mesura que avanci l'algorisme farem menys comprovacions i menys iteracions.

L'algorisme funciona de la següent manera:

Després de fer una assignació hem en el backtracking cronològic cridem a la propagació de possibilitats amb l'aula, el dia, la hora de l'assignació acabada de realitzar, la sessió i l'estructura de dades on guardem totes les possibilitats de totes les sessions.

Llavors per a cada sessió efectuem les diferents restriccions i si alguna ens dóna que l'aula que ens ha entrat com a paràmetre no hauria d'estar com a possibilitat de la sessió la esborrem. Parem de mirar restriccions i continuem. Si en un moment arribem a tenir que una sessió no té cap possibilitat parem i retornem que no es pot seguir fent l'horari d'aquesta manera. Altrament acabem retornant les possibilitats amb la propagació efectuada.

PROP - Primera entrega

Pseudocodi del forward checking:

```
propagarPossibilitats(aula, dia, hora, HashMap<Sessió, ArrayList<possibilitats>> pos)
for(i->totes les sessions)
  forAll restriccions:
    b = comprovarRestriccio()
    if(b ens dóna que hem d'esborrar l'aula del dia i hora)
      La esborrem dintre les possibilitats i parem de comprovar restriccions
    if(la sessió acaba no tenint cap possibilitat)
      return false "no es pot continuar"
return true "es pot continuar"
```