Liste des fonctions Damien

# JFrame\_Structure

add\_ECUE(code\_UE : String, libelle\_ECUE : String, coeff\_ECUE : float) : null //Permet d’ajouter une ECUE dans la liste de l’UE de code « code\_UE »

add\_UE(code\_Semestre : String, libelle\_UE : String, ects\_UE : integer) : integer

add\_Semestre(code\_Etape : String, libelle\_Semestre : String, ects\_Semestre : integer) : integer

put\_Coeff\_ECUE(code\_ECUE : String) : null //Affiche le coefficient de l’ECUE

put\_ECTS\_UE(code\_UE : String) : null

put\_ECTS\_Semestre(code\_Semestre : String) : null

put\_ECTS\_Etape(code\_Etape : String) :null

put\_Libelle\_ECUE(code\_ECUE : String) : null //Affiche le coefficient de l’ECUE

put\_Libelle\_UE(code\_UE : String) : null

put\_Libelle\_Semestre(code\_Semestre : String) : null

put\_Libelle\_Etape(code\_Etape : String) :null

delete\_ECUE(code\_ECUE : String) : null /Supprimer l’ECUE

delete\_UE(code\_UE : String) : null

delete\_Semestre(code\_Semestre : String) : null

affiche\_Liste\_Structure(utilisateur : String) : null //Affiche la liste des structures de l’user

# JFrame\_Eleve

affiche\_Liste\_Structure(utilisateur : String) : null //Affiche la liste des structures de l’user

# JFrame\_Enseignant

affiche\_Liste\_Structure(utilisateur : String) : null //Affiche la liste des structures de l’user

# JFrame\_Note

affiche\_Liste\_Structure(utilisateur : String) : null //Affiche la liste des structures de l’user

# Facade\_Structure

add\_ECUE(code\_UE : String, libelle\_ECUE : String, coeff\_ECUE : float) : integer //Renvoie 0 si l’ajout c’est bien passé, -1 sinon

add\_UE(code\_Semestre : String, libelle\_UE : String, ects\_UE : integer) : integer

add\_Semestre(code\_Etape : String, libelle\_Semestre : String, ects\_Semestre : integer) : integer

get\_Coeff\_ECUE(code\_ECUE : String) : float //Récupère le coeff d’une ECUE

get\_ECTS\_UE(code\_UE : String) : integer

get\_ECTS\_Semestre(code\_Semestre : String) : integer

get\_ECTS\_Etape(code\_Etape : String) :integer

get\_Libelle\_ECUE(code\_ECUE : String) : String //Récupère le libelle d’une ECUE

get\_Libelle\_UE(code\_UE : String) : String

get\_Libelle\_Semestre(code\_Semestre : String) : String

get\_Libelle\_Etape(code\_Etape : String) : String

delete\_ECUE(code\_ECUE : String) : null //Supprimer une ECUE (+note ECUE mais pas prof ni elève)

delete\_UE(code\_UE : String) : null //Supprime une UE(+toutes ses ECUE + note + pts jury)

delete\_Semestre(code\_Semestre : String) : null //Supprimer un semestre + UE + note + pts jury

# Façade\_Menu

get\_All\_Etape\_User(utilisateur : String) : vector<String> //Récupère toutes les étapes auxquelles à accès l’utilisateur (il pourra voir toutes les données de l’étape (y compris les semestre + UE + ECUE)

get\_All\_Semestre\_User(utilisateur : String, v\_Etape : vector<String>) : vector<String> //Récupère les semester de l’user (au cas ou il dirige pas une étape, il peut voir tous sous le (ou les) semester (donc pas l’étape)

get\_All\_UE\_User(utilisateur : String, v\_Etape : vector<String>, v\_Semestre : vector<String) : vector<String> //Même chose un cran plus bas, il ne peut voir que son (ou ses)UE et leurs ECUE

get\_All\_ECUE\_User(utilisateur : String, v\_ Etape : vector<String>, v\_Semestre : vector<String, v\_UE : vector<String>) : vector<String> //ne vois que son ou ses ECUE

# Gestion\_Structure

add\_ECUE(code\_UE : String, libelle\_ECUE : String, coeff\_ECUE : float) : integer //Renvoie 0 si l’ajout c’est bien passé, -1 sinon

add\_UE(code\_Semestre : String, libelle\_UE : String, ects\_UE : integer) : integer

add\_Semestre(code\_Etape : String, libelle\_Semestre : String, ects\_Semestre : integer) : integer

get\_Coeff\_ECUE(code\_ECUE : String) : null

get\_ECTS\_UE(code\_UE : String) : null

get\_ECTS\_Semestre(code\_Semestre : String) : null

get\_ECTS\_Etape(code\_Etape : String) :null

get\_All\_Etape\_User(utilisateur : String) : vector<String> //Cf Façade Structure

get\_All\_Semestre\_User(utilisateur : String, v\_Etape : vector<String>) : vector<String>

get\_All\_UE\_User(utilisateur : String, v\_Etape : vector<String>, v\_Semestre : vector<String) : vector<String>

get\_All\_ECUE\_User(utilisateur : String, v\_ Etape : vector<String>, v\_Semestre : vector<String, v\_UE : vector<String>) : vector<String>

delete\_ECUE(code\_ECUE : String) : null //Supprimer une ECUE (cf Façade)

delete\_UE(code\_UE : String) : null

delete\_Semestre(code\_Semestre : String) : null

get\_Libelle\_ECUE(code\_ECUE : String) : String

get\_Libelle\_UE(code\_UE : String) : String

get\_Libelle\_Semestre(code\_Semestre : String) : String

get\_Libelle\_Etape(code\_Etape : String) : String

# Abstract\_Structure

get\_Code() : String

get\_Libelle() : String

# Abstract\_Etape

get\_Semestre\_Etape() : vector<Semestre> //Renvoie la liste des semestres de l’Etape

get\_ECTS\_Etape() : integer

add\_Semestre(semestre : Semestre) : null

delete\_Note\_Etape() : null

# Abstract\_Semestre

get\_UE\_Semestre() : vector<UE> //Renvoie la liste des UE du semestre

get\_ECTS\_Semestre() : integer

add\_UE(ue : UE) : null

delete\_Note\_Semestre() : null

delete\_Points\_Jury\_Semestre() : null

# Abstract\_UE\_

get\_ECUE\_UE() : vector<ECUE> //Renvoie la liste des UE du semestre

get\_ECTS\_UE() : integer

add\_ECUE(ecue : ECUE) : null

delete\_Note\_UE() : null

delete\_Points\_Jury\_UE() : null

# Abstract\_ECUE

get\_Coeff\_ECUE() : float

delete\_Note\_ECUE() : null

# Abstract\_Factory

create\_ECUE(code\_ECUE : String, libelle\_ECUE : String, coeff\_ECUE : float) : ECUE //Créer une ECUE

create\_UE(code\_UE : String, libelle\_UE : String, ects\_ECUE : integer) : UE

create\_Semestre(code\_Semestre :String, libelle\_Semestre :String, ects\_Semestre :integer) : Semestre

create\_Etape(code\_Etape : String, libelle\_Etape : String, ects\_Etape : integer) : Etape

save\_ECUE(ecue : ECUE) : integer //Sauvegarde une ECUE (en BDD ou texte)

save\_UE(ue : UE) : integer //Sauvegarde une UE et ses ECUE

save\_Semestre(semestre : Semestre) : integer

save\_Etape(etape : Etape) : integer

create\_Etape\_User(utilisateur : String) : vector<Etape> //Créer une liste d’étape (qui peut être vide) auxquelles à acces l’utilisateur

create\_Semestre\_User(utilisateur : String, v\_Etape : vector<String>) : vector<Semestre>

create\_UE\_User(utilisateur : String, v\_Etape : vector<String>, v\_Semestre: vector<Semestre>) : vector<UE>

create\_ECUE\_User(utilisateur : String, v\_Etape : vector<String>, v\_Semestre: vector<Semestre>, v\_UE : vector<UE>) : vector<ECUE>

delete\_Note\_ECUE(ecue : ECUE) : integer

delete\_Note\_UE(ue : UE) : integer

delete\_Note\_Semestre(semestre : Semestre) : integer

delete\_Note\_Etape(etape : Etape) : integer

delete\_ECUE(ecue : ECUE) : integer

delete\_UE(ue : UE) : integer

delete\_Semestre(semestre : Semestre) : integer

delete\_Etape(etape : Etape) : integer

delete\_Points\_Jury\_UE(ue : UE) : integer

delete\_Points\_Jury\_Semestre(semestre : Semestre) : integer