Standards de nomenclature et règles

# Interface graphique :

Variables = frmDetSysExp, frmGesSysExp, frmAccAdmin, frmAccUtil

Det : fenêtre details (pop-up)

Gest : fenêtre de gestion

Acc : fenêtre d’accueil

Nom visible = « Détails – Système d’exploitation », « Gestion des systèmes d’exploitation »

Norme des emplacements des boutons

Nouveau/enregistrer | Modification/Details | Annulation/supprimer

# Demandes de modification DBA

Nom du fichier word : Demande\_Modifications\_DBA\_ + id

Ex : Demande\_Modifications\_DBA\_DA1

id : 2 premières lettres de son prénom + #

Ex : DA8 ou GU27

# Base de données

Contraintes :

FK = FK\_tblJeu\_tblGenre\_IdGenre

PK = PK\_tblJeu\_IdJeu

UNIQUE = UN\_tblJeu\_IdJeu

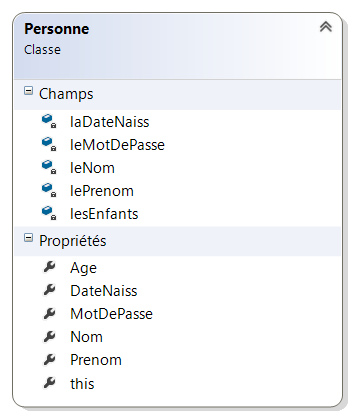
Trigger nom fichier = TRG\_Nom\_Package

Nom trigger = Package.TrigInstblNameNomChamp ex: Jeux.trigInstblJeuTag

1. Classes

Pour les classes vous utiliserez le CamelCase.

<https://msdn.microsoft.com/fr-fr/library/ff926074.aspx>



1. Tables

Pour les tables vous utiliserez aussi le CamelCase sauf pour le nom de la table qui devra commencer par tbl ce qui donnera tblPersonne.

1. Documents

* Concernant vos documents d’analyse, vous devez garder les mêmes standards vus dans le cours d’Analyse de la session passée.
* Pour tout document généré, vous devrez automatiquement ajouter les numéros de versions à la fin de la nomenclature utilisée (XXXXX-V1, XXXXX-V2) si nécessaire.
* À l’intérieur d’un même *sprint*, tous les noms de documents devront commencer par le numéro de package sur deux chiffres **PXX** (P01, P07)

Use case

* Les noms des documents contenant les *use case* auront la nomenclature suivante : **PXX-UC-XX**

Description narrative

* Les noms des documents contenant les descriptions narratives auront la nomenclature suivante : **PXX-DN-XX**

Diagramme de classes

* Les noms des documents contenant les diagrammes de classes auront la nomenclature suivante :
  + Diagramme de classe d’analyse : **PXX-DCA**
  + Diagramme de classe d’objet (facultatif): **PXX-DCO**
  + Diagramme de classe relationnel : **PXX-DCR**

Diagramme de séquence

* Les noms des documents contenant les diagrammes de séquence auront la nomenclature suivante : **PXX-DS-XX**. Plus de détails vous seront annexés ultérieurement.

Interfaces

* Les noms des documents contenant les interfaces auront la nomenclature suivante : **PXX-IHM-XX**. Plus de détails vous seront annexés ultérieurement.

Documents de tests

* Les noms des documents contenant les tests auront la nomenclature suivante : **PXX-TXX-XX (P01-TTU-01)**. Plus de détails vous seront annexés ultérieurement concernant la nature des tests.

Documents de gestion de projet

* Les noms des documents associés à la gestion de projet auront la nomenclature suivante : **Numero\_Sprint-AAMMJJ-Nom\_du\_document : S1-150826-CRR**.
* Le choix des noms se fera d’une manière cohérente et homogène.

Scripts de bases de données

1. Fonctionnement

DBA

\* Chaque équipe devra désigner un DBA qui gardera ce rôle tout le long du projet. Il sera responsable du bon fonctionnement de la base de données mais il ne pourra prendre les décisions qu’après entente avec au moins 2 autres membres de l’équipe.

Modifications

\* Pour toute demande de modification, les autres membres de l’équipe devront remplir une « demande de service » adressée au DBA. La gestion de ces demandes devra se faire par l’intermédiaire du document Excel en annexe.

Équipe développeurs

\* Les autres membres de l’équipe auront aussi à créer des scripts et des requêtes notamment pour les tests.

\* Dans certaines circonstances vous pourriez travailler en mode graphique à des fins de tests unitaires.

2. Consignes de création de scripts

Règle 1

Vous n’avez pas le droit de générer des scripts à partir de tables crées en mode graphique. En d’autres termes, vous devrez utilisez exclusivement des scripts.

Règle 2

Pour chaque package, vous devrez minimalement créer:

\* Un script de création contenant toutes les requêtes de création des tables.

\* Un script de contraintes contenant toutes les contraintes des tables.

\* Un script d’insertion de base : avec des données minimales de tests

\* Un script d’insertion complet : contenant toutes les combinaisons possibles en plusieurs occurrences à des fins de tests de volumes et de production de rapports.

Si l’utilisation de vues est nécessaire, vous devriez créer un ou plusieurs scripts associés à ces-dites vues.

Règle 3

\* Le nom d’un script SQL sera constitué de :

o Trois lettres indiquant s’il agit d’un script de :

\* Création : CRE\_nom\_significatif

\* Contraintes : CTE\_ nom\_significatif

\* Insertion de base : INSB / INSB\_ nom\_significatif

\* Insertion complet : INSC / INSC\_ nom\_significatif

o 8 caractères pour la date

o Exemples :

\* CRE\_Jeu\_Plateforme\_20150825

\* CRE\_Package\_Jeu\_20150825

\* INSB\_ Jeu\_Plateforme\_20150825

Règle 4

\* Les développeurs seront responsables de générer les scripts d’insertion des données à des fins de tests. En équipe, le DBA appuyé d’au moins deux autres membres devront s’assurer de la validité de ces scripts. Le DBA procédera ensuite à l’exécution, la génération et la mise à jour.

\* Le DBA peut à tout moment retourner les scripts avec plusieurs irrégularités aux développeurs.

\* Le DBA est responsable de vérifier la conformité des versions finales scripts.

Règle 5

\* Pendant le développement, le DBA sera autorisé à utiliser les autres instructions SQL telles que UPDATE, ADD, RENAME, etc. Cependant, les scripts finaux de chaque sprint devront utiliser les instructions de bases : CREATE et INSERT.