**Mission 1 : Mise en place des ateliers**

**Tâche 1 : Contrainte sur les dates**

Afin d’obliger la date de fin de la table Vacation à être supérieur à la date de début il existe différentes solutions telles que les contraintes de type check, les triggers et enfin les contrôles côté langage. La première solution soit la contrainte de type check, permettrait ici lors d’une insertion, de vérifier une condition au niveau d’une même table. Soit, que la date de fin soit supérieure à la date de début ce qui donnerai HEUREFIN>HEUREDEBUT. Si l’insertion ne remplis pas la condition, alors, une erreur Oracle se déclenchera afin de prévenir que la contrainte n’est pas respectée. Tandis que la seconde, les triggers, permettra de revenir en arrière en cas de non respect du contrôle permettant de signaler si la date de fin est supérieure ou non à la date de début. Cette action se déclenchera sur chaque insertion d’une nouvelle occurrence dans la table Vacation. Cette dernière pourra en cas d’erreur, remonter une exception dans l’application et ainsi afficher un message précis. Et enfin la troisième, soit le contrôle côté application permet d’afficher directement un message précis concernant l’erreur.

Ici, notre choix s’orientera sur la contrainte check car cela apporte une meilleure sécurité du fait que la vérification s’effectue à l’insertion même. De plus, un trigger demandera beaucoup trop de ressources pour un simple contrôle comme celui-ci. La contrainte check s’impose alors comme étant la meilleure solution pour ce problème.

ALTER TABLE Vacation ADD CONSTRAINT CKC\_VACATION\_HEURE CHECK (HEUREFIN > HEUREDEBUT) ;

**Tâche 2 : Gestion des ajouts et mises à jour**

*(Voir le fichier/dossier : …)*

CREATE OR REPLACE  
 PROCEDURE AJOUT\_ATELIER(pLibelleAtelier IN ATELIER.LIBELLEATELIER%TYPE , pNbPlacesMaxi IN ATELIER.NBPLACESMAXI%TYPE) IS  
BEGIN  
 INSERT INTO ATELIER(id, libelleatelier, nbplacesmaxi) VALUES (SEQATELIER.nextval, pLibelleAtelier, pNbPlacesMaxi);  
 COMMIT;  
EXCEPTION  
 WHEN NO\_DATA\_FOUND THEN raise\_application\_error(-20000, “Aucunes données trouvées.”);   
 WHEN OTHERS THEN raise\_application\_error(-20001, “Erreur détectée.”);  
END;

CREATE OR REPLACE  
 PROCEDURE AJOUT\_THEME(pIdAtelier IN THEME.IDATELIER%TYPE, pNumero IN THEME.NUMERO%TYPE , pLibelleTheme IN THEME.LIBELLETHEME%TYPE) IS  
BEGIN  
 INSERT INTO THEME(idatelier, libelleatelier, nbplacesmaxi) VALUES (pIdAtelier, pNumero, pLibelleTheme);  
 COMMIT;  
EXCEPTION  
 WHEN NO\_DATA\_FOUND THEN raise\_application\_error(-20000, “Aucunes données trouvées.”);   
 WHEN OTHERS THEN raise\_application\_error(-20001, “Erreur détectée.”);  
END;

**Tâche 3 : Construction des interfaces**

*(Voir le fichier/dossier : …)*

**Tâche 4 : Codage des fonctionnalités**

*(Voir le fichier/dossier : …)*

**Tâche 5 : Rapport de tests**

Pour contrôler le code ajouté dans les tâches précédentes, nous avons dû créer plusieurs tests :

* Le premier concerne …, c’est un test de type …, nous attendions comme résultat … celui que nous avons obtenu est … . Pour ce dernier nous pouvons en conclure que … .
* Le second s’établit sur …, c’est un test de type …, nous attendions comme résultat … celui que nous avons obtenu est … . En conclusion, on peut dire que … .

**Tâche 6 : Documentation technique**

*(Voir le fichier/dossier : …)*

**Mission 4 : Statistiques des avis des participants**

**Tâche 1 : Conception de la partie « statistiques »**

*(Voir le fichier/dossier : …)*

**Tâche 2 : Enregistrement des avis**

*(Voir le fichier/dossier : …)*

**Tâche 3 : Evolution de l’enregistrement des avis**

*(Voir le fichier/dossier : …)*

Le système actuel d’enregistrement des avis étant assez lourd à gérer, nous avons pensé à diverses solutions qui pourraient permettre de remplacer et d’améliorer ce dernier en permettant une saisie directe de la part des participants.  
 Tout d’abord, une solution que l’on pourrait envisagée est celle de … . Celle-ci comporte évidemment quelques avantages et inconvénients. D’un point de vue technique, elle … mais … . De plus, au niveau financier celle-ci … . Enfin, pour le domaine de l’organisationnel … .

**Tâche 4 : Impression graphique des statistiques**

*(Voir le fichier/dossier : …)*

**Tâche 5 : Rapport de tests**

Pour contrôler le code ajouté dans les tâches précédentes, nous avons dû créer plusieurs tests :

* Le premier concerne …, c’est un test de type …, nous attendions comme résultat … celui que nous avons obtenu est … . Pour ce dernier nous pouvons en conclure que … .
* Le second s’établit sur …, c’est un test de type …, nous attendions comme résultat … celui que nous avons obtenu est … . En conclusion, on peut dire que … .

**Tâche 6 : Documentation technique**

*(Voir le fichier/dossier : …)*