ENSET Mohammedia, Université Hassan II de Casablanca, Filière d’ingénieur GLSID, Pr. Mohamed YOUSSFI

**Examen Blanc Design Pattern et Programmation Orientée Aspect Durée : 3H | Documents Autorisés**

**Travail à rendre (Code source et Rapport) à envoyer par email : m.youssfi@enset-media.ac.ma** On souhaite concevoir et développer une application qui permet manipuler des dessins. Un Dessin contient un ensemble de figures géométriques qui peuvent être soit des rectangles ou des cercles. Pour chaque figure on souhaite calculer son périmètre et sa surface et on souhaite définir une méthode qui permet de dessiner la figure (Afficher l’état de la figure). Un cercle est défini par son centre qui est un point et par son rayon R. Un rectangle est défini par un point qui représente le coin supérieur gauche, sa largeur L et sa hauteur H. Un point est définit par ses coordonnées X et Y. Pour le cercle, sa surface veut π \* R2 alors sa surface vaut 2 \* π \*R. Pour le rectangle, sa surface veut L \* H alors sa surface vaut 2 \*(L+H).

Chaque figure est définit également par des attributs de type entiers comme l’épaisseur du contour, la couleur du contour et la couleur de remplissage.

Dans ce modèle, on voudrait aussi respecter les critères suivants :

• Donner la possibilité de Créer un groupe de figures. Chaque groupe peut contenir d’autres groupes.

• Nous introduisons un Objet Paramétrage dont l’état est défini par attributs par défaut comme l’épaisseur du contour, la couleur du contour et la couleur de remplissage. On voudrait qu’à chaque fois que l’état de cet objet change, toutes les figures doivent être notifiées pour se mettre à jour.

• Pour l’objet dessin, on voudrait définir les opérations suivantes :

o Une méthode « traiter » qui permet de traiter le contenu du dessin en utilisant une famille d’algorithmes qui sont interchangeables dynamiquement. Ce signifie qu’au moment de l’exécution, l’objet dessin pour changer d’algorithme avec un autre de la même famille. Chaque algorithme peut évoluer indépendamment de la classe dessin qui les utilise.

o Les méthodes qui permettent d’ajouter et supprimer des figures du dessin. o Une méthode qui permet d’afficher toutes les figures du dessin.

o Une méthode qui permet de sérialiser le dessin dans un fichier binaire.

• Une fois que l’application est implémentée et testée, on voudrait ajouter aspects techniques sans modifier le code source de l’application sauf si c’est nécessaire d’ajouter des annotations relatives à des aspects techniques de l’application :

P a g e 1 | 2

ENSET Mohammedia, Université Hassan II de Casablanca, Filière d’ingénieur GLSID, Pr. Mohamed YOUSSFI

o On voudrait journaliser toutes les méthodes de l’application en loguant un message au début de la méthode et un message à la sortie de la méthode en loguant aussi la durée d’exécution de la méthode.

o On voudrait sécuriser l’accès à l’application en obligeant l’utilisateur à s’identifier avec un username et un mot de passe. Chaque utilisateur possède des rôles et on voudrait protéger les méthodes de l’objet dessin avec des rôles prédéfinis.

**Travail à Faire :**

1. Etablir un Diagramme de classe du modèle en appliquant les design patterns appropriés en justifiant les designs patterns appliqués.

2. Faire une implémentation du modèle en utilisant un projet Maven sans prendre en considération des aspects techniques.

3. Effectuer des Tests du modèle

4. Créer les aspects techniques permettant la journalisation et la sécurité de l’application en utilisant AspectJ ou Spring AOP

5. Tester l’application avec ses aspects techniques.

6. Ajouter des fonctionnalités susceptibles d’améliorer la qualité de votre application en ajoutant d’autres design patterns bien appropriés, une interface graphique JavaFX de l’application, etc.

P a g e 2 | 2