

Devoir 3 : Les inscriptions à l'université

(sujet en huit pages)

Le devoir a été conçu pour une durée maximale de deux heures, quinze minutes pour lire le sujet, une heure trois quarts pour répondre aux questions. Si vous utilisez un outil pour « dessiner » les différents diagrammes, il convient de doubler le temps prévu (ceci est simplement dû aux manipulations à faire pour créer les schémas, puis les importer dans le document contenant votre réponse). Celle-ci prendra la forme d'un fichier attaché (en format pdf exclusivement) à un message adressé à votre tuteur.

N'oubliez pas de mettre votre nom sur la réponse (le fait que le message soit identifié ne suffit pas).

Ce devoir part de l'observation d'une situation réelle, celle qui prévaut dans une université, lors de l'inscription des étudiants à un ensemble d'unités de valeurs. Elle amène les apprenants à travailler simultanément sur plusieurs types de diagrammes (classes, séquences, états-transitions) qu'il convient d'affiner et de rendre cohérents.

Partie I : contexte

Le devoir va porter sur un cas, et, plus précisément, sur une ébauche de spécification UML qu'il va falloir faire évoluer, tout en assurant la cohérence entre les différents points de vue. Le texte ci-après décrit le cas et présente un diagramme de classes incomplet, un diagramme de séquences (incomplet, lui aussi) et deux automates "à trous". Il est demandé de boucher les trous, de fournir l'automate manquant, d'affiner la présentation du diagramme de séquences, d'éliminer les incohérences qui auraient pu se glisser dans la solution.

Soit le texte ci-après, extrait d'un ouvrage à paraître :

Dans une université, l'annonce du début des inscriptions est faite par affichage. Un étudiant peut s'inscrire à plusieurs filières de formation et à plusieurs unités de valeurs (UV). Il n'y a pas d'étudiant qui ne soit ni inscrit à une UV, ni inscrit à une filière. Puis, un jour, la date de fin des inscriptions est annoncée, par une affiche qui précise que "après cette date, tout étudiant devra avoir choisi une et une seule filière et s'être inscrit à au moins une unité de valeur et pas à plus de 4". Après affichage de cette information et jusqu'à la date de fin, on n'acceptera plus d'inscriptions ne respectant pas la règle qui vient d'être énoncée.

La direction de l'université demande la liste des unités de valeur pour lesquelles il y a moins de dix inscrits. Les étudiants inscrits à ces unités de valeurs sont convoqués à une réunion, sorte de marché où certains sont amenés à faire des concessions : "J'annule telle inscription pour m'inscrire à telle UV si toi, de ton côté, tu t'inscris à telle UV." Seuls ceux qui ont été convoqués peuvent faire des modifications de leurs choix d'unités de valeur. Un étudiant non convoqué pourra éventuellement remettre en cause un de ses choix si cela va dans le sens des objectifs du marché. Les choix doivent avoir lieu avant la date de début des cours. A cette date, toutes les unités de valeur conservées dans la base devront avoir au moins dix inscrits. Si, peu de temps avant cette date, on n'a pas atteint les objectifs du marché, l'administration procède à la suppression des inscriptions ne respectant pas les règles.

En analysant ce texte, un certain nombre d'étapes apparaissent, pendant lesquelles des opérations sont permises, d'autres interdites. Des invariants peuvent également être mis en évidence et reliés à ces étapes.

• Le premier s'étend du premier affichage au second. Pendant ce laps de temps, tout étudiant mémorisé dans le système est soit inscrit à une (ou plusieurs) filière(s), soit à une (ou plusieurs) UV, soit aux deux. Pendant ce temps, un étudiant peut s'inscrire à une filière (sans s'inscrire à une ou plusieurs UV), à une première UV (sans le faire à une filière), à une UV supplémentaire (tout en étant non-inscrit à une filière), ...

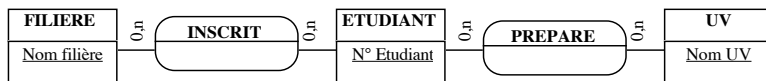


Figure n° 1 : Invariant n° 1

• Le deuxième part de ce deuxième affichage et va jusqu'à la date de fin des inscriptions. Le nouvel invariant garantit que tous les étudiants qui s'inscrivent durant ce laps de temps sont inscrits à une filière et une seule, ainsi qu'à une ou plusieurs UV (avec un maximum de 4).

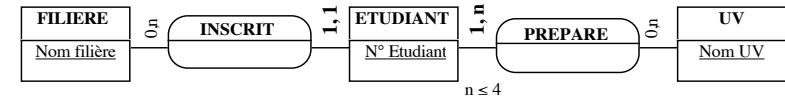


Figure n° 2 : Invariant n° 2

• Le troisième démarre en même temps que les cours. Il garantit que tout étudiant est inscrit à une et une seule filière et à au moins une UV et à au plus 4. Il garantit, d'autre part, qu'aucune UV a moins de 10 étudiants.

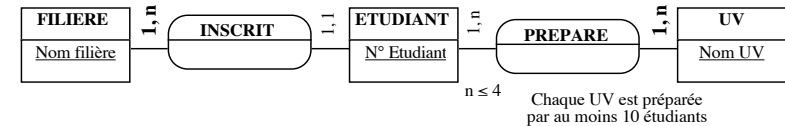


Figure n° 3 : Invariant n° 3

Cette brève énumération montre, si besoin en était, qu'il est assez facile de trouver des cas concrets de systèmes multi-invariants. Décrire précisément de tels systèmes nécessite la fourniture d'autant de MCD qu'il y a d'invariants ainsi qu'une description détaillée des transitions entre les étapes. Il faut toutefois avouer que rares sont les cas où un tel travail est fait. Un seul MCD global (le moins contraignant possible) est souvent fourni. Il est vrai que les aides proposées par les différentes méthodes dans ce domaine sont peu développées.

Partie II : diagrammes états-transitions

Nous allons étudier ce "système", à l'aune de la méthode UML. Plus précisément, nous allons travailler sur son comportement. Auparavant, quelques précisions sont nécessaires. Etudiants, unités de valeur et filières sont décrites par le diagramme de classe suivant :

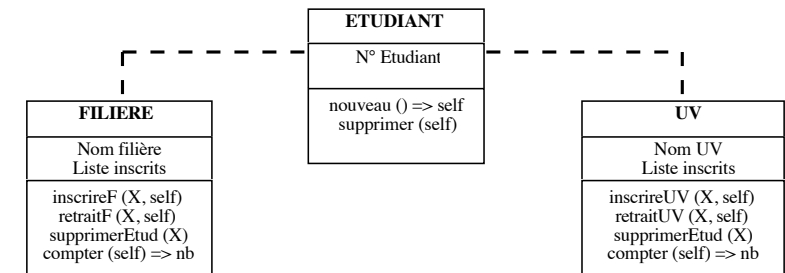


Figure n° 4 : Diagramme de classe (partiel) du système Inscriptions

Ce diagramme est incomplet (il suffit cependant pour répondre aux questions). La nature exacte des deux associations reliant FILIERE et ETUDIANT d'un côté et UV et ETUDIANT de l'autre n'est pas définie. Cela tient surtout au fait qu'il y a trois invariants successifs et qu'une seule modélisation ne suffit pas à traduire cette multiplicité. Les classes ne sont pas complètes, propriétés et méthodes devant être précisées.

NB : la méthode *nouveau()* => *self* retourne le numéro de l'étudiant qui vient d'être instancié. La méthode *inscrireF(X, self)* ajoute l'étudiant ayant pour numéro X à sa propre liste d'étudiants inscrits. Les méthodes *inscrireUV*, *retraitF* et *retraitUV* ont des comportements similaires. *supprimerEud(X)* supprime l'étudiant X. *compter(self)* => *nb* fournit le cardinal de la liste d'étudiants inscrits.

Les inscriptions (des étudiants aux filières, des étudiants aux UV) sont gérées au niveau des filières et des UV, mais pas à celui des étudiants. Il y a donc deux listes, une dans UV et une dans FILIERE. Hormis ces trois classes, nous utiliserons une classe INTERFACE (pour prendre en charge le dialogue) et une classe HORLOGE (pour surveiller le calendrier). En première approximation, le dialogue entre l'acteur externe et le système est le suivant :

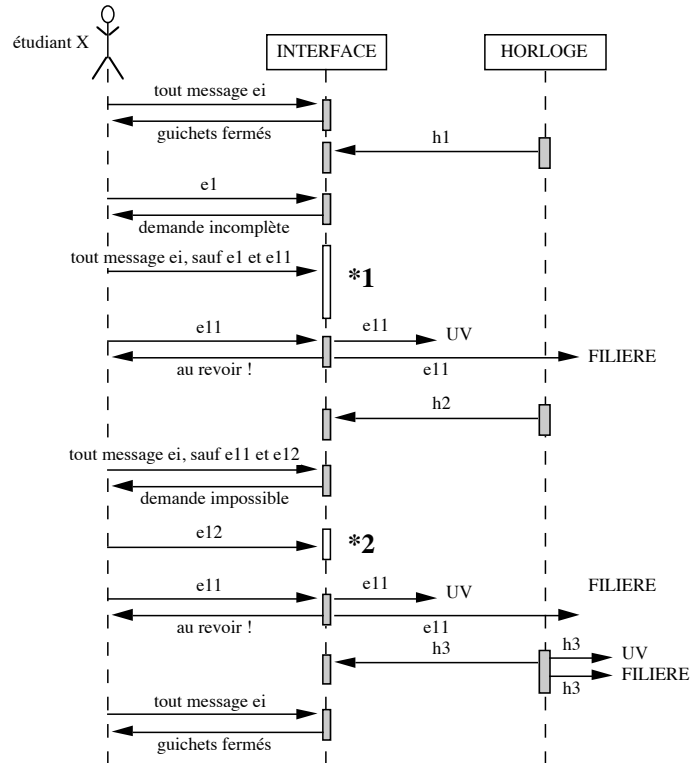


Figure n° 5 : Diagramme de séquences (partiel) du système Inscriptions

NB : les deux parties non hachurées situées "dans" la colonne correspondant à la classe INTERFACE font l'objet d'une question.

Les différents messages mentionnés sur le diagramme de séquences précédent ont la signification suivante :

Numéro du message	Messages adressés par l'horloge au système	Réaction du système (méthode(s) déclenchée(s))	Classe concernée
h1	les inscriptions peuvent débuter		INTERFACE
h2	les inscriptions sont "sous contrôle"		INTERFACE
h3	les cours commencent		INTERFACE et FILIERE // UV

Pour ce qui concerne les messages envoyés par l'étudiant (ils sont notés ei sur le diagramme), ils provoquent des réactions diverses, selon la nature des messages envoyés par l'horloge. Rappelons que le message h3 provoque le rejet de tous les messages. Pour les deux autres (h1 et h2), le tableau ci-après les présente toutes, classées dans l'ordre chronologique (première ligne : h1 reçu, deuxième ligne : h2 reçu) :

Numéro du message	Messages adressés par l'étudiant X au système	Réaction du système (méthode(s) déclenchée(s))	Classe concernée
e1	je veux m'inscrire, mais je ne sais pas encore à quoi.	rejet (X)	INTERFACE
		rejet (X)	INTERFACE
e2	je veux m'inscrire à l'UV Y	nouveau (X) puis inscrireUV(X, Y)	ETUDIANT et UV
		rejet (X)	INTERFACE
e3	je veux m'inscrire dans la filière Y	nouveau (X) puis inscrireF (X, Y)	ETUDIANT et FILIERE
		rejet (X)	INTERFACE
e4	je veux m'inscrire dans la filière Y et à l'UV Z	nouveau (X) puis (inscrireF (X, Y) // inscrireUV (X, Z))	ETUDIANT et (FILIERE // UV)
		rejet (X)	INTERFACE
e5	je suis déjà inscrit, je veux une inscription supplémentaire à l'UV Y	inscrireUV (X, Y)	UV
		rejet (X)	INTERFACE
e6	je suis déjà inscrit, je veux une inscription supplémentaire à la filière Y	inscrireF (X, Y)	FILIERE
		rejet (X)	INTERFACE
e7	je suis déjà inscrit, je veux supprimer mon inscription à l'UV Y	retraitUV (X, Y)	UV
		rejet (X)	INTERFACE
e8	je suis déjà inscrit, je veux supprimer mon inscription à la filière Y	retraitF (X, Y)	FILIERE
		rejet (X)	INTERFACE
e9	je suis déjà inscrit, je veux supprimer mon inscription à l'UV Y et à la filière Z	(retraitUV (X, Y) // retraitF (X, Z))	(UV // FILIERE)
		rejet (X)	INTERFACE

e10	je suis déjà inscrit, j'échange mon inscription à l'UV Y contre l'inscription à l'UV Z	retraitUV (X, Y) // inscritUV (X, Z)	UV
e11	j'en ai marre, je m'casse !	rejet (X)	INTERFACE
		supprimerEtud (X)	(UV // FILIERE) et ETUDIANT
e12	je veux m'inscrire dans la filière Y et aux UV Z1, Z2, Z3 et Z4	nouveau (X) puis (inscrireF (X, Y) // inscrireUV (X, Z1) // // inscrireUV (X, Z2) // inscrireUV (X, Z3) // inscrireUV (X, Z4)	ETUDIANT et (UV // FILIERE)
		nouveau (X) puis (inscrireF (X, Y) // inscrireUV (X, Z1) // // inscrireUV (X, Z2) // inscrireUV (X, Z3) // inscrireUV (X, Z4)	ETUDIANT et (UV // FILIERE)

Le comportement de la classe UV peut être modélisé assez simplement par un automate comprenant un état initial, deux états intermédiaires et deux états finaux :

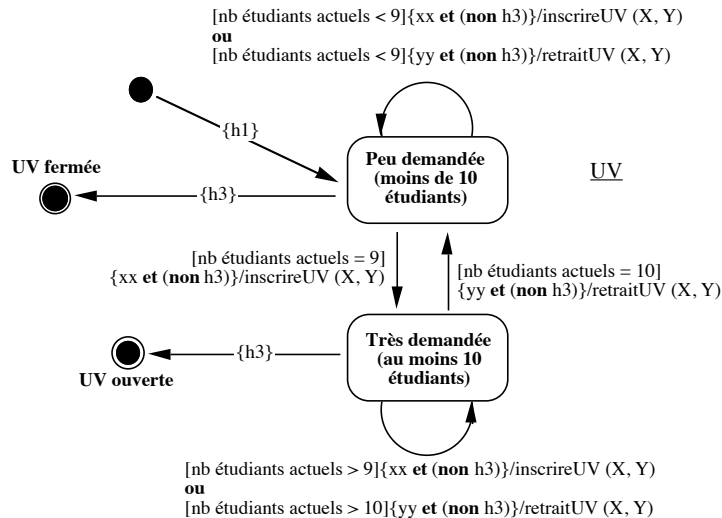


Figure n° 6 : Automate (inachevé) décrivant le comportement de la classe UV du système Inscriptions

Rappelons, avant d'aller "plus loin", la signification des notations¹ que nous avons utilisé dans ce schéma : chaque transition est composée de trois parties. La première [elle est entre crochets] est la condition sous laquelle la transition peut avoir lieu. La deuxième {elle est entre accolades} correspond aux noms des événements déclencheurs. La troisième (elle est précédée d'un /) nomme la méthode qui est exécutée durant la transition.

Question n° 1a : 2 points **TEMPS SUGGÉRÉ : 10 mn**

Dans l'automate précédent, remplacez *xx* et *yy* par les numéros des messages correspondants. Signalez et corrigez les éventuelles incohérences.

Le comportement de la classe FILIERE peut être décrit grâce à un automate qui comprend un état initial, deux états intermédiaires et deux états finaux :

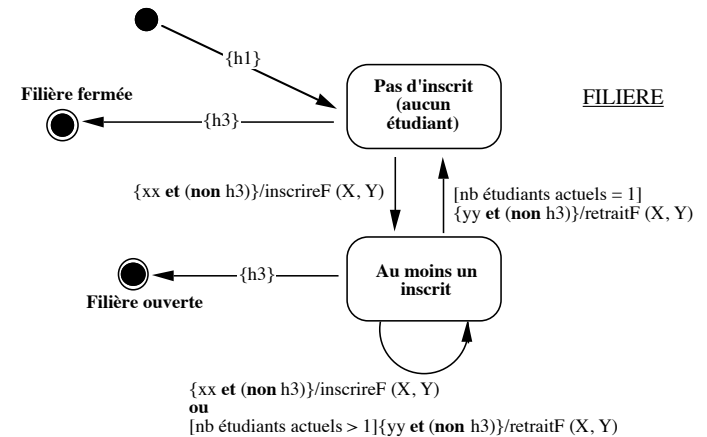


Figure n° 7 : Automate (inachevé) décrivant le comportement de la classe FILIERE du système Inscriptions

Question n° 1b : 2 points **TEMPS SUGGÉRÉ : 10 mn**

Dans l'automate précédent, remplacez *xx* et *yy* par les numéros des messages correspondants. Signalez et corrigez les éventuelles incohérences.

La classe ETUDIANT a un comportement un peu plus délicat. En particulier, les inscriptions aux UV et aux filières se font manifestement en parallèle (sauf la première fois). On peut suivre la démarche qui consiste à définir toutes les combinaisons² d'états ("inscrit à une UV et pas à une filière", "inscrit à une filière et pas à une

¹ Elle est légèrement différente de celle d'UML.

² La technique mise en œuvre est clairement celle du calcul d'un produit cartésien d'automates communicants.

UV", "inscrit ...), mais ce n'est pas si simple et le risque est grand d'en oublier. On peut procéder autrement et, de façon très élégante, exprimer le parallélisme inhérent au problème en prenant appui sur la notation UML. L'automate qui en résulte est composé de deux niveaux, le second (celui qui est le plus à l'intérieur) mettant en œuvre le parallélisme :

ETUDIANT

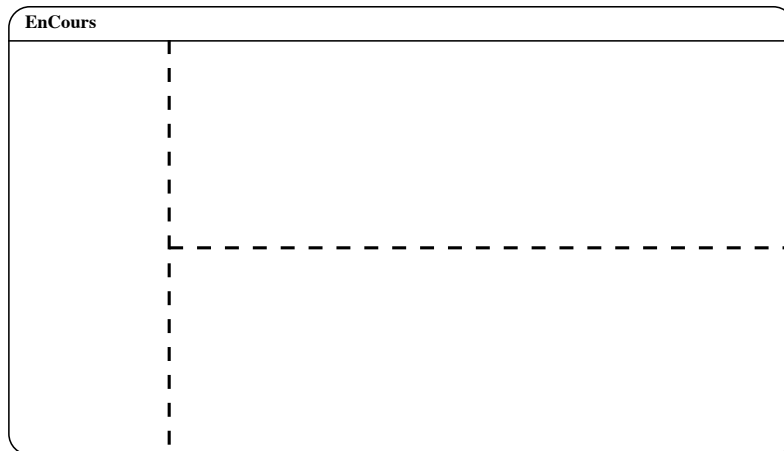
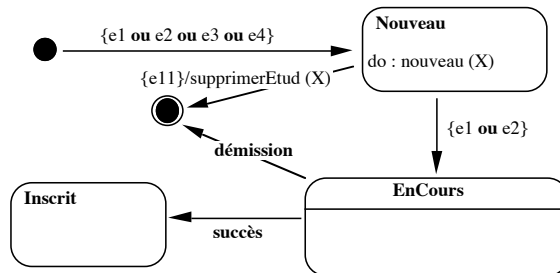


Figure n° 8 : Automate (partiel) décrivant le comportement de la classe ETUDIANT du système Inscriptions

Question n° 1c : 6 points

TEMPS SUGGÉRÉ : 30 mn

Définissez le super-état *Encours*. Signalez et corrigez les éventuelles incohérences.

Partie III : diagrammes de classes et diagrammes de séquences

On va maintenant s'intéresser de nouveau au comportement du système global.

Question n° 2 : 10 points (2, 5 et 3)

TEMPS SUGGÉRÉ : 50 mn

- Reprenez le diagramme de séquences présenté en figure 5. À la lumière de ce qui précède, précisez les deux parties numérotées *1 et *2.
- Reprenez ce même schéma et introduisez-y les classes ETUDIANT, UV et FILIERE.
- Une sixième classe semble nécessaire pour régler les problèmes de décision (fermeture de filières ou d'unités de valeur notamment). Incorporez-la au schéma précédent.

-----fin du texte-----