

EXAMEN D'INGENIERIE DES SYSTEMES D'INFORMATION

PREMIERE PARTIE : MODELISATION UML

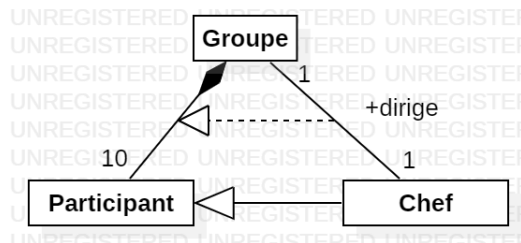
Exercice 1. RELATIONS STATIQUES – 7 POINTS

Modéliser les phrases suivantes à l'aide d'un diagramme de classes.

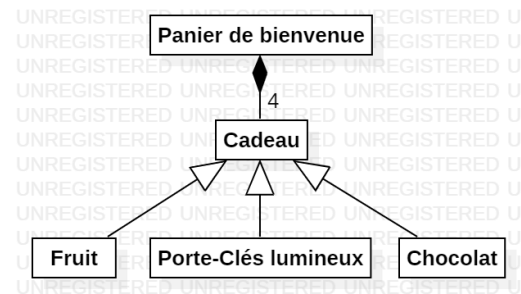
(a) 1 point – Les enfants sont des personnes.



(b) 1,5 points – Les participants sont répartis en groupes de 10 personnes, et chaque groupe choisit parmi ses membres un chef de groupe.



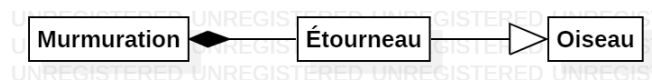
(c) 1,5 points – Le panier de bienvenue est constitué de quatre cadeaux, à choisir parmi des fruits, des chocolats, ou des porte-clés lumineux.



(d) 1,5 points – Ce livre contient 12 chapitres de 50 pages chacun.



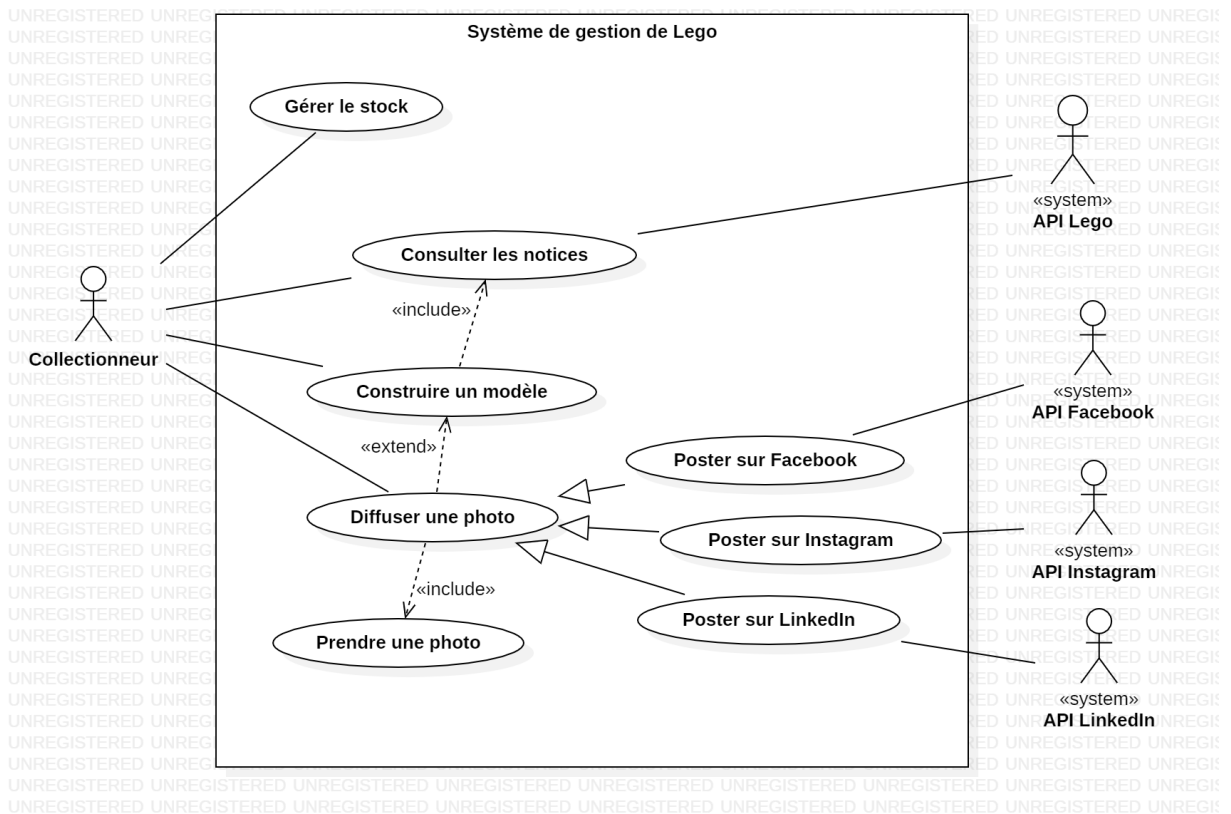
(e) 1,5 points – Les étourneaux, qui sont des oiseaux, peuvent voler en formations appelées murmurations.



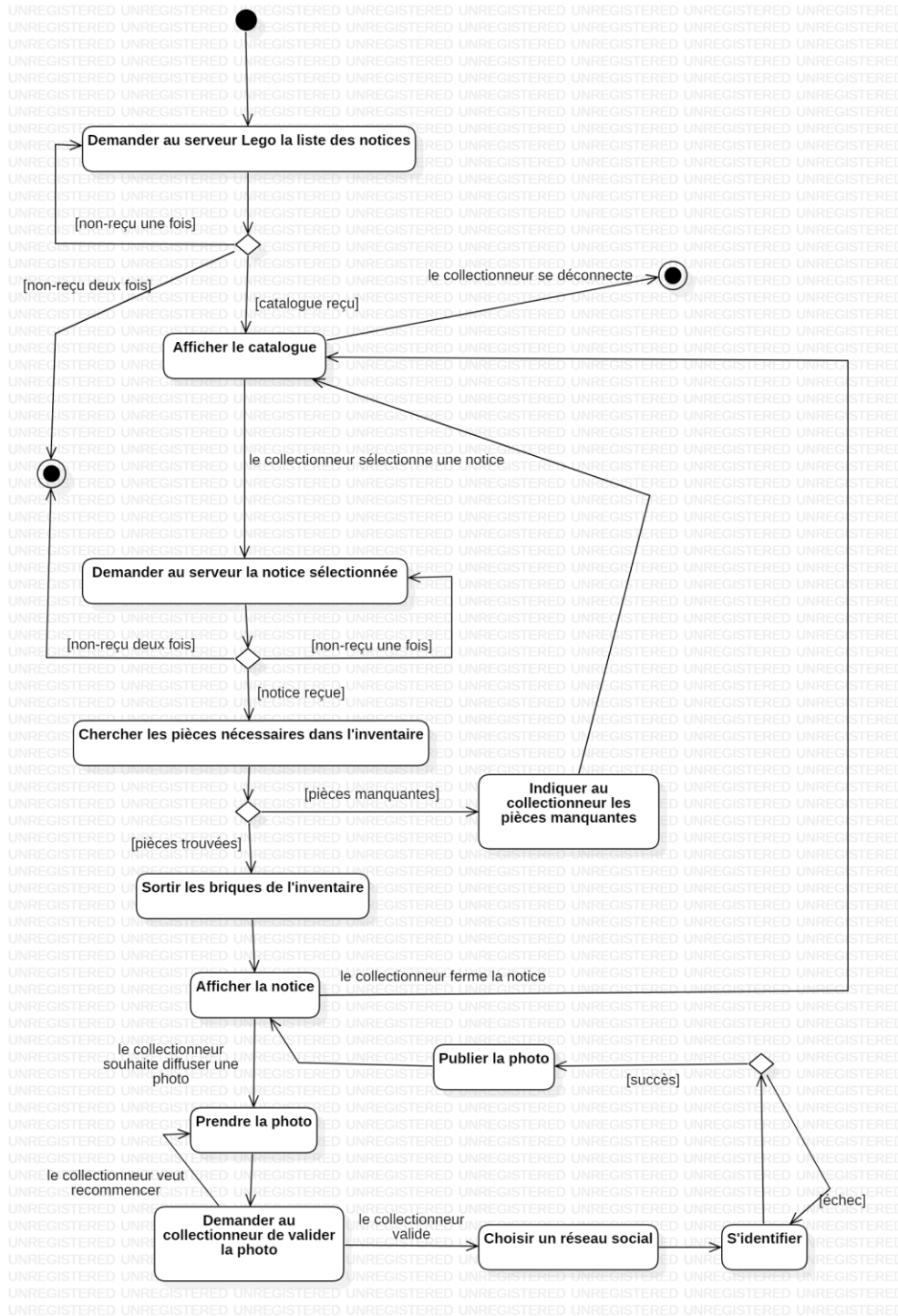
Exercice 2. UN SYSTEME DE GESTION DE LEGO – 9 POINTS

Un collectionneur souhaite organiser ses briques lego à l'aide d'un système d'information. Ce système doit permettre au collectionneur gérer son stock de briques (voir l'état courant du stock, retirer des briques, ou ajouter des briques), et de consulter les notices des différents modèles (disponibles par une API fournie par le constructeur). Il sera aussi possible, au cours de cette consultation, de sélectionner une notice pour construire le modèle. Le système devra alors sortir de l'inventaire les briques nécessaires à la construction. Enfin, à chaque fois qu'un modèle est construit par le collectionneur, il pourra demander au système d'en capturer une photo, et de la poster sur un réseau social. Le système supporte Instagram, Twitter, et LinkedIn.

Question 1. 2 points – Réaliser un diagramme de cas d'utilisation pour ce système. Attention à bien identifier les acteurs et les cas d'utilisation.



Question 2. 2 points – Présenter sous forme d'un diagramme d'activités système la consultation d'une notice à l'aide du système, en incluant les différents scénarios (en particulier la sélection de notice).



Question 3. 3 points – Décrire en détails la publication de photos. On attend une description textuelle structurée, ainsi qu'un diagramme de séquence système d'un scénario nominal.

Nom du cas d'utilisation : Diffuser une photo.

Objectif : Capturer et diffuser la photo d'une construction réalisée par le collectionneur sur le réseau social de son choix.

Acteurs : Collectionneur (principal), API Réseau social (Facebook, Instagram ou LinkedIn).

Précondition : Ce cas d'utilisation commence à l'étape « afficher la notice » du C.U. « construire un modèle ».

Besoins matériels : Le système doit être muni d'un dispositif photographique ainsi que d'une connexion internet.

Scénario nominal :

1. Le collectionneur clique sur le bouton « Publier une photo ».
2. Le système lui demande de positionner sa construction devant le dispositif photographique.
3. Une fois la construction en place, le collectionneur demande au système de prendre la photo.
4. Le système capture une image.
5. Le système affiche l'image pour validation.
6. Le collectionneur valide l'image.
7. Le système affiche la liste des réseaux sociaux pris en charge, et demande au collectionneur d'en choisir un.
8. Le collectionneur obtempère.
9. Le système demande les identifiants de connexion pour le réseau choisi.
10. Le collectionneur les fournit.
11. Le système établit une connexion avec le réseau social et s'y identifie.
12. Le réseau accepte les identifiants.
13. Le système envoie l'image au réseau social pour publication.
14. Le réseau confirme la publication.
15. Le système informe le collectionneur de la publication faite, et retourne à l'étape « Afficher notice » du C.U. « construire un modèle ».

Scénarios alternatifs :

1-10a. Le collectionneur annule la publication.

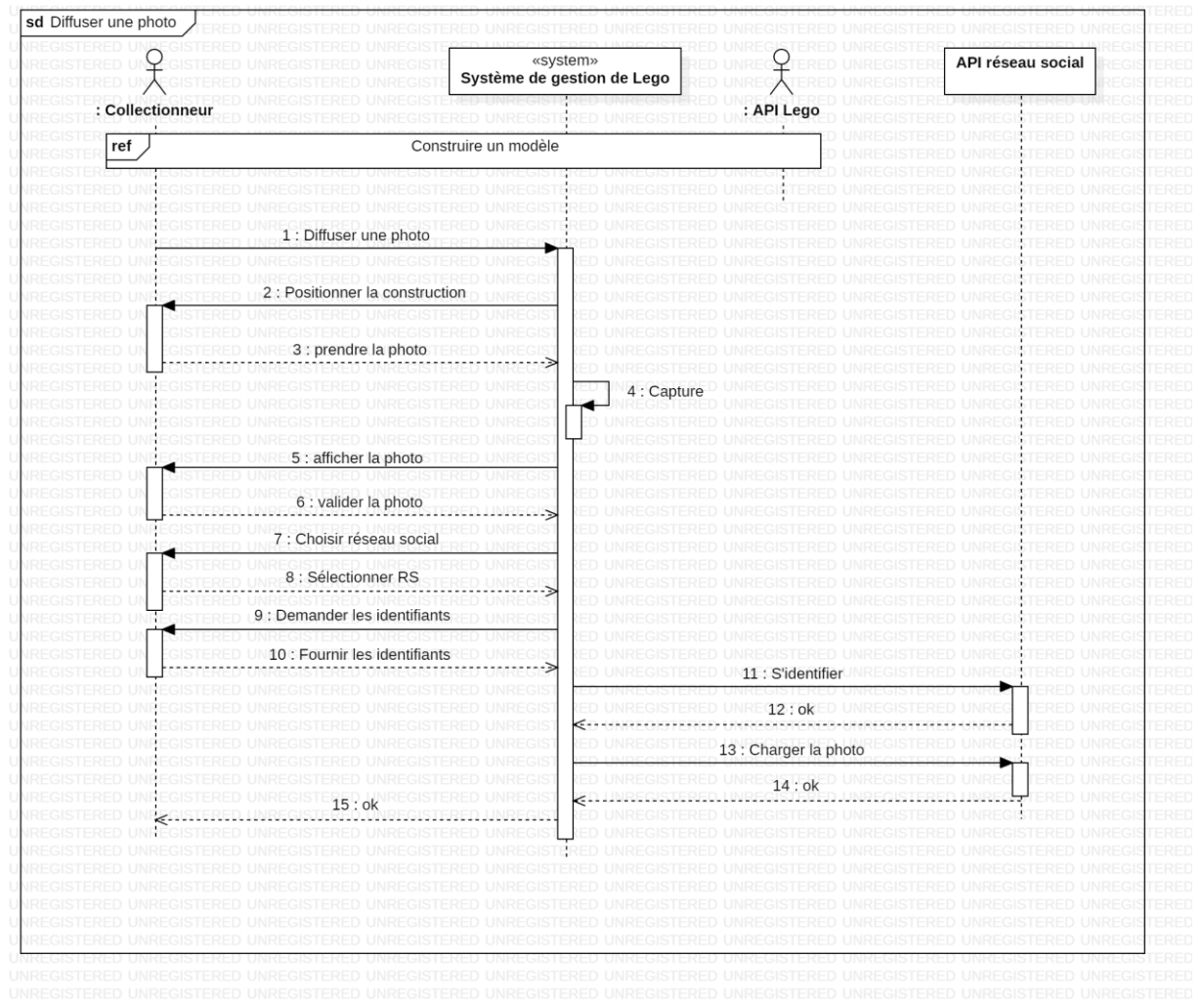
1. Retour à l'étape « Afficher notice » du C.U. « construire un modèle ».

6b. Le collectionneur ne valide pas la photo.

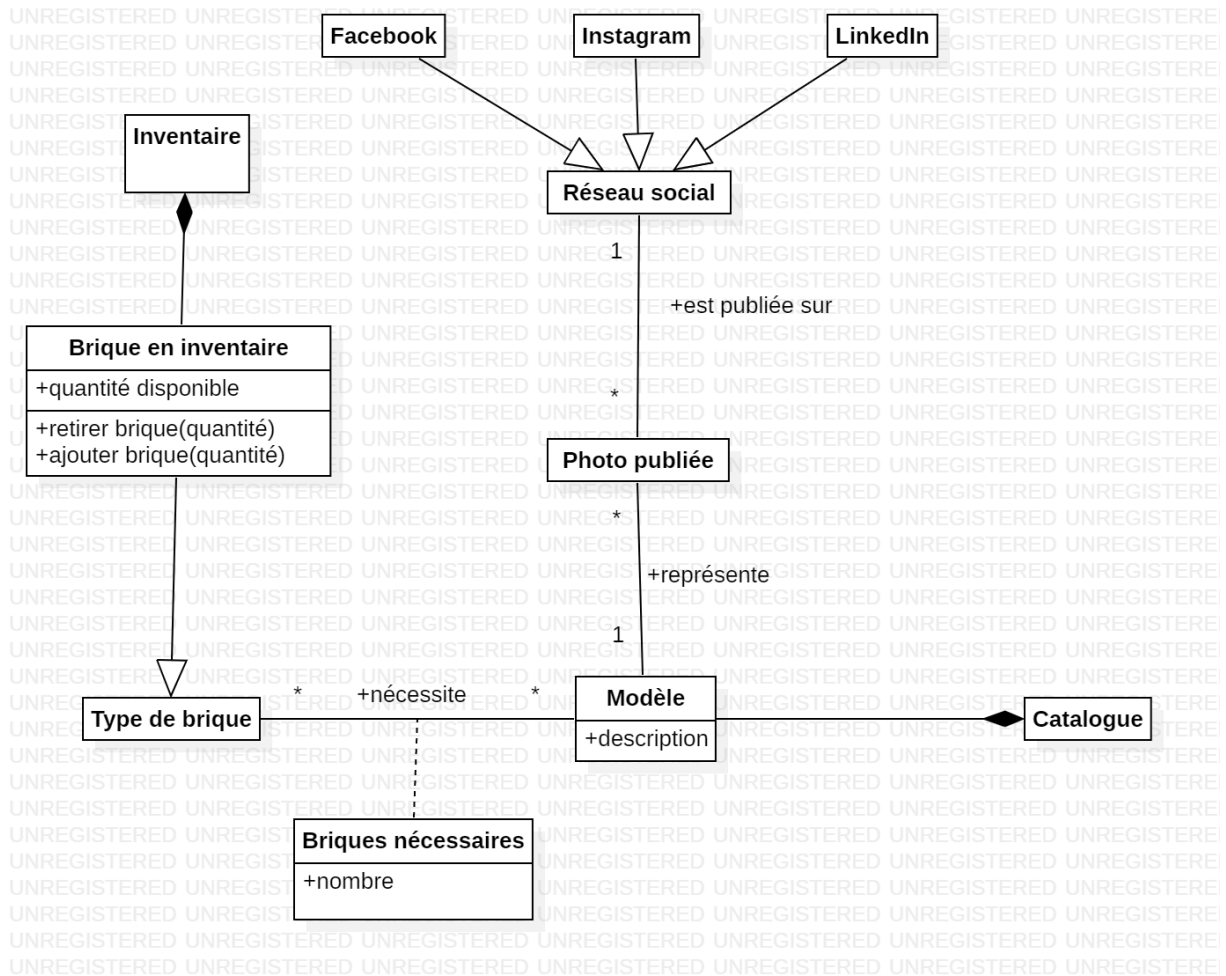
1. La photo est effacée, retour à l'étape 2.

12b. Le réseau rejette les identifiants.

1. Le système informe le collectionneur de la réponse du réseau social.
2. Retour à l'étape 9.



Question 4. 2 points – Établir un diagramme de classes métier pour ce système.



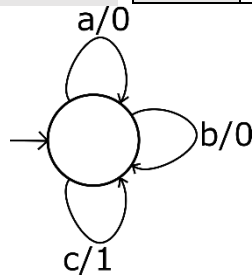
DEUXIEME PARTIE : MACHINES DE MEALY ET APPRENTISSAGE DE MODELES

Exercice 3. MACHINES DE MEALY – 4,5 POINTS

Proposer des machines de Mealy réalisant chacune des fonctions suivantes :

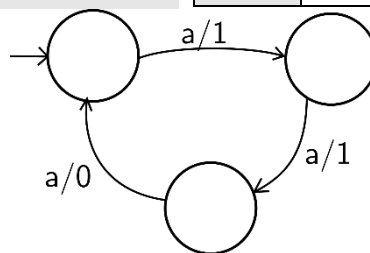
(a) 1 point – Sur l’alphabet {a,b,c}, remplace les a et les b par 0, et les c par 1.

Exemple :	Entrée	a	a	b	a	b	c	a	b	c	c	b
aababcabccb ↦ 00000100110	Sortie	0	0	0	0	0	1	0	0	1	1	0



(b) 1,5 points – Sur l’alphabet {a}, sort un 0 chaque position divisible par 3, et un 1 sinon.

Exemple : aaaaaa ↦ 1101101	Entrée	a	a	a	a	a	a	a	a	a	a
	Sortie	1	1	0	1	1	0	1	1	0	1

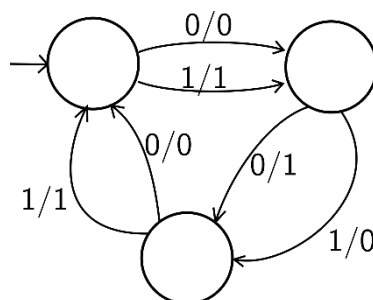


(c) 2 points – Sur l’alphabet {0,1}, encode l’entrée avec le motif 010 comme suit : on itère le motif jusqu’à atteindre la longueur de l’entrée (par exemple pour une entrée de taille 7 on prend 0100100), puis on effectue un « ou exclusif » bit à bit entre entrée et motif itéré.

Exemple : pour 11010101, on prend le motif itéré 01001001, et on obtient 10011100, comme illustré par le tableau suivant :	Entrée	1	1	0	1	0	1	0	1
	Motif	0	1	0	0	1	0	0	1
	Sortie	1	0	0	1	1	1	0	0

Rappel : l’opérateur « ou exclusif » (aussi appelé xor) est une opération binaire sur les booléens, qui renvoie 1 si et seulement si un de ses arguments vaut 1 et l’autre 0. La table de vérité de cet opérateur est donnée ci-contre.

a	b	a xor b
0	0	0
0	1	1
1	0	1
1	1	0



Exercice 4. REGISTRE PARTAGE – 5,5 POINTS

Considérons un registre partagé par n processus p_1, \dots, p_n . Ce registre peut prendre k valeurs, notées v_1, \dots, v_k . Il contient initialement la valeur v_1 . Chacun des processus peut à tout moment lire la valeur contenue dans le registre : pour cela, p_i peut exécuter l'instruction « $read(i)$ », et observer en sortie un message « $val(x)$ », où x est la valeur courante du registre. Pour éviter des problèmes liés aux accès concurrents au registre, celui-ci a été équipé d'un verrou. Le registre est initialement déverrouillé. Lorsqu'il souhaite modifier le contenu du registre, le processus p_i doit d'abord verrouiller le registre avec l'instruction « $lock(i)$ ». Si le registre était déjà verrouillé, un message « $error$ » est produit en sortie. En revanche si le registre était déverrouillé, un message « ok » est produit et p_i peut écrire une nouvelle valeur y dans le registre, via l'instruction « $write(i, y)$ », qui dans ce cas produit également un message « ok ». Lorsque le registre est ainsi verrouillé, seul p_i peut le déverrouiller avec l'instruction « $unlock(i)$ ». Toute tentative de verrouillage, ou de déverrouillage ou d'écriture par un autre processus, donne lieu à un message « $error$ ».

On souhaite modéliser ce registre par une machine de Mealy.

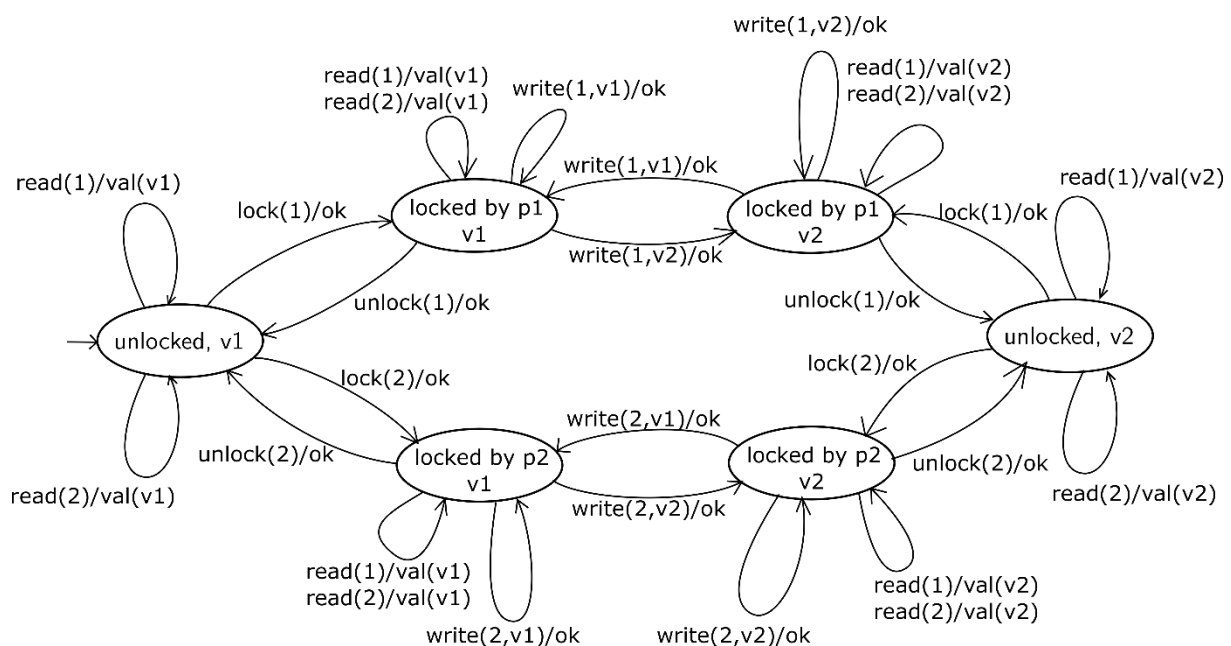
Question 1. 0,5 points – Quel sont les alphabets d'entrée et de sortie de ce système ?

Alphabet d'entrée :

$\{read(i) | i \in \{1, \dots, n\}\} \cup \{lock(i) | i \in \{1, \dots, n\}\} \cup \{unlock(i) | i \in \{1, \dots, n\}\} \cup \{write(i, y) | i \in \{1, \dots, n\}, y \in \{v_1, \dots, v_k\}\}$

Alphabet de sortie : $\{error, ok\} \cup \{val(x) | x \in \{v_1, \dots, v_k\}\}$

Question 2. 2 points – Construire une machine de Mealy pour ce système avec $n = k = 2$.



Pour ne pas surcharger le dessin, on a omis les transitions dont la sortie est le symbole « $error$ ». Ces transitions peuvent être retrouvées comme suit : à partir d'un état q , étant

donné un symbole d'entrée a , si aucune transition n'est dessinée à partir de cet état et avec ce symbole d'entrée, alors il y a une transition bouclant sur cet état avec l'étiquette $a/error$. Par exemple, sur l'état « $unlocked, v_1$ », il y a une boucle non représentée avec l'étiquette « $write(1, v_1)/error$ ».

Question 3. 3 points – Décrire les états et les transitions de la machine dans le cas général.

États : $\{(unlocked, y) | y \in \{v_1, \dots, v_k\}\} \cup \{(locked, i, y) | i \in \{1, \dots, n\}, y \in \{v_1, \dots, v_k\}\}$.

Il y a donc $k \times (1 + n)$ états. L'état initial est $(unlocked, v_1)$.

Table de transitions :

$$\begin{aligned}
 (unlocked, y) &\xrightarrow{read(i)/val(y)} (unlocked, y) \\
 (unlocked, y) &\xrightarrow{lock(i)/ok} (locked, i, y) \\
 (unlocked, y) &\xrightarrow{write(i,z)/error} (unlocked, y) \\
 (unlocked, y) &\xrightarrow{unlock(i)/error} (unlocked, y) \\
 (locked, i, y) &\xrightarrow{read(j)/val(y)} (locked, i, y) \\
 (locked, i, y) &\xrightarrow{unlock(i)/ok} (unlocked, y) \\
 (locked, i, y) &\xrightarrow{unlock(j)/error} (locked, i, y) \quad \forall j \neq i \\
 (locked, i, y) &\xrightarrow{write(i,z)/ok} (locked, i, z) \\
 (locked, i, y) &\xrightarrow{write(j,z)/error} (locked, i, y) \quad \forall j \neq i \\
 (locked, i, y) &\xrightarrow{lock(j)/error} (locked, i, y)
 \end{aligned}$$

Exercice 5. DEVINER LA MACHINE MYSTERE – 5 POINTS

Construire une machine de Mealy sur l'alphabet $\{a,b\}$ avec aussi peu d'états que possible, et telle que l'image des mots de taille 6 est comme dans le tableau suivant :

aaaaaa	BBBBBB	ababaa	BABABB	babaaa	ABABBB	babbaa	ABABBB
aaaaab	BBBBBA	ababab	BABABA	babaab	ABABBA	babbab	ABABBA
aaaaba	BBBBAB	ababba	BABABB	bababa	ABABAB	babbba	ABABAB
aaaabb	BBBBAB	ababbb	BABABA	bababb	ABABAB	babbbb	ABABAA
aaabaa	BBBABB	abbaaa	BABBBB	babbba	ABABBB	bbaaaa	AABBBB
aaabab	BBBABA	abbaab	BABBBB	babbab	ABABBA	bbaaab	AABBBB
aaabba	BBBABB	abbaba	BABBAB	babbbb	ABABAB	bbabaa	AABBAB
aaabbb	BBBABA	abbabb	BABBAB	babbbb	ABABAA	bbbaab	AABBAB
aabaaa	BBABBB	abbbaa	BABABB	baaaaa	ABBBBB	bbabab	AABABA
aabaab	BBABBA	abbbab	BABABA	baaaab	ABBBBA	bbabba	AABABB
aababa	BBABAB	abbbaa	BABAAB	baaaba	ABBBAB	bbabbb	AABABA
aababb	BBABAB	abbbbb	BABAAA	baaabb	ABBBAB	bbbbaa	AABBBB
aabbaa	BBABBB	baaaaa	ABBBBB	baabaa	ABBABB	bbbbaa	AABBBB
aabbab	BBABBA	baaaab	ABBBBA	baabab	ABBABA	bbbbaa	AABBBB
aabbba	BBABAB	baaaba	ABBBAB	baabba	ABBABB	bbbaba	AABBAB
aabbbb	BBABAA	baaabb	ABBBAB	baabbb	ABBABA	bbbabb	AABBAB
abaaaa	BABBBB	baabaa	ABBABB	babaaa	ABABBB	bbbbaa	AAAABB
abaaaab	BABBBB	baabab	ABBABA	babaab	ABABBA	bbbbaa	AAAABB
abaaab	BABBBB	baabab	ABBABA	bababa	ABABAB	bbbbaa	AAAABB
abaaaba	BABBAB	baabba	ABBABB	bababb	ABABAB	bbbbaa	AAAABB
abaaabb	BABBAB	baabbb	ABBABA			bbbbaa	AAAABB

Pas possible de corriger un mystère ! Ça gâche le plaisir...