

2243.2 GÉNIE LOGICIEL

Contrôle Principal 1

Groupes ISC2il-a et ISC2il-b

Lundi 24 janvier 2022, de 10h00 à 11h35

Groupes ISC2id-a, ISC2ie-a et ISC2ie-b

Jeudi 27 janvier 2022, de 12h30 à 14h00

INSTRUCTIONS

1. Chaque bonne réponse vaut 1 point.
2. Chaque mauvaise réponse vaut -1 point.
3. Indiquez vos nom et prénom au début du fichier à rendre.
4. Une fois terminé, envoyez votre fichier par email à benoit.lecallennec@he-arc.ch avec comme sujet [2243.2 Génie Logiciel].
5. Les documents ne sont pas autorisés.
6. Aucune communication n'est autorisée.

QUESTIONS

Question 1

La charge d'une tâche représente la quantité de travail nécessaire :

- (A) indépendamment du nombre de personnes.
- (B) multipliée par le nombre de personnes.
- (C) en CHF.

Question 2

La durée d'une tâche représente :

- (A) le coût en CHF.
- (B) le temps effectivement pris pour effectuer la tâche.
- (C) la même chose que la charge d'une tâche.

Question 3

Lorsque l'on parle de l'évaluation des risques, on peut utiliser l'équation suivante :

- (A) $\text{Impact} = \text{Probabilité d'occurrence} \times \text{Criticité}$.
- (B) $\text{Probabilité d'occurrence} = \text{Criticité} \times \text{Impact}$.
- (C) $\text{Criticité} = \text{Probabilité d'occurrence} \times \text{Impact}$.

Question 4

La revue de code est l'examen systématique d'un code source afin d'améliorer :

- (A) la vitesse du programme.
- (B) la taille mémoire du programme.
- (C) la qualité du code.

Question 5

Durant la phase d'initiation d'un projet, les documents suivants peuvent / doivent être produits :

- (A) une documentation utilisateur, un diagramme de PERT, une étude de faisabilité, un diagramme de Gantt, une demande de projet.
- (B) un diagramme de PERT, une étude de faisabilité, un diagramme de Gantt.
- (C) une étude de faisabilité, une demande de projet.

Question 6

Le diagramme 1 est un diagramme :

- (A) PERT.
- (B) de Gantt.
- (C) WBS.

Question 7

Le diagramme 2 est un diagramme :

- (A) PERT.
- (B) de Gantt.
- (C) WBS.

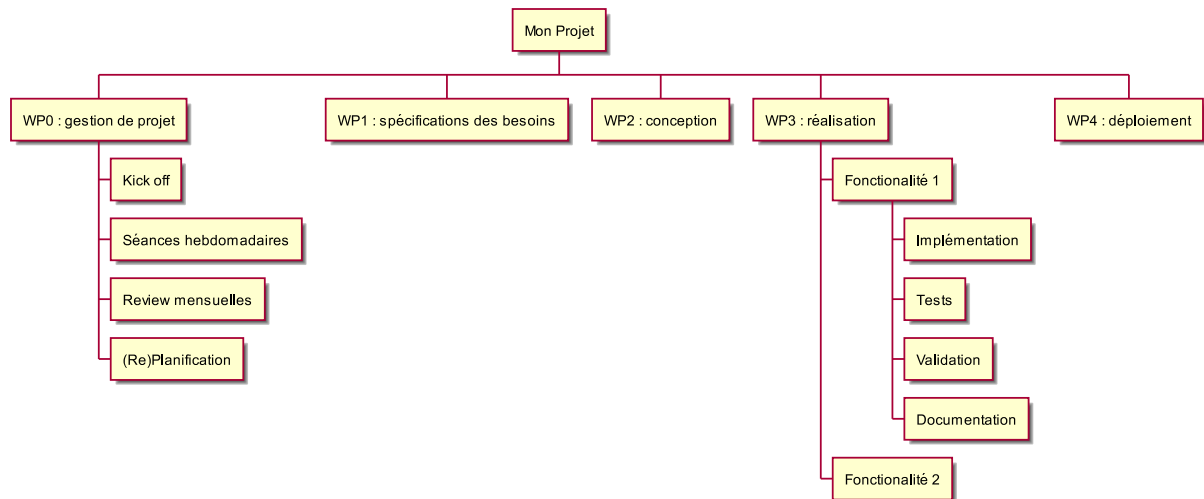


Figure 1: Diagramme 1

Question 8

Dans le diagramme 2, que représente le carré **Initial Data Collected** ?

- (A) Un jalon (ou milestone).
- (B) Une note pour la documentation.
- (C) Une tâche à effectuer en priorité.

Question 9

En gestion de projet, WP signifie :

- (A) World Package.
- (B) Work Package.
- (C) Well Played.

Question 10

Selon les bonnes pratiques, quelle est en général la tâche WP0 ?

- (A) Gestion de projet.
- (B) Conception.
- (C) Spécification des besoins.

Question 11

Les phases du cycle de vie d'un logiciel sont dans l'ordre :

- (A) Initiation, Analyse, Conception, Développements, Tests, Déploiement, Maintenance.
- (B) Maintenance, Analyse, Conception, Développements, Tests, Déploiement, Initiation.
- (C) Initiation, Analyse, Conception, Tests, Déploiement, Maintenance.

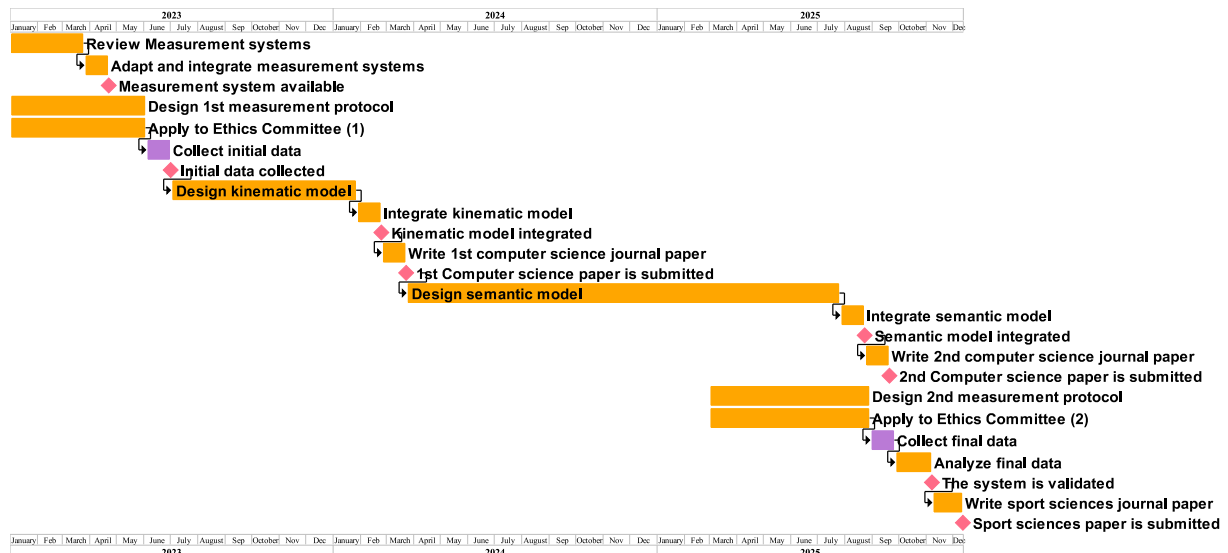


Figure 2: Diagramme 2

Question 12

Durant la hiérarchisation des besoins, on utilise parfois la classification MoSCow (basée sur les lettres MSCW). Mais que signifient ces lettres ?

- (A) Must, Should, Could, Want.
- (B) Must, Should, Could, Won't.
- (C) Must, Should, Could, Will.

Question 13

Il existe 3 types de maintenance : adaptative, évolutive et :

- (A) Corrective.
- (B) Destructive.
- (C) Réparatrice.

Question 14

La phase d'analyse des besoins détaille ce que doit faire le logiciel, ainsi que comment il doit le faire.

- (A) VRAI.
- (B) FAUX.
- (C) Faut voir.

Question 15

Le but de l'identification des tâches est de découper le projet en tâches plus précises, moins complexes et plus faciles à estimer.

- (A) VRAI.

(B) FAUX.

(C) Faut voir.

Question 16

Les phases d'un projet sont bien définies et ne se recouvrent pas. Chaque phase commence lorsque la précédente se termine, et se termine lorsque la suivante commence.

(A) VRAI.

(B) FAUX.

(C) Faut voir.

Question 17

Dans la phase de conception, on détaille comment mettre en œuvre les modèles définis dans l'analyse.

(A) VRAI.

(B) FAUX.

(C) Faut voir.

Question 18

La phase de déploiement, aussi appelée phase John « Hannibal » Smith se déroule généralement sans accroc. Il est donc rarement nécessaire de prévoir un délai important pour l'activité de déploiement.

(A) VRAI.

(B) FAUX.

(C) Faut voir.

Question 19

Les tests par boîte blanche :

(A) ont accès au code source et se basent sur la structure et la logique du code.

(B) se basent uniquement sur la structure et la logique du code.

(C) N'ont pas accès au code source.

Question 20

Les tests par boîte noire :

(A) ont accès au code source et se basent sur la structure et la logique du code.

(B) se basent uniquement sur la structure et la logique du code.

(C) n'ont pas accès au code et se basent uniquement sur les entrées-sorties.

Question 21

Le backlog représente :

- (A) la base de données utilisées pour un serveur web.
- (B) la liste des tâches à effectuer et des bugs à corriger pour un projet donné.
- (C) un fichier généré par une application lorsqu'elle s'arrête involontairement.

Question 22

En SCRUM, le Product Owner :

- (A) Définit les fonctionnalités de l'application.
- (B) Gère la partie vente et marketing de l'application.
- (C) Développe l'application.

Question 23

En SCRUM, une équipe est de préférence composée de 5 à 10 développeurs :

- (A) à temps partiel sur le projet.
- (B) à temps plein sur le projet.
- (C) ne travaillant pas sur le projet.

Question 24

En SCRUM, une équipe est de préférence composée de 5 à 10 développeurs :

- (A) regroupant toutes les compétences requises pour le développement.
- (B) se focalisant sur quelques compétences essentielles uniquement.
- (C) ce n'est pas important.

Question 25

Dans le diagramme 3, la relation entre le cas **Enregistrement** et le cas **Login** est :

- (A) Obligatoire.
- (B) Optionnelle.
- (C) Ni l'un, ni l'autre.

Question 26

Dans le diagramme 3, la relation entre le cas **Publier une annonce** et le cas **Login** est :

- (A) Obligatoire.
- (B) Optionnelle.
- (C) Ni l'un, ni l'autre.

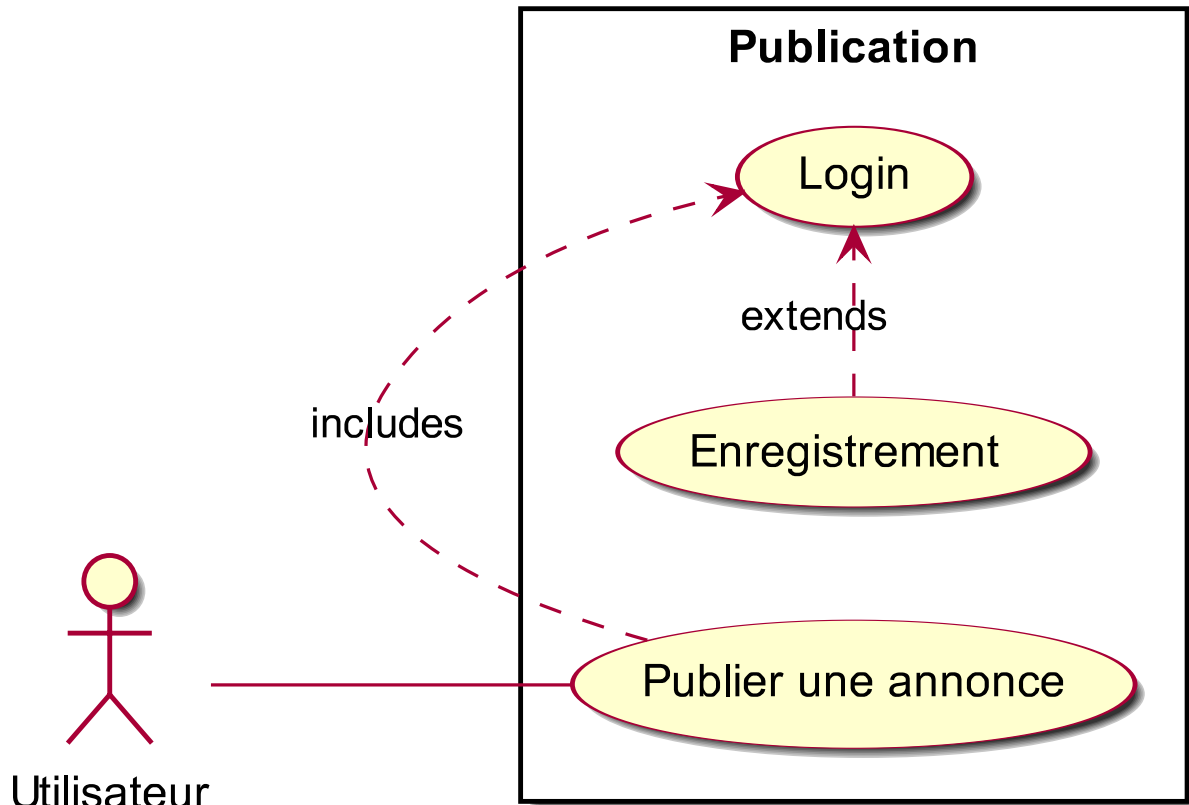


Figure 3: Diagramme 3

Question 27

Dans le diagramme 3, **Utilisateur** est appelé :

- (A) Un bonhomme.
- (B) Un comédien.
- (C) Un acteur.

Question 28

Dans le diagramme 4, la relation entre **Participant** et **Member** est :

- (A) une composition.
- (B) une aggrégation.
- (C) une généralisation.

Question 29

Dans le diagramme 5, la relation entre **Voiture** et **Pneu** est :

- (A) une composition.
- (B) une aggrégation.
- (C) une généralisation.

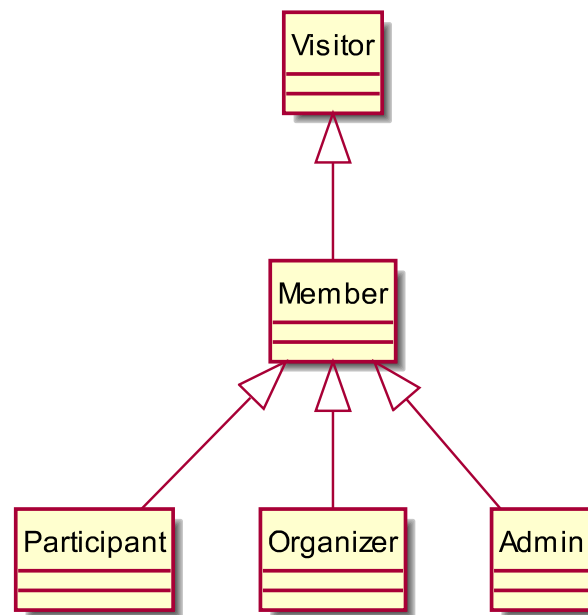


Figure 4: Diagramme 4

Question 30

Dans le diagramme 5, que se passe-t'il lorsque **Voiture** est détruite :

- (A) les **Pneus** associés sont aussi détruits.
- (B) les **Pneus** associés NE sont PAS détruits.
- (C) on ne peut pas savoir.

Question 31

Dans le diagramme 5, la relation entre **Voiture** et **Chassis** est :

- (A) une composition.
- (B) une aggrégation.
- (C) une généralisation.

Question 32

Dans le diagramme 5, que se passe-t'il lorsque **Voiture** est détruite :

- (A) le **Chassis** associé est aussi détruit.
- (B) le **Chassis** associé N'est PAS détruit.
- (C) on ne peut pas savoir.

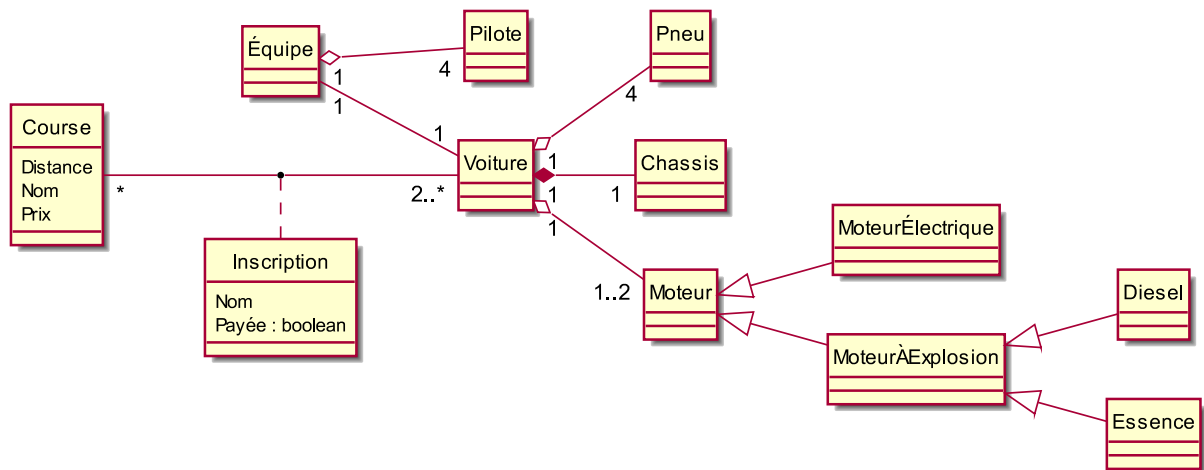


Figure 5: Diagramme 5

Question 33

Dans le diagramme 5, combien de moteurs une voiture peut avoir ?

- (A) 1 exactement.
- (B) 2 exactement.
- (C) 1 ou 2.

Question 34

Dans le diagramme 5, comment s'appelle la relation représentée par **Inscription** ?

- (A) Une classe dérivée.
- (B) Une classe action.
- (C) Une classe association.

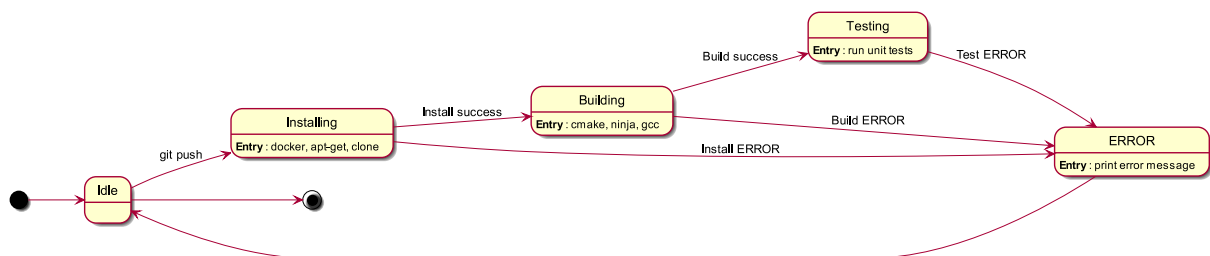


Figure 6: Diagramme 6

Question 35

Le diagramme 6 est :

- (A) un diagramme de séquences.
- (B) un flowchart.
- (C) un diagramme d'états.

Question 36

Dans le diagramme 6, que se passe-t'il après l'état **ERROR** :

- (A) On affiche un message d'erreur.
- (B) On reste dans **ERROR**.
- (C) On transite vers **Idle**.

Question 37

Que fait la commande `git checkout master` ?

- (A) Elle passe sur la branche `master`.
- (B) Elle fait une copie de la branche `master`.
- (C) Elle vérifie la branche `master`.

Question 38

Que fait la commande `git rebase master` ?

- (A) Elle change la base de `master`.
- (B) Elle rejoue les commits de la branche courante par dessus `master`.
- (C) Elle fusionne les commits de la branche courante dans `master`.

Question 39

Que fait la commande `git checkout -b dev` ?

- (A) Elle passe sur la branche `dev`.
- (B) Elle crée une branche `dev` puis passe dessus.
- (C) Elle vérifie la branche `dev`.

Question 40

Que fait la commande `git merge --ff-only dev` ?

- (A) Elle fusionne la branche `dev` dans la branche courante.
- (B) Elle fusionne la branche `dev` dans la branche courante uniquement si les 2 branches sont alignées.
- (C) Elle fusionne la branche `dev` dans la branche courante et quitte prématurément.