

TE1 3,9
TE2 4,0

Poids

2

1

Moy. Théorique

4,0

TE Génie Logiciel

Nom : Eleonore d'Agostino

Date : 9 Juin 2015

| | | | | | | | |
|---|-----|-----|---|---|-----|---|-----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1,5 | 2,5 | 2 | 1 | 8,5 | 4 | 3,5 |
| 1 | 2 | 4 | 2 | 4 | 11 | 8 | 8 |

24

40

4,0

EXERCICE 1 - 1 POINT

Méthodes agiles

Que signifie « Le processus de développement doit être adaptatif » ?

Que le processus doit être capable de s'adapter à des changements de besoins du client, ou de changements dans l'équipe, sans compromettre le projet

Oui (la question concerne spécifiquement le processus)

EXERCICE 2 - 2 POINTS

Méthode XP

Citer au moins 3 tâches essentielles que la méthode XP confie au client.

- décider des ajouts/changements à chaque itération
- décider des scénarios

EXERCICE 3 - 4 POINTS (-1.5 POINT PAR FAUTE)

Méthode XP comparée à UP

Cochez les propositions qui vous paraissent vraies

- ☒ XP comme UP sont deux méthodes « conduites par les cas d'utilisation ».
- ☒ Avec XP, contrairement à UP, les scénarios (user stories) sont écrits au fur et à mesure des itérations plutôt qu'en une fois, avant de commencer le développement
- ☒ Au fil des itérations, module par module, XP comme UP complètent le code présenté au client en s'appuyant sur une architecture centrale élaborée dans l'une des premières phases du développement.
- ☒ Avec XP comme avec UP, il est préconisé d'écrire l'ensemble des classes métier (modèle de domaine) avant de commencer les itérations à proprement parler.

EXERCICE 4 - 2 POINTS

Méthode XP pure

Que signifie « La méthode est pilotée par les tests » ?

Cela signifie que le but principal soit que le programme passe les tests qui aient été définis pour cette itération

7

EXERCICE 5 – 4 POINTS (– 1.5 POINT PAR FAUTE)

Méthodes agiles – Le vrai et le faux

Parmi les assertions suivantes, cochez celles qui reflètent correctement la notion méthode agile.

- 3
11
1/4
-1.5
-1.5
- ✓ ☒ La planification des itérations est un processus évolutif.
☒ L'observation de la vélocité d'une équipe de développement peut entraîner une révision de la planification des itérations
☒ Contrairement à UP, les méthodes agiles préconisent un développement itératif incrémental
☒ Les méthodes agiles souhaitent que l'équipe soit constituée de programmeurs « seniors » (avec expérience), des programmeurs suffisamment « agiles », susceptibles d'écrire directement leur code de manière propre et définitive.

EXERCICE 6 – 11 POINTS – MODÉLISATION DE DOMAINE

C'est les vacances !

Un club de vacances dispose d'un système de réservations d'activités. Voici un résumé de l'expertise métier réalisée par l'analyste :

1. Le club planifie différentes activités.
2. Le club tient compte des employés, on distingue les employés du club (engagés à 100%) des employés temporaires. Les employés du club ainsi que les employés temporaires sont tous caractérisés par un nom.
3. Chaque activité est sous la responsabilité d'un employé du club. Une activité est par ailleurs animée par un employé du club (qui n'est pas forcément le même que le responsable).
4. Chaque activité nécessite l'utilisation d'un certain nombre de types de ressources matérielles (p.exemple. « Œuf », « Mousse à raser, etc..). Chaque type de ressource matérielle est caractérisé par un nom. On souhaite pouvoir enregistrer le fait qu'une activité nécessite 10 œufs et 3 tubes de mousse à raser.
5. Les vacanciers – tous caractérisés par un nom – sont de 2 types : les juniors et les seniors.
6. Les seniors peuvent inscrire tel ou tel vacancier (senior ou junior) à telle ou telle activité pour une date donnée. Chaque inscription (opérée par un senior) concerne une activité pour un vacancier donné.
7. Une inscription peut être annulée ou confirmée par le club.

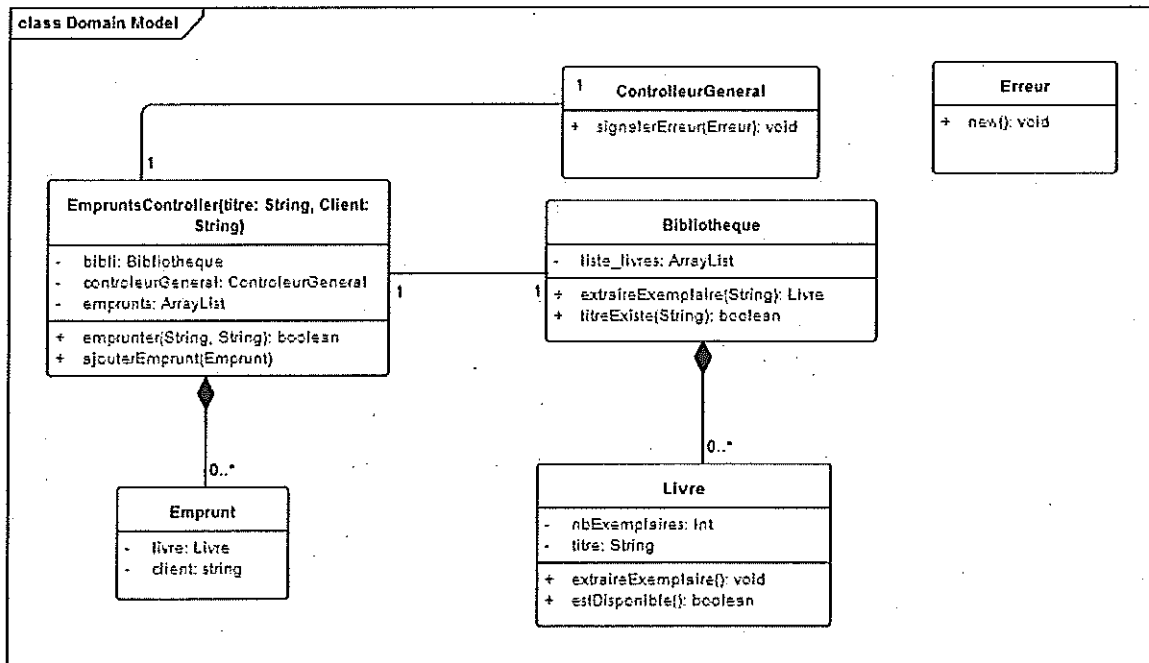
Remarques générales concernant les exercices de modélisation

1. Ne pas rajouter d'informations qui ne se trouvent pas exprimées dans l'énoncé.
2. Toutefois, si l'énoncé vous paraît ambigu (cela peut arriver..), vous pouvez le compléter. Indiquez-le par une remarque.
3. Utiliser l'agrégation ou la composition partout où cela s'y prête. Accompagner votre schéma d'une justification (pourquoi agrégation plutôt que composition).
4. Nommez les associations simples
5. Utiliser la « Généralisation-Spécialisation » partout où cela s'y prête.
6. Utiliser des associations ternaires partout où cela s'y prête.
7. Indiquer la cardinalité de manière explicite (ne pas utiliser de valeurs par défaut)
8. Compléter votre schéma de modélisation par la liste des contraintes que la syntaxe des diagrammes ne permet pas d'exprimer.

EXERCICE 7 – 8 POINTS

DIAGRAMME DE SÉQUENCE

Une bibliothèque



En analysant le code montré ci-dessous, dessiner le diagramme de séquence illustrant le fonctionnement de la méthode `emprunter` (de la classe `EmpruntsControleur`).

Le diagramme fera intervenir les lignes de vie des objets suivants :

- . `EmpruntsControleur`
- . `Bibliotheque`
- . `Emprunt`
- . `Livre`
- . `Erreur`
- . `ControleurGeneral`

```
class EmpruntsControleur
{
    private Bibliotheque bibli;
    private ArrayList<Emprunt> emprunts;
    private ControleurGeneral controleurGeneral;
    [...]
    public Livre emprunter(String titre, String client)
    {
        Livre livre = bibli.extraireExemplaire(titre) ;

        if (livre != null)
            ajouterEmprunt(new Emprunt(client, livre)) ;
        else
            controleurGeneral.signalerErreur (new Erreur("Indisponibilite"));
    }
    private void ajouterEmprunt(Emprunt emprunt)
    { /* Code non montré */ }
}
```

```

class Bibliotheque {
    private ArrayList<Livre> liste_livres;
    [...]
    public Livre extraireExemplaire(String titre)
    {
        Livre livre = titreExiste(titre);
        if (livre!=null)
        {
            if (livre.estDisponible())
                livre.extraireExemplaire();
            else
                livre = null;
        }
        return livre;
    }
    public Livre titreExiste(String titre)
    {
        /* Code non montré.*/
        /* Retourne null si le titre n'existe pas */
        /* Retourne le livre en question si ce dernier existe */
    }
}

class Livre
{
    private String titre;
    private int nbExemplaires;
    [...]

    public boolean estDisponible()
    {
        return nbExemplaires >= 1;
    }

    public void extraireExemplaire()
    {
        nbExemplaires -= 1;
    }
}

class Emprunt
{
    private String client;
    private Livre livre;

```

```

    public Emprunt (String client, Livre livre){
        /* Code non montré */
    }
}

class ControleurGeneral
{
    public void signalerErreur(Erreur erreur)
    {
        /* Code non montré */
    }
}

class Erreur {
    String texte;
    public Erreur(String texte)
    {
        /* Code non montré */
    }
}

```

EXERCICE 8 – 8 POINTS

Dresser le diagramme d'activité correspondant au traitement du processus du choix d'un travail de bachelor, dans GAPS, par un étudiant de l'école d'ingénieurs.

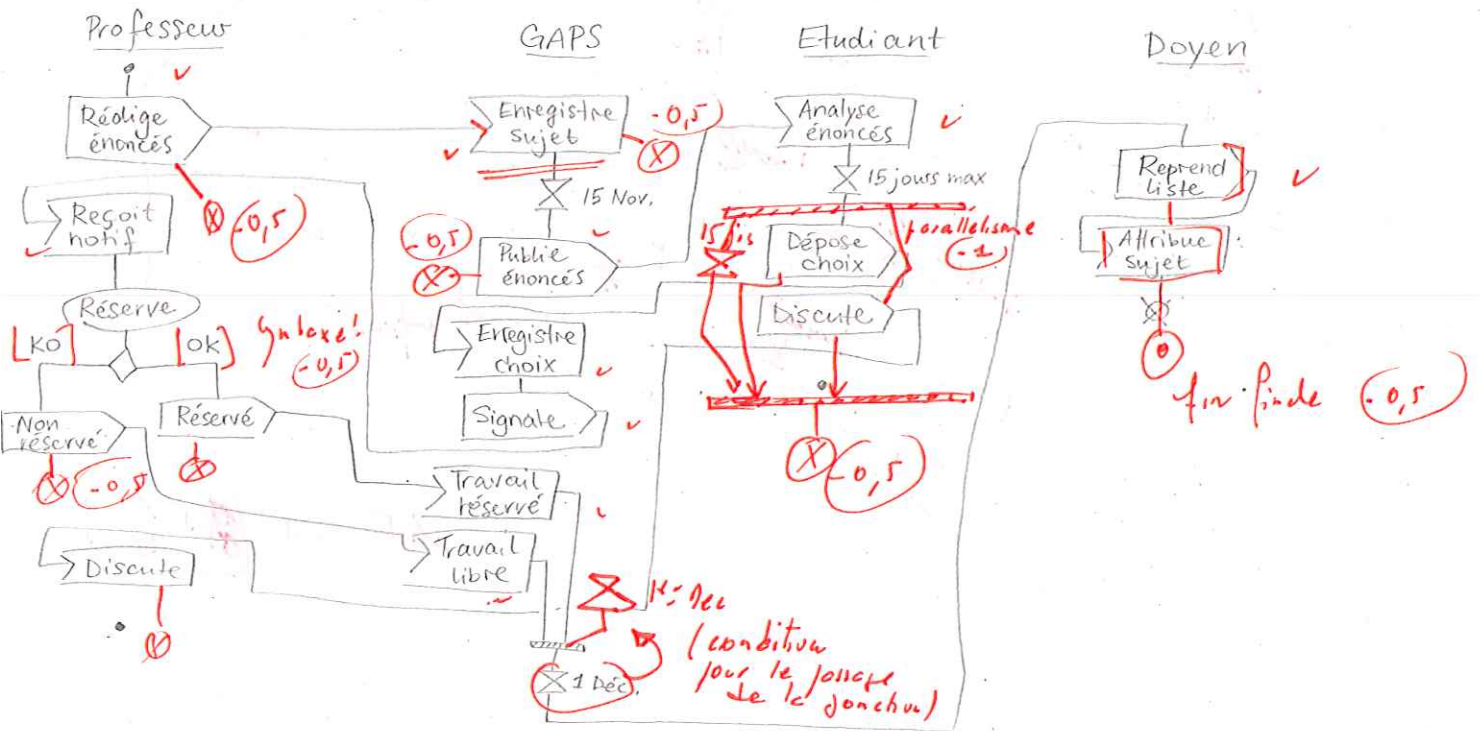
Ce diagramme mettra en jeux 4 couloirs d'activité: le professeur, GAPS, l'étudiant, le doyen

Description du processus

Note importante: Toutes les communications d'informations sont opérées dans le diagramme de manière asynchrone

- Au départ du processus, le professeur rédige ses énoncés de bachelor, puis les dépose sur GAPS qui enregistre le sujet.
- Le 15 Novembre, GAPS publie les énoncés qui ont été déposés. L'étudiant analyse ces énoncés et, pendant 15 jours, jusqu'au 30 Novembre au plus tard, simultanément, dépose ses choix sur GAPS en indiquant une priorité, et discute avec les professeurs.
- L'étudiant peut déposer jusqu'à 3 choix au maximum. A chaque fois qu'il dépose un choix, GAPS le signale au professeur concerné. S'il le désire, le professeur peut alors réserver sur GAPS le travail de bachelor pour l'étudiant qui a montré son intérêt. Le sujet de diplôme est alors définitivement réservé pour l'étudiant.
- Le 1^{er} Décembre, le doyen reprend la liste des étudiants pour lesquels aucun diplôme n'a encore été réservé et attribue à chacun un sujet en fonction des intérêts qu'il a exprimés. Le processus global est alors terminé.

Exercice 8



Eleonore
d'Agostino



Exercice 7

| Emprunts Contrôleur | Bibliothèque | Livre | Emprunt | Erreur | Contrôle Général |
|---------------------|-----------------|-------|---------|--------|------------------|
| | Librairie Exile | | | | |

