

Devoir n°2 (2h00) - correction : thèmes 2 et 4 - Questions 3 et 9

Date : **21/10/2017** Elève :

Note :	Commentaire :
/20	

Ce sujet comporte 13 pages, dont 5 annexes. Les annexes B, D et E sont à compléter. Vos réponses sont à inscrire sur votre copie et le sujet est à rendre dans son intégralité. Les supports de cours sont interdits. L'usage de la calculatrice est autorisé. Vous tâcherez d'apporter des réponses complètes et de vous appliquer sur la qualité de votre rédaction.

Exercice 1 : gestion de projet (9 points)

Sujet :

Jusqu'à fin 2016, la société STAY COOL disposait d'un logiciel « maison » afin d'assurer sa gestion de paie. Ce logiciel avait été réalisé sur mesure par un prestataire extérieur. Ce logiciel lui permettait de produire les bulletins de salaires de ses salariés ainsi que de réaliser les déclarations sociales et fiscales liées aux salaires versés. La société qui a réalisé le logiciel étant éloignée, il s'est avéré difficile de former les utilisateurs au logiciel. L'évolution du logiciel s'est elle aussi avérée compliquée.

A l'occasion de l'entrée en vigueur de nouvelles réglementations, et du fait qu'il n'est pas satisfait de son précédent prestataire, le directeur général de la société a décidé de se munir d'un nouveau logiciel de paie. Le logiciel est à destination du service paie de l'entreprise. Il devra être installé et disponible sur le réseau. Il sera interconnecté au PGI qu'utilise la société et des règles de gestion spécifiques devront être prises en compte. A ce titre, le projet comportera un volet « développement spécifique ». Par ailleurs, le logiciel devrait requérir un certain nombre de paramétrages.

Le prestataire qui développe le PGI qu'utilise STAY COOL, la société CAELUM, propose son propre logiciel de paie. Cependant, le directeur général a souhaité mettre en concurrence CAELUM avec d'autres prestataires. Aussi, il a décidé de lancer un appel d'offres. Finalement, le prestataire concurrent, à savoir l'ESN DEVNOOB, a été retenu.

Travail à réaliser :

En vous servant des annexes A à C, répondre aux questions suivantes :

1.1.	En annexe B, établir la durée des tâches, le diagramme de Gantt et la durée du projet. <i>2,5 points</i> Voir annexe B.
1.2.	Définir la notion de « chemin critique » puis préciser le chemin critique de ce projet. <i>1,5 Point</i> Le chemin critique est l'ensemble (ou la succession) des tâches dont la modification de la durée

	<p>impacte immédiatement la durée totale du projet. Le chemin critique est ici la succession de tâches : A, B, C puis F.</p>
1.3.	<p>Déterminer la date de fin du projet et/ou sa durée calendaire en supposant que le projet ait commencé exactement le 04/01/2017 au matin. <i>1 Point</i> N.B. : si vous n'avez pas trouvé la durée du projet à la question 1.1, indiquer sur votre copie « on suppose que la durée du projet est de 30 jours » et répondre à la question en supposant que la durée du projet est de 30 jours. Le projet a une durée de 29 jours (ouvrés). Si le projet a commencé le 04/01/2017 au matin, en comptant les jours sur le calendrier en annexe C (voir annexe C), on constate que le projet s'est terminé normalement le lundi 13/02/2017 au soir. La durée calendaire du projet est de 41 jours.</p>
1.4.	<p>Etablir le coût de chaque tâche en annexe B afin de retrouver le coût total du projet. <i>1 point</i> Voir annexe B. Le coût total du projet s'élève à $12000 + 30000 + 5000 + 5000 + 3000 + 1000 = 56000\text{€}$.</p>

Une fois le logiciel mis en place, le directeur général a informé le service paie que leur nouveau logiciel de paie, ergonomique et intuitif, est désormais disponible. Il a fait remarquer au service paie qu'ils devaient désormais utiliser ce nouveau logiciel. Après 1 mois, le directeur général constate que le nouveau logiciel de paie a été vaguement utilisé la première semaine mais que cela fait déjà 3 semaines qu'il n'a pas été utilisé.

1.5.	<p>Expliquer la ou les raisons qui ont pu conduire les employés du service paie à ne pas utiliser le nouveau logiciel. Proposer une ou plusieurs solutions qui auraient évité ce problème. <i>1 point</i> En annexe A, on constate que la société DEVNOOB n'a intégré ni présentation du logiciel ni formation des utilisateurs au logiciel. De plus, les utilisateurs finaux (le service paie), semblent avoir tout juste été informés à la fin qu'ils devaient prendre en main le nouveau logiciel par eux-mêmes, sans qu'ils n'aient été impliqués dans le projet. Ce manque d'implication a pu conduire à la réticence voire à l'incapacité des utilisateurs à pouvoir prendre en main le logiciel. Plus encore, n'ayant pu donner leur avis, les utilisateurs finaux trouvent peut-être en ce logiciel un outil qui ne répond pas à leurs attentes. Aucune conduite du changement ne semble avoir été entreprise afin que les utilisateurs puissent s'approprier le nouveau logiciel dans de bonnes conditions. Il aurait été convenable d'impliquer les utilisateurs et, au moins, de leur présenter le logiciel et de leur dispenser une formation.</p>
-------------	--

Après 6 mois, le service paie utilise finalement le nouveau logiciel de paie depuis maintenant 3 mois. Les utilisateurs ont rencontré et continuent à rencontrer de nombreux bogues qu'ils signalent régulièrement à DEVNOOB. Ces anomalies ainsi que des demandes complémentaires formulées par le service paie ont engendré jusque-là un coût supplémentaire de 22000€, facturé par DEVNOOB, et ce sans compter le temps perdu et la gêne occasionnée pour le service paie.

1.6.	<p>Proposer une solution qui aurait pu limiter ou éviter les problèmes rencontrés par le service paie. Aurait-elle eu une incidence financière ? Conclure. <i>2 points</i> La société DEVNOOB semble n'avoir intégré aucune phase de tests et de recette au projet. Il aurait donc fallu mettre en place de réels tests afin de s'assurer du bon fonctionnement du</p>
-------------	--

logiciel ce qui aurait probablement évité la majorité de bogues rencontrés. Par ailleurs, le fait d'impliquer les utilisateurs dès le début du projet aurait pu éviter la multiplication des demandes complémentaires. Bien entendu, la réalisation d'une phase de tests aurait occasionné un travail supplémentaire et donc un coût supplémentaire. Peut-être le directeur général se serait-il alors immédiatement aperçu que la proposition de DEVNOOB n'est pas financièrement plus avantageuse. En fin de compte, le projet coûte jusque-là $56000+22000 = 78000\text{€}$. En conclusion, la solution de DEVNOOB risque de coûter bien plus chère à STAY COOL que la solution de CAELUM ne lui aurait coûté.

Exercice 2 : programmation VBA (5 points)

Sujet :

La société STAY COOL a un fort taux de roulement du personnel. Afin de conserver ses employés, le service paie a proposé de valoriser le travail et la « fidélité » des employés en mettant en place un système de prime annuelle en fonction du salaire annuel brut et de l'ancienneté des salariés. Le directeur général a accepté cette proposition et a établi le barème pour l'année prochaine (voir capture d'écran ci-dessous). Il a demandé à ce que la prime soit versée le 15 janvier de chaque année.

Le directeur général a demandé au service paie de créer et de lui présenter un outil Excel afin d'automatiser le calcul de la prime. Il a précisé que le barème, du moins le pourcentage versé, pourrait varier d'une année à l'autre afin que la prime soit proportionnée au regard de la situation financière de l'entreprise.

Le service paie a proposé la maquette suivante :

	A	B	C	D	E	F	G	H	I	J
1										
2			Calculateur de prime d'ancienneté					Barème		
3					Salaire annuel brut			Ancienneté (années)	Pourcentage (%)	
4					35 000,00 €			0 à 3 ans	0,00%	
5					Ancienneté			4 à 5 ans	1,00%	
6					4			6 à 10 ans	2,50%	
7					Prime :			11 ans et plus	5,00%	
8					350,00 €					
9				Calculer						
10										
11										
12										
13										
14										

Le fonctionnement de l'application peut être résumé comme ceci :

- Quand le barème change, l'utilisateur peut modifier les pourcentages (cellules I4 à I7) ;
- L'utilisateur peut saisir le salaire annuel brut (nombre à virgule dans la cellule E5) ;
- L'utilisateur peut saisir l'ancienneté (nombre entier dans la cellule E7) ;
- Quand l'utilisateur clique sur « calculer », la procédure « `calculer_prime()` » est appelée. Elle calcule puis affiche (cellule C10) la prime en fonction du salaire annuel, de l'ancienneté et des pourcentages.

Travail à réaliser :

- 2.1.** Quelle(s) solution(s) peut-on mettre en place afin d'éviter les erreurs de saisie ? *1 point*
Afin d'éviter les erreurs de saisies, on pourrait mettre en place des validations de données. On pourrait ainsi forcer l'utilisateur à bien saisir un pourcentage (compris entre 0 et 100%) dans les

	<p>cellules I4 à I7, un nombre à virgule (supérieur à 0) dans la cellule E5 et un nombre entier (supérieur à 0) dans le cellule E7. On pourrait également protéger la feuille et/ou le classeur au moyen d'un mot de passe et restreindre les saisies afin que l'utilisateur ne puisse modifier que les cellules E5, E7 et I4 à I7.</p>
2.2.	<p>En vous aidant de l'annexe D, compléter la procédure « <code>calculer_prime()</code> » en annexe E afin qu'elle effectue le traitement décrit ci-avant. <i>4 points</i> Voir annexe E.</p>

Exercice 3 : question de gestion (6 points)

En une à deux pages, à partir de vos connaissances et en vous appuyant le cas échéant sur des situations de gestion diverses, répondre de façon cohérente et argumentée à la question suivante :

**La réalisation d'un projet de création de logiciel se résume-t-elle
seulement à sa programmation, à son codage ?**

Explication du sujet :

Le sujet suggérait que la mise en place d'un logiciel ne se résume clairement pas au seul fait de programmer le logiciel à mettre en place. La question posée suggère en effet qu'il y a d'autres étapes essentielles dans le cadre d'un projet de création de logiciel.

Ainsi, l'élève était invité à préciser brièvement, dès l'introduction, que le seul fait de programmer ne permet certainement pas d'aboutir à un projet de logiciel réussi.

Il convenait ensuite d'avancer quelques arguments. On pouvait par exemple chercher à montrer qu'un projet logiciel ne peut être réussi que si d'autres étapes sont effectuées correctement. Il convenait alors d'expliquer l'intérêt de ces autres étapes.

L'on s'attendait naturellement à ce que l'utilisateur mobilise des connaissances acquises et/ou des exemples étudiés en cours.

Proposition de corrigé :

Remarque : il s'agit d'un corrigé indicatif. A l'examen, l'on n'attend pas de vous une réponse aussi complète. Si vous le faites, tant mieux... Au demeurant, on valorise en outre les éléments suivants :

- *Argumentaire structuré : 1 introduction + 2 ou 3 arguments détaillés + 1 conclusion ;*
- *Arguments pertinents : des arguments clairs répondant effectivement à la problématique ;*
- *Mobilisation de connaissances : utilisation des termes et notions appris en cours ;*
- *Illustration : utilisation pertinente d'exemple étudiés ;*
- *Présentation et syntaxe : copie lisible, phrases claires et argumentation logique.*

Un logiciel, encore appelé « programme », est constitué de « codes » qu'une machine informatisée peut interpréter afin d'offrir diverses fonctionnalités à des utilisateurs. On pourrait dès lors croire que, pour réaliser un projet de logiciel, il suffit de rédiger ces fameux « codes » sources dont le logiciel est constitué. Aussi, l'on peut se poser la question : la réalisation d'un projet de création de logiciel pour un client se résume-t-elle seulement à de la programmation, à du codage ? Au travers de plusieurs arguments, nous montrerons que la réalisation d'un tel projet requiert bien d'autres tâches que le seul codage d'un logiciel.

Tout d'abord, avant de programmer, il faut encore savoir ce qu'il faut programmer. Il convient avant toute chose d'identifier le besoin ce qui nécessite une phase de spécification débouchant en outre sur un cahier des charges et/ou une spécification qui définit clairement la solution à réaliser, à programmer. Cette phase initiale de conception, prise en charge par la MOA (maîtrise d'ouvrage), occupe une place importante dans un projet. Cette étape peut prendre du temps, et même un temps absolument non négligeable au regard de celui nécessaire pour développer le logiciel. C'est d'abord le donneur d'ordre, le client, que l'on cherche à satisfaire ! Aussi, il semble essentiel de discuter avec lui afin d'établir un cahier des charges répondant de manière pertinente à son ou à ses besoins. Au contraire, programmer immédiatement conduirait inévitablement à une solution hasardeuse répondant probablement mal au besoin du client.

Egalement, le prestataire qui réalise la solution se doit aussi de rentrer dans ses frais. Aussi, il ne peut mobiliser des ressources inutilement. Il doit avoir une idée claire du temps qu'il faudra pour réaliser le projet, non seulement pour respecter les contraintes de qualité, de coût et de délai du client, mais encore tout simplement pour gagner de l'argent afin que sa société fonctionne. A quoi bon se lancer dans la programmation si l'on n'a pas encore idée des ressources humaines à mobiliser ? Pour les connaître, il convient donc non seulement de savoir ce qu'il faut faire mais encore de savoir combien de temps il faut pour le faire. Le projet doit alors être divisé en tâches élémentaires puis chacune des tâches doit être chiffrée, c'est-à-dire que la durée de chacune des tâches doit être évaluée. La durée et le coût du projet peuvent seulement alors être estimés.

Par ailleurs, une fois le projet « en route », il est nécessaire d'assurer la bonne conduite du projet. Au-delà de la programmation, l'équipe va devoir être coordonnée par un chef de projet afin que tous les membres de l'équipe projet puissent travailler efficacement de manière collaborative. En effet, sans cohésion et sans travail collaboratif, on risque d'obtenir un résultat décousu... Plus encore, le chef de projet se doit d'être l'interface entre le client et le prestataire qu'il représente. Effectivement, pour qu'un projet réussisse, il faut aussi un travail collaboratif entre le prestataire et le client. Le chef de projet peut ainsi être amené à discuter avec ce dernier de l'état d'avancement du projet ou encore de corrections à apporter à la réalisation en cours afin de s'assurer de la satisfaction du client. De fait, la réussite du projet requiert l'implication du client dans la réalisation de son propre projet. Il en va de la qualité de la réalisation. On notera encore que l'implication du client et la qualité d'un logiciel passent par la formation des utilisateurs et par la rédaction d'une ou plusieurs documentations. Après tout, ce ne sont pas les développeurs (les programmeurs) qui utilisent le logiciel !

De surcroît, le seul fait de programmer un logiciel n'en assure pas pleinement la qualité intrinsèque. Plus que « coder », il faut encore « tester » le logiciel ! La mise en place de tests, éventuellement de tests automatisés (tests unitaires) va permettre de s'assurer du bon fonctionnement du logiciel. Cela va permettre d'éviter la prolifération de bogues qui peuvent soit occasionner une petite gêne, dans le meilleur des cas, soit causer un dysfonctionnement du logiciel. Dans cette lignée, il convient de recetter le logiciel au travers d'une étape appelée « recette ». Celle-ci va permettre de s'assurer de la conformité du logiciel au regard du cahier des charges. Dans le cas où des anomalies sont rencontrées, des

corrections seront apportées. Dans ces conditions seulement, on a une plus grande certitude de déboucher sur un logiciel répondant aux attentes du client.

Finalement, un projet logiciel n'est jamais complètement terminé... Il est ainsi nécessaire d'assurer la maintenance du logiciel car le client peut quand même être amené à rencontrer des bogues ou encore à souhaiter des évolutions afin que le logiciel continue à répondre ou réponde de mieux en mieux à ses attentes.

En conclusion, si la réalisation d'un logiciel requiert inévitablement une phase de programmation, nombreuses sont les étapes et tâches accomplir autour de cette phase afin de pouvoir espérer obtenir dans les délais un logiciel abouti, qualitatif et répondant effectivement aux besoins du client.

Barème :

- *Argumentaire structuré : 1 point ;*
- *Arguments pertinents et/ou illustrations pertinentes : 2 points ;*
- *Mobilisation de connaissances : 2 points ;*
- *Présentation et syntaxe : 1 point.*

Annexe A : offres commerciales

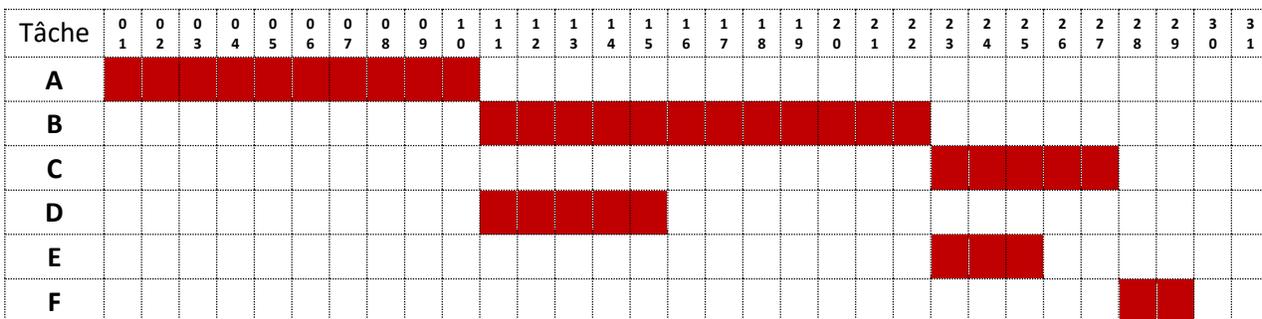
On énumère ci-après les principales caractéristiques des propositions commerciales transmises respectivement par CAELUM et par DEVNOOB suite à l'appel d'offres lancé par le directeur général de la société STAY COOL :

Société CAELUM	Société DEVNOOB
Logiciel standard « CAELUM PAIE » développé par CAELUM.	Logiciel sur mesure développé par DEVNOOB.
<u>Description</u> : le logiciel CAELUM PAIE est un logiciel standard et paramétrable développé et maintenu par la société CAELUM depuis plus de 20 ans. Vous y trouverez toutes les fonctionnalités dont vous pouvez avoir besoin !	<u>Description</u> : le logiciel sera développé et paramétré spécifiquement pour STAY COOL afin qu'il réponde au mieux à vos attentes. Une réalisation unique à prix unique ! Vous y trouverez uniquement les fonctionnalités dont vous-même avez besoin !
<u>Prix incluant</u> : Mises à jour incluses Logiciel : 120€/mois et par utilisateur Formation : 1200€ par jour (déplacement inclus) Développement spécifique : 18000€ (1 ^{ère} année) Paramétrage : 9000€ (1 ^{ère} année)	<u>Prix incluant</u> : Définition du besoin Développement logiciel Interconnexion avec le PGI Rédaction de la spécification technique Paramétrage Installation et configuration des postes Garantie sur 3 mois contre toute imperfection
<u>Prix annuel</u> : 10 utilisateurs (14400€/an), 2 jours de formation par an (2400€/an) - 1 ^{ère} année : 43800€ - Années suivantes : 16800€/an	<u>Prix annuel</u> : 56000€
<u>Investissement sur 3 ans</u> : 77 400€	<u>Investissement sur 3 ans</u> : 56000€

Annexe B (à compléter) : charge de travail et diagramme de Gantt

La société DEVNOOB a établi l'ordonnancement suivant :

Tâche	Intitulé	Charge en j.h	Ressources humaines	Durée en jours	Tâche(s) antérieure(s)
A	Définition du besoin	30	3	10	
B	Développement logiciel	60	5	12	A
C	Interconnexion avec le PGI	10	2	5	B
D	Rédaction de la spécification technique	10	2	5	A
E	Paramétrage	6	2	3	B
F	Installation et configuration des postes	4	2	2	C, E



N.B. : vous pouvez tracer un trait plein proprement et lisiblement plutôt que de colorier les cases unes à unes.

La durée du projet est de : **29 jours**

A partir de l'ordonnancement ci-dessus, la société DEVNOOB a établi le chiffrage suivant :

Tâche	Intitulé	Coût par j.h	Coût de la tâche
A	Définition du besoin	400€	400 x 30 = 12 000€
B	Développement logiciel	500€	500 x 60 = 30 000€
C	Interconnexion avec le PGI	500€	500 x 10 = 5 000€
D	Rédaction de la spécification technique	500€	500 x 10 = 5 000€
E	Paramétrage	500€	500 x 6 = 3 000€
F	Installation et configuration des postes	250€	250 x 4 = 1000€

Annexe D (à compléter) : rappels de VBA

Déclaration de variables :

```
' Déclaration d'une variable nommée "maVariable" de type entier long, c'est-à-dire un « grand entier »  
Dim maVariable As Long
```

Quelques types de variables : *Integer* (entier court), *Long* (entier long), *Double* (nombre à virgule), *String* (chaîne de caractères).

Récupération de la valeur d'une cellule :

```
maVariable = Range("C6").Value
```

Modification de la valeur d'une cellule :

```
Range("J12").Value = 12  
Range("J13").Value = maVariable
```

Structure conditionnelle :

```
If condition1 Then  
...  
Elseif condition2 Then  
...  
Else  
...  
End If
```

Opérateurs de comparaison :

Une condition peut être construite en utilisant des opérateurs de comparaison. En VBA, les principaux opérateurs de comparaison sont : = (égal à), <> (différent de), >= (supérieur ou égal à), <= (inférieur ou égal à), < (inférieur à), > (supérieur à).

Opérateurs arithmétiques :

On peut effectuer des calculs au moyen d'opérateurs arithmétiques. En VBA, les opérateurs les plus courants sont : + (addition), * (multiplication), - (soustraction), / (division) et & (concaténation).

Annexe E (à compléter) : procédure calculer_prime()

Sub calculer_prime()

' Déclaration des variables

' Salaire annuel brut du salarié

Dim salaireAnnuel **As** Double

' Ancienneté du salarié

Dim anciennete **As** Integer

' Pourcentage de prime en fonction de l'ancienneté

Dim pourcentage **As** Double

' Prime du salarié

Dim prime **As** Double

' Récupération du salaire annuel et de l'ancienneté *0,75 point*

salaireAnnuel = Range("E5").Value

anciennete = Range("E7").Value

' Récupération du pourcentage de prime *1,5 points*

If anciennete <= 3 **Then**

 pourcentage = Range("I4").Value

Elseif anciennete <= 5 **Then**

 pourcentage = Range("I5").Value

Elseif anciennete <= 10 **Then**

 pourcentage = Range("I6").Value

Else

 pourcentage = Range("I7").Value

End If

' Calcul de la prime du salarié *1 point*

prime = salaireAnnuel * pourcentage

' Affichage de la prime calculée *0,75 point*

Range("C10").Value = prime

End Sub