## N° du projet : 41

## Noms des élèves : Baptiste HUBERT, Antoine GATARD, Eric KHELLA

**Nom de l’évaluateur : PAJUSCO Patrice Date : 22/03/2024**

|  |  |  |
| --- | --- | --- |
| **Commentaires généraux :** |  | |
| A noter un démarrage réel sur la partie codage avec des échanges intéressants. Par contre, le document ne reflète pas du tout :  sur la forme la formulation est pompeuse (contexte de haute technicité où chaque détail compte, réalisation d'un système à la hauteur des enjeux stratégiques, en utilisant des technologie de programmation Python...), pas de numéros de pages, des champs par défaut non renseignés. Bref beaucoup de choses maladroites, du blabla mais pas de vraiment concret significatif.  sur le fond :   * le contexte est faux. La SNCF n'est pas le client de ce programme et cela n'a aucun rapport avec les mesures faites par IRIS. Vous ne parlez pas du contexte FRMCS, des projets collaboratifs, ni des expérimentations radios d'ou viennent les contraintes... * il y a deux type de fichiers avec des output différents, ce n'est pas clairement expliqué. Les traitements sont différents. Un exemple avec une image de chaque fichier aurait été beaucoup plus explicite pour présenter le besoin et la problématique * Je croyais que XML ne servait à rien, ce n'est plus clair après avoir lu le cahier des charges * le format de sortie n'est pas détaillé, un fichier csv...v oui mais quel contenu, quel unité... * La problématique de l'interpolation n'est pas clairement expliquée (c'est un des problèmes important) * la réalisation d'une classe pour l'accès direct à une image à un instant arbitraire n'est pas indiqué. | |  |

## Principe général d’utilisation de la grille :

Le schéma suivant résume le principe de lecture du contenu de la grille d’évaluation **pour chaque critère** :

**Tous les points   
de la colonne « = » sont vérifiés ?**

**Un point   
de la colonne « + » est vérifié ?**

**OUI**

**OUI**

**NON**

**NON**

**-**

**=**

**+**

Globalement, **les éléments de la colonne « = » indiquent ce que le livrable doit vérifier**, alors que les éléments des colonnes « + » et « - » proposent une liste **non exhaustive** d’exemples pouvant justifier l’une de ces deux évaluations.

**Consignes pour l’évaluation :**

**L’évaluateur doit obligatoirement choisir l’un des 3 niveaux « + », « = » ou « - » pour CHAQUE critère** en changeant la couleur de fond de la cellule (par exemple en vert) ou par tout autre moyen permettant d’identifier sans ambiguïté, le niveau retenu.

**Pour le niveau « = »**, si vous estimez que tous les points du niveau sont **particulièrement bien réalisés**, changer également la couleur de fond de la cellule **« Niveau très satisfaisant »**.

Si des éléments isolés apparaissent dans les colonnes voisines (voire dans la colonne choisie), ne pas hésiter à les surligner pour les indiquer aux élèves.

| **Critères** | **Niveaux** | | | |
| --- | --- | --- | --- | --- |
| **-** | **=** | | **+** |
| **1** – Le **contexte** présenté est-il pertinent par rapport au projet ? | Absence des raisons du besoin ou présence de raisons erronées.  Le contexte présenté ne sert pas à justifier le besoin (trop vague, trop global). | Le pourquoi du besoin est exprimé de manière **pertinente** (le contexte vient justifier ce besoin).  Le besoin est positionné explicitement par rapport à un existant ou une absence d’existant. | Niveau très satisfaisant | Le repositionnement explicite n’est pas simplement qualitatif mais fait intervenir certains aspects **quantifiés**. |
| **2** – Le **périmètre** du système à développer est-il pertinent vis-à-vis du besoin ? | La définition des acteurs (clients / utilisateurs / composants externes) ou des entrées/sorties n’est pas suffisamment précise ou bien elle est incomplète, ce qui ne permet pas de définir le périmètre du système à développer en rapport avec le besoin du client.  Le périmètre ne correspond pas au besoin exprimé. | Le périmètre du système est explicité en spécifiant les acteurs (utilisateurs, clients et composants externes) qui interagissent avec le système, les entrées/sorties du système par rapport à son environnement. Une clarification du type « Qu’est-ce qui n’est pas dans le système ? » est aussi possible.  **Le périmètre envisagé correspond au besoin du client** mais cette correspondance n’est pas décrite explicitement. | Niveau très satisfaisant | Le périmètre envisagé est **justifié** explicitement par rapport au besoin du client. |
| **3** – L’expression des **fonctionnalités** (fonctions principales) attendues est-elle pertinente ? | Ce ne sont pas des fonctionnalités du système qui sont exprimées ; par exemple, expression d’une contrainte, … | L’expression de chaque fonctionnalité est explicite (verbe + complément). Les entrées/sorties de chaque fonction sont indiquées (nature et rôle).  Le rôle de chaque fonction est **pertinent** par rapport au besoin. | Niveau très satisfaisant | Les plages d’acceptation des « valeurs » des entrées/sorties sont indiquées et **justifiées**.  Lajustification des valeurs données s’appuie sur des informations **réalistes** et vérifiées. |
| **4** – Le **découpage fonctionnel** couvre-t-il l’ensemble du besoin exprimé (et juste celui-ci) ? | La couverture du besoin est partielle (il manque une ou des fonctions pour couvrir un élément de besoin).  La couverture du besoin est inadéquate (il y a des fonctions qui ne correspondent à aucun besoin) | Le découpage proposé couvre **l’ensemble** du besoin exprimé sans fonctions superflues. | Niveau très satisfaisant | La couverture proposée par le découpage est **expliquée** (même partiellement). |
| **5** – Les **contraintes** associées au système sont-elles correctement exprimées ? | Il n’existe aucune contrainte ou bien il manque de manière flagrante, certaines contraintes relatives à des aspects essentiels du système.  Des contraintes sont indiquées mais sans quantification ou avec une quantification floue ou trop vague ou trop large ou **inadéquate** par rapport au besoin. | Les contraintes sont explicites (verbe + complément) avec une quantification **acceptable** pour le problème posé (par exemple, « le système doit garantir un temps de réponse < 10s pour le traitement d’une image de 10 Mo) | Niveau très satisfaisant | Les valeurs associées à chaque contrainte sont **justifiées** par rapport à des éléments explicites, issus de l’expression du besoin ou de la présentation du contexte, ou bien la logique qui permet d’aboutir à ces valeurs est décrite et argumentée. |
| **6** – Les **cas d’utilisation** du système sont-ils représentatifs du besoin réel ? | Pas de cas d’utilisation.  S’il existe des cas d’utilisation, ceux-ci ne correspondent pas aux cas d’utilisation les plus **pertinents** / standards / **fréquents** du système. | La description des cas d’utilisation classiques ou standards du système indiquant le rôle et les actions des différents acteurs ainsi que les résultats attendus, correspondent à des **cas** **réalistes/concrets** d’utilisation du système à développer. | Niveau très satisfaisant | Justification que l’ensemble des cas d’utilisation donné couvre les besoins exprimés.  Les cas « limite » ou les cas particuliers (réaction du système dans le cadre d’un environnement non standard) évoqués sont **plausibles**, tant dans leur occurrence que dans leurs réponses. |
| **7** – Est-ce que les **scénarios** proposés correspondent aux cas d’utilisation présentés (et donc aux fonctions du système) ? | Absence d’exigence dynamique du système.  Les scénarios sont non pertinents pour les cas d’utilisation du système. | Chaque scénario explicite l’enchaînement des interactions / étapes nécessaires pour répondre à un cas d’utilisation (au moins pour un fonctionnement nominal). Cet enchaînement est **pertinent**. | Niveau très satisfaisant | Des cas « limite » ou des cas particuliers d’utilisation sont prévus et les scénarii envisagés sont cohérents et en phase avec les attentes du client.  La démarche pour **s’assurer de cette adéquation** est explicitée et mise en œuvre. |
| **8** – Les fonctions et contraintes ont-elles des **priorités** associées, conformes à l’attente du client ? | Absence ou présence partielle de priorité | Une priorité est associée à chaque fonction et contrainte.  Cette priorité est **conforme** au besoin du client. | Niveau très satisfaisant | La priorisation des fonctions et contraintes est justifiée explicitement.  Cette priorisation descend au niveau des scénarios illustrant les cas d’utilisation.  La démarche pour **valider** l’adéquation entre la priorisation proposée et les attentes du client est explicitée et mise en œuvre. |
| **9** – **Validation** du système / « produit » attendu | L’aspect « validation » du produit n’est pas décrite ou de manière partielle. | Le document indique comment il est envisagé concrètement de valider le produit par rapport à **toutes** les fonctionnalités et contraintes indiquées.  Les éléments présentés sont **pertinents** pour s’assurer que le produit correspond effectivement à toutes les attentes du client (spécifiées dans le CdCF). | Niveau très satisfaisant | Les modalités du processus de validation envisagé résultent d’une **interaction spécifique avec le client** et ont reçu son **accord explicite**. |

**Glossaire (au sens de l’analyse du besoin client vue en cours)**

La description fonctionnelle (statique) du produit / du système se réalise au travers d’une série de « fonctions de service » qui répondent au besoin du client. Elles s’expriment donc **exclusivement en termes de finalité**. Deux types de fonctions de service sont explicités :

* « fonction principale » ou **fonction** dans la grille : c’est une fonctionnalité du produit / du système (une réponse à « pourquoi le produit est créé ? »)
* « fonction contrainte » ou **contrainte** dans la grille : elle traduit une contrainte particulière à laquelle le produit doit faire face / répondre.

La description du comportement (dynamique) du produit / du système se réalise au travers d’une série de « cas d’utilisation ». Chaque cas d’utilisation décrit, pour un service rendu par le système, les interactions du système avec son environnement. Les éléments de l’environnement avec lesquels le système interagit sont les **acteurs** (utilisateurs, autres systèmes, …).

* Ainsi chaque « cas d’utilisation » se décline en un ensemble de scénarios (nominal, alternatif, échec) décrivant les interactions du système avec les acteurs qui constituent l’environnement.
* Chaque scénario peut être décrit sous forme de « diagrammes de séquence » qui indiquent l’enchaînement chronologique des interactions entre les acteurs et le système.