

**SUJET D'EXAMEN HM 40 – 26 juin 2024 – durée 2h**  
(Aucun document autorisé)

**Exercice 1. Intentionnalité.**

**1) Perception (1 pt).** Trois grandes approches philosophiques traditionnelles de la perception sont la thèse du réalisme direct, la thèse représentative, et le phénoménalisme. Expliquer brièvement en quoi consistent ces trois thèses et en quoi elles diffèrent l'une de l'autre. On pourra illustrer chaque cas par un schéma simple faisant intervenir le percevant, l'expérience visuelle, la donnée sensible, l'objet de la perception. Indiquer quelles sont les arguments et objections émis par ces différentes approches pour justifier de leur validité.

**2) Intentionnalité intrinsèque/relative (1 pt).** Donner une liste hiérarchisée de systèmes dits « intelligents » dont les capacités intentionnelles sont manifestement intrinsèques au système. Donner une liste hiérarchisée de systèmes dits « intelligents » dont les capacités intentionnelles sont manifestement relatives à un agent extérieur au système. Pour rappel, l'intentionnalité est la capacité pour un système de posséder des états mentaux, des buts, et d'agir par rapport à une finalité.

**3) Etats mentaux (2 pts).** Un système interactif doit être adapté aux capacités perceptives, motrices et cognitives de ses utilisateurs : l'utilisabilité d'un système renvoie alors aux capacités humaines. Ainsi, l'IHM se trouve à un carrefour singulier entre l'informatique, la biologie et les sciences humaines et sociales (philosophie, psychologie). Par exemple, le concepteur peut analyser la tâche utilisateur dans les termes de l'intentionnalité.

Présenter par un schéma simple le cycle des activités mentales et physiques en jeu dans l'accomplissement d'une tâche avec un ordinateur/écran/clavier/souris. Mettre en évidence les états mentaux, les actions physiques, les perceptions et l'interprétation. Mettre en évidence la décomposition entre niveaux physiques/syntaxiques/sémantiques. Que doit faciliter la réalisation d'une interface de qualité.

**Exercice 2. Détruire un fichier (3pts).** Nous considérons quatre façons différentes de détruire un fichier dans une interface de type « Explorateur de fichiers » :

- (a) La première solution consiste à sélectionner l'icône du fichier à détruire puis glisser-déposer avec la souris l'icône du fichier à détruire dans la corbeille.
- (b) La deuxième solution consiste à sélectionner l'icône du fichier à détruire puis à sélectionner l'option « supprimer » dans le menu contextuel activé avec le bouton droit de la souris.
- (c) La troisième solution consiste à sélectionner l'icône du fichier à détruire puis à sélectionner l'icône de suppression marqué d'une croix rouge dans le menu « accueil » affiché sous forme de toolbox.
- (d) La quatrième solution consiste à sélectionner l'icône du fichier à détruire puis à appuyer au clavier sur la touche « suppr ».

1) Encoder avec KEYSTROKE les quatre méthodes décrites ci-dessus.

2) Calculez les temps de réalisation des quatre méthodes et les classer en termes de performance, en considérant les valeurs moyennes suivantes : Opérateur K (frappe de touche) : 200 ms. Opérateur P (Pointing) : 1100 ms. Opérateur H (Homing) : 400 ms. Opérateur M (Penser) : 1300 ms.

3) En pratique expliquez pourquoi vous n'utilisez généralement pas telle ou telle méthode et à quel niveau les aspects perception font la différence.

### Exercice 3. Structure d'écran

1) (2 pts) Commentez selon les principes de la structuration d'écran des deux affichages suivants d'un message d'erreur sous deux navigateurs lorsqu'un site WWW n'est pas atteignable. Indiquer les défauts et améliorations entre les deux versions. Donner les diagrammes de structure des écrans et préciser les sujets pragmatiques. Indiquer à l'aide du graphe de transitions l'intérêt des modifications apportées aux écrans entre les deux navigateurs.

Navigateur 1 :



Navigateur 2 :



#### Adresse introuvable

Firefox ne peut trouver le serveur à l'adresse www.libe.fr.

- Veuillez vérifier la syntaxe de l'adresse (saisie de **ww**.exemple.com au lieu de **www**.exemple.com par exemple) ;
- Si vous n'arrivez à naviguer sur aucun site, vérifiez la connexion au réseau de votre ordinateur ;
- Si votre ordinateur ou votre réseau est protégé par un pare-feu ou un proxy, assurez-vous que Firefox a l'autorisation d'accéder au Web.

Réessayer

2) (1pt) Que pensez-vous du pop-up de confirmation suivant. Justifier votre critique, positive ou négative. Quel est le nom de l'effet visuel « irrépessible » qui intervient ici.



3) (1 pt) On veut évaluer les 2 écrans suivants permettant la saisie d'un montant à rembourser d'un crédit bancaire. Préciser votre critique des 2 écrans selon les principes de structuration, Gestalt, règles d'ergonomie.

Ecran 1 :

**Crédit Renouvelable**

Titulaire : Mr

Synthèse Description Souscription

1 Entrez **le montant** que vous voulez rembourser :

Date d'effet : 18/05/2024

Veillez entrer un montant multiple de 10 € avec un minimum de 50 € ou « T » pour un remboursement total de votre crédit renouvelable.

2 Veuillez **valider** ou annuler votre demande.

Valider Annuler

Ecran 2 :

**Montant à rembourser**

Depuis : N°

Montant

€

Veillez saisir un montant multiple de 10 €, minimum de 50 €.

Rembourser

[Rembourser la totalité des 417.66 €](#)

**Exercice 4 (3pts).** Les diagrammes état/transition imbriqués, encore appelés machines à état ou « statecharts » sont un outil visuel commode pour représenter des techniques d'interaction graphiques avec souris/clavier. Et parfois des situations simples méritent une attention particulière pour être décrites correctement. On pourra utiliser les événements *enter* et *leave* qui correspondent à l'entrée et la sortie du curseur dans la zone d'un bouton. Les événements *down* et *up* pour un appui avec bouton souris et relâchement bouton souris respectivement. Lorsque l'on appelle "*arm<ms>*" dans une action, l'évènement *timeout* est émis à l'expiration du délai (exprimé en millisecondes). On pourra utiliser des booléens comme *armed* = true/false pour indiquer que le timeout est armé ou encore *entered* = true/false pour indiquer que l'on est sur la zone d'un objet. Par exemple, l'évènement *down[not armed]* signifie click souris avec temporisateur désarmé. On peut aussi définir des événements spécifiques comme *click-on* pour *down[entered]*. Donner un automate état/transition, avec les événements et actions appropriées, pour les tâches et dispositifs suivants :

- 1) Construction d'un polygone fermé dans une zone de dessin avec la souris. Il s'agit d'entrer une suite de points à la souris en terminant par le premier point entré. Toute action non autorisée, ou l'absence d'action pendant 5s annule la commande.
- 2) Interaction avec un bouton. L'entrée et la sortie du curseur dans la zone du bouton entraîne une accentuation du rendu du bouton. L'action associée au bouton s'exécute sur le relâchement du bouton de sélection et on rappelle que si l'on relâche le bouton de la souris alors que l'on est en-dehors de la zone du bouton, la sélection est annulée.
- 3) Construire la machine à états qui détecte des simple-clics et des double-clics sur un icône de fichier et exécute l'action correspondante. Un double-clic est une séquence de deux clics séparés de moins de 200 ms.

**Problème de modélisation (6pts).** On veut réaliser un système d'information sur les transports aériens. Le système peut être interrogé en précisant la compagnie de transport, les villes origines/destinations, les dates, et autres contraintes pour connaître les différents vols possibles. L'utilisateur peut visualiser les réponses puis ensuite réserver le vol qui l'intéresse et en régler le prix. Le noyau fonctionnel est une base de données sur les transports aériens accessible par des requêtes SQL. On peut également utiliser un module de reconnaissance vocale, ou d'écriture de phrases en langage naturel pour construire ces requêtes. Le système est conçu suivant le principe du bureau avec une barre d'outils principale et des fenêtres multiples autonomes correspondant aux différentes fonctionnalités offertes.

1. Préciser les caractéristiques du/des postes de travail. Spécifier les fonctionnalités du système en termes de buts à atteindre et/ou tâches à réaliser en donnant un modèle HTA. Préciser un modèle des données via un modèle de classes UML précisant le vocabulaire du domaine associé.
2. Donner un modèle de tâches SADT de l'application en plusieurs étapes, en partant du diagramme de contexte de niveau 0 et par raffinement successifs couvrant les différentes fonctions offertes.
3. Proposez un modèle PAC de l'interface homme/machine. Le détail de la partie Contrôle n'est pas demandé.