

Rapport de laboratoire

|  |  |
| --- | --- |
| **No de laboratoire** | 4 |
| **Étudiant(s)** | Sébastien Lago  Marc-André Allard  Israël Hallé |
| **Code(s) permanent(s)** | LAGS04128102  HALI17049101  ALLM09029106 |
| **Cours** | LOG430 |
| **Session** | Hiver 2014 |
| **Groupe** | 02 |
| **Professeur** | B. Galarneau |
| **Chargés de laboratoire** |  |
| **Date de remise** | avril 2014 |

Table des matières

[Attributs de qualité choisis 3](#_Toc382920331)

[Scénarios d’attribut de qualité 3](#_Toc382920332)

# Attributs de qualité choisis

**Disponibilité**

Justification: Si le système (Mission Editor) tombe en panne pendant que l’utilisateur est en train de travailler dessus, il doit être capable de revenir à son état précédent le plus rapidement possible en minimisant les pertes de données. La disponibilité est la qualité la plus importante pour le système Mission Editor 2.0.

**Performance**

Justification: Les utilisateurs du système étant parfois contraints par le temps et la quantité de mémoire vive étant limitée à seulement 150 mégaoctets, le système Mission Editor 2.0 doit être conçu de manière à utiliser les ressources d’une manière efficace pour exécuter les tâches souhaitées dans des délais raisonnables tout en restant en deçà de la limite imposée par la quantité de mémoire vive.

**Convivialité**

Justification: Il est important que le logiciel soit conçu afin que les utilisateurs puissent l’utiliser efficacement. De plus, les nouveaux utilisateurs doivent être capables d’apprendre rapidement à utiliser le logiciel Mission Editor 2.0.

# Scénarios d’attribut de qualité

**Disponibilité**

* **Scénario complet** : L’utilisateur modifie une mission et une panne survient. Il doit pouvoir reprendre son travail le plus rapidement possible.
* **Source** : Interne. L’éditeur de mission Mission Editor 2.0
* **Stimulus** : Crash. Le système ne répond plus.
* **Artefact** : L’éditeur de mission Mission Editor 2.0
* **Environnement** : Système en opération normale.
* **Réponse** : Lancer une exception contenant l’identifiant et le message de l’erreur.
* **Mesure** : Temps moyen pour réparer le système est moins de 30 secondes (MTTR). Temps moyen entre deux pannes (MTTF) ne dépasse pas une panne au trois heures.

**Disponibilité**

* **Scénario complet** : L’utilisateur essais de modifier une mission et le système plante. Il veut pouvoir continuer à travailler sans perdre toutes les modifications qu’il avait fait.
* **Source** : Interne. L’éditeur de mission Mission Editor 2.0
* **Stimulus** : Crash. Le système ne répond plus.
* **Artefact** : L’éditeur de mission Mission Editor 2.0
* **Environnement** : Système en opération normale.
* **Réponse** : Restoration du système.
* **Mesure** : Temps moyen pour continuer le travail après une panne est de 20 secondes.

**Performance**

* **Scénario complet** : Un utilisateur du système veut créer une mission et il dispose de peu de temps.
* **Source** : Un utilisateur Mission Editor 2.0.
* **Stimulus** : Demande de création de mission.
* **Artefact** : Le système Mission Editor 2.0.
* **Environnement** : En mode normal ou surchargé.
* **Réponse** : Traitement des demandes de mise à jour des fichiers de configuration.
* **Mesure** : Temps de latence maximal de 3 secondes.

**Convivialité**

* **Scénario complet** : L’utilisateur veut rajouter une zone d’intérêt au bassin. Il veut pouvoir le faire en utilisant la souris le moins de clics de souris possible.
* **Source** : L’utilisateur du Mission Editor 2.0.
* **Stimulus** : L’utilisateur souhaite ajouter une zone d’intérêt au bassin.
* **Artefact** : Le logiciel Mission Editor 2.0
* **Environment**: En execution.
* **Réponse** : Possibilité de sélectionner le champ suivant en utilisant le contrôle de tabulation. Les champs sont placés selon l’ordre dans lesquels ils doivent être remplis.
* **Mesure** : Prend au maximum 5 clics de souris pour ajouter une zone d’intérêt au bassin.

**Convivialité**

* **Justification** : Un nouvel utilisateur de l’éditeur de mission doit pouvoir rapidement être en mesure de l’utiliser.
* **Scénario complet** : Un nouvel utilisateur essais de créer une nouvelle mission à partir du « Mission Editor 2.0.
* **Source** : Nouvel utilisateur.
* **Stimulus** : Volonté de créer une mission
* **Artefact** : Mission Editor 2.0
* **Environnement** : En execution.
* **Réponse** : Des valeurs par défauts sont fournis pour les champs où c’est possible.
* **Mesure** : La création d’une mission prend au maximum 15 minutes.

**Disponibilité**

* **Justification** : On doit être avertie si le temps global ne correspond pas à la somme de tous les temps alloués.
* **Scénario complet** : Un utilisateur change le temps alloués à une sous-mission. Le total des temps alloués n’est plus égal au temps global.
* **Source** : Un utilisateur
* **Stimulus** : La somme de tous les temps alloués n’est plus égale au temps global.
* **Artéfact** : Mission editor 2.0
* **Environnement** : Fonctionnement normal
* **Réponse** : Une alerte est affichée à l’écran
* **Mesure** : Erreur détectée après un maximum de 1 seconde.

**Performance**

* **Scénario complet** : Un utilisateur du système fait pivoter un bassin. La boussole doit être mise à jour en temps réel.
* **Source** : Un utilisateur Mission Editor 2.0.
* **Stimulus** : Pivoter un bassin
* **Artefact** : Le système Mission Editor 2.0.
* **Environnement** : En mode normal ou surchargé.
* **Réponse** : Mise à jour de la bousolle
* **Mesure** : Temps de latence maximal de 100 millisecondes.

**Convivialité**

* **Scénario complet** : Un utilisateur veut valider la configuration d’une mission pour vérifier sa conformité
* **Source** : Un utilisateur
* **Stimulus** : Un utilisateur clique sur le bouton de validation de mission
* **Artefact** : Mission Editor 2.0
* **Environnement** : En execution.
* **Réponse** : Message indiquant si la mission est conforme ou non.
* **Mesure** : La satisfaction de l’utilisateur face à la validation effectué.

**Modification**

* **Scénario complet** : Un utilisateur veut modifier le code source de l’état d’une sous-mission. Ce code ne doit pas avoir d’impact sur autre chose que l’état modifié.
* **Source** : Un utilisateur
* **Stimulus** : Modification du code source de l’état d’une sous-mission.
* **Artefact** : Mission Editor 2.0
* **Environnement** : Normal ou surchargé
* **Réponse** : La modification ne doit pas avoir d’impact sur autre chose que l’état modifié. Le code doit être modulaire.
* **Mesure** : Aucun impact sur les autres fonctions du système.

**Performance**

* **Scénario complet** : Un utilisateur veut rapidement trouver un état en utilisant la barre de recherche.
* **Source** : Un utilisateur Mission Editor 2.0.
* **Stimulus** : Rechercher un état
* **Artefact** : Le système Mission Editor 2.0.
* **Environnement** : En mode normal ou surchargé.
* **Réponse** : Afficher le résultat de la recherche
* **Mesure** : Temps pris pour la recherche maximal est 1 seconde.

**Convivialité**

* **Scénario complet** : Un utilisateur cherche un état sans connaitre exactement le nom de l’étiquette ou de la classe.
* **Source** : Un utilisateur Mission Editor 2.0.
* **Stimulus** : Entre du texte dans la barre de recherche
* **Artefact** : Le système Mission Editor 2.0.
* **Environnement** : En mode normal ou surchargé.
* **Réponse** : Afficher les 5 premiers états qui commencent par les lettres entrés dans la barre de recherche.
* **Mesure** : Satisfaction de l’usager

**Convivialité**

* **Scénario complet** : Un utilisateur veut voir les zones d’intérêts des bassins quand il est en mode « édition de mission ».
* **Source** : Un utilisateur du Mission Editor 2.0.
* **Stimulus** : Modifie une mission
* **Artefact** : Le système Mission Editor 2.0
* **Environnement** : En mode normal ou surchargé.
* **Réponse** : Affiche les zones d’intérêts de chaque bassin associé à la mission.
* **Mesure** : Temps gagné par l’usager