

## Knapsack, détails d'implémentation du sprint 1, acheter des items

Date début : semaine du 10 mars. Date remise et de revue: Le 31 mars

Liste des fonctionnalités : Voici la liste des fonctionnalités attendues pour le sprint 1. (ce sont les fonctionnalités et non des user-stories. Pour les US, referez-vous à votre Backlog de sprint.

**Vous devez modifier votre backlog de sprint 1 pour qu'il reflète les fonctionnalités ci-dessous.**

Énoncé des fonctionnalités	Priorité
Afficher tous les items avec les tris demandés	M
Rechercher des items selon un critère avec les tris demandés	M
Voir le détail d'un item selon son type.	M
Inscription d'un joueur et connexion de celui-ci	M
Ajouter un item au panier	M
Supprimer un item du panier	M
Modifier la quantité d'un item dans panier	M
Consulter son panier	M
Payer son panier <ul style="list-style-type: none"> <li>Mise à jour du solde du joueur</li> <li>Calculer le poids du sac à dos et mise à jour de la dextérité du joueur.</li> <li>Mise à jour de l'inventaire du joueur</li> <li>Mise à jour de la dextérité s'il y'a lieu.</li> <li>Mise à jour de la quantité en inventaire. (Items)</li> </ul>	M
Consulter le sac à dos du joueur	M
Rechercher des items selon plusieurs critères (critères combinés)	S

- Le prix d'un item est de l'ordre de 1 à 100 caps . Vous n'avez pas besoin d'une contrainte CHECK. C'est à titre indicatif seulement.
- Le montant initial d'un joueur est : 1000 caps
- Le poids maximal à transporter sans perdre de la dextérité est 50 livres.
- Le poids des Items est entre 1 et 10 livres. (à titre d'indication seulement)
- La dextérité initiale et maximale est de 100.
- Pour chaque livre dépassée le joueur perd un point de dextérité.
- Initialement le nombre de points de vie d'un joueur est 10. Vous pouvez fixer un maximum à atteindre.

Stories techniques pour le sprint 1	Détails
Vérifier la connexion au serveur PHP. Les infos de connexions seront fournies d'ici peu.	Chaque membre de l'équipe doit faire le test qu'il peut bien transférer une page sur le serveur
Créer et peupler la base de données	Vous devez vérifier que votre modèle de données répond, puis : <ol style="list-style-type: none"> <li>Créer les tables de la base de données à partir de votre modèle</li> <li>Ajouter toutes les contraintes check</li> </ol>

	<ol style="list-style-type: none"> <li>3. Créer les deux triggers s'il y'a lieu</li> <li>4. Créer les procédures stockées pour insérer des items. Ici les procédures stockées sont obligatoires pour garantir la cohérence des données.</li> <li>5. Appel des procédures pour faire les insertions. Au moins 5 items de chaque type.</li> </ol>
Configurer GitHub	<p>Chaque membre de l'équipe doit avoir un compte GitHub</p> <p>Un membre de l'équipe sera chargé de créer le dépôt dans son propre compte GitHub et doit inviter les membres de son équipe.</p>

**Les tests :** À titre d'indication, voici un ensemble non exhaustif de tests qui seront réalisés pour Knapsalk --- sprint1.

Tests le compte du joueur
<ul style="list-style-type: none"> <li>- Validation de tous les champs.</li> <li>- L'alias est unique.</li> <li>- Le courriel est unique s'il y a lieu</li> <li>- Afficher les cas d'erreur (alias dupliqué, courriel dupliqué etc..)</li> <li>- Connexion réussie : afficher les infos pertinentes du joueur. (alias, caps, dextérité, point de vie)</li> <li>- Connexion non réussie : afficher le message erreur</li> </ul>
Tests , rechercher d'item
<ul style="list-style-type: none"> <li>- Aucun critère</li> <li>- Tous les critères (un à la fois)</li> <li>- Recherche selon un critère : Exemple Nourriture. Tous les critères seront testés</li> <li>- Recherche selon deux critères. Exemple Armure et Arme. Toutes les combinaisons seront testées</li> <li>- Recherche selon trois critères. Toutes les combinaisons seront testées</li> </ul> <p><b>Pour les points précédents : La photo, le poids, le prix et la quantité en inventaire sont affichés.</b></p> <ul style="list-style-type: none"> <li>- Afficher les détails pertinents pour un item</li> <li>- Trier le résultat de la recherche.</li> <li>- Interface conviviale et intuitive. Facilité de navigation.</li> <li>- Interface adaptative.</li> </ul>
Tests, le panier
<ul style="list-style-type: none"> <li>- Ajouter un item au panier : Quantité disponible: <ul style="list-style-type: none"> <li>o Le montant s'affiche.</li> <li>o Le montant total du panier s'affiche.</li> <li>o Le poids total du panier s'affiche.</li> </ul> </li> <li>- Modifier la quantité d'un item : Quantité disponible : <ul style="list-style-type: none"> <li>o Le montant total du panier est mis à jour.</li> <li>o Le poids total du panier est mis à jour.</li> </ul> </li> <li>- Supprimer un item du panier : <ul style="list-style-type: none"> <li>o Le montant total du panier se met à jour</li> <li>o Le poids total du panier est mis à jour.</li> </ul> </li> </ul>

- Ajouter un item au panier, quantité non disponible
- Modifier la quantité du panier, quantité non disponible
- Payer son panier : le joueur a le montant.
  - o Le nombre caps du joueur est mis à jour
  - o La dextérité du joueur est mise à jour si le poids permis du sac à dos est dépassé. Pour chaque livre dépassée le joueur perd un point de dextérité.
  - o Le sac à dos du joueur est mis à jour
  - o Le magasin (la boutique) ou l'item est mis à jour.
  - o Le panier est vidé.
- Payer son panier : le joueur n'a pas le montant : indiquer que le joueur n'a pas le solde pour payer.
- Acheter sans se connecter. --> erreur

#### Tests le sac à dos du joueur

- Aucun critère : on affiche tous les items.
- (optionnel) afficher selon un critère : Exemple armes. Tous les critères seront testés
- (optionnel) afficher selon des critères combinés