



Symphony

I. Contexte et Scénario Général

Univers et Thématique

Dans un monde fantastique, la **Guilde des Héros** recrute des aventuriers pour accomplir des quêtes épiques. Chaque joueur s'inscrit, reçoit un email de confirmation, se connecte, et peut ensuite créer jusqu'à **3 héros**. Lors de la connexion, s'il possède plusieurs héros, le joueur choisit celui qu'il souhaite incarner pour la session en cours.

Une fois son héros sélectionné, le joueur accède à un **Dashboard** personnalisé dans lequel il peut :

- Voir les informations détaillées de son héros (nom, classe, niveau, expérience, biographie).
- Interagir avec un PNJ emblématique, « **Le Sage du Bosquet** », qui attribue une quête au hasard parmi un ensemble de quêtes disponibles.
- Gérer la quête active en choisissant de l'abandonner ou de la valider.
- **Validation** : Le héros reçoit alors les récompenses associées (gold, points d'expérience, etc.) et les points d'expérience s'ajoutent à son total. Si le total atteint un certain seuil, le héros monte de niveau.
- **Abandon** : Le héros peut renoncer à la quête, qui redevient alors disponible pour d'autres joueurs.

De plus, l'application doit offrir des endpoints API REST (testables via Postman) pour récupérer la liste des quêtes **disponibles** et la liste des héros existants (pour usage administratif ou complémentaire).

II. Objectifs Fonctionnels

Inscription et Connexion

- Le joueur s'inscrit via un formulaire et reçoit un email de confirmation.
- Il se connecte ensuite via un formulaire de login pour accéder à son Dashboard.

Gestion des Héros

- Chaque utilisateur peut créer **jusqu'à 3 héros**.
- Lors de la connexion, si plusieurs héros existent, l'utilisateur choisit lequel incarner pour la session.
- Chaque héros a les attributs suivants :
- **Nom** (ex. "Arthas")
- **Classe** (choix parmi : Druide, Chaman, Guerrier, Voleur, Mage)
- **Niveau** (défini initialement à 1)
- **Expérience** (commence à 0, accumulée par l'achèvement de quêtes)
- **Biographie** (optionnel, texte descriptif)
- *(Optionnel)* **CurrentQuest** : la quête actuellement assignée.

Gestion des Quêtes

- Une entité **Quest** représente une mission à accomplir avec les attributs suivants :
- **Titre** (ex. "La Quête de l'Épée Sacrée")
- **Description** (résumé de la mission)
- **Reward** (récompense en or ou points d'expérience – voir entité Reward)
- **Status** (valeurs possibles : "available", "assigned", "completed")
- **ExperienceGained** (le nombre de points d'expérience attribués lors de la validation de la quête)
- Les quêtes sont préalablement créées dans la base de données (manuellement ou via fixtures).

Gestion des Récompenses

- Une entité **Reward** gère les récompenses, avec les attributs suivants :
- **Montant** (nombre entier indiquant par exemple le nombre d'or ou points bonus)
- **Description** (texte décrivant la nature de la récompense)
- Chaque quête est associée à une récompense via une relation OneToOne.

1. Interaction avec le PNJ – Attribution de Quête

- Le PNJ, nommé « **Le Sage du Bosquet** », distribue une quête aléatoire parmi celles dont le statut est "available".
- Quand un joueur clique sur « Rencontrer le Sage du Bosquet » dans le Dashboard, le système :
- Sélectionne au hasard une quête disponible.
- Change son statut en "assigned".
- L'associe au héros actif (champ `currentQuest` de l'entité Hero).
- Le joueur peut alors choisir de :

Valider la quête : Le système marque la quête comme "completed", attribue la récompense et ajoute les points d'expérience au héros. Si l'expérience accumulée atteint le seuil requis, le niveau du héros augmente.

Abandonner la quête : La quête est désassignée (le statut repasse à "available") et le héros n'obtient pas de récompense ni d'expérience, permettant de tenter une nouvelle attribution.

Dashboard Personnalisé

- Le Dashboard (accessible via `/dashboard` et protégé pour les utilisateurs authentifiés) affiche :
- Le profil du héros sélectionné (nom, classe, niveau, expérience, biographie).
- Les détails de la quête active (titre, description, récompense, points d'expérience attribués).
- Des boutons d'action permettant de **valider** ou **abandonner** la quête.
- Un lien pour rencontrer à nouveau le PNJ pour recevoir une nouvelle quête si aucune quête n'est active ou si la quête en cours a été validée/abandonnée.

Endpoints API (via Postman) :

- GET `/api/quests/available` : renvoie la liste des quêtes dont le statut est "available" en JSON.
- GET `/api/heroes` : renvoie la liste de tous les héros créés en JSON (pour usage administratif ou test).

Bonus – Email de Confirmation

- Lors de l'inscription, après la création du compte utilisateur, un email de confirmation est envoyé à l'adresse fournie.
- L'email comporte un message de bienvenue et, éventuellement, un lien d'activation (si implémenté).

III. Classes et Entités à Créer (Détails des Attributs et Relations)

1. Entité User

Attributs :

- `id` (integer, auto-incrémenté)
- `email` (string, unique)
- `password` (string, haché)
- `roles` (json)

Relation :

- Un utilisateur peut avoir **jusqu'à 3 héros** (relation OneToMany vers Hero).

2. Entité Hero

Attributs :

- `id` (integer, auto-incrémenté)
- `name` (string, 255) – Nom du héros.
- `class` (string, 100) – La classe du héros (Druide, Chaman, Guerrier, Voleur, Mage).
- `level` (integer) – Niveau, initialisé à 1.
- `experience` (integer) – Points d'expérience accumulés, initialisé à 0.
- `bio` (text, nullable) – Description du héros.

Relations :

- **User** : Relation ManyToOne (chaque héros appartient à un utilisateur) avec contrainte de maximum 3 héros par utilisateur.
- **CurrentQuest** : Relation ManyToOne vers Quest (optionnelle) – représente la quête active du héros.

3. Entité Quest

Attributs :

- `id` (integer, auto-incrémenté)
- `title` (string, 255) – Titre de la quête.
- `description` (text, nullable) – Détails de la mission.
- `reward` (integer) – Récompense en or ou points.
- `status` (string, 50) – Statut, par défaut "available" (autres valeurs : "assigned", "completed").
- `experienceGained` (integer) – Points d'expérience attribués lors de la validation de la quête.

Relation :

- **Reward** : Relation OneToOne vers l'entité Reward.

4. Entité Reward

Attributs :

- `id` (integer, auto-incrémenté)
- `amount` (integer) – Montant de la récompense (en or, par exemple).
- `description` (string ou text) – Description de la récompense.

IV. Vues et Interfaces Utilisateur (Pages et Templates)

Page d'Inscription (Register)

- **URL** : `/register`
- **Contenu** :

- Formulaire d'inscription (champs : email, mot de passe, confirmation de mot de passe).
- Message indiquant qu'un email de confirmation a été envoyé.
- **But** : Permettre à un nouveau joueur de créer un compte.

Page de Connexion (Login)

- **URL** : `/login`
- **Contenu** :
- Formulaire de connexion (champs : email, mot de passe).
- **But** : Authentifier l'utilisateur et accéder au Dashboard.

Page de Sélection des Héros

- **URL** : `/dashboard/select-hero`
- **Contenu** :
- Liste des héros créés par l'utilisateur (affichage du nom, classe, niveau).
- Option pour sélectionner un héros actif pour la session.
- **But** : Permettre à l'utilisateur de choisir quel héros incarner parmi les 3 maximum.

Page de Création de Héros

- **URL** : `/dashboard/create-hero`
- **Contenu** :
- Formulaire de création de héros avec les champs : nom, choix de classe (liste déroulante : Druide, Chaman, Guerrier, Voleur, Mage), et optionnellement une biographie.
- **Règle** : Un utilisateur ne peut pas créer plus de 3 héros.
- **But** : Créer un nouveau héros et l'associer au compte utilisateur.

Dashboard Principal

- **URL** : `/dashboard`
- **Contenu** :
- Affichage du profil du héros actuellement sélectionné (nom, classe, niveau, expérience, biographie).

- Affichage de la quête active, si elle existe, avec ses détails : titre, description, récompense (via Reward), et points d'expérience à gagner.
- Boutons d'action pour **valider** ou **abandonner** la quête active.
- Lien pour "Rencontrer le Sage du Bosquet" si aucune quête n'est assignée.
- Option de changer de héros (renvoie vers la page de sélection des héros).

Page d'Interaction avec le PNJ (Le Sage du Bosquet)

- **URL :** `/dashboard/meet-pnj`
- **Contenu :**
- Un message du PNJ (ex. "Je suis Le Sage du Bosquet, je vais vous confier une quête épique !").
- Un bouton "Recevoir une quête".
- **Fonctionnalité :**
- Lorsqu'un joueur clique, le système sélectionne aléatoirement une quête avec le statut "available", change son statut en "assigned" et l'associe au héros actif (champ `currentQuest`).
- Affiche un message de confirmation (ex. "Vous avez reçu la quête : La Quête de l'Épée Sacrée !").

1. Endpoints API REST (pour tests via Postman)

- **Endpoint 1 : Quêtes Disponibles**
- **URL :** `/api/quests/available`
- **Fonction :** Renvoie en JSON la liste des quêtes dont le statut est "available".
- **Endpoint 2 : Liste des Héros**
- **URL :** `/api/heroes`
- **Fonction :** Renvoie en JSON la liste de tous les héros créés (pour usage administratif ou tests).

V. Scénario de Jeu et Flux Utilisateur

Inscription et Confirmation :

- Le joueur accède à `/register`, remplit le formulaire d'inscription et soumet.
- Le système crée le compte utilisateur et envoie un email de confirmation (bonus).
- Le joueur valide son email et se connecte via `/login`.

Création et Sélection du Héros :

- À la première connexion, s'il n'a aucun héros, le joueur est redirigé vers `/dashboard/create-hero`.
- S'il possède déjà 1 à 3 héros, il est invité à choisir le héros qu'il souhaite incarner via `/dashboard/select-hero`.
- Le héros sélectionné devient le héros actif pour la session.

Dashboard et Interaction PNJ :

- Le Dashboard affiche le profil du héros actif et, si aucune quête n'est assignée, propose un lien pour "Rencontrer le Sage du Bosquet".
- Le joueur clique sur ce lien et est redirigé vers `/dashboard/meet-pnj`.
- Le système sélectionne aléatoirement une quête disponible et l'attribue au héros (status de la quête passe à "assigned").
- La quête devient active et est affichée dans le Dashboard.
- Le joueur peut alors choisir de **valider** la quête, ce qui lui confère la récompense (via l'entité Reward) et lui accorde des points d'expérience. Une fois suffisamment d'expérience accumulée, le héros monte de niveau.
- Alternativement, le joueur peut **abandonner** la quête, la réinitialisant à "available", afin de recevoir une autre quête par la suite.

Endpoints API pour Tests :

- Un administrateur ou l'élève peut utiliser Postman pour envoyer une requête GET à `/api/quests/available` et récupérer la liste des quêtes disponibles.

- De même, une requête GET à `/api/heroes` renvoie la liste de tous les héros.

VI. Critères d'Évaluation et Livrables

Livrables attendus :

- **Code complet du projet Symfony** comprenant :
 - Les entités : **User**, **Hero**, **Quest**, **Reward** (avec les colonnes et relations décrites).
 - Les contrôleurs pour :
 - L'inscription et la connexion (`RegistrationController`, `SecurityController`).
 - La gestion des héros (création via `HeroFormType`, sélection via une page dédiée).
 - Le Dashboard (`DashboardController`) qui affiche le profil du héros et la quête active.
 - L'interaction avec le PNJ (méthode pour "Rencontrer le Sage du Bosquet").
 - Les endpoints API (`ApiQuestController`, `ApiHeroController`, si applicable).
 - Les formulaires :
 - **RegistrationFormType** pour l'inscription.
 - **HeroFormType** pour la création du héros.
 - Les templates Twig pour :
 - La page d'inscription (`/register`).
 - La page de connexion (`/login`).
 - La page de création de héros (`/dashboard/create-hero`).
 - La page de sélection des héros (`/dashboard/select-hero`).
 - Le Dashboard (`/dashboard`).
 - La page d'interaction avec le PNJ (`/dashboard/meet-pnj`).
 - **Configuration de sécurité** dans `security.yaml` pour gérer l'authentification (Register, Login) et la protection du Dashboard (accessible uniquement aux utilisateurs authentifiés).
 - **Envoi d'un email de confirmation** lors de la création du compte utilisateur (bonus).
 - **Endpoints API REST** testables via Postman pour récupérer les quêtes disponibles et, éventuellement, la liste des héros.

Critères d'Évaluation :

Fonctionnalité et Intégration :

- Le système d'inscription, avec email de confirmation, doit fonctionner correctement.
- Un utilisateur peut créer jusqu'à 3 héros maximum.
- Lors de la connexion, l'utilisateur doit pouvoir sélectionner le héros qu'il souhaite incarner.
- Le Dashboard doit afficher les informations du héros et la quête active.
- L'interaction avec le PNJ doit attribuer une quête aléatoire et permettre au héros de valider ou d'abandonner la quête pour gagner de l'expérience et monter de niveau.
- Les endpoints API REST doivent renvoyer les données au format JSON.

Cohérence et Originalité de l'Univers Fantasy :

- Les noms et descriptions des héros, quêtes et du PNJ doivent être immersifs et cohérents avec l'univers heroic fantasy.
- L'expérience utilisateur (UX) doit être intuitive et claire.

Qualité et Organisation du Code :

- Respect de l'architecture MVC et de la structure du projet (entités dans `src/Entity`, contrôleurs dans `src/Controller`, templates dans `templates/`).
- Bonne gestion des relations entre entités (User-Hero, Hero-Quest, Quest-Reward).
- Commentaires et clarté dans le code.

Utilisation de Docker :

- Toutes les commandes Symfony doivent être exécutées via `docker compose exec php php bin/console <commande>`.
- L'application doit fonctionner correctement dans l'environnement Docker fourni.

Originalité et Créativité :

- Intégration d'éléments originaux dans l'univers fantasy (noms, descriptions, messages du PNJ, etc.).
- Ajout éventuel d'améliorations (par exemple, animation simple dans le Dashboard ou interface interactive).

VII. Synthèse du Scénario

« **La Guilde des Héros** » est une application web dans un univers heroic fantasy qui doit permettre aux joueurs de :

- **S'inscrire** (avec email de confirmation) et de se connecter.
- **Créer jusqu'à 3 héros** et sélectionner celui qu'ils souhaitent incarner lors de leur connexion.
- **Recevoir des quêtes** d'un PNJ emblématique, **Le Sage du Bosquet**, qui attribue une quête aléatoire parmi un ensemble prédéfini.
- **Gagner de l'expérience** grâce à la validation des quêtes, ce qui permet à leur héros de monter de niveau et d'obtenir des récompenses gérées par une entité Reward.
- **Consulter un Dashboard** qui affiche le profil du héros, la quête active et propose des actions (valider ou abandonner la quête, rencontrer à nouveau le PNJ).
- **Tester l'API REST** via Postman pour récupérer les quêtes disponibles et la liste des héros.