

Projet Java 1			
Binôme:			
Rapport			
Fonctionnalités		Description des fonctionnalités	
Technique		Génération de la JavaDoc Réalisation du diagramme de classes	
Inspection code			
Génie logiciel		Modélisation en classes Mise en place de l'encapsulation Factorisation du code (p.ex. utilisant héritage, classes abstraites) Utilisation du polymorphisme Gestion des exceptions	
Bonnes pratiques		Lisibilité /Commentaires Création de Packages Mise en place de classes de tests Génération d'un Jar exécutable	
Fonctions:			
Prototype de base	S'inscrire / Se connecter	S'enregistrer Saisir ses notes et générer les caractéristiques de départ de l'avatar Créer l'avatar et affecter une liste aléatoire de questions Vérifier login/pwd Saisir de nouvelles notes	
	Lancer un défi	Sélectionner un utilisateur Afficher la liste des questions disponibles Poser une question (QCM) Gérer les points associés aux questions et aux joueurs Ajouter les questions répondues correctement au joueur Gérer le délai de réponse	
Fonctionnalités Avancées	Ajout de nouvelles caractéristiques	Intégration de nouvelles caractéristiques ou règles par rapport aux avatars Prise en compte de la notion de domaine/discipline pour les questions et les avatars Intégrer un niveau de difficulté pour les questions et un système de bonus/malus	
	Possibilité de suivre des tests	Création d'un test Suivre un test et acquérir de nouvelles questions Suivre un test et améliorer les caractéristiques de l'avatar	
	Espace administrateur	Modifier les informations d'un joueur Supprimer un joueur Saisir de nouvelles questions/réponses Permettre au joueur de proposer des questions et les valider ou non	
	Vérifier les notes et améliorer leur intégration	Uploader un bulletin de notes pdf Extraire les notes du fichier pdf Gérer la diversité des intitulés d'UE	
	Ajouter la possibilité de se regrouper	Créer un groupe de joueurs Lancer un défi à un autre groupe	
	Recevoir des notifications	Envoyer automatiquement un mail à un joueur défié Recevoir un mail de notification (p.ex. pour rejoindre un groupe)	
	Intégrer la localisation des joueurs	Faire un moteur de recherche des joueurs basé sur la localisation Afficher une carte des joueurs Ajouter une caractéristique à l'avatar pour limiter la portée de la recherche Localiser des tests avec des bonus spéciaux	
	Afficher des graphiques statistiques /	Afficher les statistiques du joueur Afficher des courbes Afficher les performances de tous les joueurs	
	Introduire la notion de compétences	Ajout de nouvelles capacités (avantage ou pénalités) dans le jeu Ajout de capacités à un joueur Intégration des capacités lors des défis	
Remarques/ Suppléments	IHM design ++ Autres fonctionnalités intégrées :		

A rendre :

- Un rapport décrivant (avec des captures d'écran) les fonctionnalités développées
- Le code source commenté, intégrant des classes de tests pour chaque classe développée
- La documentation technique Javadoc
- Le diagramme des classes au format UML.

Votre application devra a minima faire usage de tous les concepts abordés en classe.