

QUALITY ASSURANCE TESTING
Projet AlgoFlex
ESP MSC2 2022

Numéro ticket	Description	Etapes	Résultats	STATUS
AL-15	Création du pipeline CI/CD	<ul style="list-style-type: none"> - Création du pipeline - Mise en place des github actions - Mise en place de SonarCloud - Ajout déploiement Netlify 	Le projet compile, lance les tests lors d'un push et est mis en production lors de la création d'une nouvelle version	Terminé
AL-16	Front/Back - Créer serveur Node/React	<ul style="list-style-type: none"> -Lancement back & front end -Ping API par ip -Affichage page web standard front end 	Les projets du front et du back doivent être créer et doivent pouvoir compiler avec les dépendances	Terminé
AL-17	Architecture et création de la base de donnée	<ul style="list-style-type: none"> - Document montrant la strucutre de la BDD - Script base de donn�� en.sql 	Cr��ation d'un document montrant la structure de la base de donn��e ainsi que la cr��ation des scripts pour cr��er celle-ci	Termin��
AL-18	Infra PCR	<ul style="list-style-type: none"> - Json envoy���� API - .cpp cr���� - .exe cr���� - retour avec temps d'ex��cution et r��sultat 	<p>Le docker API recoit une requ��te en json.</p> <p>Le Parser parse la requete et cr��er un fichier .cpp avec le code contenu</p> <p>Le Compileur cr��er un ex��cutable �� partir du .cpp</p> <p>Le Runner ex��cute l'ex��cutable et renvoi le temps d'ex��cution ainsi que le r��sultat du programme</p>	Termin��
AL-24	Login/Cr��ation de compte	<ul style="list-style-type: none"> - Cr��ation de compte via firebase - Connexion via firebase - Oublie mot de passe - Gestion utilisateur (photo profil, pseudo etc) - Affichage d'un IDE 	Un utilisateur doit pouvoir cr��er et se connecter �� l'application.	Termin��
AL-26	IDE	<ul style="list-style-type: none"> - Edition du code sur l'IDE - Autocompl��tion sur l'IDE - Envoie du code sur le serveur - Compilation du code sur le serveur - Execution du code sur le serveur - Affichage d'un retour de la compilation - Affichage d'un retour de l'execution 	Un utilisateur doit pouvoir se entrer du code dans une fen��tre web, tout en ayant des suggestion d'autocompl��tion dans le langage choisi	Termin��
AL-28	Compilation et ex��cution �� distance	<ul style="list-style-type: none"> - D��ploiement sur Azure - Ansible playbook - NavBar - Accueil - Profil utilisateur - Affichage de la liste de th��mes - Affichage de la liste d'exercices 	Un utilisateur doit pouvoir envoyer du code sur un serveur, puis recevoir les erreurs de compilation et le r��sultat de l'execution de son code	Termin��
AL-31 to 34	D��veloppement du front-end	<ul style="list-style-type: none"> - D��ploiement Netlify -Fil twitter 	Les diff��rentes pages du sites doivent pouvoir ��tre navigables tout en ayant un design attractif et doivent respecter la charte graphique	Termin��

AL-45	Développement du back-end	<ul style="list-style-type: none"> - API Rest - Déploiement Base de donnée - Mise en place Swagger -Déploiement sur Heroku - Injection entrée - Récupération des sorties - Comparaison avec résultats attendus - Envoi Echec / Réussite 	<p>Une api doit être mise en place pour que les utilisateurs puissent itérer avec les données du site qui leur sont disponible. Une documentation swagger sera disponible</p> <p>Les sousmissions utilisateurs pour un exercice sont validé lors de l'exécution du programme. Ils passent un certain nombre de test (prédéfini) afin de valider l'algorithme soumis par l'utilisateur.</p>	Terminé
AL-67	Validation Exercice (Serveur exécution)			Terminé
AL-13	Interface Jeu Validation	<ul style="list-style-type: none"> - Informe l'utilisateur de manière claire le status de résolution de l'exercice à chaque soumission - Modifie l'apparence de la liste d'exercice en fonction des éléments terminés - Modifie l'apparence de la liste thème en fonction des listes d'exercices terminés 	Un suivi dynamique de l'évolution de l'utilisateur dans la résolution des exercices réussi / commencés / ratés	En cours