

Stratégie de test

Scénarios prévus

- Inscription d'un nouvel utilisateur
- Connexion d'un utilisateur
 - o Via mot de passe simple
 - o Via double authentification
 - o Via authentification biométrique
- Téléversement d'un document
- Accès au tableau de bord
- Accès au suivi de consommation
- Téléchargement des relevés
- Utilisation de conseiller virtuel
- Vérification des appels API

Méthodes de test adaptées

- Tests unitaires des composants par le dev front : Test de l'affichage des composants + Gestion des succès ou erreur api
- Tests de nouvelles fonctionnalités / non régression via cypress ou selenium par le testeur logiciel :
 - o Inscription d'un nouvel utilisateur
 - o Connexion via mot de passe simple
 - o Téléversement d'un document
 - o Utilisation de conseiller virtuel
 - o Tests API
- Tests de nouvelles fonctionnalités / non régression par un fonctionnel ou le testeur logiciel
 - o Connexion via biométrie ou double facteur (nécessite une intervention humaine)
 - o Accès au tableau de bord (avec vérification de la cohérence des données)
 - o Accès au suivi de la consommation (avec vérification de la cohérence des données)
 - o Téléchargement des relevés (avec vérification de la cohérence des données)
- Tests exploratoires (30 minutes)
 - o Se concentrer sur l'inscription d'un nouvel utilisateur et le téléversement des documents
 - o Scénario end to end : Inscription d'un nouvel utilisateur, téléversement d'un

document, déconnexion, connexion avec le compte utilisateur,
téléchargement de son relevé, accès au suivi de consommation et
déconnexion

Ressources nécessaires

Dev front (lors du développement)

Testeur logiciel

Environnement de test dédié (base de données de test)

Jeux de données de tests pour l'appel sur les api externes + un jeu de données par type d'authentification

Excel

IDE + Accès au projet pour développement des tests automatisés

Optionnel : salarié fonctionnel

Étapes clés de la stratégie

1. Analyse des exigences.
Déjà effectuée
2. Stratégie de test.
En cours
3. Conception du cahier de recette.
2 jours afin de spécifier le cahier (en cours de rédaction)
4. Exécution du cahier de recette au niveau des nouvelles fonctionnalités (cela inclut le suivi des anomalies).
4 jours selon abaque + ½ journée pour les tests api (non chiffrable via abaque en attente de la documentation)
+ 2*1/2 journée de suivi des anomalies à programmé sur les sprints 3 et 6
Soit 5 jours et demi en tout
5. Exécution des tests de non régression
Prévoir 1 jour pour les tests de non régression après finalisation du projet et ajout de la nouvelle fonctionnalité par équipe tierce
6. Bilan et PV de campagne
1 jour pour rédaction et organisation d'une réunion bilan

Préconisations

- Les jeux de tests devront être rapidement mis à disposition, et doivent pouvoir

permettre d'appeler des données sur l'api banque de France, qui sera en charge de la récupération de ses données ?

- Les Tests automatisés développés en lien avec les sprints pourront être réutilisés pour des tests futurs de non régression et à maintenir selon les évolutions du produit.
- Les tests api fait sur l'api externe pourront être rapidement mis en place dès que la documentation sera mise à disposition, une exécution régulière +/- automatique pourront permettre une surveillance de la disponibilité des données.
- La criticité des stories est à revoir rapidement ensemble afin d'affiner le planning de tests
- Il sera important d'avoir rapidement les informations sur le projet à venir pour reprendre les scénarios de tests en conséquence et anticiper l'impact sur les développements du projet TOMSEN.