

---

## BEHATC2: Organisation et optimisation des tests Behat

### Informations

---

Durée : 2 jours soit 14 heures  
Code : BEHATC2

Niveau : Perfectionnement  
Public : Développeurs-euses

### Prérequis

---

- Maîtrise du développement Orienté Objet en PHP
- Maîtrise des fondamentaux de Behat
- Maîtrise HTML/CSS et utilisation des outils de développement des navigateurs

### Conditions matérielles

---

Le navigateur Chrome avec une connexion à Internet.

Machine Linux avec Docker installé. La machine Linux doit avoir un accès à Internet pour le téléchargement des images Docker.

### Evaluation des compétences

---

L'évaluation des compétences sera effectuée tout au long de la formation par des cas pratiques qui permettent de valider les blocs de compétence. Une auto-évaluation du stagiaire est effectuée en fin de stage.

### Moyens pédagogiques

---

- Slides de présentation en petit comité (max 6 personnes)
- Ateliers pratiques
- Remise des supports et des exercices corrigés

### Objectifs

---

Cette formation vous apprendra à manipuler l'infrastructure de l'application, organiser vos contextes, aller plus loin avec le concept de PageObject dans le cadre des tests fonctionnels automatisés avec Behat.

## Optimiser les tests Behat

- Organiser le dossier contenant le code des tests fonctionnels.
- Organiser les tests par fonctionnalité grâce au suite et au tags.
- Organiser les contextes.
- Utilisation des crochets pour manipuler l'architecture de test (hooks).

Cas pratique : Réorganiser les contextes et charger les données en base avant chaque scénario

## Optimiser les scénarii

- Utilisation de l'exécution en arrière plan (background)
- Utilisation d'un modèle de scénario
- Utilisation des transformateurs pour améliorer les arguments (transform)

Cas pratique : Optimiser les scénarii

## Aller plus loin grâce à PageObject

- Présentation de PageObject
- Installation de l'extension PageObjectExtension
- Ecrire notre premier service PageObject
- Utilisation de PageObject dans un projet complexe

Cas pratique : Refactoriser les contextes pour utiliser PageObject

## Tester une application legacy

- Comment tester
- Les tests possibles et comment améliorer leur portée

Cas pratique : Mettre en place des tests fonctionnels sur une application tierce

## Évaluation finale