Mais en vrai on test quoi ?

Drupalcamp Lannion 2017
27/10/2017

Le point de départ

Je vais tout retester

Sarah, chef de projet

Et si on se concentrait sur la valeur ajoutée ?

Un projet parfaitement adapté aux tests

- Une base de code enrichie depuis 3 / 4 mois
- Du contenu et des utilisateurs existants (site déjà en prod)
- Des fonctionnalités limitées
- 2 développeurs sur le projet

Tests à disposition

Différents types de tests automatisables

- **Tests unitaires**
- **Tests d’intégration**
- **Tests fonctionnels**
- Tests de montée en charge
- Tests d’accessibilité
- Tests de régressions
- Tests de sécurité
- …

Tests unitaires

- PHPUnit / UnitTestCase
- Tests rapides et autonomes (pas d'initialisation de base de données, peuvent être lancés depuis PHPStorm ❤)
- Exemple : si je charge une permission depuis un YAML, j'ai bien un titre, une description.
- On s'appuie sur des Mockups pour simuler les données d'entrées afin d'augmenter les performances.
- Utile pour tester vos services ou vos classes utilitaires.

Tests d’intégration

- PHPUnit / KernelTestBase
- Permet d’accéder aux fonctions d’API de Drupal
- Se lance à partir d’une base vide (sans contenu) en RAM (peuvent être initialisés à partir d’artefacts)
- Exemple : En chargeant la définition d’une propriété (*base field*) les méthodes `isReadOnly(), isComputed(), isTranslatable()` me renvoient bien les valeurs du schéma.

Tests fonctionnels

- Dépréciées : WebTestBase & KernelTestBase

- BrowserTestBase
- Drupal entièrement installé depuis un profil d’installation
- Très lent mais facile à écrire
- Exemple : Une vue m’affiche seulement les contenus pour lesquels j’ai une permission
- A privilégier pour valider des comportements liés au rendu

Tests fonctionnels

- JavascriptTestBase
- Basé sur Mink + PhantomJS
- Un vrai navigateur web à disposition
- Interactions type drag’n drop
- Sert pour tester le JS du site de prod
- Exemple : Génération du nom machine d’un champ

Tests fonctionnels

- Behat
- Test des fonctionnalités utilisateurs en écrivant des scénarios
- S’écrit en langage naturel
- S’appuie sur une bibliothèque d’instructions
- Pas fait pour tester du code
- Décrit des comportements attendus

Mise en place de behat

Outillage

Behat 3 + Drupal extension

https://www.drupal.org/project/drupalextension

https://github.com/jhedstrom/drupalextension

https://github.com/jhedstrom/DrupalDriver

```
$ composer require drupal/drupal-extension='~3.0'
```

```
$ vendor/bin/behat --init
```

Quoi tester dans mon projet

Périmètre fonctionnel

- Une implémentation personnalisée des permissions (périmètres)
- Des données remontent dans un export (partenaires)
- Les contenus publiés dans une section sont accessibles en lecture et en édition selon les périmètres d’un utilisateur (dashboard)

Fonctionnalité : Attribution des périmètres

En tant qu'administrateur des périmètres, je veux pouvoir permettre à un utilisateur de devenir contributeur ou chef de rédaction en le rattachant à une section afin qu'il puisse ne modifier que les contenus liés à cette section.

Fonctionnalité : Attribution des périmètres

Critères d’acceptation :

- Un utilisateur n’a pas de périmètre assigné par défaut
- Un utilisateur ne peut pas créer de contenu dans une section s’il n’a pas les droits pour ce périmètre
- Un utilisateur peut créer un contenu dans une section s’il possède le périmètre associé

Exemple de scénario Behat

Fonctionnalité : Je teste mes périmètres

Scénario : Un nouvel utilisateur n'a pas de périmètre associé

[Given | **Etant donné / Lorsque]** je suis connecté en tant qu'utilisateur possédant le role "Admin"

[When | Quand] je vais sur "/admin/people/create"

[Then | Alors] je devrais voir "Aucun Périmètre ajouté pour l'instant."

1 à n

Les drivers exposent des commandes

Mink + **Drupal Extension** donnent accès à une bibliothèque d'actions enrichie

```
$ bin/behat -di # Liste les actions disponibles
```

- Je visite la page <chemin>
- Je devrais voir le lien <lien>
- Je ne devrais pas voir le bouton <bouton>
- Le code de réponse HTTP devrait être <code>

- Je suis connecté comme utilisateur avec le rôle <role>
- Je clique sur le lien <lien> de la ligne <ligne>
- Je consulte un contenu de type <type> dont le titre est <titre>
- Je sais la valeur <valeur> pour le champ <champ>

Lancement d’un test behat

```
$ ./behat perimetres.feature
```

Binaire ^ ^ Fonctionnalité à tester

```
@api
Fonctionnalité: Attribution des périmètres
  En tant qu'administrateur des permissions, je veux pouvoir permettre à un utilisateur de
  devenir contributeur, chef de rédaction, le rattacher au marketing d'une section
  particulière afin qu'il puisse ne modifier que les contenus liés à cette section.

Scénario: Un nouvel utilisateur n'a pas de périmètre associé
  Etant donné je suis connecté en tant qu'utilisateur possédant le role "Admin"
  Quand je vais sur "/admin/people/create"
  Alors je devrais voir "Aucun Périmètre ajouté pour l'instant."
```

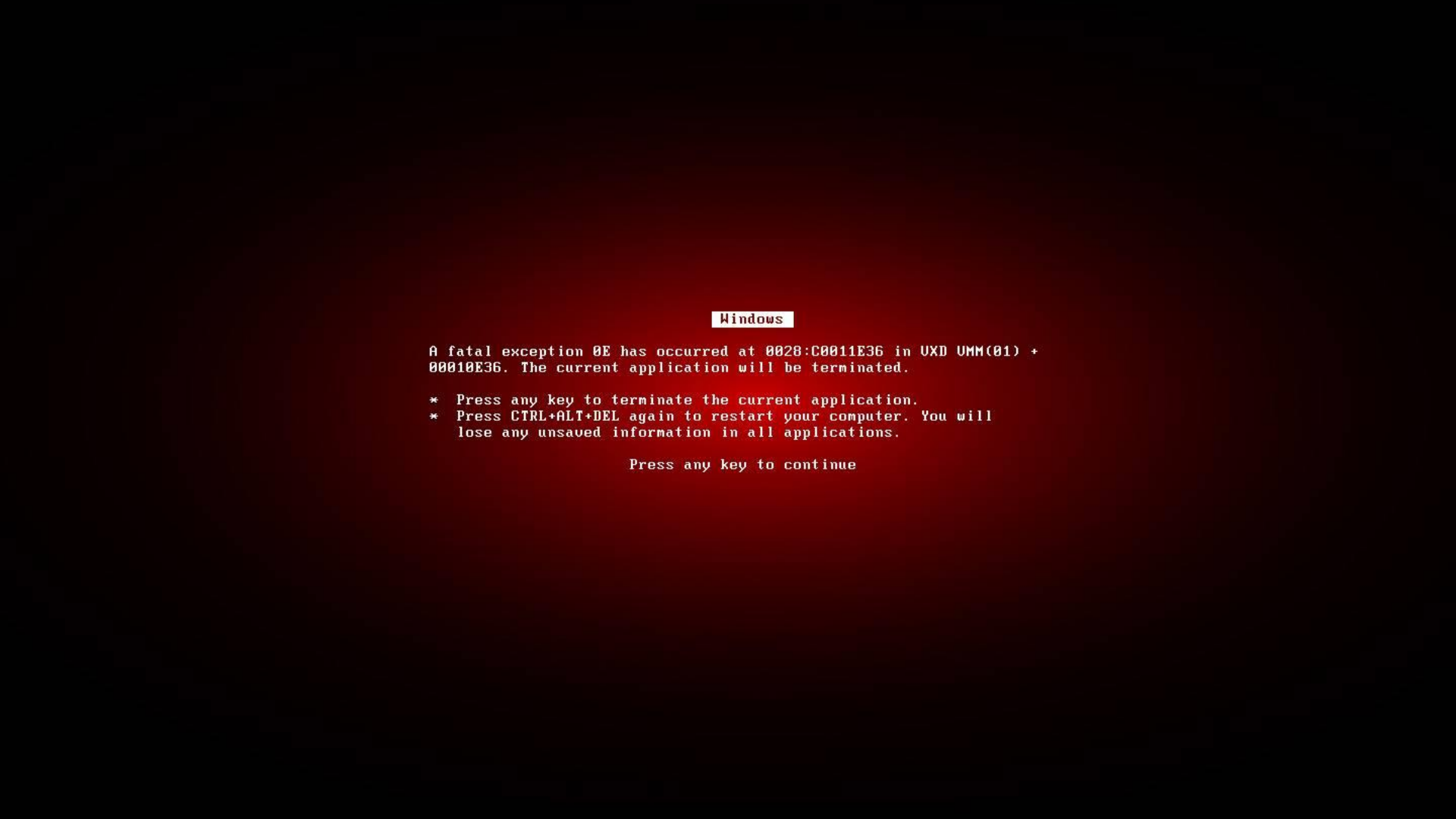

```yaml
default:
  suites:
    default:
      contexts:
        - FeatureContext
        - Drupal\DrupalExtension\Context\DrupalContext
        - Drupal\DrupalExtension\Context\MinkContext
        - Drupal\DrupalExtension\Context\MessageContext
        - Drupal\DrupalExtension\Context\DrushContext
  extensions:
    Behat\MinkExtension:
      goutte: ~
      selenium2: ~
      base_url: http://artusamak.gazette.fr
    Drupal\DrupalExtension:
      blackbox: ~
      api_driver: drupal
      text:
        username_field: Nom d'utilisateur
        password_field: Mot de passe
        log_in: Se connecter
        log_out: Se déconnecter
      drupal:
        drupal_root: "/var/www/gazette/web"
```

Lancement des tests dans l'IC

```
./vendor/bin/behat # Exécutable Behat
  --lang=fr # Instructions en français
  --config=./../behat/behat.localdev.yml # Fichier de conf de l'environnement
  ./../behat # Dossier des features à lancer
  --format=progress # Format le résultat de façon condensée
```

- Lancer tous les tests à chaque build

Et un jour, le saint graal...


Windows
A fatal exception 0E has occurred at 0028:C0011E36 in VXD VMM(01) +
00010E36. The current application will be terminated.
* Press any key to terminate the current application.
* Press CTRL+ALT+DEL again to restart your computer. You will
lose any unsaved information in all applications.
Press any key to continue

Il est possible d'ajouter ses propres commandes

- Une instruction est **une chaîne de caractères** passée dans **une expression régulière** associée à **une méthode d'une classe** à laquelle on peut passer des arguments.

« Je peux créer un dossier dans la section "La gazette Magazine" »

```
/**
 * @Then je peux créer un dossier dans la section :section
 * @Then l'utilisateur :name peut créer un dossier dans la section :section
 */
public function userCanCreateDossierInSection($section, $name = '');
```

```php
public function userCanCreateDossierInSection($section, $name = '') {
  // Connexion si l'utilisateur demandé n'est pas l'utilisateur courant.
  if (!empty($name)) {
    $user = $this->drupalContext->getUserManager()->getUser($name);
    $this->drupalContext->login($user);
  }
  $this->drupalContext->getSession()->visit('/node/add/dossier');
  $this->minkContext->assertResponseStatus(200);

  // Chargement de la section pour obtenir son ID.
  $section = $this->loadSectionFromName($section);

  // Parcours des options disponibles pour le champ sections.
  $selector = 'select[name="field_sections"] option';
  $available_sections = $this->drupalContext->getSession()->getPage()->findAll('css', $selector);
  foreach ($available_sections as $available_section) {
    // On compare l'ID de la section car son nom peut être précédé d'un "-"
    // s'il y a des termes parents dans la liste des sections.
    if ($available_section->getValue() == $section->id()) {
      return TRUE;
    }
  }
  throw new Exception(sprintf("Droits insuffisants pour publier dans la section '%s'", $section->getName()));
}
```


ten

Snippets pour accéder aux contextes

```
/**
 * Gather any contexts needed.
 *
 * @BeforeScenario
 */
public function gatherContexts(BeforeScenarioScope $scope) {
  $environment = $scope->getEnvironment();
  $this->drupalContext = $environment->getContext('Drupal\DrupalExtension\Context\DrupalContext');
  $this->minkContext = $environment->getContext('Drupal\DrupalExtension\Context\MinkContext');
}
-----
$this->minkContext->pressButton('Ajouter Périmètre');
$this->minkContext->getSession()->getPage()->findAll('css', 'select[name$="field_metiers"]');$sites =
\Drupal::entityTypeManager()->getStorage('taxonomy_term')->loadTree('sites', 0, 1);
```

Appliquer la même recette pour tous les critères


```
#language: fr
@api

Fonctionnalité: Attribution des périmètres
En tant qu'administrateur des permissions, je veux pouvoir permettre à un utilisateur de devenir
contributeur, chef de rédaction, le rattacher au marketing d'une section particulière afin qu'il
puisse ne modifier que les contenus liés à cette section.

Scénario: Un nouvel utilisateur n'a pas de périmètre associé
Etant donné je suis connecté en tant qu'utilisateur possédant le role "Admin"
Quand je vais sur "/admin/people/create"
Alors je devrais voir "Aucun Périmètre ajouté pour l'instant."

Scénario: Un utilisateur ne peut pas créer de dossier dans une autre section que celle pour
laquelle il a les droits
Etant donné un utilisateur nommé "behat"
Et je suis connecté en tant qu'utilisateur "behat"
Alors je ne peux pas créer un dossier dans la section "La gazette Magazine"
Et je ne peux pas créer un dossier dans la section "Chrono Sport"
Quand un administrateur m'assigne un périmètre dans le pôle "Pôle Actu" avec le rôle "Chargé
d'édition"
Alors je peux créer un dossier dans la section "La gazette Magazine"
Et je ne peux pas créer un dossier dans la section "Chrono Sport"
```

Pour aller plus loin

Fonctionnalités Behat complémentaires

- « Backgrounds » si des actions préparatoires sont à effectuer avant de lancer des scénarios (contexte partagé qui évite la répétition)
- Chaines multilignes avec une syntaxe particulière
- Tableaux de données
- Boucles avec les « scenario outlines »

Attention à certains points

- Mutualiser les actions
- Créer une librairie qui fait sens
- Driver VS API

```
\Drupal\Driver\Cores\Drupal8::nodeCreate() !== \Drupal\Node\Entity\Node::create()
```

Poursuite et remerciements

- Votre meilleur allié : http://mink.behat.org/en/latest/index.html
- Travail en cours pour migrer tous les tests en PHPUnit
  https://www.drupal.org/node/2735005
  Relativement accessible pour des débutants
- PR de traduction FR https://github.com/jhedstrom/drupalextension/pull/374

- Inspiré de la présentation de Pieter Frensen (CE) présentation DDD Séville.