

# Тест план

## Виды и уровни тестирования

### *Функциональное тестирование*

**На системном уровне** - протестировать операции в File Browser:

- добавление/переименование/удаление каталогов;
- добавление/переименование/удаление файлов;
- добавление данных в существующий файл;
- восстановление файлов и каталогов из корзины;
- ролевая модель, в том числе разграничение прав доступа
- негативные сценарии, в т.ч. проверка валидации.

**На интеграционном уровне** - WebHDFS REST API:

- добавление каталога;
- запрос данных о статусе каталога;
- запрос данных о статусе всех каталогов и файлов в заданном каталоге;
- запрос основной информации по каталогу;
- запрос данных о домашнем каталоге текущего пользователя;
- переименование каталога;
- удаление каталога.
- установление прав доступа на каталог и файл
- установление владельца каталога и файла

### *Нефункциональное тестирование*

- Тестирование производительности (объемное, нагрузочное)
- Тестирование на отказ и восстановление
- Стресс тестирование
- Конфигурационное тестирование

## Риски и ограничения

Интеграционное тестирование для проверки взаимодействия со смежными сервисами в текущем тест плане не учтено.

## Инструментарий

PyCharm (pytest + selenium + requests)

## Автотесты

ID	Тест-кейс	Шаги	Тестовые данные
1_1	Создание каталога (File Browser)	<p><b>hdfs/test/test_add_directory.py</b></p> <p><i>Предварительные действия:</i> удалить папки и файлы в текущей директории.</p> <ol style="list-style-type: none"> <li>1. Получить список элементов в текущем каталоге old_list</li> <li>2. Создать новый каталог</li> <li>3. Получить список элементов в текущем каталоге new_list</li> <li>4. К списку old_list добавить каталог из шага №2</li> <li>5. Сравнить два отсортированных списка с элементами – old_list и new_list.</li> </ol> <p><i>Ожидаемый результат:</i> списки old_list и new_list совпали.</p>	<p>Получать новый набор тестовых данных перед каждым запуском автотеста при помощи генератора данных hdfs/generator/directory.py.</p> <p>Тестовые данные сохраняются в hdfs/data/directory.json.</p>
1_2	Добавление файла (File Browser)	<p><b>hdfs/test/test_add_file.py</b></p> <p><i>Предварительные действия:</i> удалить папки и файлы в текущей директории.</p> <ol style="list-style-type: none"> <li>1. Получить список элементов в текущем каталоге old_list</li> <li>2. Добавить файл</li> <li>3. Получить список элементов в текущем каталоге new_list</li> <li>4. К списку old_list добавить файл из шага №2</li> <li>5. Сравнить два отсортированных списка с элементами – old_list и new_list.</li> </ol> <p><i>Ожидаемый результат:</i> списки old_list и new_list совпали.</p>	<p>Тестовые данные определены в hdfs/data/file.py.</p> <p>Количество файлов – 19.</p> <p>Загружаемые файлы размещены в hdfs/data/files.</p>
1_3	Восстановление файла из корзины (File Browser)	<p><b>hdfs/test/test_restore_from_trash.py</b></p> <p><i>Предварительные действия:</i> удалить папки и файлы в текущей директории.</p> <ol style="list-style-type: none"> <li>1. Получить список элементов в текущем каталоге old_list</li> <li>2. Добавить файл</li> <li>3. Удалить файл</li> <li>4. Перейти в корзину</li> <li>5. Восстановить файл/каталог</li> <li>6. Получить список элементов в текущем каталоге new_list</li> </ol>	

ID	Тест-кейс	Шаги	Тестовые данные
		<p>7. Сравнить количество элементов в списках old_list и new_list.</p> <p><i>Ожидаемый результат:</i> количество элементов в списках old_list и new_list совпало.</p>	
1_4	Создание каталога (REST API)	<p><b>hdfs/test/test_webhdfs_rest_api.py</b></p> <p>HTTP PUT request: http://&lt;HOST&gt;:&lt;PORT&gt;/&lt;PATH&gt;?op=MKDIRS[&amp;permission=&lt;OCTAL&gt;]</p> <p><i>Ожидаемый результат:</i> получен ответ True.</p>	<p>path1 = '/user/cloudera/diri'</p> <p>http://127.0.0.1:50070/webhdfs/v1%s?user.name=hdfs&amp;op=MKDIRS' % path1</p>
1_5	Получение данных о статусе каталога (REST API)	<p><b>hdfs/test/test_webhdfs_rest_api.py</b></p> <p>HTTP GET request: http://&lt;HOST&gt;:&lt;PORT&gt;/webhdfs/v1/&lt;PATH&gt;?op=GETFILESTATUS</p> <p><i>Ожидаемый результат:</i> наличие всех атрибутов 'accessTime', 'blockSize', 'childrenNum', 'fileId', 'group', 'length', 'modificationTime', 'owner', 'pathSuffix', 'permission', 'replication', 'storagePolicy', 'type'.</p>	<p>path1 = '/user/cloudera/diri'</p> <p>http://127.0.0.1:50070/webhdfs/v1%s?user.name=hdfs&amp;op=GETFILESTATUS' % path1</p>
1_6	Получение данных о статусах элементов в заданном каталоге (REST API)	<p><b>hdfs/test/test_webhdfs_rest_api.py</b></p> <p>HTTP GET request: http://&lt;HOST&gt;:&lt;PORT&gt;/webhdfs/v1/&lt;PATH&gt;?op=LISTSTATUS</p> <p><i>Ожидаемый результат:</i> наличие всех атрибутов для каждого из элементов 'accessTime', 'blockSize', 'childrenNum', 'fileId', 'group', 'length', 'modificationTime', 'owner', 'pathSuffix', 'permission', 'replication', 'storagePolicy', 'type'.</p>	<p>path = '/user/cloudera'</p> <p>http://127.0.0.1:50070/webhdfs/v1%s?user.name=hdfs&amp;op=LISTSTATUS' % path</p>
1_7	Получение основных данных заданного каталога (REST API)	<p><b>hdfs/test/test_webhdfs_rest_api.py</b></p> <p>HTTP GET request: http://&lt;HOST&gt;:&lt;PORT&gt;/webhdfs/v1/&lt;PATH&gt;?op=GETCONTENTSUMMARY</p> <p><i>Ожидаемый результат:</i> количество директорий равно двум.</p>	<p>path = '/user/cloudera'</p> <p>http://127.0.0.1:50070/webhdfs/v1%s?user.name=hdfs&amp;op=GETCONTENTSUMMARY' % path</p>
1_8	Определение домашнего каталога текущего пользователя (REST API)	<p><b>hdfs/test/test_webhdfs_rest_api.py</b></p> <p>HTTP GET request: http://&lt;HOST&gt;:&lt;PORT&gt;/webhdfs/v1/?op=GETHOMEDIRECTORY</p> <p><i>Ожидаемый результат:</i> для пользователя hdfs /user/hdfs.</p>	<p>path1 = '/user/cloudera/diri'</p> <p>http://127.0.0.1:50070/webhdfs/v1%s?user.name=hdfs&amp;op=GETHOMEDIRECTORY' % path1</p>

ID	Тест-кейс	Шаги	Тестовые данные
1_9	Переименование каталога (REST API)	<b>hdfs/test/test_webhdfs_rest_api.py</b>  HTTP PUT request: http://<HOST>:<PORT>/webhdfs/v1/<PATH>?op=RENAME&destination=<PATH>  <i>Ожидаемый результат:</i> получен ответ True.	path1 = '/user/cloudera/dir1' path2 = '/user/cloudera/dir2'  http://127.0.0.1:50070/webhdfs/v1%s?user.name=hdfs&op=RENAME&destination=%s' % (path1, path2)
1_10	Удаление каталога (REST API)	<b>hdfs/test/test_webhdfs_rest_api.py</b>  HTTP DELETE request: http://<host>:<port>/webhdfs/v1/<path>?op=DELETE[&recursive=<true false>]  <i>Ожидаемый результат:</i> получен ответ True.	path2 = '/user/cloudera/dir2'  http://127.0.0.1:50070/webhdfs/v1%s?user.name=hdfs&op=DELETE&recursive=true' % path2

# Тест кейсы для ручного тестирования

ID тест-кейса	1_1
Наименование тест-кейса	Создание Directories в File Browser
Задача тест-кейса	Убедиться, что в File Browser создаются Directories
Предварительные действия	Пройдена авторизация в HUE UI под пользователем cloudera

№	Шаг тест-кейса	Тестовые данные	Ожидаемый результат
1	Перейти в File Browser, кликнув на кнопку «Manage HDFS».		Осуществлен переход в File Browser.
2	Кликнуть на кнопку «New», в выпадающем списке выбрать «Directory».		Открылось всплывающее окно «Create Directory» с активным полем Directory name.
3	В поле Directory name ввести наименование, кликнуть на кнопку «Create».	1. Cats 2. Cats_43953 3. Cats! 43953 4. _43953_Cats 43953	В открытом каталоге созданы Directory с наименованием: 1. Cats 2. Cats_43953 3. Cats! 43953 4. _43953_Cats 43953

ID тест-кейса	1_2
Наименование тест-кейса	Создание Directories в File Browser (имя новой Directories совпадает с уже существующим)
Задача тест-кейса	Убедиться, что в File Browser невозможно присвоить имя Directories аналогичное уже присвоенному.
Предварительные действия	Пройдена авторизация в HUE UI под пользователем cloudera; В File Browser созданы 4 Directories из п.3 тест кейс 1_1.

№	Шаг тест-кейса	Тестовые данные	Ожидаемый результат
1	Перейти в File Browser, кликнув на кнопку «Manage HDFS».		Осуществлен переход в File Browser.
2	Кликнуть на кнопку «New», в выпадающем списке выбрать «Directory».		Открылось всплывающее окно «Create Directory» с активным полем Directory name.
3	В поле Directory name ввести наименование, кликнуть на кнопку «Create».	1. Cats 2. Cats_43953 3. Cats! 43953 4. _43953_Cats 43953	Сообщение об ошибке «... already exists», при нажатии на кнопку «Create» создание Directory не происходит.

ID тест-кейса  
Наименование тест-кейса  
Задача тест-кейса  
Предварительные действия

**2\_1**  
**Действия (Actions) с Directories – Rename**  
Убедиться, что в File Browser Directory можно переименовать  
Пройдена авторизация в HUE UI под пользователем cloudera;  
В File Browser созданы 4 Directories из п.3 тест кейс 1\_1.

№	Шаг тест-кейса	Тестовые данные	Ожидаемый результат
1	Перейти в File Browser, кликнув на кнопку «Manage HDFS».		Осуществлен переход в File Browser.
2	Кликнуть на чекбокс слева от Directory.		Кнопка «Actions» стала активной.
3	Кликнуть на кнопку «Actions», в выпадающем списке выбрать «Rename».		Открылось всплывающее окно «Renaming наименование каталога» с активным полем New name.
4	В поле New name ввести наименование, кликнуть на кнопку «Rename».	<ol style="list-style-type: none"><li>1. Cats → Dogs</li><li>2. Cats_43953 → Dogs_43953</li><li>3. Cats! 43953 → Dogs! 43953</li><li>4. _43953_Cats 43953 → _43953_Dogs 43953</li></ol>	В открытом каталоге созданы Directory с наименованием: <ol style="list-style-type: none"><li>1. Dogs</li><li>2. Dogs_43953</li><li>3. Dogs! 43953</li><li>4. _43953_Dogs 43953</li></ol>

ID тест-кейса  
Наименование тест-кейса

Задача тест-кейса

Предварительные действия

**2\_2**  
**Действия (Actions) с Directories – Rename (имя новой Directories совпадает с уже существующим)**  
Убедиться, что в File Browser невозможно присвоить имя Directories аналогичное уже присвоенному (через Rename).  
Пройдена авторизация в HUE UI под пользователем cloudera;  
В File Browser созданы 4 Directories из п.5 тест кейс 2\_1.

№	Шаг тест-кейса	Тестовые данные	Ожидаемый результат
1	Перейти в File Browser, кликнув на кнопку «Manage HDFS».		Осуществлен переход в File Browser.
2	Кликнуть на чекбокс слева от Directory.		Кнопка «Actions» стала активной.
3	Кликнуть на кнопку «Actions», в выпадающем списке выбрать «Rename».		Открылось всплывающее окно «Renaming наименование каталога» с активным полем New name.
4	В поле New name ввести наименование, кликнуть на кнопку «Rename».	<ol style="list-style-type: none"><li>1. Dogs</li><li>2. Dogs_43953</li><li>3. Dogs! 43953</li><li>4. _43953_Dogs 43953</li></ol>	Сообщение об ошибке «... already exists», при нажатии на кнопку «Rename» переименование Directory не происходит.

ID тест-кейса  
Наименование тест-кейса  
Задача тест-кейса  
Предварительные действия

**2\_3**  
**Действия (Actions) с Directories – Move (пустая Directory)**  
Убедиться, что в File Browser Directories можно перемещать в каталоге.  
Пройдена авторизация в HUE UI под пользователем cloudera;  
В File Browser созданы 4 Directories из п.4 тест кейс 2\_1.

№	Шаг тест-кейса	Тестовые данные	Ожидаемый результат
1	Перейти в File Browser, кликнув на кнопку «Manage HDFS».		Осуществлен переход в File Browser.
2	Кликнуть на чекбокс слева от Directory.		Кнопка «Actions» стала активной.
3	Кликнуть на кнопку «Actions», в выпадающем списке выбрать «Move».		Открылось всплывающее окно «Move to» содержащее структуру каталога и активное поле для поиска.
4	В структуре каталога выбрать Directories для переноса.	Dogs Dogs_1990	Directory перенесена в выбранное место в структуре каталога.

ID тест-кейса  
Наименование тест-кейса  
  
Задача тест-кейса  
Предварительные действия

**2\_4**  
**Действия (Actions) с Directories – Move (Directory содержит в себе другую Directory)**  
Убедиться, что в File Browser Directories можно перемещать в каталоге.  
Пройдена авторизация в HUE UI под пользователем cloudera;  
В File Browser созданы 4 Directories из п.4 тест кейс 2\_1.  
В Directory «Gena» содержится Directory «Gena\_1990» (п. 4 тест кейс 2\_3).

№	Шаг тест-кейса	Тестовые данные	Ожидаемый результат
1	Перейти в File Browser, кликнув на кнопку «Manage HDFS».		Осуществлен переход в File Browser.
2	Кликнуть на чекбокс слева от Directory.		Кнопка «Actions» стала активной.
3	Кликнуть на кнопку «Actions», в выпадающем списке выбрать «Move».		Открылось всплывающее окно «Move to» содержащее структуру каталога и активное поле для поиска.
4	В структуре каталога выбрать Directories для переноса.	Dogs! 1990 Dogs Dogs_1990	Directory со всем содержимым перенесена в выбранное место в структуре каталога.