

Тест-кейсы для системы обработки данных о юридических лицах

Выполнил начинающий тестировщик

Дашкин С.М.

Оглавление

Тесты конфигурационного файла obrabotkasobytie.properties	3
Тесты обработки событий	5
Тесты задачи "Получение данных из СХДЮЛ"	8
Тесты изменений в таблице ADDRESS	11
Тесты интерфейса.....	13
Тесты логирования	15
Негативные тесты	16

Тесты конфигурационного файла obrabotkasobytie.properties

ТС_001 - Включение обработки данных

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_001

Название тест-кейса: Проверка включения обработки данных при Put1=true

Шаги теста:

1. Установить в конфигурационном файле параметр Put1=true
2. Отправить запрос на обработку данных клиента ЮЛ
3. Проверить, что запрос обрабатывается системой

Предусловия: Конфигурационный файл obrabotkasobytie.properties доступен для изменения

Ожидаемый результат: Запрос успешно обрабатывается системой

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система готова к обработке следующих запросов

Метки: Positive, Configuration

ТС_002 - Отключение обработки данных

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_002

Название тест-кейса: Проверка отключения обработки данных при Put1=false

Шаги теста:

1. Установить в конфигурационном файле параметр Put1=false
2. Отправить запрос на обработку данных клиента ЮЛ
3. Проверить, что запрос не обрабатывается и не возвращает ошибку

Предусловия: Конфигурационный файл obrabotkasobytie.properties доступен для изменения

Ожидаемый результат: Запрос не обрабатывается, ошибка не возвращается

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система не обрабатывает входящие запросы

Метки: Positive, Configuration

ТС_003 - Настройка префиксов пользователей

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_003

Название тест-кейса: Проверка обработки событий с префиксом QWE

Шаги теста:

1. Установить в конфигурационном файле параметр `userNamePrefix=QWE`
2. Отправить событие с `username`, имеющим префикс QWE
3. Проверить, что событие обрабатывается согласно алгоритму

Предусловия: `Put1=true`, `userNamePrefix` настроен

Ожидаемый результат: Событие с префиксом QWE обрабатывается

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система корректно фильтрует события по префиксу

Метки: Positive, Configuration

ТС_004 - Пустой префикс пользователей

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_004

Название тест-кейса: Проверка поведения при пустом `userNamePrefix`

Шаги теста:

1. Установить в конфигурационном файле параметр `userNamePrefix=` (пустое значение)
2. Отправить событие с `username`, имеющим префикс QWE
3. Проверить поведение системы

Предусловия: `Put1=true`, `userNamePrefix` пуст

Ожидаемый результат: События с префиксом QWE не обрабатываются или обрабатываются в зависимости от реализации

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система работает согласно логике обработки префиксов

Метки: Boundary, Configuration

Тесты обработки событий

TC_005 - Поиск клиента в CROSS_LINKS (найден)

Автор: Дашкин С.М.

Идентификатор тест-кейса: TC_005

Название тест-кейса: Проверка обработки события при наличии клиента в CROSS_LINKS

Шаги теста:

1. Подготовить данные: создать запись в таблице CROSS_LINKS с SYS_ID='testuser' и USER_ID='12345'
2. Отправить событие с username='QWEtestuser'
3. Проверить, что выполняется только ШАГ 4 алгоритма

Предусловия: Префикс QWE настроен, клиент существует в CROSS_LINKS

Ожидаемый результат: userName принимает значение

CROSS_LINKS.USER_ID, выполняется ШАГ 4

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Данные клиента корректно обработаны

Метки: Positive, Event Processing

TC_006 - Поиск клиента в USERS (найден, статус NEW)

Автор: Дашкин С.М.

Идентификатор тест-кейса: TC_006

Название тест-кейса: Проверка обработки события при наличии клиента в USERS со статусом NEW

Шаги теста:

1. Подготовить данные: клиент отсутствует в CROSS_LINKS, но есть в USERS со статусом NEW
2. Отправить событие с username='QWEtestuser'
3. Проверить обновление статуса и создание записи в CROSS_LINKS

Предусловия: Клиент в USERS со статусом NEW, отсутствует в CROSS_LINKS

Ожидаемый результат: userName=USERS.USERS_ID, связанная ссылка удаляется, STATUS=NEW, создается запись в CROSS_LINKS

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Данные обновлены, дальнейшие шаги не выполняются

Метки: Positive, Event Processing

TC_007 - Поиск клиента в USERS (найден, статус UPDATE)

Автор: Дашкин С.М.

Идентификатор тест-кейса: TC_007

Название тест-кейса: Проверка обработки события при наличии клиента в USERS со статусом UPDATE

Шаги теста:

1. Подготовить данные: клиент в USERS со статусом UPDATE
2. Отправить событие с username='QWEtestuser'
3. Проверить изменение статуса на ERROR и установку кода ошибки

Предусловия: Клиент в USERS со статусом UPDATE

Ожидаемый результат: STATUS=ERROR, ERROR_CODE="Отсутствуют необходимые данные", связанная ссылка удаляется, создается запись в CROSS_LINKS

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Статус клиента изменен на ERROR

Метки: Negative, Event Processing

TC_008 - Создание нового клиента

Автор: Дашкин С.М.

Идентификатор тест-кейса: TC_008

Название тест-кейса: Проверка создания нового клиента при его отсутствии в USERS

Шаги теста:

1. Убедиться, что клиент отсутствует в таблицах CROSS_LINKS и USERS
2. Отправить событие с username='QWEtestuser'
3. Проверить создание новой записи в USERS и CROSS_LINKS

Предусловия: Клиент отсутствует в системе

Ожидаемый результат: Создается запись в USERS с USER_ID=userName, STATUS=NEW, создается запись в CROSS_LINKS

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Новый клиент создан в системе

Метки: Positive, Event Processing

TC_009 - Обработка клиента со статусом DEDUPE

Автор: Дашкин С.М.

Идентификатор тест-кейса: TC_009

Название тест-кейса: Проверка обработки клиента со статусом DEDUPE

Шаги теста:

1. Подготовить данные: клиент в USERS со статусом DEDUPE и заполненным USER_ID2
2. Выполнить обработку согласно [ШАГ 4](#)
3. Проверить, что userName принимает значение связанной ссылки

Предусловия: Клиент найден со статусом DEDUPE

Ожидаемый результат: userName принимает значение USERS.USER_ID2

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Используется связанная ссылка клиента

Метки: Positive, Event Processing

TC_010 - Обработка клиента со статусом DELETED

Автор: Дашкин С.М.

Идентификатор тест-кейса: TC_010

Название тест-кейса: Проверка обработки клиента со статусом DELETED

Шаги теста:

1. Подготовить данные: клиент в USERS со статусом DELETED и заполненным USER_ID2
2. Выполнить обработку согласно [ШАГ 4](#)
3. Проверить, что userName принимает значение связанной ссылки

Предусловия: Клиент найден со статусом DELETED

Ожидаемый результат: userName принимает значение USERS.USER_ID2

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Используется связанная ссылка удаленного клиента

Метки: Positive, Event Processing

Тесты задачи "Получение данных из СХДЮЛ"

ТС_011 - Настройка автоматической процедуры

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_011

Название тест-кейса: Проверка включения автоматической процедуры

Шаги теста:

1. Установить параметр action=Включена в конфигурации
2. Проверить запуск автоматической процедуры
3. Убедиться, что процедура выполняется согласно расписанию

Предусловия: Задача настроена в конфигурации

Ожидаемый результат: Автоматическая процедура запускается согласно расписанию

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Автоматическая процедура активна

Метки: Positive, Task Configuration

ТС_012 - Режим загрузки через Kafka

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_012

Название тест-кейса: Проверка загрузки данных через Kafka

Шаги теста:

1. Установить параметр mode=kafka
2. Подготовить JSON данные в очереди Kafka
3. Проверить успешную обработку данных из очереди

Предусловия: Подключение к Kafka настроено, данные в очереди

Ожидаемый результат: Данные успешно получены и обработаны из Kafka

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Данные импортированы в систему

Метки: Positive, Kafka Integration

ТС_013 - Режим загрузки через директорию

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_013

Название тест-кейса: Проверка загрузки данных через директорию

Шаги теста:

1. Установить параметр mode=path
2. Настроить пути директорий (path.directory, path.archive, path.error)
3. Поместить корректный JSON файл в path.directory
4. Проверить обработку файла и его перемещение в path.archive

Предусловия: Директории настроены, JSON файл подготовлен

Ожидаемый результат: Файл обработан и перемещен в архивную директорию

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Файл обработан и архивирован

Метки: Positive, File Processing

ТС_014 - Обработка файла с ошибкой

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_014

Название тест-кейса: Проверка обработки некорректного JSON файла

Шаги теста:

1. Установить параметр mode=path
2. Поместить некорректный JSON файл в path.directory
3. Проверить обработку ошибки и перемещение файла в path.error

Предусловия: Директории настроены, некорректный JSON файл подготовлен

Ожидаемый результат: Файл с ошибкой перемещен в директорию ошибок

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Файл с ошибкой изолирован

Метки: Negative, File Processing

ТС_015 - Проверка ограничения размера файла

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_015

Название тест-кейса: Проверка ограничения максимального размера файла

Шаги теста:

1. Установить параметр path.size=100 (Мб)
2. Поместить файл размером больше 100 Мб в path.directory
3. Проверить обработку файла согласно ограничению

Предусловия: Ограничение размера настроено, большой файл подготовлен

Ожидаемый результат: Файл размером больше лимита не обрабатывается или

обрабатывается с ошибкой

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Ограничение размера соблюдено

Метки: Boundary, File Processing

ТС_016 - Проверка кодировки UTF-8

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_016

Название тест-кейса: Проверка корректной обработки файлов в кодировке UTF-8

Шаги теста:

1. Подготовить JSON файл с русскими символами в кодировке UTF-8
2. Поместить файл в path.directory
3. Проверить корректную обработку специальных символов

Предусловия: Файл в кодировке UTF-8 подготовлен

Ожидаемый результат: Файл обработан корректно, специальные символы не искажены

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Кодировка обработана корректно

Метки: Positive, Encoding

ТС_017 - Проверка настройки расписания

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_017

Название тест-кейса: Проверка выполнения задачи каждые 5 минут

Шаги теста:

1. Установить параметр Expression="Каждые 5 минут"
2. Активировать автоматическую процедуру
3. Проверить выполнение задачи с интервалом 5 минут

Предусловия: Расписание настроено, процедура активна

Ожидаемый результат: Задача выполняется каждые 5 минут

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Расписание работает корректно

Метки: Positive, Scheduling

Тесты изменений в таблице ADDRESS

TC_018 - Сохранение данных адреса

Автор: Дашкин С.М.

Идентификатор тест-кейса: TC_018

Название тест-кейса: Проверка сохранения данных адреса в таблице ADDRESS

Шаги теста:

1. Подготовить JSON с данными адреса (country, regionCode)
2. Обработать JSON через систему
3. Проверить сохранение данных в таблице ADDRESS

Предусловия: Таблица ADDRESS настроена, JSON данные подготовлены

Ожидаемый результат: Данные адреса сохранены в соответствующих полях таблицы

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Данные адреса сохранены в БД

Метки: Positive, Database

TC_019 - Проверка типов данных полей

Автор: Дашкин С.М.

Идентификатор тест-кейса: TC_019

Название тест-кейса: Проверка соответствия типов данных в полях таблицы ADDRESS

Шаги теста:

1. Подготовить данные различных типов для полей COUNTRY, REGION
2. Попытаться сохранить данные превышающие максимальную длину
3. Проверить обработку некорректных типов данных

Предусловия: Таблица ADDRESS настроена

Ожидаемый результат: Данные сохраняются согласно типам полей, превышающие лимит обрезаются или вызывают ошибку

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Валидация типов данных работает корректно

Метки: Boundary, Database

TC_020 - Автоматическое заполнение CHANGE_DATE

Автор: Дашкин С.М.

Идентификатор тест-кейса: TC_020

Название тест-кейса: Проверка автоматического заполнения поля
CHANGE_DATE

Шаги теста:

1. Добавить новую запись в таблицу ADDRESS
2. Проверить автоматическое заполнение поля CHANGE_DATE
3. Убедиться, что время соответствует времени операции

Предусловия: Таблица ADDRESS настроена

Ожидаемый результат: Поле CHANGE_DATE автоматически заполняется
текущим временем

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Время изменения зафиксировано

Метки: Positive, Database

Тесты интерфейса

ТС_021 - Отображение данных в форме "Справочник клиенты ЮЛ"

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_021

Название тест-кейса: Проверка отображения данных о клиентах ЮЛ в интерфейсе

Шаги теста:

1. Импортировать данные о клиенте через систему
2. Открыть форму "Справочник клиенты ЮЛ"
3. Проверить отображение импортированных данных

Предусловия: Данные о клиенте импортированы в систему

Ожидаемый результат: Данные корректно отображаются в форме справочника

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Пользователь может видеть актуальные данные

Метки: Positive, UI

ТС_022 - Отображение блока "Адрес юридического лица"

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_022

Название тест-кейса: Проверка отображения блока данных об адресе

Шаги теста:

1. Импортировать данные с информацией об адресе
2. Открыть форму "Справочник клиенты ЮЛ"
3. Проверить отображение колонок "Страна" и "Регион (номер региона)"

Предусловия: Данные об адресе импортированы

Ожидаемый результат: Блок "Адрес юридического лица" отображается с колонками "Страна" и "Регион"

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Данные адреса доступны пользователю

Метки: Positive, UI

ТС_023 - Скрытие поля CHANGE_DATE

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_023

Название тест-кейса: Проверка скрытия поля CHANGE_DATE в интерфейсе
Шаги теста:

1. Открыть форму "Справочник клиенты ЮЛ"
2. Проверить отсутствие отображения поля CHANGE_DATE
3. Убедиться, что поле не видно пользователю

Предусловия: Форма настроена согласно требованиям

Ожидаемый результат: Поле CHANGE_DATE не отображается в интерфейсе

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Служебная информация скрыта от пользователя

Метки: Positive, UI

Тесты логирования

ТС_024 - Логирование ошибок обработки JSON

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_024

Название тест-кейса: Проверка логирования ошибок в файл sys1.log

Шаги теста:

1. Отправить некорректный JSON для обработки
2. Проверить создание записи об ошибке в файле sys1.log
3. Убедиться, что запись содержит достаточную информацию для диагностики

Предусловия: Система логирования настроена

Ожидаемый результат: Ошибка зафиксирована в файле sys1.log с детальной информацией

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Ошибка задокументирована для анализа

Метки: Positive, Logging

ТС_025 - Формат записи в логе

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_025

Название тест-кейса: Проверка формата записи ошибок в логе

Шаги теста:

1. Вызвать различные типы ошибок при обработке JSON
2. Проверить формат записей в sys1.log
3. Убедиться в читаемости и информативности записей

Предусловия: Система логирования активна

Ожидаемый результат: Записи в логе имеют единый читаемый формат с временными метками

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Лог пригоден для анализа ошибок

Метки: Positive, Logging

Негативные тесты

ТС_026 - Обработка невалидного JSON

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_026

Название тест-кейса: Проверка обработки некорректного JSON

Шаги теста:

1. Отправить JSON с синтаксическими ошибками
2. Проверить обработку ошибки системой
3. Убедиться, что система не падает и логирует ошибку

Предусловия: Система готова к обработке данных

Ожидаемый результат: Ошибка корректно обработана, система продолжает работу

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система стабильна после ошибки

Метки: Negative, Error Handling

ТС_027 - Обработка пустого JSON

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_027

Название тест-кейса: Проверка обработки пустого JSON объекта

Шаги теста:

1. Отправить пустой JSON объект {}
2. Проверить поведение системы
3. Убедиться в корректной обработке

Предусловия: Система готова к обработке данных

Ожидаемый результат: Пустой JSON обработан без критических ошибок

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система обработала граничный случай

Метки: Boundary, Error Handling

ТС_028 - Недоступность внешних систем

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_028

Название тест-кейса: Проверка поведения при недоступности Kafka

Шаги теста:

1. Отключить доступ к Kafka
2. Попытаться обработать данные в режиме mode=kafka
3. Проверить обработку ошибки подключения

Предусловия: Система настроена на работу с Kafka

Ожидаемый результат: Ошибка подключения корректно обработана и залогирована

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система готова к восстановлению подключения

Метки: Negative, Integration

ТС_029 - Недоступность директорий

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_029

Название тест-кейса: Проверка поведения при недоступности рабочих директорий

Шаги теста:

1. Удалить или заблокировать доступ к path.directory
2. Попытаться обработать файлы в режиме mode=path
3. Проверить обработку ошибки доступа к директории

Предусловия: Система настроена на работу с директориями

Ожидаемый результат: Ошибка доступа к директории корректно обработана и залогирована

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система готова к восстановлению доступа

Метки: Negative, File System

ТС_030 - Переполнение базы данных

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_030

Название тест-кейса: Проверка поведения при переполнении таблиц БД

Шаги теста:

1. Максимально заполнить таблицы USERS или CROSS_LINKS
2. Попытаться добавить новые записи
3. Проверить обработку ошибки переполнения

Предусловия: Таблицы БД близки к максимальному заполнению
Ожидаемый результат: Ошибка переполнения корректно обработана
Фактический результат: *Заполняется при/после тестирования*
Статус: *Заполняется при/после/до тестирования*
Постусловия: Система стабильна при ограничениях БД
Метки: Negative, Database

ТС_031 - Длительная обработка данных

Автор: Дашкин С.М.
Идентификатор тест-кейса: ТС_031
Название тест-кейса: Проверка обработки большого объема данных
Шаги теста:

1. Подготовить большой JSON файл (близкий к лимиту 100 Мб)
2. Загрузить файл для обработки
3. Проверить стабильность системы и время обработки

Предусловия: Подготовлен большой корректный JSON файл
Ожидаемый результат: Файл обработан успешно в разумное время
Фактический результат: *Заполняется при/после тестирования*
Статус: *Заполняется при/после/до тестирования*
Постусловия: Система обработала большой объем данных
Метки: Performance, Stress Testing

ТС_032 - Конкурентная обработка

Автор: Дашкин С.М.
Идентификатор тест-кейса: ТС_032
Название тест-кейса: Проверка одновременной обработки нескольких файлов
Шаги теста:

1. Поместить несколько JSON файлов одновременно в path.directory
2. Проверить корректную обработку всех файлов
3. Убедиться в отсутствии конфликтов

Предусловия: Несколько корректных JSON файлов подготовлены
Ожидаемый результат: Все файлы обработаны корректно без конфликтов
Фактический результат: *Заполняется при/после тестирования*
Статус: *Заполняется при/после/до тестирования*
Постусловия: Система обработала конкурентные запросы
Метки: Positive, Concurrency

ТС_033 - Обработка специальных символов

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_033

Название тест-кейса: Проверка обработки специальных символов в username

Шаги теста:

1. Отправить событие с username, содержащим специальные символы
2. Проверить корректную обработку в таблицах БД
3. Убедиться в отсутствии SQL-инъекций

Предусловия: Система готова к обработке событий

Ожидаемый результат: Специальные символы корректно обработаны и экранированы

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система защищена от специальных символов

Метки: Security, Input Validation

ТС_034 - Восстановление после сбоя

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_034

Название тест-кейса: Проверка восстановления обработки после сбоя системы

Шаги теста:

1. Остановить систему во время обработки файла
2. Перезапустить систему
3. Проверить корректное восстановление и обработку оставшихся файлов

Предусловия: Система в процессе обработки файлов

Ожидаемый результат: После перезапуска система корректно продолжает работу

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система восстановила работоспособность

Метки: Recovery, Reliability

ТС_035 - Валидация JSON структуры

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_035

Название тест-кейса: Проверка валидации структуры JSON согласно схеме

Шаги теста:

1. Подготовить JSON с корректной структурой, но неожиданными полями
2. Отправить JSON для обработки
3. Проверить поведение системы при неизвестных полях

Предусловия: Определена ожидаемая структура JSON

Ожидаемый результат: Неизвестные поля игнорируются или вызывают предупреждение

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Система обработала JSON с дополнительными полями

Метки: Boundary, Validation

ТС_036 - Тестирование регистрозависимости

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_036

Название тест-кейса: Проверка обработки префиксов с разным регистром

Шаги теста:

1. Отправить события с username, имеющим префикс в разных регистрах (QWE, qwe, Qwe)
2. Проверить обработку согласно настройкам userNamePrefix
3. Убедиться в корректной обработке регистра

Предусловия: Настроен префикс QWE

Ожидаемый результат: Система корректно обрабатывает регистр префиксов

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Обработка регистра работает согласно спецификации

Метки: Boundary, Case Sensitivity

ТС_037 - Проверка транзакционности

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_037

Название тест-кейса: Проверка целостности данных при ошибке обработки

Шаги теста:

1. Инициировать обработку события, которое должно изменить несколько таблиц
2. Вызвать ошибку в середине обработки
3. Проверить откат всех изменений

Предусловия: Система поддерживает транзакции
Ожидаемый результат: При ошибке все изменения откатываются
Фактический результат: *Заполняется при/после тестирования*
Статус: *Заполняется при/после/до тестирования*
Постусловия: Целостность данных сохранена
Метки: Negative, Data Integrity

ТС_038 - Мониторинг производительности

Автор: Дашкин С.М.
Идентификатор тест-кейса: ТС_038
Название тест-кейса: Проверка производительности обработки данных
Шаги теста:

1. Подготовить набор тестовых данных известного размера
2. Измерить время обработки
3. Проверить соответствие производительности требованиям

Предусловия: Тестовые данные подготовлены, система под нагрузкой
Ожидаемый результат: Производительность соответствует требованиям
Фактический результат: *Заполняется при/после тестирования*
Статус: *Заполняется при/после/до тестирования*
Постусловия: Производительность измерена и задокументирована
Метки: Performance, Load Testing

ТС_039 - Проверка безопасности доступа

Автор: Дашкин С.М.
Идентификатор тест-кейса: ТС_039
Название тест-кейса: Проверка ограничений доступа к конфигурационным файлам
Шаги теста:

1. Попытаться получить доступ к obrabotkasobytie.properties от имени обычного пользователя
2. Проверить права доступа к директориям обработки
3. Убедиться в корректной настройке безопасности

Предусловия: Система развернута в рабочей среде
Ожидаемый результат: Доступ к критичным файлам ограничен
Фактический результат: *Заполняется при/после тестирования*
Статус: *Заполняется при/после/до тестирования*
Постусловия: Безопасность доступа проверена
Метки: Security, Access Control

ТС_040 - Тестирование обновления конфигурации

Автор: Дашкин С.М.

Идентификатор тест-кейса: ТС_040

Название тест-кейса: Проверка применения изменений конфигурации без перезапуска

Шаги теста:

1. Изменить параметры в obrabotkasobytie.properties
2. Проверить применение изменений без перезапуска системы
3. Убедиться в корректной работе с новыми параметрами

Предусловия: Система работает с начальной конфигурацией

Ожидаемый результат: Изменения конфигурации применены без перезапуска

Фактический результат: *Заполняется при/после тестирования*

Статус: *Заполняется при/после/до тестирования*

Постусловия: Конфигурация обновлена в рабочей системе

Метки: Configuration, Hot Reload

* Шаг 4. Если найденный клиент имеет статус DEDUPE (USERS.STATUS= DEDUPE) или DELETED (т.е. USERS.STATUS= DELETED), то userName = (USERS.USER_ID2).*