

Описание

Модуль работы с утилитами

Назначение

Блок утилит позволяет легко опубликовать любую бизнес-логику в виде сервиса, с описанием входных и выходных параметров

Общие настройки

Для публикации класса в качестве утилиты, необходимо пометить его интерфейсом `mp.jprime.utils.JPUtil` и аннотацией `@JPUtilLink` с указанием урла вызова, прав доступа и списка метаклассов, для которых актуальная утилита

```
@JPUtilLink(  
    authRoles = "AUTH_ACCESS",  
    code = "test/testutil",  
    jpClasses = "testClass",  
    title = "Тестовая утилита"  
)
```

Выполняющий метод (или методы утилиты в случае многошаговости) помечаются аннотацией `@JPUtilModelink`

```
@JPUtilModelink(  
    code = "check",  
    title = "Проверка",  
    outClass = Out.class  
)  
public Mono<Out> check(JPUtilCheckInParams in, AuthInfo authInfo) {  
    check(in.getId(), authInfo);  
    return new Out();  
}
```

с указанием кода выполняемого шага

Настройки JPUtilLink

Код	Описание
jpClasses	Список классов, для которых доступна утилита
authRoles	Роли, имеющиеся доступ к этой утилите
title	Название утилиты
qName	QName утилиты
code	Уникальный код утилиты

Код	Описание
Настройки	JPUtilModelLink

Код	Описание
authRoles	Роли, имеющиеся доступ к этому шагу
title	Название шага
qName	QName шага
code	Уникальный код утилиты
type	Тип доступности шага (список/объект/произвольно)
jpAttrs	Настройки доступа на атрибутах
confirm	Сообщение перед запуском шага
inParams	Список входящих параметров
outClass	Выходной класс параметров
outCustomParams	Список кастомных исходящих параметров

Описание методов

Обязательно наличие бинов с входными и выходными параметрами, корректно преобразующихся из/в JSON

Кроме того, опционально в методе можно описать

- AuthInfo - Данные авторизации `mp.jprime.security.AuthInfo`
- ServerWebExchange - Данные http запроса `org.springframework.web.server.ServerWebExchange`

Метод check

С помощью этого метода можно управлять предварительной проверкой возможности запуска утилиты. Для запрета запуска: * метод должен бросить любое исключение (не рекомендуется) * вернуть `JPUtilCheckOutParams` с `denied == true` (рекомендовано)

Если в утилите нет метода, помеченного `code = "check"`, то по умолчанию код с `mode/check` отработает с 200 кодом и результатом `denied: false`

Входные параметры

Класс входящих параметров представляет собой реализацию `mp.jprime.utils.JPUtilInParams`, обычно наследника от `mp.jprime.utils.BaseJPUtilInParams`, и может содержать произвольные параметры. Для корректной публикации и работы утилиты в реалтайме все входящие параметры необходимо описать в интерфейсе `@JPUtilModelLink.inParams`

Особые параметры

`objectClassCode` и `objectIds` заполняются данными объектов, для которых вызывается утилита

`rootObjectClassCode` и `rootObjectId` заполняются данными корневого объекта

- `rootObjectClassCode` - кодовое имя метакласса корневого объекта
- `rootObjectId` - идентификатор корневого объекта
- `objectClassCode` - кодовое имя метакласса объекта/ов
- `objectIds` - идентификатор или идентификаторы объектов через запятую, в случае списка

Например, при вызове утилиты по списку документов личного дела, `root*` параметры будут заполнены идентификатором и кодом ЛД, а `object*` параметры идентификаторами и кодовым именем документов

Выходные параметры

Класс входящих параметров представляет собой реализацию `mp.jprime.utils.JPUtilOutParams` и может содержать произвольные параметры. Для корректной публикации и работы утилиты в реалтайме все исходящие параметры необходимо описать в интерфейсе `@JPUtilModelLink.outParams`

Типовые реализации `mp.jprime.utils.JPUtilOutParams`

Код	Название	Код resultType	Описание
JPUtilCheckOutParams		check	результат check-проверки утилиты
JPUtilMessageOutParams		message	результатом является только строка-описание
JPUtilJPIdOutParams		jpId	результатом является JPId объекта
JPUtilJPObjectOutParams		jpObject	результатом является объект JPObject
JPUtilJPObjectListOutParams		jpObjectList	результатом является список объектов JPObject
JPUtilCustomOutParams		custom	результатом является произвольный набор данных
JPUtilVoidOutParams		void	отсутствие результата

Привязка утилиты

Свойство	Описание
object	Обработка 1 объекта
list	Обработка списка объектов
uni	Обработка объекта и списка объектов
custom	Произвольная

Настройка доступа

Запуск шага утилиты разрешен ролям, указанным в `authRoles` метода или же в `authRoles` утилиты, если настройки метода пусты. При пустых `authRoles` в обеих аннотациях запуск разрешен всем

Использование утилит

Использование в прикладном коде

Получение утилиты по ее коду осуществляется методом

```
@Autowired
private IJPUtilService jpUtilService;
...
IJPUtil util = jpUtilService.getUtil("test/testutil");
```

Вызов определенного шага утилиты из кода

```
In in = new In();
in.setValue("test");
Mono<Out> out = jpUtilService.apply("test/testutil", "print", in);
```

Вызов по REST

Детальнее см. `jprime-utils-rest`

```
POST utils/v1/<код утилиты>/mode/<код шага>
```