



Микросервисы системы автоматизации и роботизации исполнения процессов

Выполнил:

**Студент 3 курса механико-математического факультета, группы ПМИ-3-20
Семенов Сергей Павлович**

Научный руководитель:

к.т.н. Селетков Илья Павлович

Введение

- В настоящее время развитие промышленности требует всё большей автоматизации процессов, в том числе процессов управления и обработки информации.
- Согласно исследованию McKinsey [1], в 2020 году автоматизацию проводили 66% опрошенных компаний, что говорит о росте по сравнению с 57% двумя годами до этого.
- С другой стороны в текущих условиях возросли риски, связанные с использованием ПрЭВМ западного производства.
- Представленные на рынке отечественные системы решают, в первую очередь, задачу документооборота, нежели обработки информации.
- В связи с этим существует потребность в разработке инструментов, позволяющих упростить автоматизацию рабочих процессов управления, принятия решений и обработки информации.

Цель работы

Цель

Упростить процедуру автоматизации рабочих и бизнес-процессов и повысить эффективность их исполнения.

Объект исследования

Рабочие и бизнес-процессы, подлежащие автоматизации.

Предмет исследования

Структура и алгоритмы работы **серверной части** системы, позволяющей автоматизировать рабочие процессы.

Постановка задачи

- Разработать сервис для управления рабочими процессами.
- Предложить готовое ПО, которое может быть сервисом-исполнителем программного кода.
- Протестировать сервис путём разработки автоматизированных тестов и/или ручную.
- Оформить техническую документацию на разработанный сервис.

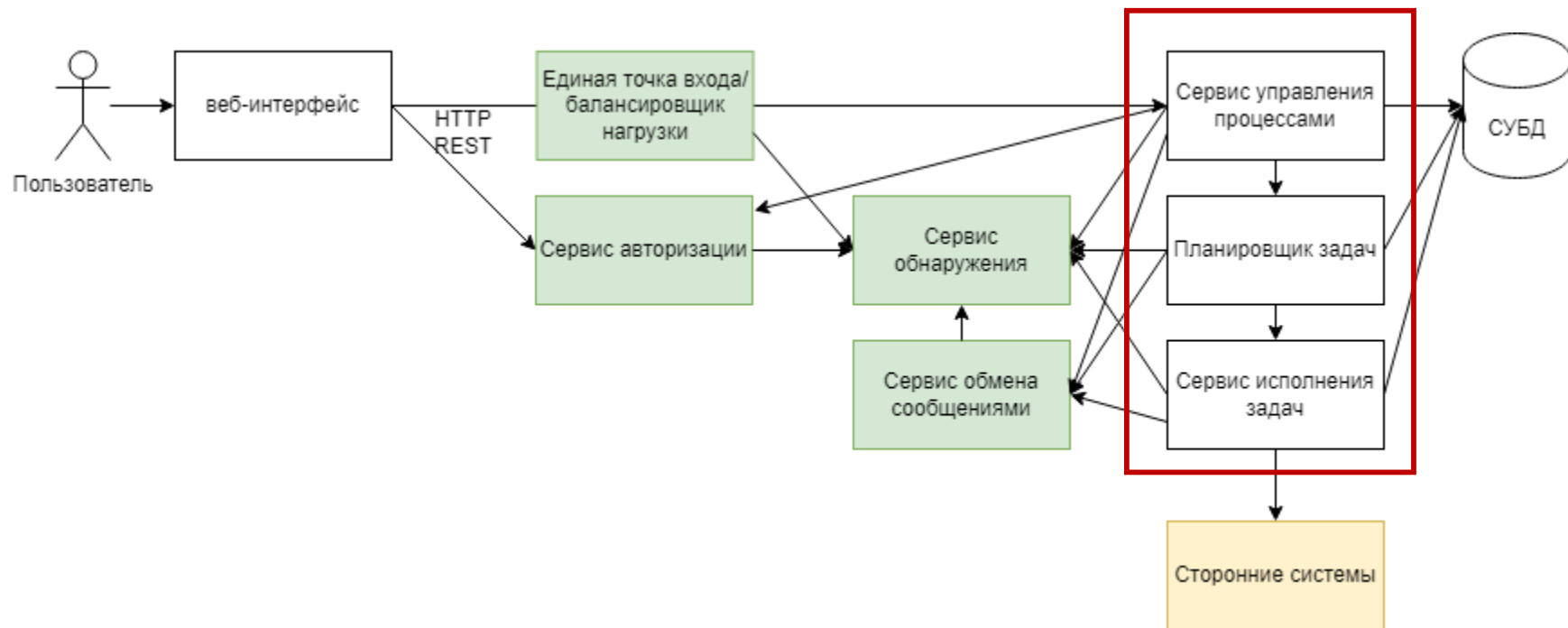
Функциональные требования:

- Загрузка процессов из файла в формате bpmn 2.0.
- Управление списком рабочих процессов: построение, редактирование, удаление.
- Изменение программного кода отдельных задач рабочего процесса.
- Запуск экземпляров рабочих процессов на исполнение через вызов API и по расписанию.
- Отслеживание состояний экземпляров рабочих процессов.

Нефункциональные требования:

- Взаимодействие с web-интерфейсом по протоколу HTTP REST.
- Реализация сервиса на языке Kotlin с использованием фреймворка Spring Boot. 4/19

Структура системы



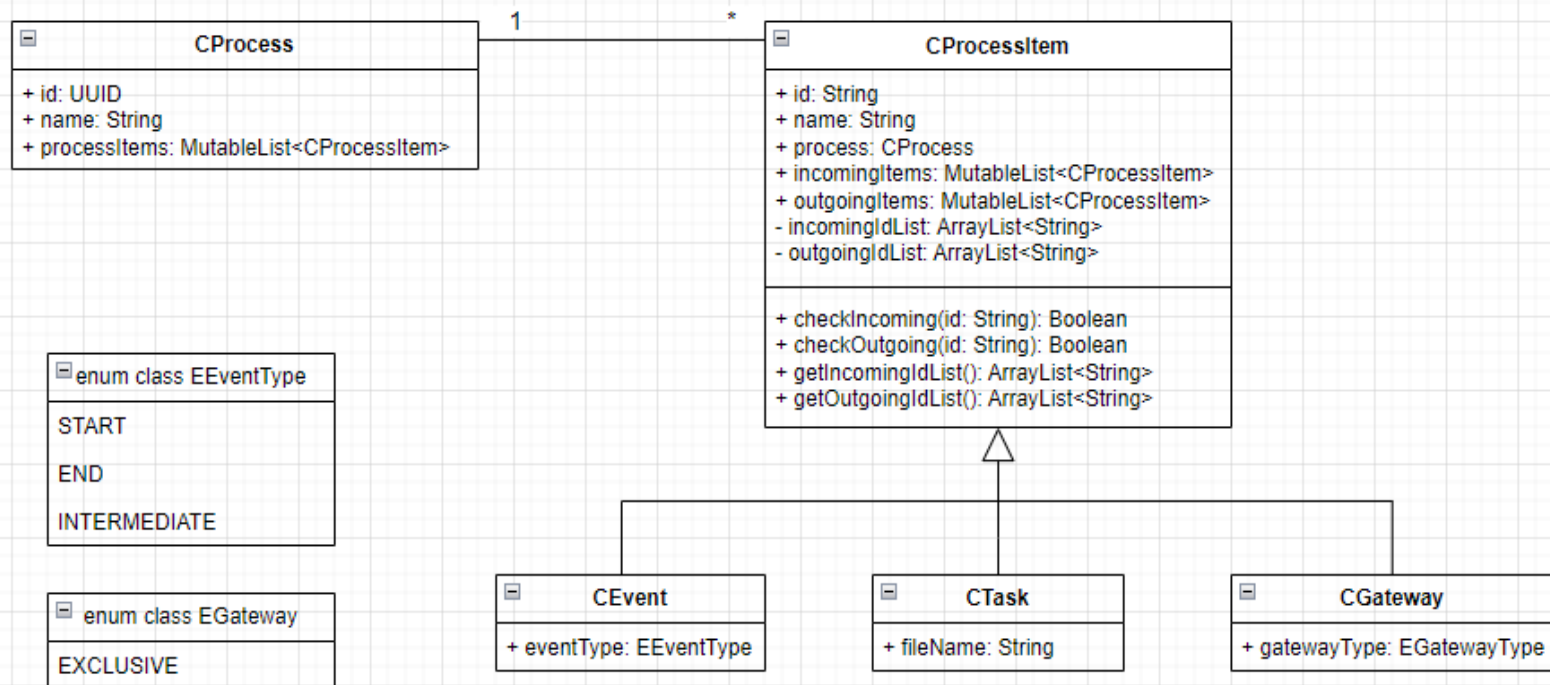
Обзор существующих решений

ПрЭВМ	Преимущества	Недостатки
ARIS APG	Хорошо развитый графический интерфейс, поддержка мощного хранилища данных	Зарубежный вендор, Высокая цена продукта
Comindware Business Application Platform 4	Возможность выбора необходимого набора услуг	Высокая цена продукта
Docsvision	Широкие интеграционные возможности	Не оптимизирован под многопользовательскую работу, зависит от платформы Microsoft
LanDocs	Возможность настроить процесс любой сложности	Высокий порог входа
1С: Документооборот	Заранее заготовленный набор из 9 шаблонов базовых процессов	Частые сбои в системе, в связи с новыми обновлениями

Проработка и детализация решения

- Изучить язык Kotlin, структуру проекта Spring Boot.
- Подготовить проект REST API: список запросов, форматы передаваемых данных.
- Спроектировать модель данных.
- Разработать алгоритм получения информации из *.BPMN файла.
- Разработать контроллер/сервисы для работы с процессами.
- Разработать контроллер/сервисы для работы с элементами процессов.
- Выбрать сервис исполнения Python кода.
- Произвести тестирование.
- Подготовить документацию.

Модель данных предметной области

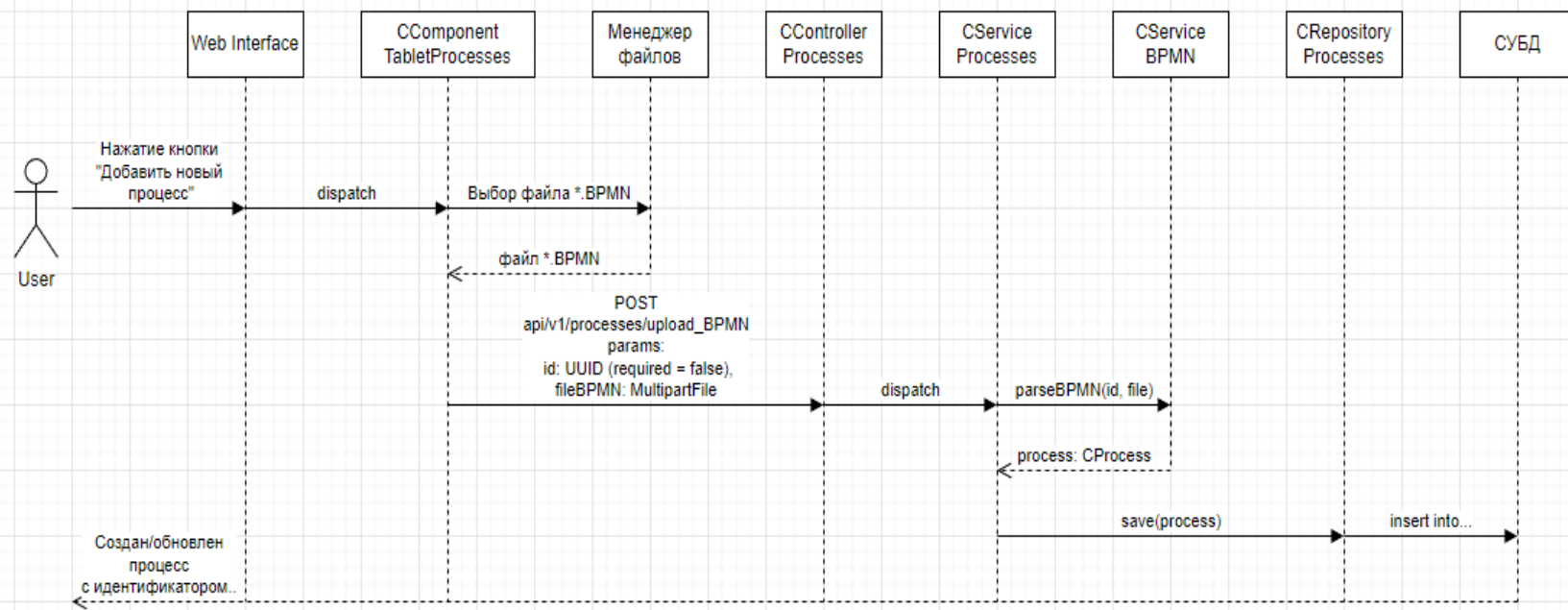


Проект программного интерфейса (API)

Тип запрос	Путь	Параметры	Формат тела запроса	Описание	Формат ответа
GET	/api/v1/processes/			Возвращает список всех процессов	<pre>{ { "id": "68a988b8-6246-413c-bd52-df7c206c1cc4", "name": "Процесс 1" }, { "id": "a8ab2aba-3c34-40e0-b886-9c6b291131f1", "name": "Публикация данных" } }</pre>
GET	/api/v1/processes/	process_id: UUID		Получение процесса по идентификатору	<pre>{ "id": "68a988b8-6246-413c-bd52-df7c206c1cc4", "name": "Процесс 1" }</pre>
POST	/api/v1/processes/	process_id: UUID (не обязательно)	file: MultiPart	Загрузка процесса, если id не указан, или в базе нет процесса с таким id, создаётся новый процесс, если id указан, создаётся новая версия процесса с этим id.	<pre>{ "id": "68a988b8-6246-413c-bd52-df7c206c1cc4", "name": "Процесс 1" }</pre>
DELETE	/api/v1/processes/	process_id: UUID		Удаление процесса по id	Удален процесс с идентификатором: id/ ResponsEntity.NotFound.build()
DELETE	/api/v1/processes/		<pre>{ "id": "68a988b8-6246-413c-bd52-df7c206c1cc4" }</pre>	Удаление процесса по сущности	Удален процесс с идентификатором: id/ ResponsEntity.NotFound.build()
DELETE	/api/v1/processes/			Удаление всех процессов	Удалены все процессы/ResponsEntity.NotFound.build()
GET	/api/v1/processes/execute	process_id: UUID		Запуск процесса с process_id	<pre>Новый экземпляр { "id": "fadf642b-118c-4364-9333-0eab33a7a185", "created_at": "2023-03-22T23:12:00", "status": { "id": "9542b833-3b38-4519-9320-168b39ecd1cb", "name": "ожидание" } }</pre>
GET	/api/v1/tasks/source	task_id: UUID		Получение кода по id задачи	
POST	/api/v1/tasks/source	task_id: UUID, pythonScript: MultipartFile		Сохранение кода по задаче с id задачи	python script text

Взаимодействие сервисов

Загрузка *.BPMN файла



Выбор сервиса исполнения программного кода python

ПрЭВМ	Преимущества	Недостатки
Celery	Бесплатное, многопоточное	Сложность настройки/отладки
django-carrot	Работа фоновом режиме, распределение задач	Привязана к Django
Jython	Интеграция с Java, доступ к библиотекам Java и Python	Нет поддержки Python 3
Chaquopy	Возможность использовать сторонние библиотеки Python	Ограниченность поддержки, разработан под Android
ProcessBuilder – Java класс	Универсальность, простота, возможность передавать аргументы внешним процессам	Ограниченность обмена данными, низкая производительность

ProcessBuilder - простое и удобное решение для запуска Python кода в Java приложении. Будет использоваться в дальнейшем для выполнения задач.

Используемый инструментарий

Среда разработки	IntelliJ IDEA Community Edition
Система сборки	Gradle
Язык программирования	Kotlin
Программная платформа	Spring Boot
Ключевые библиотеки	ProcessBuilder
Тестирование API	Postman
СУБД	PostgreSQL
Сервис хранения файлов	MinIO
Хостинг git-репозитория/boards/code review	JetBrains Space.

Структура проекта

- ▼ kotlin
 - ▼ ru.psu.workflow.serviceworkflows
 - ▼ controllers
 - CControllerProcesses
 - CControllerTasks
 - ▼ model
 - CEvent
 - CExecuteInfo.kt
 - CGateway
 - CProcess
 - CProcessItem
 - CTask
 - EEventType
 - EGatewayType
 - ▼ repositories
 - IRepositoryProcesses
 - IRepositoryTasks
 - ▼ services
 - CServiceBPMN
 - CServiceProcesses
 - CServiceTasks
 - IServiceBPMN
 - IServiceProcesses
 - IServiceTasks

Примеры реализации методов контроллеров

Обработка запроса `api/v1/processes?id=id`

```
@GetMapping (params = ["id"])
fun getById (@RequestParam id: UUID): ResponseEntity<CProcess>
{
    return serviceProcesses.getById(id)
}
```

Логика запроса, обращение к репозиторию процессов

```
override fun getById(id: UUID): ResponseEntity<CProcess>
{
    return repositoryProcesses.findById(id)
        .map { process -> ResponseEntity.ok(process) }
        .orElse(ResponseEntity.notFound().build())
}
```

Репозиторий процессов

```
@Repository
interface IRepositoryProcesses: CrudRepository<CProcess, UUID>
```

Сохранение обработанного файла *.BPMN в БД

Обработка запроса api/v1/processes/upload_bpmnПарсинг *.BPMN файла -> создание процесса

@PostMapping("upload_bpmn")

fun uploadBPMN(

 @RequestParam("id", required = false) id: UUID?,

 @RequestParam("bpmn") fileBPMN: MultipartFile

): CProcess

{

 return serviceProcesses.save(serviceBPMN.parseBPMN(id, fileBPMN))

}

override fun parseBPMN(id: UUID?, fileBPMN: MultipartFile): CProcess

{

 var process: CProcess? = null

 val dbf = DocumentBuilderFactory.newInstance()

 val doc = dbf.newDocumentBuilder().parse(fileBPMN.inputStream)

 val processNode = doc.getElementsByTagName("semantic:process").item(0)

 if (id != null) {

 process = CProcess(id,

 processNode.attributes.getNamedItem("name").nodeValue)

 } else {

 process = CProcess(UUID.randomUUID(),

 processNode.attributes.getNamedItem("name").nodeValue)

 }

 val processChildList = processNode.childNodes

 val processItemMap = getProcessItemsMap(processChildList, process)

 addConnections(processItemMap)

 makeProcess(process, processItemMap)

 return process

}

Тестирование

Загрузка файла “create_process.BPMN”. Ответ в формате JSON, данные успешно сохранены в БД.

The screenshot shows a REST client interface with a sidebar on the left containing a list of workflow actions. The main panel displays a POST request to the endpoint `localhost:8080/v1/processes/upload_bpmn/bpmn`. The request body is a file named `create_process.BPMN`. The response is shown in JSON format:

```
{
  "id": "c0c3af36-7113-45b5-ad36-b8a033982f7e",
  "name": "Создание исполняемого процесса"
}
```

The status bar at the bottom indicates a 200 OK status, 188 ms response time, and a 377 B response size.

Добавление python кода к задаче загруженного процесса

The screenshot shows the same REST client interface. The sidebar highlights the action "Добавить файл скрипта к задаче". The main panel displays a POST request to the endpoint `localhost:8080/v1/tasks/sourced_task_id=314d36d8-d56c-71ad-7db6-5254003963f7/python_script`. The request body is a file named `telegram_bot.py`. The response is shown in JSON format:

```
{
  "id": "c0c3af36-7113-45b5-ad36-b8a033982f7e",
  "name": "Создание исполняемого процесса"
}
```

The status bar at the bottom indicates a 200 OK status, 255 ms response time, and a 133 B response size.

Заключение

- Получены навыки разработки на языке программирования Kotlin на платформе Spring Boot.
- Реализованы обработка *.BPMN файла и последующее сохранение в БД.
- Реализован базовый набор HTTP REST API запросов.
- Реализована возможность добавления Python кода к задаче при помощи средств объектного хранилища MinIO.
- Подготовлена первая версия сервиса управления рабочими процессами.

Реализованно
е количество
классов
> 10

Анализ
публикаций
> 30

Количество
ручных тестов
> 100

Список литературы

1. Современная workflow-система. Возможности, инструменты, Value Stream : сайт. – URL: <https://it-guild.com/info/blog/> (дата обращения: 10.02.2023).
2. Как работает система Workflow в компании : сайт. – URL: <https://www.comindware.ru/blog/> (дата обращения: 10.02.2023).
3. Рейтинг Workflow в российских СЭД : сайт. – URL: <https://www.tadviser.ru/index.php/> (дата обращения: 11.03.2023).
4. Postman API Platform : платформа API, позволяющая разработчикам проектировать, создавать, тестировать и повторять свои API / ООО «ПОСТМЭН». – Москва 2017.

Спасибо за внимание!

Архитектура БД

processe

	id [PK] uuid	name character varying (255)
1	078c7017-2090-438f-8a9b-e3c42b517ebf	Проверка отчёта
2	b5409ba5-bc32-4f8b-94e3-ec7ea6de727c	Запуск телеграм бота

process_item

	type character varying (31)	id [PK] character varying (255)	name character varying (255)	type character varying (255)	process_id uuid	file_name character varying (255)
1	CEvent	09f595e4-d56b-11ed-7bb9-5254003963f7	Необходимость сдачи отчёта возникла	START	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
2	CTask	09f595e7-d56b-11ed-7bb9-5254003963f7	Оформление текста отчёта	[null]	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
3	CTask	09f595f6-d56b-11ed-7bb9-5254003963f7	Создание запроса на проверку	[null]	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
4	CTask	09f59600-d56b-11ed-7bb9-5254003963f7	Проверка корректности оформления	[null]	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
5	CTask	09f59604-d56b-11ed-7bb9-5254003963f7	Проверка наличия текстовых заимствований	[null]	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
6	CTask	09f59608-d56b-11ed-7bb9-5254003963f7	Проверка наличия графических заимствований	[null]	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
7	CTask	09f5960c-d56b-11ed-7bb9-5254003963f7	Информирование о проблеме	[null]	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
8	OSGateway	314d30c2-d56b-11ed-7bb9-5254003963f7	Gateway	EXCLUSIVE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
9	CEvent	314d30cc-d56b-11ed-7bb9-5254003963f7	Информация о проверке получена	END	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
10	CTask	9b5688f7-d56b-11ed-7bb9-5254003963f7	Оценка отчёта	[null]	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
11	CTask	9b5688f3-d56b-11ed-7bb9-5254003963f7	Информирование об оценке	[null]	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
12	OSGateway	9b5688f7-d56b-11ed-7bb9-5254003963f7	Gateway	EXCLUSIVE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
13	CEvent	9b5688f6-d56b-11ed-7bb9-5254003963f7	Проблемы выявлены	INTERMEDIATE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
14	CEvent	9b568900-d56b-11ed-7bb9-5254003963f7	Оформление корректно	INTERMEDIATE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
15	OSGateway	9b568903-d56b-11ed-7bb9-5254003963f7	Gateway	EXCLUSIVE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
16	CEvent	9b568906-d56b-11ed-7bb9-5254003963f7	Текстовые заимствования отсутствуют	INTERMEDIATE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
17	OSGateway	9b568909-d56b-11ed-7bb9-5254003963f7	Gateway	EXCLUSIVE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
18	CEvent	9b56890c-d56b-11ed-7bb9-5254003963f7	Графические заимствования отсутствуют	INTERMEDIATE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
19	CEvent	9b56890f-d56b-11ed-7bb9-5254003963f7	Отчёт проверен	INTERMEDIATE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
20	CEvent	9b568911-d56b-11ed-7bb9-5254003963f7	Информация о проблеме получена	INTERMEDIATE	078c7017-2090-438f-8a9b-e3c42b517ebf	[null]
21	OSGateway	way_in4r2on	Запуск бота	EXCLUSIVE	b5409ba5-bc32-4f8b-94e3-ec7ea6de727c	[null]

process_item_relatio

	source_id character varying (255)	target_id character varying (255)
1	09f595e4-d56b-11ed-7bb9-5254003963f7	314d30c2-d56b-11ed-7bb9-5254003963f7
2	314d30c2-d56b-11ed-7bb9-5254003963f7	09f595e7-d56b-11ed-7bb9-5254003963f7
3	09f595e7-d56b-11ed-7bb9-5254003963f7	09f595f6-d56b-11ed-7bb9-5254003963f7
4	09f595f6-d56b-11ed-7bb9-5254003963f7	09f59600-d56b-11ed-7bb9-5254003963f7
5	09f59600-d56b-11ed-7bb9-5254003963f7	9b5688f7-d56b-11ed-7bb9-5254003963f7
6	9b5688f7-d56b-11ed-7bb9-5254003963f7	9b568900-d56b-11ed-7bb9-5254003963f7
7	9b5688f7-d56b-11ed-7bb9-5254003963f7	9b5688f6-d56b-11ed-7bb9-5254003963f7
8	9b568900-d56b-11ed-7bb9-5254003963f7	09f59604-d56b-11ed-7bb9-5254003963f7
9	09f59604-d56b-11ed-7bb9-5254003963f7	9b568903-d56b-11ed-7bb9-5254003963f7
10	9b568903-d56b-11ed-7bb9-5254003963f7	9b568906-d56b-11ed-7bb9-5254003963f7
11	9b568906-d56b-11ed-7bb9-5254003963f7	9b56890f-d56b-11ed-7bb9-5254003963f7

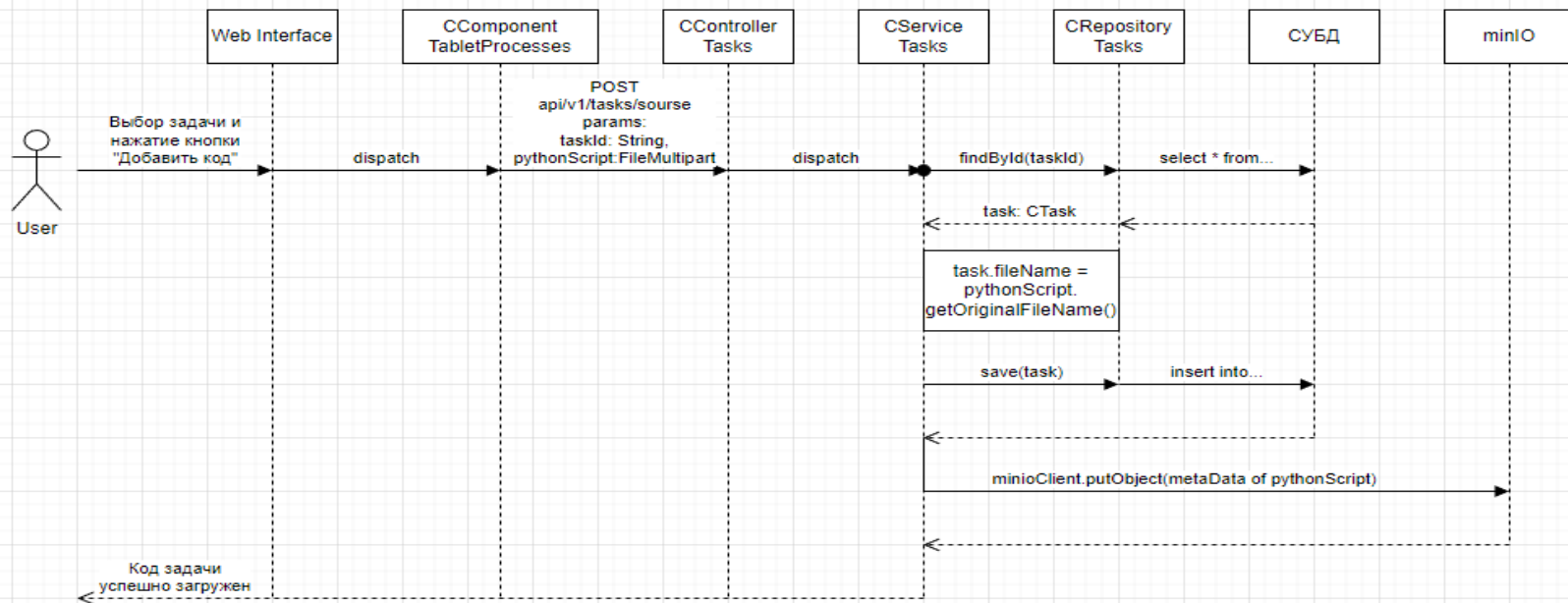
Доска задач в системе JetBrains Space

The screenshot displays the JetBrains Space task board interface. At the top, there's a header with 'Workflow / Доска задач' and filters for 'Assignee', 'Created by', and 'Tag'. The board is organized into three main columns: 'Open' (2 tasks), 'In progress' (8 tasks), and 'Done' (7 tasks). Each task card includes a title, a brief description, an assignee, and a status (e.g., 'Unscheduled').

Column	Task ID	Title	Description	Assignee	Status
Open (2)	WF-T-15	Репозиторий в web-интерфейсе с запросами к контроллеру процессов в серверной части	Даниил Саломатов	Unscheduled	
	WF-T-16	Репозиторий в web-интерфейсе с запросами к контроллеру задач в серверной части	Даниил Саломатов	Unscheduled	
In progress (8)	WF-T-5	Архитектура приложения	Илья Селетков	Unscheduled	
	WF-T-4	Нарисовать макеты окон интерфейса	Даниил Саломатов	Unscheduled	
	WF-T-8	Презентация (backend)	Сергей Семенов (me)	Unscheduled	
	WF-T-1	Логическая модель программы	Илья Селетков	Unscheduled	
	WF-T-9	Презентация (Front end)	Даниил Саломатов	Unscheduled	
	WF-T-14	Структура REST API	Сергей Семенов (me)	Unscheduled	
	WF-T-13	Разработка боковой панели с кнопками загрузки файла и выбора уже загруженных. Связь с базой	Даниил Саломатов	Unscheduled	
	WF-T-6	Презентация (Front end)	Даниил Саломатов	Unscheduled	
Done (7)	WF-T-3	Запустить пустой проект Spring Boot	Сергей Семенов (me)	Unscheduled	
	WF-T-2	Запустить пустой проект VueJS	Даниил Саломатов	Unscheduled	
	WF-T-10	Постановка задачи (Front end)	Илья Селетков	Unscheduled	
	WF-T-11	Постановка задачи (Back end)	Илья Селетков	Unscheduled	
	WF-T-8	Написание парсера файла *.bpmn 2.0	Сергей Семенов (me)	Unscheduled	
	WF-T-7	Реализация загрузки, обработки и сохранения в БД файла *.BPMN	Сергей Семенов (me)	Unscheduled	
	WF-T-12	Примеры файлов *.BPMN	Илья Селетков	Unscheduled	

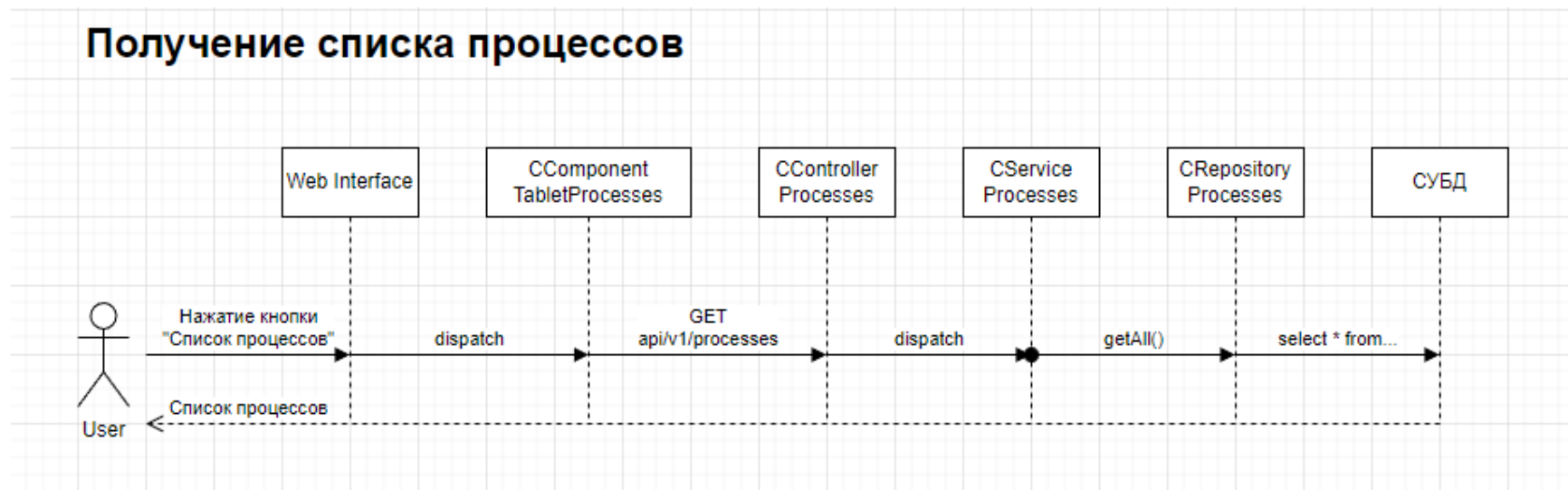
Логическая модель программы (взаимодействие сервисов)

Сохранение кода по задаче



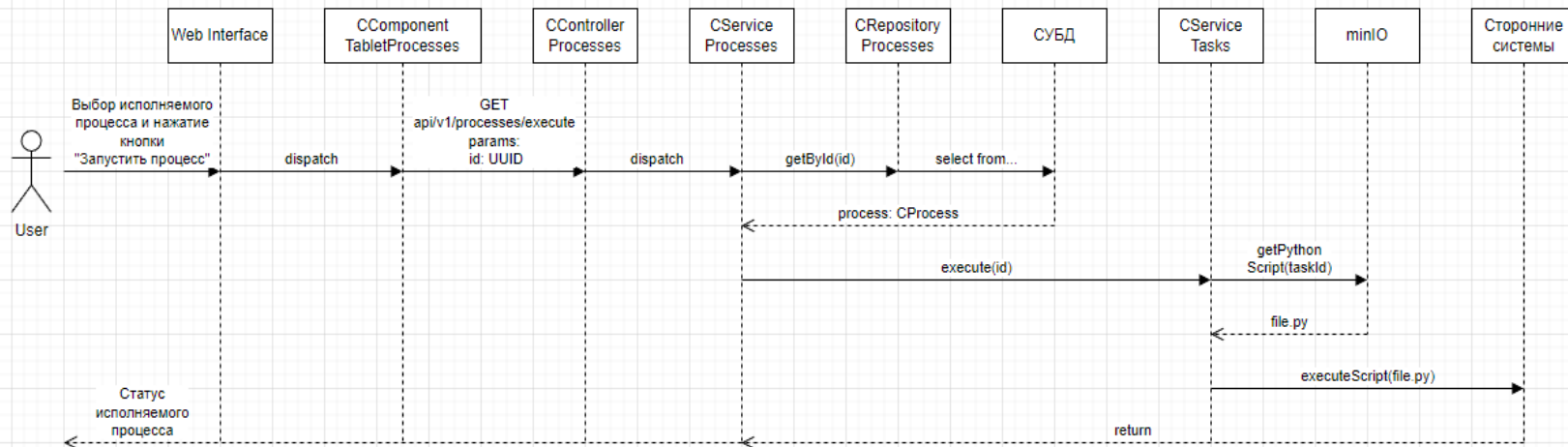
Логическая модель программы (взаимодействие сервисов)

Получение списка процессов



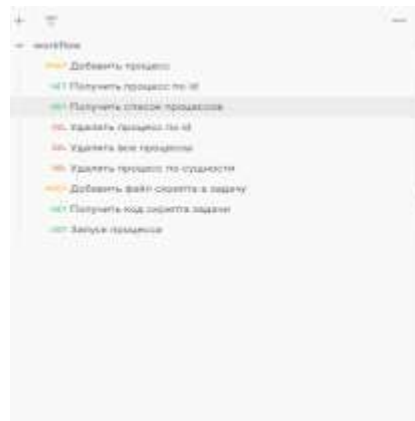
Логическая модель программы (взаимодействие сервисов)

Запуск процесса



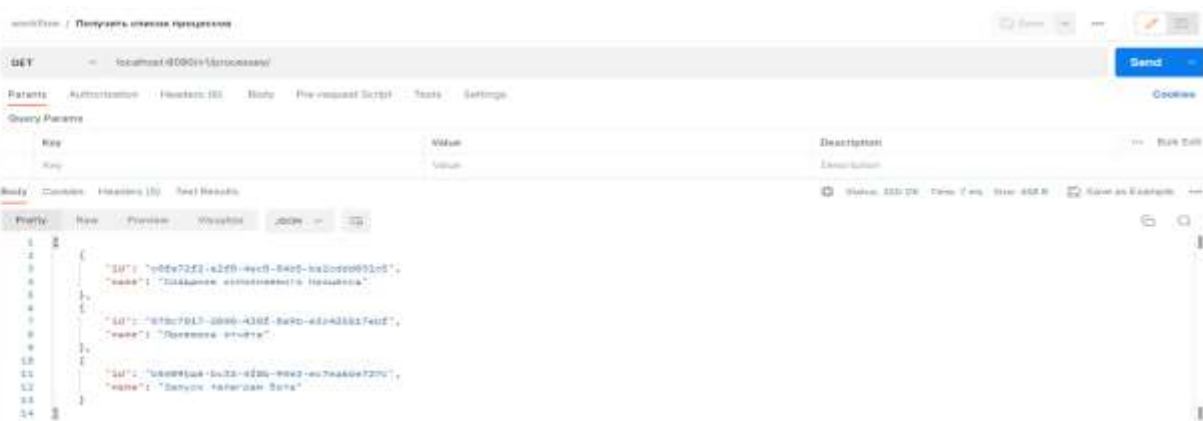
Тестирование

Получение списка процессов/удаление процесса по идентификатору



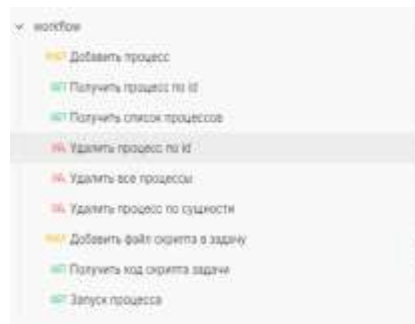
Workflow editor showing a list of steps:

- Добавить процесс
- Получить процесс по id
- Получить список процессов**
- Удалить процесс по id
- Удалить все процессы
- Удалить процесс по существу
- Добавить файл скрипта в заднюю
- Получить код скрипта задачи
- Запуск процесса



REST client showing a GET request to `localhost:8080/v1/processes/`. The response is a JSON array of process objects:

```
[{"id": "c0fe72f2-a2f8-4ec8-84b5-ba2c0d031c5", "name": "Задача по существу"}, {"id": "87bc7d17-2896-43d5-8a9d-a2a438a740f2", "name": "Процесс в уме"}, {"id": "c0fe72f2-a2f8-4ec8-84b5-ba2c0d031c5", "name": "Задача по существу"}]
```



Workflow editor showing a list of steps:

- Добавить процесс
- Получить процесс по id
- Получить список процессов
- Удалить процесс по id**
- Удалить все процессы
- Удалить процесс по существу
- Добавить файл скрипта в заднюю
- Получить код скрипта задачи
- Запуск процесса



REST client showing a DELETE request to `localhost:8080/v1/processes/c0fe72f2-a2f8-4ec8-84b5-ba2c0d031c5`. The response is a 200 OK status:

```
Удален процесс с идентификатором: c0fe72f2-a2f8-4ec8-84b5-ba2c0d031c5
```

Тестирование

Запуск исполняемого процесса

The screenshot displays a REST client interface. On the left, a 'workflow' sidebar lists several actions, with 'Запуск процесса' (Start process) highlighted at the bottom. The main area shows a GET request to `localhost:8080/v1/processes/execute?id=N6e321db-9b86-4fc1-9443-686a262f477b`. The 'Query Params' table lists the 'id' parameter. The 'Body' tab is active, showing a JSON response in 'Pretty' format. The response status is 200 OK, and the body contains process details.

Workflow:

- Добавить процесс
- Получить процесс по id
- Получить список процессов
- Удалить процесс по id
- Удалить все процессы
- Удалить процесс по существу
- Добавить файл скрипта в задачу
- Получить код скрипта задачи
- Запуск процесса**

Request: GET `localhost:8080/v1/processes/execute?id=N6e321db-9b86-4fc1-9443-686a262f477b`

Query Params:

Key	Value	Description
<input checked="" type="checkbox"/> id	N6e321db-9b86-4fc1-9443-686a262f477b	

Response (JSON):

```
{
  "processId": "f9e321db-9b86-4fc1-9443-686a262f477b",
  "createDate": "2023-04-23T05:10:22.4731430",
  "status": {
    "id": "1704635398",
    "name": "Ожидание"
  }
}
```