

Описание

Модуль конвертации файлов в формат PDF.

API

Входные параметры: * InputStream/File - стрим из исходного файла либо непосредственно файл * тип исходного файла (AllowedFileType) * временный файл для конвертации (PDF)

Выходные параметры: * Path - файл результата конвертации в PDF

Для конвертации файл -> PDF рекомендуется использовать `mp.jprime.services.FileToPdfConverter`, передавая в метод `toPDF()` стрим из исходного файла (либо непосредственно исходный файл), тип исходного файла (AllowedFileType) и временный файл для конвертации (PDF).

```
/**
 * Пул {@link FileToPdfConverter конвертеров}
 */
private PdfConverterPool converterPool;

@Autowired
private void setConverterPool(PdfConverterPool converterPool) {
    this.converterPool = converterPool;
}

private Mono<Void> writeConvertedTo(..., AllowedFileType fileType, ..., InputStream inputStream, ...) {
    //...
    Path tmpFile;
    try {
        tmpFile = Files.createTempFile("file-preview-converted_", ".pdf");
        tmpFileRef.set(tmpFile);
    } catch (Exception e) {
        throw new JPRuntimeException(e);
    }
    Path pdfPath = converterPool.getConverter(fileType).toPdf(sourceFile.toFile(), fileType, pdfFile.toFile());
    //...
}
```

Конвертация на основе `org.jodconverter` описана в классе `mp.jprime.services.FileToPdfJodConverter`. При необходимости использовать другие способы конвертации — имплементируем интерфейс `mp.jprime.services.FileToPdfConverter`.

Каждый конвертер знает список форматов, которые он может конвертировать (имплементации интерфейса `mp.jprime.files.AllowedFileType`). Например, для реализации основанной на `org.jodconverter` — этот список находится в enum `mp.jprime.files.JodAllowedFileType`. При необходимости добавить поддержку конвертации из нового типа файла нужно добавить его расширение в список поддерживаемых типов одного из имеющихся конвертеров, проверить работоспособность, либо создать свою имплементацию `mp.jprime.services.FileToPdfConverter`, которая будет конвертировать новый тип.

Конвертация HTML -> PDF

Внешние HTML файлы могут приходить в различных кодировках (в т.ч. без указания кодировки), с битыми тэгами и другими проблемами, что крайне усложняет корректную конвертацию через Java.

Однако браузеры без проблем отображают подобные файлы. Поэтому для конвертации HTML ->PDF в HtmlToPdfConverter используется обращение к браузеру Chrome/Chromium. Через командную строку запускается браузер без интерфейса и выполняет сохранение переданного html файла в переданный pdf файл.

- Настройки:
jprime.html-converter.*

Настройка	По умолчанию	Описание
enabled	false	Признак активации конвертации через Chrome
path	-	Путь к Chrome/Chromium (включая исполняемый файл)

Настройки OfficeManager

jprime.openoffice.*

- Для обеспечения стабильности работы сервиса настройку disableOpengl выставить true.
- PureJavaProcessManager - этот процесс-менеджер работает в среднем *в 1,3 раза быстрее*, как показали тесты. И он не зависит от платформы - поэтому он выбран как менеджер по умолчанию. Однако он не умеет получать PID процесса, что может быть проблемой, поэтому рекомендуется в aplicacion-distrib.yml передавать специфичный для конкретной ОС менеджер (напр. org.jodconverter.local.process.UnixProcessManager).

Настройка	По умолчанию	Описание
path	-	Место установки LibreOffice
taskExecutionTimeout	120000	Задаёт максимальное время, необходимое для обработки задачи. Если время обработки задачи превышает этот тайм-аут, эта задача будет прервана и будет обработана следующая.
taskQueueTimeout	30000	Это свойство используется для установки максимального времени жизни задачи в очереди на конвертацию. Задача будет удалена из очереди, если время ожидания превышает этот тайм-аут, и будет

Настройка	По умолчанию	Описание OfficeException
processTimeout	120000	Время в миллисекундах на попытку выполнить вызов офисного процесса (start/terminate).
maxTasksPerProcess	200	Максимальное количество задач, которые может выполнить офисный процесс до перезапуска
disableOpengl	true	Отключает использование OpenGL при конвертации. Рекомендуется true во избежание падений процесса LibreOffice.
ports	2002	Порты, на которых будут запущены локальные процессы Office для конвертации.
afterStartProcessDelay	0	указывает задержку в миллисекундах после попытки запустить процесс office, прежде чем делать что-либо еще. В некоторых ОС отсутствие этой задержки может вызывать зависание, разработчики JODConverter не знают, почему
processManager	<code>org.jodconverter.local.process.PureJavaProcessManager</code>	Менеджер процессов

Deployment

Так как для работы конвертера кроме непосредственно Java-приложения запускаются так же внешние процессы (напр. LibreOffice, Chromium) необходимо выделить им достаточно памяти. В случае запуска в докер-контейнере рекомендуется поднять выделенную докеру память, а так же снизить `MaxRAMPercentage` .