



123426.1.10!20190923T13:15-0000021

Драфт кейса для кейс-лаборатории / Java

Создание системы документооборота с минимальной функциональностью.

Требования к используемым технологиям:

Клиентская часть:

Angular 2+ - рекомендуемое

Angular 1.x - возможное

Серверная часть:

Spring boot - рекомендуемое

Spring, java EE - возможное

База данных:

Никаких ограничений. Рекомендуемое PostgreSQL, как самая простая в установке на рабочем месте. Также возможен MySQL.

Функциональные требования:

- 1 Система должна позволять импортировать следующие форматы файлов - PDF, JPG, TIFF.
- 2 Система должна позволять вносить изменения в сформированные пакеты документов. Перемещать страницы между пакетами, удалять страницы из пакетов, создавать новые пакеты, удалять пакеты документов, редактировать ориентацию страницы.
- 3 Система должна предоставлять возможность предпросмотра и редактирования импортированного комплекта документов. Должна быть возможность применять изменение как к одной странице, так и ко всему комплекту документов.
- 4 Система должна предоставлять возможность просмотра ранее импортированных комплектов документов, которые были отправлены во внешнюю систему.
- 5 Система должна иметь возможность, сформированный пакет отправлять на сервер в виде единого PDF файла.
- 6 Система должна предоставлять возможность изменять язык для каждого пользователя индивидуально.
- 7 Должна быть возможность настройки постобработки страниц пакета документов: сжатие, перевод в ч/б формат. *- опционально.

Нефункциональные требования:

- 1 Система должна работать под операционной системой Windows / под семейством Unix-подобных операционных систем на базе ядра Linux. Можно выбрать.
- 2 Система должна работать с использованием «тонкого» клиента. Тонкий клиент должен корректно работать в браузерах Internet Explorer 11, Google Chrome, браузерах на базе Chromium.
- 3 Система должна предоставлять возможность базовой аутентификации (логин/пароль).
- 4 В Системе должен быть реализован инструмент интеграции с системой оповещения пользователей посредством электронной почты. Система должна отправлять пользователям почтовые уведомления в согласованном формате.
- 5 Система должна предоставлять API для взаимодействия с внешними системами.
- 6 Система должна иметь совмещенный пользовательский и административный интерфейс. Доступ к функциям должен определяться ролью пользователя.
- 7 Система должна иметь возможность поддержки мульти-язычных пользовательских интерфейсов.
- 8 Система должна иметь встроенный мониторинг, который информирует о доступности системы.
- 9 Система должна поддерживать аудит как действий пользователей, так и ключевых изменений в системе, в объеме достаточном для анализа. Логирование.
- 10 Система должна функционировать в трехзвенной архитектуре.

- 11 Проект при необходимости может быть собран на любом внешнем сервере. Результат сборки может быть запущен на любом сервер-приложении, например, в Tomcat.



123426.1.10!20190923T13:15-000002

Кейс проблематика



Автоматизация процессов

- Уменьшение количества человеческих ошибок
- Избавление пользователей от рутинной работы
- Повышение эффективности работы пользователя



Монолитная архитектура

- Сбои компонент систем сказываются на работоспособности всей системы
- Обновление системы трудозатрат
- Обновление требует простоя всей системы

Кейс описание

Создание системы манипулирования контентом

- Использование штрихкодирования в качестве первичной обработки контента
- Реализация пользовательского интерфейса для дополнительной обработки контента
- Реализация возможности задавать правила первичной обработки контента — опционально (универсализация разработки, применимость не только в рамках конкретной задачи)





123426.1.10!20190923T13:15-00000

Предлагаемые технологии

Клиентская часть	Серверная часть	База данных
Angular 2+ — рекомендуемое	Spring boot — рекомендуемое	Без ограничений
Angular 1.x — возможное	Java EE, Spring — возможное	Postgre SQL — рекомендуемое
		MySql — возможное

Для всех операций с PDF – Apache PDFBox <https://pdfbox.apache.org/>

Для чтения штрих-кодов - ZXing (Zebra Crossing) barcode scanning library

<https://github.com/zxing/zxing>

Деление многостраничного TIFF на странички – умеет сама Java через JAI.

Средства разработки

- Maven, idea — рекомендуется, java 8, gitlab
- Grunt — сборка приложения с angular 1.x, less — опционально

