

Техническое Задание: Разработка плагина NocoBase для генерации документов Word

1. Общая цель

Целью является разработка плагина для системы NocoBase, который позволит менеджерам образовательных программ генерировать документы Microsoft Word (заявления на зачисление, договоры) на основе predetermined шаблонов. Генерация должна происходить для конкретного зачисления (Enrollment), связанного со сделкой (Deal), с использованием данных слушателя (Person) и образовательной программы/курса (Course), ассоциированных с этим зачислением и сделкой.

2. Бизнес-контекст

- Используется CRM-система на базе NocoBase.
- Основные сущности (коллекции): Пользователи (Users), Роли (Roles), Сделки (Deals), Слушатели/Персоны (Persons), Курсы (Courses), Зачисления (Enrollments), Шаблоны (Templates), Сгенерированные документы (generatedDocuments).
- Пользователи с ролью "manager" отвечают за определенные курсы.
- Менеджерам необходимо формировать документы для слушателей в рамках процесса зачисления.
- В коллекции "Deals" существуют поля для выбора шаблонов Word для заявления (`selectedAppTemplate`) и договора (`selectedContractTemplate`).
- Сгенерированный документ должен быть доступен для скачивания менеджером сразу после генерации и отправки слушателю.

3. Основные Требования к Плагину

Плагин должен реализовать следующую функциональность:

- **Интеграция с UI NocoBase:** Добавить две кнопки ("Сгенерировать заявление", "Сгенерировать договор") в интерфейс просмотра/редактирования записи "Зачисление" (Enrollment), который открывается из связанной "Сделки" (Deal).
- **API Endpoint:** Создать серверный маршрут (API endpoint) в рамках плагина, который будет принимать запрос на генерацию документа.
- **Получение данных:** API endpoint должен уметь извлекать необходимые данные на основе переданных идентификаторов:
 - Данные шаблона (из коллекции `templates` по ID шаблона).
 - Данные слушателя (из коллекции `persons` по ID слушателя из зачисления).
 - Данные курса (из коллекции `courses` по ID курса из сделки или зачисления).
 - Данные сделки (из коллекции `deals` по ID сделки).

- Данные зачисления (из коллекции `enrollments` по ID зачисления).
- **Генерация документа:** Использовать библиотеку для работы с `.docx` файлами (например, `docx-templates` для Node.js или аналогичную), чтобы заполнить данными выбранный шаблон Word.
- **Сохранение документа:** Сохранить сгенерированный `.docx` файл:
 - Создать новую запись в существующей коллекции `generatedDocuments`.
 - Заполнить поля новой записи: `name` (например, "Договор для [ФИО слушателя]"), `templateID` (ID использованного шаблона), `dealID` (ID сделки), `personID` (ID слушателя), `enrollmentID` (ID зачисления - *возможно, потребуется добавить это поле в схему `generatedDocuments`*).
 - Загрузить сгенерированный файл в поле `generatedDocument` (тип `attachment`) этой новой записи.
- **Скачивание документа:** После успешной генерации и сохранения, инициировать скачивание сгенерированного файла браузером пользователя.
- **Отправка документа:** После успешной генерации и сохранения, инициировать отправку сгенерированного файла, связанному пользователю слушателю по email с шаблонным текстом (общим для всех программ).

4. Пользовательский Сценарий (User Flow)

1. Менеджер открывает карточку "Сделки" (Deal).

Пользователи

Программы

Заявки и договоры

Люди

Шаблоны

Зачисления

Заявки и договоры

Таблица

Канбан

Фильтр

Удалить

Экспортировать

Добавить новый

	Действия	Программа	Контакты	Статус	Стоимость	Тип заказчика	Форма реализации	Тип программы	Когда создано	Когда изменено	Стат
1	Удалить Изменить Подробнее	EMBA (тест)	8316454@gmail.com - Крючков Василий...	Документы запрошены	900,000	физлицо	Очно-заочная	пп	2025-04-02	2025-04-02	не о
2	Удалить Изменить Подробнее	EMBA (тест)	suslov_ck@mailwqlke.rr - Суслов Михаил	Заявка	900,000	физлицо	Очно-заочная	пп	2025-04-02	2025-04-02	не о
3	Удалить Изменить Подробнее	EMBA (тест)	vasyan.super@nf.ew - Васнецов Виктор	Договор получен	900,000	физлицо	Очно-заочная	пп	2025-04-02	2025-04-02	не о
4	Удалить Изменить Подробнее	EMBA (тест)	shisha.vanya@ds.ew - Шишкин Иван	Заявка	900,000	физлицо	Очно-заочная	пп	2025-04-02	2025-04-02	не о
5	Удалить Изменить Подробнее	EMBA (тест)	levik.bomba@few.rw - Левитан Исаак	Заявка	900,000	физлицо	Очно-заочная	пп	2025-04-02	2025-04-02	не о
6	Удалить Изменить Подробнее	EMBA (тест)	serik.paint@das.ew - Серов Валентин	Заявка	900,000	физлицо	Очно-заочная	пп	2025-04-02	2025-04-02	не о
7	Удалить Изменить Подробнее	EMBA (тест)	sdfgfdsg@fdsa.rt - Васечкин Иван Андр...	Заявка	900,000	физлицо	Очно-заочная	пп	2025-04-02	2025-04-05	не о

Всего 7 элементов < 1 > 20 / стр. ▾

Подробности

Зачисления

Зачислить слушателя :

Зачислить

	Действия	Слушатель	Статус	ID
1	Подробнее	8316454@gmail.com - Крючков Василий...	Не зачислен	7
2	Подробнее	suslov_ck@mailwqlke.rr - Суслов Михаил	Не зачислен	9

Всего 2 элементов < 1 > 20 / стр. ▾

- В карточке сделки менеджер видит связанные "Зачисления" (Enrollments).
- Менеджер выбирает и открывает конкретное "Зачисление".
- В интерфейсе просмотра "Зачисления" менеджер видит две новые кнопки: "Сгенерировать заявление" и "Сгенерировать договор".

Подобности

Подобности

☺ Зачислить слуш

Зачислить

Действия

1 Подобности

2 Подобности

Фамилия :

Крючков

Имя :

Василий

Отчество :

Андреевич

Телефон :

+7 434 232-32-23

Email :

8316454@gmail.com

Когда изменено :

2025-04-02

Generate doc

5. Менеджер нажимает одну из кнопок (например, "Сгенерировать договор").
6. Система использует ID шаблона, указанный в соответствующем поле *родительской Сделки*.
7. Система собирает данные слушателя (`studentId` из Зачисления -> `persons`), курса (`courseId` из Сделки -> `courses`) и другие необходимые данные.
8. Плагин генерирует `.docx` файл на основе шаблона и данных, заменяя в шаблоне переменные в формате `${variable}` или аналогичные соответствующими данными из БД Nocobase.

Заявление

Прошу зачислить меня на обучение по программе повышения квалификации «Актуальные проблемы теории и практики управления и экономики» экономического факультета МГУ имени М.В. Ломоносова с «3» марта 2025 года по «2» марта 2026 года.

Контактные и личные данные

Дата рождения: `${birthDateStudent}`

Е-mail: `${emailStudent}`

Телефон: `${phoneStudent}`

СНИЛС: `${SnilsNumberStudent}`

Гражданство: `${citizenshipStudent}`

Образование: `${EducationLevelStudent}`

Серия документа о высшем образовании: `${diplomaSeriesStudent}`

Номер документа о высшем образовании: `${diplomaNumberStudent}`

Кем и когда выдан диплом: `${diplomaIssuedFromStudent}`

Фамилия в дипломе: `${diplomaLastNameStudent}`

Ученая степень: `${academicDegreeStudent}`

Ученое звание: `${academicTitleStudent}`

Общий стаж (лет): `${workExperienceStudent}`

Педагогический стаж (лет): `${teachingExperienceStudent}`

9. Плагин создает запись в коллекции `generatedDocuments`, сохраняет в хранилище связанный файл.
10. В карточке зачисления появляется новая запись в списке сгенерированных файлов для зачисления с возможностью этот файл скачать пользователем NocoBase или отправить файл слушателю на email из зачисления

5. Технические Детали Реализации

- **API Endpoint:**
 - Создать маршрут, например: `POST /api/template-manager:generate`
 - Маршрут должен принимать в теле запроса (или параметрах) как минимум: `enrollmentId`, `dealId`, `personId`, `courseId`, `templateId`.
 - Реализовать логику получения данных из коллекций NocoBase по этим ID.
 - Реализовать логику генерации документа и сохранения в хранилище NocoBase с использованием библиотеки (например, `docx-templates`).
 - Реализовать логику сохранения записи в `generatedDocuments` и загрузки файла.
 - В случае ошибки, вернуть соответствующий HTTP-статус и сообщение об ошибке.
- **Обработка Шаблонов Word:**
 - Плагин должен уметь работать с файлами `.docx` и `.doc`.
 - Необходимо определить формат плейсхолдеров (заполнителей) в шаблонах Word, которые плагин будет заменять данными. Например: `{lastName}`, `{firstName}`, `{patronymic}`, `{course.fullName}`, `{course.price}`, `{deal.id}` и т.д.
 - Предполагается, что шаблоны уже загружены в коллекцию `templates` и связаны с курсами. Плагин *не* должен реализовывать загрузку или управление шаблонами.
- **Тестирование:**
 - Первоначально протестировать работу API endpoint напрямую (например, через Postman или curl), передавая тестовые ID.
 - После успешного тестирования API, интегрировать кнопки в UI и протестировать полный пользовательский сценарий.

6. Существующая Структура Данных (Схема Коллекций NocoBase)

- **trainingGroups**
 - `createdAt`: string (\$date-time)
 - `updatedAt`: string (\$date-time)
 - `name`: string

- endDate: string
- id: integer (\$int64)
- lmsGroupld: string
- courseId: integer (\$int64)
- startDate: string
- createdByld: integer (\$int64)
- updatedByld: integer (\$int64)
- **templates**
 - createdAt: string (\$date-time)
 - updatedAt: string (\$date-time)
 - name: string
 - description: string
 - type: string
 - isActive: string
 - file: attachment
 - id: integer (\$int64)
 - courseID: integer (\$int64)
 - createdByld: integer (\$int64)
 - updatedByld: integer (\$int64)
- **enrollments**
 - dealld: integer (\$int64)
 - createdAt: string (\$date-time)
 - updatedAt: string (\$date-time)
 - id: integer (\$int64)
 - trainingGroupld: integer (\$int64)
 - studentld: integer (\$int64)
 - status: string
 - createdByld: integer (\$int64)
 - updatedByld: integer (\$int64)
- **deals**
 - createdAt: string
 - updatedAt: string
 - status: string
 - templateID: integer (\$int64)
 - id: integer (\$int64)
 - clientTyp: string
 - paymentStatus: string
 - notes: string
 - courseId: integer (\$int64)
 - trainingGroupld: integer (\$int64)
 - formComment: string
 - f_4ud089mte33: integer (\$int64)
 - createdByld: integer (\$int64)
 - updatedByld: integer (\$int64)

- **generatedDocuments**
 - sentAt: string
 - templateID: integer (\$int64)
 - id: integer (\$int64)
 - dealID: integer (\$int64)
 - createdAt: string (\$date-time)
 - updatedAt: string (\$date-time)
 - name: string
 - generatedDocument: attachment
 - personID: integer (\$int64)
 - sentByEmail: string
 - createdById: integer (\$int64)
 - updatedById: integer (\$int64)
- **persons**
 - id: integer (\$int64)
 - createdAt: string (\$date-time)
 - updatedAt: string (\$date-time)
 - firstName: string
 - lastName: string
 - patronymic: string
 - phoneNumber: string
 - email: string
 - personNotes: string
 - fullName: string
 - createdById: integer (\$int64)
 - updatedById: integer (\$int64)
- **courses**
 - id: integer (\$int64)
 - createdAt: string (\$date-time)
 - updatedAt: string (\$date-time)
 - fullName: string
 - shortName: string
 - type: string
 - form: string
 - price: string
 - lmsModule: string
 - url: string
 - createdById: integer (\$int64)
 - updatedById: integer (\$int64)

8. Ожидаемые Результаты (Deliverables)

- Исходный код плагина NocoBase в формате .tar, готовый к установке через UI.
- Краткое описание по добавлению кнопок для генерации документов в интерфейсе NocoBase для вызова плагина (доступны для добавления для всех коллекций или только зачислений или еще с какими-то ограничениями).
- Список плейсхолдеров, которые плагин ожидает найти в шаблонах Word для замены данными и пояснения, как этот список можно расширить при необходимости.