**Техническое задание на разработку интеграции с сервисом автоматизации Albato**

**Дата:** -
**Заказчик: livedigital**
**Исполнитель: -**
**Дата начала работ: [укажите дату].
Ожидаемая дата завершения работ: [укажите дату].**

### **1. Цели и задачи проекта**

Целью работы является разработка интеграции Albato с сервисом ВКС livedigital: Albato предоставляет обширный инструментарий для гибкой автоматизации реализации и интеграции бизнес-процессов.

### **1.1. Целевая аудитория**

1. Сотрудники livedigital;

2. Клиенты livedigital.

### **2. Требования функциям (задачам), выполняемым плагином**

#### **2.1. Основные требования**

**Описание:**

1. Автоматизация процессов обмена данными между сервисом видеоконференций livedigital и Albato.
2. Возможность передачи информации о созданных конференциях, пользователях, записях и событиях (например, начало или окончание конференции).
3. Настройка Webhook для моментального уведомления о событиях.
4. Поддержка возможности управления функционалом видеоконференций через API (например, создание группы, конференций или вебинаров и т.д.).

### **3. Требования к функциональным возможностям**

#### **3.1. Авторизация через API Token**

**Описание:**Пользователь (администратор) должен выполнять авторизацию используя свой API-токен, который именуется как Personal Token. Personal token не имеет ttl, используется в дальнейшем как шаблон для создания связок.

**Функциональность:**

1. Пользователь самостоятельно создаёт токен в личном кабинете livedigital.
2. Пользователь вводит ранее созданный токен в поле авторизации Albato, получая доступ к интеграции.

#### **3.2. Действие: возможность создать новый space (группу) для проведения в нем мероприятия**

**Описание:**Пользователь (администратор) имеет возможность из интерфейса Albato создать новую группу комнат (space) и указать её наименование.

**Функциональность:**

1. В процессе настройки подключения связки пользователь имеет возможность создать space (группу) для проведения в ней мероприятия при помощи метода [сreateSpace](https://moodhood-api.livedigital.space/doc/#/Spaces/createSpace).

#### **3.3. Действие: возможность удалить ранее созданный space (группу)**

**Описание:**Пользователь (администратор) имеет возможность из интерфейса Albato удалить необходимую группу (space).

**Функциональность:**

1. Удаление группы производится только после её создания при помощи метода [deleteSpace](https://moodhood-api.livedigital.space/doc/#/Spaces/deleteSpace).

#### **3.4. Действие: возможность создать новую комнату для проведения в ней конференции или вебинара**

**Описание:**Пользователь (администратор) имеет возможность из интерфейса Albato создать новую комнату с необходимым типом (вебинар или конференция) и настройками.

**Функциональность:**

1. В процессе настройки подключения связки пользователь имеет возможность создать комнату для проведения в ней мероприятия при помощи метода [createRoom](https://moodhood-api.livedigital.space/doc/#/SpaceRooms/createRoom).
2. У пользователя должна быть возможность выбрать группу (spaceId и/или spaceName), в которой создастся комната.
3. В процессе создания комнаты нужно добавить обязательные поля настроек: name, type (lesson/webinar) и необязательные поля: isAutoRecordingAllowed (Автоматическая запись при входе администратора в комнату), isMicrophonePublishingAllowed (Выключение всех микрофонов участников true/false ), isCameraPublishingAllowed (Выключение всех камер участников true/false), isScreenMediaPublishingAllowed (Блокировка доступа к демонстрации экрана для участников встречи true/false ), waitingRoomAudience (Зал ожидания true/false), redirectUrl (Переадресация после завершения звонка в формате URL link).

#### **3.5. Действие: изменить настройки созданной комнаты**

**Описание:**Пользователь (администратор) должен иметь возможность изменить настройки ранее созданной комнаты.

**Функциональность:**

1. Настройки комнаты могут быть изменены пользователем при выполнении действия PUT updateRoom
2. Новые настройки задаются параметрами: name, type (lesson/webinar), и необязательные поля: isAutoRecordingAllowed (Автоматическая запись при входе администратора в комнату), isMicrophonePublishingAllowed (Выключение всех микрофонов участников true/false ), isCameraPublishingAllowed (Выключение всех камер участников true/false), isScreenMediaPublishingAllowed (Блокировка доступа к демонстрации экрана для участников встречи true/false ), waitingRoomAudience (Зал ожидания true/false), redirectUrl (Переадресация после завершения звонка в формате URL link).

#### **3.6. Действие: удалить ранее созданную комнату**

**Описание:**Пользователь (администратор) должен иметь возможность удалить ранее созданную комнату.

**Функциональность:**

1. Удаление производится через метод [deleteRoom](https://moodhood-api.livedigital.space/doc/#/SpaceRooms/deleteRoom).

#### **3.7. Действие: создать обычную (не персонализированную) пригласительную ссылку на встречу**

**Описание:**Пользователь (администратор) создает пригласительную ссылку для комнаты.

**Функциональность:**

1. В процессе настройки подключения связки пользователь имеет возможность создать ссылку для комнаты с ролью role\_room\_user или role\_room\_moderator.

#### **3.8. Действие: создать персонализированную пригласительную ссылку на встречу**

**Описание:**Пользователь (администратор) имеет возможность создать пригласительную ссылку на комнату, которую укажет пользователь. Ссылка должна содержать индивидуальные параметры участника: имя (username), роль (moderator/user), externalUserId (необязательное поле), externalMeetingId (необязательное поле), ttl (максимальное значение 604800 секунд), email, phone. Используется метод [roomGenerateAccess](https://moodhood-api.livedigital.space/doc/#/RoomGenerateAccess/roomGenerateAccess).

**Функциональность:**

1. Для каждой комнаты пользователь (администратор) должен иметь возможность создать ссылку с индивидуальными параметрами.
2. Индивидуальные параметры должны быть заполнены вручную, либо автоматически при создании связки с другим сервисом.

#### **3.9. Действие: выгрузить аналитический отчёт в формате xlsx и json**

**Описание:**Пользователь должен иметь возможность выгрузить по завершенному звонку аналитический отчёт в формате xlsx и json

**Функциональность:**

1. По окончанию встречи срабатывает вебхук, формируется аналитический отчёт.
2. Аналитический отчёт выгружается по действию пользователя.
3. При загрузке отчёта используется callId (идентификатор корректного звонка в вебхуке).

#### **3.9. Действие: выгрузить аналитический отчёт по проведенному автовебинару**

**Описание:**Пользователь должен иметь возможность выгрузить по завершенному автовебинару аналитический отчёт в формате xlsx.

**Функциональность:**

1. Реализовать получение аналитики после прохождения пользователем автовебинара. Используется метод: [getPlaybackSessionAnalyticsReport](https://moodhood-api.livedigital.space/doc/#/PlaybackAnalytics/getPlaybackSessionAnalyticsReport).

#### **3.10. Действие: выгрузить видеозапись в формате mp4**

**Описание:**Пользователь должен иметь возможность выгрузить по завершенному звонку аналитический отчёт в формате xlsx.

**Функциональность:**

1. По окончанию встречи формируется видеозапись, которой присваивается параметр recordId.
2. recordId используется при загрузке видеозаписи. Его также можно получить в webhook.

#### **3.11. Действие: создать автовебинар из существующей видеозаписи звонка**

**Описание:**Добавить механизм создания автовебинаров из существующих записей на платформе livedigital.

**Функциональность:**

1. Создание сущности автовебинара из существующих записей как отдельное действие.
2. При создании комнаты типа “автовебинар” используется метод создания сущности автовебинара в API livedigital: [createPlayback](https://moodhood-api.livedigital.space/doc/#/Playbacks/createPlayback). Интегрируемые поля: name (название автовебинара).
3. При настройки связки пользователь указывает из какой группы (space) и комнаты (room) парсить запись для использования в качестве материала для автовебинара.
4. После выбора записи подставляем её id в автовебинар при помощи [updatePlayback](https://moodhood-api.livedigital.space/doc/#/Playbacks/updatePlayback).
5. После создания сущности необходимо задать время проведения автовебинара: [createPlaybackEvent](https://moodhood-api.livedigital.space/doc/#/PlaybackEvents/createPlaybackEvent) (
6. Опубликовать автовебинар с помощью [updatePlayback](https://moodhood-api.livedigital.space/doc/#/Playbacks/updatePlayback).
7. Получить [eventId](https://moodhood-api.livedigital.space/doc/#/PlaybackEvents/getPlaybackEvents) и сформировать приглашение на автовебинар по форме "[https://edu.livedigital.space/event/ваш-eventID](https://edu.livedigital.space/event/%D0%B2%D0%B0%D1%88-eventID)”

#### **3.12. Триггер: начало звонка**

**Описание:**Добавить триггер начала звонка (call\_started) для парсинга и маппинга данных. Полученные данные можно использовать при загрузки аналитического отчёта по мероприятию.

**Функциональность:**

1. После старта звонка платформа отдает данные о звонке: eventName roomId spaceId callId.

####

#### **3.13. Триггер: конец звонка**

**Описание:**Добавить триггер окончания звонка (call\_finished) для парсинга и маппинга данных. Полученные данные можно использовать при загрузки аналитического отчёта по мероприятию.

**Функциональность:**

1. После завершения звонка платформа отдает данные о звонке: eventName, roomId, spaceId, callId.

#### **3.14. Триггер: начало записи звонка**

**Описание:**Добавить триггер старта записи звонка (record\_started) для парсинга и маппинга данных о записи. Полученные данные можно использовать при загрузки записи.

**Функциональность:**

1. После завершения записи звонка платформа отдает данные о звонке: eventName, recordId, roomId, spaceId, callId.

#### **3.15. Триггер: конец записи звонка**

**Описание:**Добавить триггер окончания записи звонка (record\_finished) для парсинга и маппинга данных о записи. Полученные данные можно использовать при загрузки записи.

**Функциональность:**

1. После завершения записи звонка платформа отдает данные о звонке: eventName, recordId, roomId, spaceId, callId.

#### **3.16. Добавить статусы ошибок для каждого действия и триггера**

**Описание:**Пользователю необходимо иметь информацию о том, почему так или иная связка не смогла запуститься.

**Функциональность:**

1. Добавить основные виды ошибок: Unauthorized (ошибка авторизации) и Invalid request params (невалидное тело запроса).

**Техническая составляющая

Создание Space и Room**

Получение списка Space <https://moodhood-api.livedigital.space/doc/#/Spaces/getSpacesList>

Создание Space <https://moodhood-api.livedigital.space/doc/#/Spaces/createSpace>

{

 "isPublic": true,

 "name": "Some space name", - **название комнаты**

 "description": "Space where my team will have all future calls",

 "logo": "string"

}



Удаление Space

<https://moodhood-api.livedigital.space/doc/#/Spaces/deleteSpace>

Получение списка комнат <https://moodhood-api.livedigital.space/doc/#/SpaceRooms/getRoomsList>

Создание комнаты
<https://moodhood-api.livedigital.space/doc/#/SpaceRooms/createRoom>

{

 "name": "string", - **название комнаты**

 "templateId": "string",

 "isPublic": true,

 "isChatAllowed": true,

 "isRecordAllowed": true,

 "isAutoRecordingAllowed": false, - **автоматическая запись**

 "isMicrophonePublishingAllowed": true,

 "isScreenMediaPublishingAllowed": true,

 "isCameraPublishingAllowed": true,

 "isAvatarsAllowed": true,

 "isSelfRenamingAllowed": true,

 "isHandRaisingAllowed": true,

 "isScreensharingAllowed": true,

 "type": "lesson", - **тип комнаты**

 "waitingRoomAudience": "nobody", - **зал ожидания**

 "parentRoomId": "60d55c0eb9ef88ab17b0aabb"

}


Обновление настроек комнаты <https://moodhood-api.livedigital.space/doc/#/SpaceRooms/updateRoom>

Удаление комнаты:

<https://moodhood-api.livedigital.space/doc/#/SpaceRooms/deleteRoom>

**Вебхуки**

1. По полученному токену от пользователя сервис подвязывает вебхуки на spaceId

[https://moodhood-api.livedigital.space/doc/#/Spaces/getSpacesList](https://moodhood-api.teleboss.ru/doc/#/Spaces/getSpacesList)
[https://moodhood-api.livedigital.space/doc/#/WebHook/CreateWebHook](https://moodhood-api.teleboss.ru/doc/#/WebHook/CreateWebHook)

2. После получения вебхука со Space о старте звонка мы получаем callId, roomId и spaceId и создаем запись в списке мероприятий со статусом “В процессе”

Пример вебхука:

{

 "signature": "d0f834f6b741a053ca266a31f49aea5a52b22c25a8ee1f5defb79ac5e65",

 "body": {

 "eventName": **"call\_started"/ "call\_finished"**,

 "roomId": "6694ea10eaaf52c5ca846225",

 "spaceId": "662cc96d46ac941cda787127",

 "callId": "66a234fdb37e8469c87cf622"

 }

}

3. Сервис получает отчет после окончания звонка используя метод и полученные идентификаторы

<https://moodhood-api.livedigital.space/doc/#/SpaceAnalytics/getRoomAnalyticsReport>

**Получение записей и ссылки на запись**1. При старт/стоп записи мы также получим вебхуки record\_started/record\_finished которые будут содержать маршрут до записи, а именно spaceId, roomId и callIdПример вебхука:

{

 "signature": "1f37f6f34afef89bbc7db107e513ffb6040552b07cd90171091c000ba45609c6",

 "body": {

 "eventName": **"record\_finished"/"record\_started"**,

 "roomId": "66d1d55d1cf5d9c045c0d1c4",

 "spaceId": "662cc96d46ac941cda787127",

 "recordId": "66d546c02b949f07df0e96ae",

 "callId": "66d546bf2b949f39f80e9697"

 }

}

2. Используя метод [**DownloadRoomRecord**](https://moodhood-api.teleboss.ru/doc/#/RoomRecords/RedirectToDownloadRoomRecord) мы можем предоставить пользователю ссылку на скачивание записи

[**https://moodhood-api.livedigital.space/doc/#/RoomRecords/RedirectToDownloadRoomRecord**](https://moodhood-api.teleboss.ru/doc/#/RoomRecords/RedirectToDownloadRoomRecord)

**Получение аналитического отчета**

После получения вебхука об окончании звонка, мы можем получить ссылку на скачивание XLSX отчета по идентификаторам в вебхуке

[https://moodhood-api.livedigital.space/doc/#/SpaceAnalytics/getRoomAnalyticsReport](https://moodhood-api.teleboss.ru/doc/#/SpaceAnalytics/getRoomAnalyticsReport)

Пример вебхука:

{

 "signature": "d0f834f6b741a053ca266a31f49aea5a52b22c25a8ee1f5defb79ac5e65",

 "body": {

 "eventName": **"call\_started"/ "call\_finished"**,

 "roomId": "6694ea10eaaf52c5ca846225",

 "spaceId": "662cc96d46ac941cda787127",

 "callId": "66a234fdb37e8469c87cf622"

 }

**Создание автовебинара**

1. Используется метод создания сущности автовебинара в API livedigital

<https://moodhood-api.livedigital.space/doc/#/Playbacks/createPlayback>

2. В настройках элемента пользователь указывает из какого space и room парсить запись для использования в качестве материала для автовебинара

1. <https://moodhood-api.livedigital.space/doc/#/Spaces/getSpacesList>
2. <https://moodhood-api.livedigital.space/doc/#/SpaceRooms/getRoomsList>
3. <https://moodhood-api.livedigital.space/doc/#/RoomRecords/get_spaces__spaceId__rooms__roomId__records>

После выбора записи подставляем ее id в автовебинар

1. <https://moodhood-api.livedigital.space/doc/#/Playbacks/updatePlayback>
2. После создания сущности необходимо задать время проведения автовебинара
<https://moodhood-api.livedigital.space/doc/#/PlaybackEvents/createPlaybackEvent>

 Request body для постоянного вебинара, который каждый раз открывается при нажатии на ссылку:

* {name: "Имя автовебинара", recurrence: null, startDate: null, endDate: null, timezone: "Europe/Moscow"}

Request body для однократно повторяющегося вебинара, который доступен только в определенное время

* endDate: null
* name: "Имя автовебинара"
* playbackId: "a06b59a7-8529-42f7-bb0e-52e571de3488"
* recurrence: null
* startDate: "2025-02-06T11:50:00.000Z"
* timezone: "Europe/Moscow"

**endDate** - дата завершения;

**name** - имя автовебинара;

**playbackId** - id записи;

**recurrence** - цикличность мероприятия, рекомендуем оставлять значение “null”

**startDate** - дата и время начала мероприятия в формате 2025-01-01T12:11:22.000Z

1. После окончания проведения автовебинара собираем аналитику методом
<https://moodhood-api.livedigital.space/doc/#/PlaybackAnalytics/getPlaybackSessionAnalyticsReport>