Бизнес логика с стороны менеджера.

1. Есть стороннее решение, в котором менеджер имеет возможность оформить заявку (частный случай бизнес задачи). Менеджер производит оформление заявки заполняя форму на сайте, форма отправляется на микросервис, который предоставляет для этого эндпоинт. Данные формы должны валидироваться на микросервисе так же в зависимости от атрибутов полей (required or not).
2. Микросервис принимает данные – валидирует и заполняет таблицу этими данными.
3. После вставки данных срабатывает событие ADD на таблице и данные должны отправляться на webhook для формирования пакета данных в формате json и отправки данных в 1С.
4. Данные обрабатываются 1С и присылаются обратно к нам и модифицируют определенное поле в таблице (номер заявки). Для избежания дублирования данных (должна быть проверка, что это поле уже заполнено для конкретной заявки).

Бизнес логика с стороны cron задач.

Есть список менеджеров (он будет динамический).

1. Продумать механизм создания таблиц для каждого из менеджеров в разрезе месяц / день.
2. Продумать механизм, который позволит создавать в конце рабочего дня новую таблицу на завтра с конкретными полями (работа с cron).
3. Продумать механизм создания нового документа, который будет идентифицировать месяц, в котором работает менеджер в данный момент.
4. Структура orgs/workspace/docs/tables
5. C звездочкой:

* Для менеджера блокируется таблица за прошлый день (может видеть, но менять данные не может), при этом есть админ, который может и видеть, и модифицировать данные в любой из таблиц у любого менеджера. Некоторые поля могут быть использованы для подсчета данных.
* Если другой менеджер заходит не в свою таблицу (он может видеть таблицы другого менеджера) и модифицирует данные, должна быть возможность отследить это событие – либо через webhook, либо внутренний механизм Grist tables.