*Перед тем, как начинать работу Анна Александровна посоветовала прочитать о таких вещах:*

*- ER*

*-Crow Foot*

*-Primary Key*

*-First Key*

*- Составной ключ*

ЧТО НЕОБХОДИМО ИЗМЕНИТЬ:

*Примечание: 1)Связываем ID// это наверно уже больше замечание для меня*

*2) тип данных memo=Text в некоторых программах*

1. **Разорвать связь между таблицами СИТУАЦИЯ и ПОЛЬЗОВАТЕЛЬ (ситуации и пользователи не связаны, неудачный скриншот).**
2. **Таблица ПОЛЬЗОВАТЕЛЬ**:

* добавить поле active (bool тип данных) – это поле показывает, что пользователь активен. ТО есть, например, он уволился и эту должность больше не занимает. Сделано! Для пользователей поле active, для персонала если дата увольнения заполнена, то значит человек уволен с данной должности, необходимости active нет.
* добавить поле pas либо password (для хранения хэша пароля) Сделано!
* добавить связь с таблицей CITY (Связь была)

1. **Таблица STAFF**:

* Убрать id INT (если его убрать получится составной ключ из двух полей и эти два поля будут “прилетать в таблицу заявки” где итак много полей). Но есть и плюс в таблице заявки будет храниться сразу и ссылка на отвественного и ссылка на его должность. Тут нужно подумать.
* Ключом сделать user\_id INT (нельзя, т.к. вдруг у нас один человек может работать на двух должностях или в двух подразделениях, поэтому обязательно должен быть ключом и user\_id и position\_id).

1. **Таблица ЗАЯВКА**:

* gps NVARCHAR (1000) заменить на два поля latitute VARCHAR и longitute VARCHAR
* executor переименовать на executor\_id
* location – какое его назначение? Не дублирует ли оно поле address и gps
* content заменить на тип memo (так как может не хватить)
* добавить поле city\_id (связать с таблицей Cities)
* добавить поле active (проверяем обработана заявка или нет)

1. **Таблица ApplicationType**:

* добавить поле active (возможно дли для выбора тип заявки, например) ??

1. **Таблица STATUS**:

* добавить поле active (через 2 недели может быть необработано) ??

1. **Таблица Application\_has\_status**:

* Id INT убрать (связка по полю) (составной ключ не совсем удобно)
* Добавить staff\_id (связано с Таблицей STAFF) Сделано!

1. **Таблица Categories**:

* добавить поле active (зачем?)

1. **Таблица Application\_has\_solution**:

* Id INT убрать (связка по полю) (составной ключ не совсем удобно)
* Добавить staff\_id (связано с Таблицей STAFF) Сделано!
* Rating INT убрать (оценка для конкретной заявки и решения а не в целом для решения, поэтому стоит отсавить тут)

1. **Таблица solutions**:

* Rating INT добавить (нет, так как решения это набор готовых решений. И некоторые решения могут использоватся по несколько раз (я так предполагал). Если все таки какждое решение в таблице Solution будет использоваться только по одному разу, то тогда целесообразно перенести rating в Solutions. ВОПРОС: Решение в таблице Solutions индивидуально для каждой заявки или в таблице могут содержаться общие решения (например «По вашему запросу проведена проверка в соответсвии со статьями раз два и три.»?
* Content Изменить тип данных на memo (Не нашел такого типа данных в MySQL WorkBench)

1. **Таблицу** надо подумать как и какую создать, чтобы показать **вход через VK, yandex, mail.ru** и другие соцсети Сделано (таблица social\_services).

* Связать с таблицей Пользователь
* Добавить таблицу с полями soc\_type, soc\_id + конечно, подумать над другими полями

В пользователей нужно ли добавить булевое поле social\_service (если true, то аутентифицироваться пользователь должен через соцсеть и хэш пароля на сайте должен остаться пустым, так как аутентификация будет происходить через соцсеть)?

PS: прилагаю фото таблицы с исправлениями