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

*- ER*

*-Crow Foot*

*-Primary Key*

*-First Key*

*- Составной ключ*

**ССЫЛКА НА GITHUB:** [**https://github.com/solopchenko/tom**](https://github.com/solopchenko/tom)

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 УБРАЛ
* 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).

Так и пусть будет составной ключ: user\_id и position\_id, просто ID – уже дублирование, его убрать СДЕЛАЛ!

1. **Таблица ЗАЯВКА**:

* gps NVARCHAR (1000) заменить на два поля latitute VARCHAR и longitute VARCHAR Сделал! то же самое сделал в организациях
* executor переименовать на executor\_id Сделал!
* location – какое его назначение? Не дублирует ли оно поле address и gps Согласен. Удалил.
* content заменить на тип memo (так как может не хватить) (Не нашел такой тип в MySQL Workbech). Это на случай, если 2000 символов не хватит, может есть тип поле text или что-то подобное (УБРАЛ ТАК КАК ЕСТЬ СВЯЗЬ С ТАБЛИЦЕЙ ФАЙЛОВ)
* добавить поле city\_id (связать с таблицей Cities) Сделал!
* добавить поле active (проверяем обработана заявка или нет) (а нельзя ее статус проверить?)

Если active=1, то не надо проверять статус ЖЦ, который может быть разным (Новая, в обработке, ожидает решения и т.д.). Тогда нужно обращаться только к 1ой таблице заявок, и не трогать связанную с ней – меньше запросов.

Кроме того, надо учитывать, что система управления должна позволять задавать различные статусы (ЖЦ, категории и т.д.), причем некоторые из них сейчас «включены» - доступны для выбора, а какие-то - нет.

1. **Таблица ApplicationType**:

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

Представим, что сейчас у нас 2 типа заявок – проблема и хорошая новость, но может появиться третий тип, например, благодарность, или жалоба на несправедливость мира. Возникла заявка с этим типом «жалоба на несправедливость мира», прошла свой ЖЦ полностью (ЖЦ – это жизненный цикл), а потом мы исключили этот тип заявок. Так что же будет с теми заявками, которые уже есть в системе? Поэтому система управления должна позволять задавать различные статусы (ЖЦ, категории и т.д.), причем некоторые из них сейчас «включены» - доступны для выбора, а какие-то - нет. СДЕЛАЛ

1. **Таблица STATUS**:

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

Представим, что сейчас у нас 7этапов ЖЦ заявок, но может появиться временный восьмой, например, «ждем Иван Петровича, специалиста по решению всех задач». И этот статус ЖЦ актуален на период с 2 июля по 10 августа для заявок категории «дороги». Поскольку наша система управления будет позволять настраивать статусы ЖЦ в том числе, мы можем включать и выключать различные статусы тогда, когда это необходимо. СДЕЛАНО

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

* Id INT убрать (связка по полю) (составной ключ не совсем удобно) почему? Сделал составной ключ (ключевыми сделать точно application\_id и status\_id, еще сделал дату ключевым)
* Добавить staff\_id (связано с Таблицей STAFF) Сделано!

1. **Таблица Categories**:

* добавить поле active

Представим, что сейчас у нас 12 категорий заявок, но может появиться еще 13 и 14, временная, актуальная в период летнего сезона. Возникла заявка с этой категорией, прошла свой ЖЦ полностью (ЖЦ – это жизненный цикл), а потом мы исключили эту категорию заявок. Так что же будет с теми заявками, которые уже есть в системе? Поэтому система управления должна позволять задавать различные статусы (ЖЦ, категории и т.д.), причем некоторые из них сейчас «включены» - доступны для выбора, а какие-то - нет. Сделано!

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

* Id INT убрать (связка по полю) (составной ключ не совсем удобно) почему? Сделал составной ключ
* 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)

Логично, что решение будет уникальным для каждой заявки, т.к. каждая заявка уникальна. Можно посмотреть процедуру обработки официальных обращений граждан – ведомство обязано ответить письменно на каждый запрос не позднее чем в 30-дневный срок. Поэтому поле Rating INT перенести в таблицу Solution.

Кстати, подумайте – может, сделать и рейтинг у заявок, чтобы их могли + и – другие юзеры, чтобы заявки с высоким рейтингом обрабатывались быстрее (больший приоритет)? Ведь может так быть, что заявку плюсанут +100500,, а решение заминусуют. ПЕРЕНЕС В SOLUTIONS

Мб стоит перенести привязку пользователя из администрации (staff\_user\_id и staff\_position\_id) в таблицу Solutions, а не держать в таблице Application\_has\_solution, раз каждое решение у нас уникально.

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.»?

Логично, что решение будет уникальным для каждой заявки, т.к. каждая заявка уникальна. Можно посмотреть процедуру обработки официальных обращений граждан – ведомство обязано ответить письменно на каждый запрос не позднее чем в 30-дневный срок. Поэтому поле Rating INT перенести в таблицу Solution.

Кстати, подумайте – может, сделать и рейтинг у заявок, чтобы их могли + и – другие юзеры, чтобы заявки с высоким рейтингом обрабатывались быстрее (больший приоритет)? Ведь может так быть, что заявку плюсанут +100500,, а решение заминусуют. СДЕЛАНО

* Content Изменить тип данных на memo (Не нашел такого типа данных в MySQL WorkBench) Это на случай, если 2000 символов не хватит, может есть тип поле text или что-то подобное (УБРАЛ ТАК КАК ЕСТЬ СВЯЗЬ С ТАБЛИЦЕЙ ФАЙЛОВ)

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)?

Для реализации нескольких способов доступа, булевое поле social\_service НЕ НАДО!!! ОК

поле name – правильнее назвать oauth\_provider,

и нужен идентификатор пользователя в этом провайдере - oauth\_id. Прочитайте про OAuth – https://ru.wikipedia.org/wiki/OAuth

Просьба посмотреть поля какие должны быть null, какие не null.

PS: прилагаю фото таблицы с исправлениями