Таблица локальных пользователей (condemned) – информация об осужденном

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Название** | **Тип** | **Описание** |
| 1 | id | Int (PK) |  |
| 2 | name | String | Имя(ник) пользователя |
| 3 | pin | String | Пин код (цифры) |
| 4 | token | String | Токен для доступа к серверу |

Таблица осужденный (condemned) – информация об осужденном

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Название** | **Тип** | **Описание** |
| 1 | id | Int (PK) |  |
| 2 | updated\_at | String | Дата последнего обновления в формате ISO 8601. Изначально получается при создании в приложении или при загрузке из базы данных сервера |
| 3 | name | String |  |
| 4 | family | String |  |
| 5 | patronymic | String |  |
| 6 | birthday | String | Дата в формате dd-MM-Y (дд-мм-гггг) |
| 7 | gender | Byte | Пол 1 – мужчина, 2 – женщина, 0 – отсутствует |
| 8 | illness\_id | Int | Наличие заболеваний, которые нужно лечить по УДО  0 – отсутствие  Иначе код/наличие |
| 9 | info | String | Дополнительная информация |
| 10 | nick | String | Псевдоним для осужденного |
| 11 | server\_id | Int | Первичный ключ на сервере (0, если запись не загружалась) |
| 12 | local\_user\_id | Int | id локального пользователя |

Таблица для всего УДО (freedom)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Название** | | **Тип** | **Описание** | |
| 1 | id | | INT (PK) |  | |
| 2 | updated\_at | | String | Дата последнего обновления в формате ISO 8601. Изначально получается при создании в приложении или при загрузке из базы данных сервера | |
| 3 | condemned\_server\_id | | FK | Карточка осужденного, внешний ключ на condemned.server\_id | |
| 4 | info | | String | Дополнительная информация по УДО? | |
| 5 | slug | | String |  | |
| 6 | enddate | | String | Дата в формате ISO 8601? | |
| 7 | server\_id | | Int | Первичный ключ на сервере (0, если запись не загружалась) | |
| 8 | status | INT | | | Статус УДО. 1 – доступно для редактирования, 2 – заблокировано, 3 - закрыто |

Каждая отметка относится к таблице точек (point)

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Название** | **Тип** | **Описание** |
| 1 | id | INT (PK) |  |
| 2 | updated\_at | String | Дата последнего обновления в формате ISO 8601. Изначально получается при создании в приложении или при загрузке из базы данных сервера |
| 3 | freedom\_server\_id | FK | Таблица УДО, внешний ключ на freedom.server\_id |
| 4 | startdate | String | Дата/время в формате ISO 8601? |
| 5 | enddate | String | Дата/время в формате ISO 8601? |
| 6 | node\_id | FK | Узел, внешний ключ на node.server\_id |
| 7 | status | Byte | Статус точки (просрочка, выполнен и т.д.) |
| 8 | info | String | Комментарий от инспектора к узлу |
| 9 | server\_id | Int | Первичный ключ на сервере (0, если запись не загружалась) |

Узлы (node) – по сути пресеты контента для точек

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Название** | **Тип** | **Описание** |
| 1 | id | INT (PK) |  |
| 2 | updated\_at | String | Дата последнего обновления в формате ISO 8601. Изначально получается при создании в приложении или при загрузке из базы данных сервера |
| 3 | title | String | Название |
| 4 | description | String | Описание узла |
| 5 | type | INT | Тип узла (информация, контрольная точка и т.д.) – реализовать через строку |
| 6 | default\_length | INT | Количество в днях на выполнение узла по дефолту |
| 7 | content | String | Контент для узла |
| 8 | server\_id | Int | Первичный ключ на сервере (0, если запись не загружалась) |

Внимательно следить за связями между записями (связь идёт по ключу server\_id, а не по id)!