

# Проектирование БД

## Система поиска потерянных вещей

### Роли и сценарии использования

1. Суперадминистратор (единственное право — генерировать инвайт-коды для создания аккаунтов администраторов). Аккаунт создаётся отдельно и может быть только один. Как root в linux
2. Администратор сервиса (модерация объявлений, например)
3. Администрация школы (директор, заместитель директора)
4. Сотрудники школы (повар, охранник, уборщик, дворник)
5. Учителя
6. Родители
7. Ученики

Также:

1. Аккаунты пользователей (кроме суперадминистратора) создаются только по одноразовым инвайт-ссылкам — каждая ссылка индивидуальна. Выпускаются инвайт-коды администратором сервиса и позволяют создать аккаунт только с определённой ролью. Благодаря такому способу создания аккаунтов мы считаем все существующие аккаунты уже верифицированными, т.е. что за каждым аккаунтом стоит человек, указанный в нём
2. Использоваться будет в ОУ (школы, вузы). Такое разделение ролей — это, в основном, задел на будущее.
3. Восстановить пароль от аккаунта суперадминистратора невозможно. Если его забыть, придётся вручную удалять аккаунт из БД и затем создавать новый через сайт

### Что может делать Суперадминистратор

1. Только генерировать инвайт-коды для регистрации администраторов сервиса

### Что могут делать Администраторы сервиса

1. Просмотр, удаление объявлений
2. Модерация объявлений (отклонение/принятие)
3. Генерация инвайт-кодов для создания любых пользователей, кроме администраторов сервиса
4. Удаление пользователей

### Что может делать Администрация школы

1. Просмотр объявлений
2. Публикация, редактирование, удаление своих объявлений
3. Сбрасывать пароль от своего аккаунта

### Что могут делать Сотрудники школы

1. Просмотр объявлений
2. Публикация, редактирование, удаление своих объявлений
3. Сбрасывать пароль от своего аккаунта

### Что могут делать Учителя

1. Просмотр объявлений
2. Публикация, редактирование, удаление своих объявлений
3. Сбрасывать пароль от своего аккаунта

### Что могут делать Родители

1. Просмотр объявлений
2. Публикация, редактирование, удаление своих объявлений
3. Привязывать/отвязывать детей к/от своему(-го) аккаунту(-а)
4. Редактировать своих детей

5. Просмотр объявлений своих детей, классов своих детей
6. Сбрасывать пароль от своего аккаунта

### **Что могут делать Ученики**

1. Просмотр объявлений
2. Публикация, редактирование, удаление своих объявлений
3. Просмотр объявлений своих родителей, своего класса
4. Сбрасывать пароль от своего аккаунта

### **Таблицы для пользователей (подобие RBAC)**

1. `superadmin` — логин и хеш пароля единственного аккаунта суперадминистратора
2. `user` — адреса электронной почты и хеши паролей всех пользователей, кроме суперадминистратора
3. `role` — роли пользователей (администратор сервиса, администрация школы, сотрудник и т.д.)
4. `permission` — права доступа (`can_view_posts` — право просматривать все объявления, `can_add_post` — право создать своё объявление, `can_accept_post` — право принять объявление (у модератора, напр.), `can_reject_post` — право отклонить объявление, `can_create_invite_code` — право создать приглашительный код (только для администратора сервиса) и т.д.)
5. `role_permission` — связь ролей и прав (связующая таблица; многие ко многим)
6. `user_role` — связь пользователей и их ролей (связующая таблица; многие ко многим)
7. `school_administrator` — данные представителей администрации школы, связь с `user` один к одному
8. `staff` — данные сотрудников школы, связь с `user` один к одному
9. `room` — аудитории, учебные кабинеты
10. `students_group` — классы/группы учеников/студентов (1А, 1А гр. 1, гр. 1234)
11. `subject` — преподаваемые предметы
12. `teacher` — данные учителей, связь с `user`, `room` и `students_group` — один-к-одному
13. `teacher_subject` — предметы, преподаваемые учителем; связь `teacher-subject` многие-ко-многим
14. `student` — данные учеников, связь с `user` один к одному, связь `student-students_group` многие-к-одному
15. `parent` — данные родителей, связь с `user` один к одному.
16. `parent_student` — связующая таблица для родителей и их детей, связь `parent-student` многие-ко-многим (двое родителей могут иметь несколько детей)

### **Остальные таблицы**

1. `post` — объявление о потерянной вещи. Здесь `pk` — `UUID` (для возможности использования в ссылке на странице статуса вещи), `fk` на `user.id` (автор)
2. `invite_code` — приглашительный код. Ссылка вида `https://domain.ru/invite?code=<UUID>`, где `UUID` — это код (`pk` в БД). В таблице коду соответствует роль, которая назначается аккаунту, созданному с помощью этого кода. Для генерации кода достаточно добавить в таблицу `role_id` и получить сгенерированный РК новой записи, т.е. сам код. После использования кода запись с ним удаляется из БД: `DELETE FROM invite_code WHERE code='<UUID>';`