Спроектировать и разработать систему регистрации и обработки пользовательских заявок.

Пользователь посредством системы может подавать заявки оператору на рассмотрение.

Оператор может просматривать пользовательские заявки и принимать или отклонять их.

Администратор управляет правами доступа.

• Спроектировать и разработать REST back-приложение

• Спроектировать и разработать Базу данных

Функции приложения

• Создать заявку (Заявка помимо прочих системных полей состоит из статуса и текстового обращения пользователя)

• Отправить заявку оператору на рассмотрение

• Просмотреть список отправленных на рассмотрение заявок, отсортированных по дате создания

• Посмотреть заявку

• Принять заявку

• Отклонить заявку

• Просмотреть список пользователей

• Назначить права оператора

В системе предусмотрены 3 роли:

• Пользователь

• Оператор

• Администратор

У пользователя системы может быть одновременно несколько ролей, например, «Оператор» и «Администратор».

У заявки пользователя предусмотрено 4 состояния:

• черновик

• отправлено

• принято

• отклонено

Пользователь может

• создавать заявки

• просматривать созданные им заявки

• редактировать созданные им заявки в статусе «черновик»

• отправлять заявки на рассмотрение оператору.

Пользователь НЕ может:

• редактировать отправленные на рассмотрение заявки

• видеть заявки других пользователей

• принимать заявки

• отклонять заявки

• назначать права

• смотреть список пользователей

Оператор может

• Просматривать отправленные на рассмотрение заявки

• Принимать заявки

• Отклонять заявки

Оператор НЕ может

• создавать заявки

• просматривать заявки в статусе отличном от «отправлено»

• редактировать заявки

• назначать права

Администратор может

**•** смотреть список пользователей

• назначать пользователям права оператора

Администратор НЕ может

• создавать заявки

• просматривать заявки

• редактировать заявки

• принимать заявки

• отклонять заявки

Технические требования к приложению

• Java 1.8

• Использовать архитектуру REST

• Использовать Spring Boot

• Использовать Spring Security

• Использовать Hibernate

• Использовать реляционную БД (MS SQL, PostgreSQL, h2),

скрипт базы данных должен быть приложен к исходникам в системе контроля версии

• Создание пользователей и ролей не предусмотрено в этой системе.

Подразумевается, что данные об учетных записях пользователей и роли уже есть в БД

• В случае просмотра заявки оператором текст заявки выводить со знаком <-> после каждого символа.

Пример: Пользователь отправил на рассмотрение заявку с текстом «Мне нужна помощь»,

а оператор на экране видит текст в формате «М-н-е- -н-у-ж-н-а- -п-о-м-о-щ-ь».