

Руководство для системного программиста

Система учета заявок на ремонт бытовой техники

1. Назначение системы Система предназначена для автоматизации учета заявок на ремонт бытовой техники (холодильники, стиральные машины, телевизоры, микроволновые печи, пылесосы и др.). Она позволяет:
 - Автоматизировать процесс регистрации и обработки заявок
 - Контролировать выполнение работ на всех этапах
 - Управлять пользователями с разными уровнем доступа
 - Анализировать статистику и эффективность работы сервисной службы
 - Обеспечивать обратную связь с клиентами через QR-коды

2. Установка и настройка

2.1. Установка зависимостей:

Создание виртуального окружения

```
python -m venv venv
```

Активация (Windows)

```
venv\Scripts\activate
```

Активация (Linux/macOS)

```
source venv/bin/activate
```

Установка зависимостей

```
pip install -r requirements.txt
```

2.2. Настройка базы данных:

Создайте базу данных:

«python load_data.py» в консоль

И затем выбрать из перечисленного:

1. Создать новую базу данных и загрузить данные
2. Только загрузить данные из CSV файлов
3. Только создать примеры CSV файлов
4. Проверить целостность базы данных
5. Создать резервную копию базы данных
6. Выполнить все операции (создание + загрузка + проверка + резервная копия)
7. Выход

2.3. Запуск системы:

Запуск backend

uvicorn main:app --reload

Запуск frontend

streamlit run frontend.py

3. Управление пользователями и ролями

3.1. Роли пользователей:

Роль	Права
Заказчик	Создание и просмотр своих заявок, просмотр комментариев
Оператор	Просмотр всех заявок, создание и редактирование заявок
Мастер	Просмотр назначенных заявок, добавление комментариев, обновление статуса
Менеджер	Все права оператора + управление пользователями, удаление заявок
Менеджер по качеству	Просмотр всех заявок, назначение специалистов, продление сроков

3.2. Создание пользователей:

1. Войдите под ролью Менеджер (логин: `kasoo`, пароль: `root`)
2. Перейдите в раздел "Пользователи"
3. Нажмите "Новый пользователь"
4. Заполните форму:

- ФИО
- Телефон
- Логин (уникальный)
- Пароль
- Роль пользователя

3.3. Тестовые учетные записи:

Логин	Пароль	Роль
kasoo	root	Менеджер
test1	test1	Мастер
login3	pass3	Заказчик
admin	admin	Менеджер по качеству

4. Администрирование базы данных

«python load_data.py» в консоль

И затем выбрать из перечисленного:

1. Создать новую базу данных и загрузить данные
2. Только загрузить данные из CSV файлов

3. Только создать примеры CSV файлов
4. Проверить целостность базы данных
5. Создать резервную копию базы данных
6. Выполнить все операции (создание + загрузка + проверка + резервная копия)
7. Выход

5. Безопасность

1. Измените стандартные пароли перед использованием в production
2. Используйте HTTPS для защищенного соединения
3. Настройте брандмауэр для ограничения доступа к портам 8000 и 5000
4. Регулярно обновляйте зависимости:

6. Устранение неисправностей

6.1. Распространенные проблемы:

Проблема	Решение
Ошибка подключения к базе данных	Проверьте PostgreSQL и строку подключения
Модуль не найден	Проверьте зависимости
Ошибка 500 при создании заявки	Проверьте обязательные поля
QR-код не генерируется	Установите qrcode: pip install qrcode[pil]

6.2. Проверка работоспособности:

1. Откройте <http://localhost:5000>
2. Войдите под тестовым пользователем
3. Создайте тестовую заявку
4. Проверьте статистику в разделе "Статистика"
5. Проверьте генерацию QR-кода

7. Обновление системы

7.1. Процесс обновления:

1. Создайте резервную копию базы данных
2. Остановите работающие сервисы
3. Обновите код из репозитория
4. Обновите зависимости:

```
pip install --upgrade -r requirements.txt
```

5. Запустите миграции БД (если есть)
6. Запустите сервисы

7.2. Проверка после обновления:

- Все ли модули импортируются без ошибок
- Работает ли аутентификация
- Корректно ли отображаются данные
- Генерируется ли QR-код

8. Контакты для поддержки

При возникновении проблем:

1. Проверьте логи приложения
2. Проверьте подключение к базе данных
3. Убедитесь, что все зависимости установлены

Для технической поддержки обращайтесь к разработчикам системы.