### Документация для разработчика

#### Проект: Программный модуль для учета заявок на ремонт автомобилей

---

### 1. Введение

Проект включает разработку веб-приложения для учета заявок на ремонт автомобилей. Приложение позволяет добавлять, редактировать, отслеживать и управлять заявками на ремонт, а также назначать ответственных за выполнение работ.

### 2. Требования к окружению

- Python 3.x

- Flask

- SQLite

### 3. Структура проекта

```

auto\_repair/

├── app.py

├── templates/

│ ├── base.html

│ ├── index.html

│ ├── add\_request.html

│ ├── edit\_request.html

│ ├── view\_request.html

├── static/

│ ├── css/

│ │ ├── styles.css

├── auto\_repair.db

├── README.md

```

### 4. Установка и настройка

1. \*\*Клонирование репозитория:\*\*

```bash

git clone <repository\_url>

cd auto\_repair

```

2. \*\*Создание виртуального окружения и установка зависимостей:\*\*

```bash

python -m venv venv

source venv/bin/activate # On Windows use `venv\Scripts\activate`

pip install Flask

```

3. \*\*Инициализация базы данных:\*\*

Создайте файл `auto\_repair.db` и выполните следующую команду для создания таблицы:

```python

import sqlite3

conn = sqlite3.connect('auto\_repair.db')

c = conn.cursor()

c.execute('''

CREATE TABLE requests (

id INTEGER PRIMARY KEY AUTOINCREMENT,

request\_number TEXT NOT NULL,

date\_added TEXT NOT NULL,

car\_type TEXT NOT NULL,

car\_model TEXT NOT NULL,

problem\_description TEXT,

client\_name TEXT NOT NULL,

phone\_number TEXT NOT NULL,

status TEXT NOT NULL,

assigned\_mechanic TEXT

)

''')

conn.commit()

conn.close()

```

4. \*\*Запуск приложения:\*\*

```bash

flask run

```

### 5. Основные маршруты

- \*\*`/`\*\* (GET): Главная страница со списком всех заявок.

- \*\*`/add`\*\* (GET, POST): Добавление новой заявки.

- \*\*`/edit/<int:request\_id>`\*\* (GET, POST): Редактирование существующей заявки.

- \*\*`/view/<int:request\_id>`\*\* (GET): Просмотр подробной информации о заявке.

### 6. Шаблоны

Шаблоны HTML используют Jinja2 для динамического рендеринга содержимого.

#### `base.html`

Базовый шаблон, включающий общий каркас страницы и подключение стилей.

```html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Auto Repair Requests</title>

<link rel="stylesheet" href="{{ url\_for('static', filename='css/styles.css') }}">

</head>

<body>

<header>

<h1>Auto Repair Requests</h1>

<nav>

<a href="{{ url\_for('index') }}">Home</a>

<a href="{{ url\_for('add\_request') }}">Add Request</a>

</nav>

</header>

<main>

{% block content %}{% endblock %}

</main>

<footer>

<p>&copy; 2024 Auto Repair Service</p>

</footer>

</body>

</html>

```

#### `index.html`

Шаблон для отображения списка всех заявок.

```html

{% extends "base.html" %}

{% block content %}

<h2>All Requests</h2>

<table>

<thead>

<tr>

<th>Request Number</th>

<th>Client Name</th>

<th>Car Model</th>

<th>Status</th>

<th>Actions</th>

</tr>

</thead>

<tbody>

{% for request in requests %}

<tr>

<td>{{ request['request\_number'] }}</td>

<td>{{ request['client\_name'] }}</td>

<td>{{ request['car\_model'] }}</td>

<td>{{ request['status'] }}</td>

<td>

<a href="{{ url\_for('view\_request', request\_id=request['id']) }}">View</a>

<a href="{{ url\_for('edit\_request', request\_id=request['id']) }}">Edit</a>

</td>

</tr>

{% endfor %}

</tbody>

</table>

{% endblock %}

```

#### `add\_request.html`

Шаблон для добавления новой заявки.

```html

{% extends "base.html" %}

{% block content %}

<h2>Add New Request</h2>

<form method="POST">

<!-- Поля для добавления заявки -->

<label for="request\_number">Request Number:</label>

<input type="text" id="request\_number" name="request\_number" required>

<label for="date\_added">Date Added:</label>

<input type="date" id="date\_added" name="date\_added" required>

<label for="car\_type">Car Type:</label>

<input type="text" id="car\_type" name="car\_type" required>

<label for="car\_model">Car Model:</label>

<input type="text" id="car\_model" name="car\_model" required>

<label for="problem\_description">Problem Description:</label>

<textarea id="problem\_description" name="problem\_description"></textarea>

<label for="client\_name">Client Name:</label>

<input type="text" id="client\_name" name="client\_name" required>

<label for="phone\_number">Phone Number:</label>

<input type="text" id="phone\_number" name="phone\_number" required>

<button type="submit">Add Request</button>

</form>

{% endblock %}

```

#### `edit\_request.html`

Шаблон для редактирования заявки.

```html

{% extends "base.html" %}

{% block content %}

<h2>Edit Request</h2>

<form method="POST">

<label for="request\_number">Request Number:</label>

<input type="text" id="request\_number" name="request\_number" value="{{ request['request\_number'] }}" required>

<label for="date\_added">Date Added:</label>

<input type="date" id="date\_added" name="date\_added" value="{{ request['date\_added'] }}" required>

<label for="car\_type">Car Type:</label>

<input type="text" id="car\_type" name="car\_type" value="{{ request['car\_type'] }}" required>

<label for="car\_model">Car Model:</label>

<input type="text" id="car\_model" name="car\_model" value="{{ request['car\_model'] }}" required>

<label for="problem\_description">Problem Description:</label>

<textarea id="problem\_description" name="problem\_description">{{ request['problem\_description'] }}</textarea>

<label for="client\_name">Client Name:</label>

<input type="text" id="client\_name" name="client\_name" value="{{ request['client\_name'] }}" required>

<label for="phone\_number">Phone Number:</label>

<input type="text" id="phone\_number" name="phone\_number" value="{{ request['phone\_number'] }}" required>

<label for="status">Status:</label>

<select id="status" name="status">

<option value="new" {% if request['status'] == 'new' %}selected{% endif %}>New</option>

<option value="in\_progress" {% if request['status'] == 'in\_progress' %}selected{% endif %}>In Progress</option>

<option value="waiting\_for\_parts" {% if request['status'] == 'waiting\_for\_parts' %}selected{% endif %}>Waiting for Parts</option>

<option value="ready\_for\_pickup" {% if request['status'] == 'ready\_for\_pickup' %}selected{% endif %}>Ready for Pickup</option>

<option value="completed" {% if request['status'] == 'completed' %}selected{% endif %}>Completed</option>

</select>

<label for="assigned\_mechanic">Assigned Mechanic:</label>

<input type="text" id="assigned\_mechanic" name="assigned\_mechanic" value="{{ request['assigned\_mechanic'] }}">

<button type="submit">Update Request</button>

</form>

{% endblock %}

```

#### `view\_request.html`

Шаблон для просмотра заявки.

```html

{% extends "base.html" %}

{% block content %}

<h2>Request Details</h2>

<ul>

<li><strong>Request Number:</strong> {{ request['request\_number'] }}</li>

<li><strong>Date Added:</strong> {{ request['date\_added'] }}</li>

<li><strong>Car Type:</strong> {{ request['car\_type'] }}</li>

<li><strong>Car Model:</strong> {{ request['car\_model'] }}</li>

<li><strong>Problem Description:</strong> {{ request['problem\_description'] }}</li>

<li><strong>Client Name:</strong> {{

request['client\_name'] }}</li>

<li><strong>Phone Number:</strong> {{ request['phone\_number'] }}</li>

<li><strong>Status:</strong> {{ request['status'] }}</li>

<li><strong>Assigned Mechanic:</strong> {{ request['assigned\_mechanic'] }}</li>

</ul>

<a href="{{ url\_for('edit\_request', request\_id=request['id']) }}">Edit</a>

{% endblock %}

```

### 7. Поддержка и развитие

Для поддержки и дальнейшего развития проекта необходимо:

- Добавить тестирование для основных функций.

- Расширить функциональность согласно новым требованиям.

- Оптимизировать производительность и улучшить пользовательский интерфейс.

### 8. Контакты

Если у вас есть вопросы или предложения по улучшению проекта, свяжитесь с командой разработки.

---

Эта документация предоставляет основные сведения, необходимые для понимания и работы с проектом. Если у вас есть дополнительные вопросы или предложения, не стесняйтесь задавать их.