Ecorevolt Developer Guide

Introduction

Welcome to the developer guide for Ecorevolt! This guide provides a comprehensive overview of the project's architecture, technologies used, and guidelines for contributing to its development. Ecorevolt is currently 40-50% completed, utilizing Django for the backend, HTML, CSS, JavaScript for frontend development, and SQLite3 for database management.

Project Overview

Ecorevolt is a web platform aimed at connecting environmental projects with potential investors. It facilitates the submission and management of project proposals by creators and enables investors to discover, evaluate, and fund projects aligned with their interests.

Technologies Used

- **Backend**: Django framework (Python-based) for handling server-side logic, routing, authentication, and database management using SQLite3.
- **Frontend**: HTML for structure, CSS for styling, and JavaScript for interactive features and dynamic content.
- **Database**: SQLite3 for local development and testing. Consider migration to a more robust database system like PostgreSQL or MySQL for production.

Development Environment Setup

The setup is already mentioned in readme.md

Project Structure

- **ecorevolt/**: Django project directory.
    - **ecorevolt/settings.py**: Configuration settings for Django project.
    - **ecorevolt/urls.py**: URL routing configuration.

- **ecorevolt/views.py**: Views handling HTTP requests.
- **ecorevolt/templates/**: HTML templates for frontend rendering.
- **ecorevolt/static/**: Static files (CSS, JavaScript, images).
- **ecorevolt/models.py**: Database models using Django's ORM.

Contribution Guidelines

1. **Version Control**:
    - Fork the repository and create a new branch for each feature or bug fix.
    - Commit changes with clear and descriptive messages.
2. **Code Style**:
    - Follow PEP 8 guidelines for Python code.
    - Maintain consistency in HTML structure, CSS classes, and JavaScript functions.
3. **Testing**:
    - Write unit tests for backend functionality using Django's testing framework.
    - Test frontend interactions and responsiveness across different devices and browsers.
4. **Documentation**:
    - Document new features, APIs, and significant changes in README or inline comments.
    - Update this developer guide as necessary to reflect changes and additions.
5. **Code Reviews**:

- Submit pull requests for review, addressing any feedback or suggestions.

Future Development Roadmap

As Ecorevolt progresses towards completion, future development focuses on:

- Enhancing user authentication and authorization mechanisms.
- Implementing real-time notifications and messaging features.
- Optimizing database performance and scalability.
- Integrating external APIs for enhanced functionality (e.g., payment gateways).