**Sustainability Gamification App - System Architecture Document**

**Overview**

The **Sustainability Gamification App** is a Django-based web application designed to encourage users to adopt environmentally friendly habits through gamification. The system allows users to earn points for completing sustainable actions, engaging with quizzes, and visiting verified eco-friendly locations.

**System Architecture**

The application follows a **Model-View-Template (MVT)** architecture, which is Django’s adaptation of the **Model-View-Controller (MVC)** design pattern.

**1. Main Components**

1. **Frontend (Templates & Static Files)**
   * HTML, JavaScript for UI elements.
   * Uses Django templates to dynamically generate pages.
   * Basic user interactions (e.g., quiz answering, check-ins) handled with forms and JavaScript.
2. **Backend (Django Framework)**
   * Handles authentication, business logic, and point tracking.
   * Processes user interactions with gamification elements.
   * Uses Django’s built-in ORM to manage data interactions.
3. **Database (SQLite, can be replaced with PostgreSQL in production)**
   * Stores user profiles, challenges, leaderboard rankings, and check-ins.
   * Uses Django’s ORM for database management and migrations.
4. **Authentication & User Management**
   * Uses Django’s built-in authentication system for secure login and registration.
   * Custom user profiles extend Django’s User model to store points and sustainability achievements.
5. **Gamification & Leaderboard System**
   * Users earn points by completing quizzes, checking into sustainable locations, and other eco-friendly actions.
   * A leaderboard ranks users based on points earned.
6. **Location & Business Listings (QR Check-in Placeholder)**
   * Allows admins to add verified sustainable businesses to a database.
   * Users can “check in” to earn points (currently uses a placeholder QR system).

**2. Interaction Between Components**

1. **User Requests a Page (Client → Server)**
   * The frontend sends an HTTP request to the Django backend.
   * Django processes the request and retrieves relevant data from the database.
   * The template engine renders the response and sends it back to the client.
2. **User Actions (Form Submission, Check-ins, Quizzes)**
   * Users interact with forms or buttons on the frontend.
   * Data is sent to Django views, which update the database accordingly.
   * Success messages or updates are displayed on the UI.
3. **Database Operations (CRUD Operations)**
   * User registrations, profile updates, and point calculations are stored in the database.
   * Admins can add new sustainable locations via the Django admin panel.

**3. Database Structure (Key Models)**

1. **User Profile (Extends Django’s User Model)**
   * User (Django default) – Stores username, email, password.
   * Account (Custom model) – Stores points, achievements, and profile info.
2. **Leaderboard Model**
   * Tracks user rankings based on points earned.
3. **Quiz & Challenges System**
   * Stores quiz questions and tracks user answers.
   * Awards points for correct responses.
4. **Check-in System**
   * Stores which users checked into which locations.
   * Tracks timestamps for validation.
5. **Business Listings**
   * Admins can register eco-friendly businesses.
   * Stores business details and verification status.

**Security & Scalability Considerations**

* **Data Encryption**: User passwords hashed with Django’s authentication system.
* **Scalability**: SQLite is used for development, but PostgreSQL is recommended for production.
* **Future Enhancements**:
  + API integration for real-time sustainability data.
  + Improved verification system for location check-ins.

**Conclusion**

The Sustainability Gamification App is designed with a modular and scalable architecture, enabling smooth user interactions, secure data management, and an engaging gamification experience. Future iterations will enhance the system with real-time updates, more point-earning opportunities, and advanced verification features.