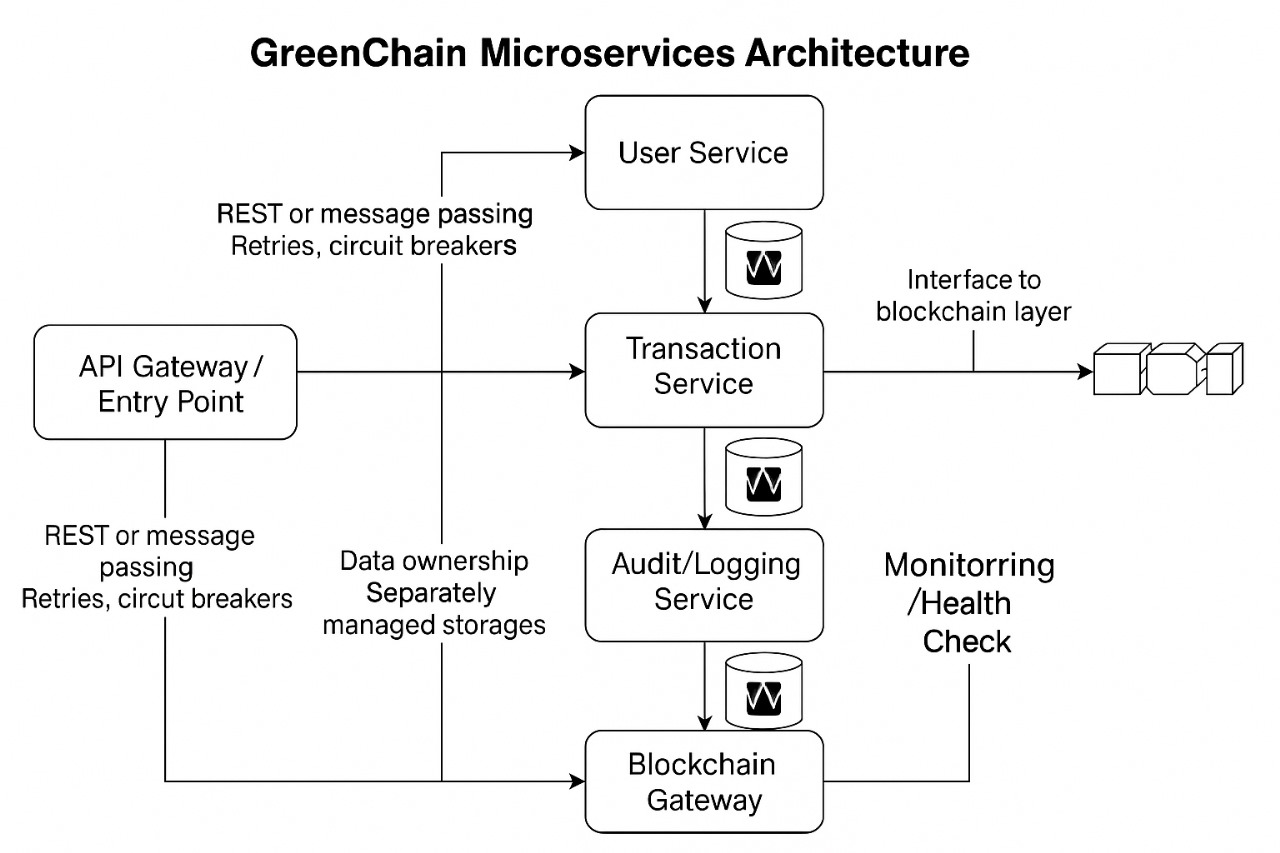
**GreenTrack Microservices**



**GreenTrack**

**1. Microservices Structure**

The architecture is broken into individual microservices, each handling a specific domain or function.

User Service

Transaction Service

Blockchain Gateway Service

**2. Communication Flow**

Arrows between services indicate RESTful HTTP communication or message passing (e.g., via queues).

Use of retry mechanisms, circuit breakers

**3. Data Ownership**

Each service manages its own data storage.

No direct database sharing—services only interact via APIs.

**4. Logging and Monitoring**

Centralized logging service is shown, capturing logs from all services.

A monitoring endpoint is to be used for system health checks.

**5. Blockchain Integration**

A Blockchain Gateway microservice acts as the interface between internal services and the blockchain layer (likely for GreenChain’s core functionality).

This decouples blockchain calls from other services, improving modularity and fault tolerance.

**6. API Gateway / Entry Point**

A single API Gateway or Frontend Entry Point routes external requests to appropriate services.

This helps with security, throttling, and load balancing.

For the blockchain integration this is the api (abi) we are using to connect the contract.sol and the ui for our application:  
