Biochar Production Tracking System Requirements

This document outlines the requirements for developing a system to track biochar production by farmers. The system includes hardware (furnaces with sensors) and a mobile app for farmer interaction. The system will automate data collection where possible and enable manual inputs where needed.

1. Furnace and Sensor Integration

The system will provide farmers with furnaces equipped with sensors to collect and transmit the following data:

- **Input Data**:
  - Weight of biomass input into the furnace
  - Date and time of biomass combustion
  - Temperature during combustion
  - Water usage
- **Output Data**:
  - Weight of biochar produced
  - Biochar storage details
  - Biochar sales details
  - Biochar application details

This data will be manually and automatically captured by sensors (wherever possible) and sent to the mobile application or central database.

2. Mobile Application Registration Process

Farmers will register themselves via the mobile app, providing the following details:

- **Farmer Registration Details**:
  - Farmer's Name
  - Address
  - Land ownership status (Yes/No)
  - Number of cattle animals
  - Carbon rights agreement (Yes/No)

Commented [1]: Need to confirm if this would be necessary

3. Land Parcel Information

Farmers will input detailed information for each land parcel they own. The system should allow farmers to register multiple parcels, with the following data for each:

- **Land Parcel Details**:

- GPS coordinates of the land
- Shape file of the land parcel
- Average crop yield for the last 3 years
- Average yield of crop residue for the last 3 years
- Average consumption of crop residue over the last 3 years
- Distance of the biochar production facility from the farm
- Distance of the storage facility from the farm
- Distance of the storage facility from the biochar unit
- Available storage area

4. Fertilizer Inventory & Seasonal Purchase Tracking:

- **Fertilizer Inventory**:
  - The system will maintain an inventory of fertilizers, tracking their quantity and usage.
  - Every fertilizer purchase will be recorded, updating the inventory in real time.
- **Seasonal Fertilizer Purchase Data**: Farmers will log the following details for each seasonal fertilizer purchase:
  - **Chemical Fertilizer**:
    - Fertilizer purchase proof (documents/photos)
    - Quantity of fertilizer purchased
  - **Organic Fertilizer**:
    - Organic fertilizer purchase proof (documents/photos)
    - Quantity of organic fertilizer purchased
- The system will also track the use of both chemical and organic fertilizers, enabling accurate monitoring of inventory levels and application patterns.
    -

5. Biomass Production Data

After seasonal harvesting, farmers will enter biomass production details for each land parcel:

- **Biomass Details**:
  - Type of crop harvested
  - Crop yield
  - Quantity of crop residue generated
  - Crop storage details (similar to fertilizer purchase data):
    - Date of storage
    - Place of storage
    - Storage proof (document upload or photo)
  - Residue storage details:
    - Date of residue storage
    - Place of residue storage
    - Residue storage proof (document upload or photo)

6. Biochar Production and Monitoring

The farmer, in coordination with furnace sensors, will record details of biomass combustion and biochar production. The system will either collect data manually or through sensors:

- **Biomass Combustion Details**:
    - Date of biomass combustion
    - Quantity of biomass prepared for combustion
    - Weight of biomass loaded into the furnace
    - Temperature readings (via sensor)
    - Time of combustion
    - Water usage
- **Biochar Production Data**:
    - Weight of biochar produced (via sensor)
    - Quantity of biochar produced

7. Biochar Storage and Usage

Once biochar is produced, farmers will input storage and application data into the system:

- **Biochar Storage Details**:
    - Location of biochar storage
    - Biochar for mixing with fertilizers
    - Organic fertilizers used for mixing
    - Details of biochar-mixed fertilizer application

System Design Considerations

1. **Automated Data Collection**: Sensors installed in furnaces will automate the collection of critical data points such as biomass weight, biochar weight, temperature, and water usage. This will reduce manual data entry by the farmers and ensure accuracy.
2. **Manual Data Entry**: Certain data, such as farm details, land parcel information, and biochar storage details, will need to be manually input by the farmer into the mobile application.
3. **User-friendly Mobile App Interface**: The mobile application should have an intuitive and user-friendly interface, enabling farmers to easily input or verify data.
4. **Data Validation**: Where possible, the system should validate inputs (e.g., GPS coordinates, storage batch dates) to reduce the likelihood of errors.
5. **Cloud Storage and Security**: Data collected from sensors and through the app should be securely stored in a cloud database. Privacy and security protocols must be in place to protect sensitive farmer and production data.
6. **Reporting and Analytics**: The system should allow for automated generation of reports on biomass production, biochar production, and other key metrics to assist both farmers and the company in tracking performance and compliance with carbon rights agreements.

This design will ensure efficient biochar production tracking and will help in optimizing biomass utilization, benefiting both the farmers and the organization.

ER Design for SQL (Relational Database)

Entities:

1. **Farmer**
2. **Land Parcel**
3. **Fertilizer Inventory**
4. **Biomass**
5. **Biochar Production**
6. **Biochar Storage**

SQL ER Diagram Breakdown:

- **Farmer**: Contains personal details, land ownership status, and carbon rights agreement.
- **Land Parcel**: Tracks parcel information for each farmer, including GPS coordinates and yields.
- **Fertilizer Inventory**: Tracks both chemical and organic fertilizers, their purchase, usage, and related proof documents.
- **Biomass**: Tracks biomass production for each land parcel, crop type, yield, and residue.
- **Biochar Production**: Tracks the biochar production process, linking with biomass and furnace data.
- **Biochar Storage**: Stores details of where the produced biochar is stored.

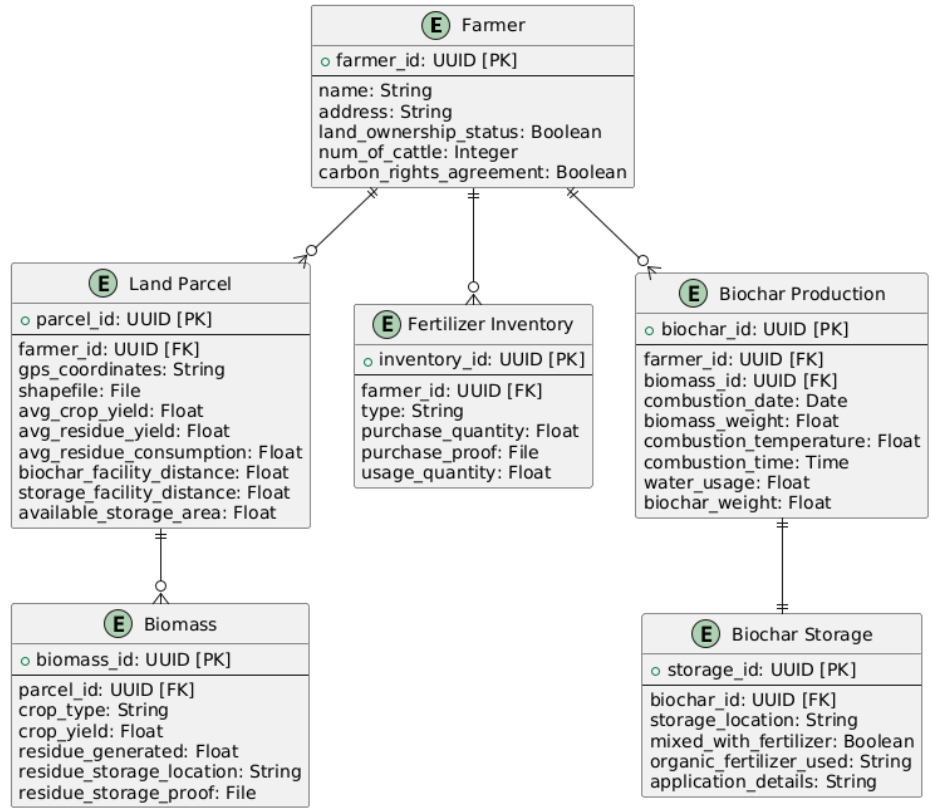
Entity Relationships:

1. **Farmer ↔ Land Parcel** (One-to-Many): A farmer can own multiple land parcels.
2. **Land Parcel ↔ Biomass** (One-to-Many): Each land parcel can produce multiple types of biomass.
3. **Farmer ↔ Fertilizer Inventory** (One-to-Many): Each farmer can have multiple fertilizer inventory records.
4. **Farmer ↔ Biochar Production** (One-to-Many): Each farmer can have multiple biochar production records.
5. **Biochar Production ↔ Biochar Storage** (One-to-One): Each biochar production record is stored in one place.

ER Design for NoSQL (Flexible Data)

Collections:

1. **Sensor Data**
   - Includes data for biomass combustion like temperature, water usage, and biochar weight.
   - Flexible schema for accommodating sensor data types from various furnaces.
2. **Fertilizer Documents**
   - Stores photos and documents for fertilizer purchase proofs.
   - Schema-less document storage (MongoDB or similar).
3. **Storage Documents**
   - Stores images and documents related to biochar and biomass storage.
   - Schema-less for flexibility with different file types.



PlantUML Script for ER Design

plantuml
Copy code

```
@startuml
entity "Farmer" as Farmer {
  +farmer_id: UUID [PK]
  ---
  name: String
  address: String
  land_ownership_status: Boolean
  num_of_cattle: Integer
  carbon_rights_agreement: Boolean
}

entity "Land Parcel" as LandParcel {
  +parcel_id: UUID [PK]
  ---
  farmer_id: UUID [FK]
  gps_coordinates: String
  shapefile: File
  avg_crop_yield: Float
  avg_residue_yield: Float
  avg_residue_consumption: Float
  biochar_facility_distance: Float
  storage_facility_distance: Float
  available_storage_area: Float
}

entity "Fertilizer Inventory" as FertilizerInventory {
  +inventory_id: UUID [PK]
  ---
  farmer_id: UUID [FK]
  type: String
  purchase_quantity: Float
  purchase_proof: File
  usage_quantity: Float
}
```

```
entity "Biomass" as Biomass {
  +biomass_id: UUID [PK]
  ---
  parcel_id: UUID [FK]
  crop_type: String
  crop_yield: Float
  residue_generated: Float
  residue_storage_location: String
  residue_storage_proof: File
}

entity "Biochar Production" as BiocharProduction {
  +biochar_id: UUID [PK]
  ---
  farmer_id: UUID [FK]
  biomass_id: UUID [FK]
  combustion_date: Date
  biomass_weight: Float
  combustion_temperature: Float
  combustion_time: Time
  water_usage: Float
  biochar_weight: Float
}

entity "Biochar Storage" as BiocharStorage {
  +storage_id: UUID [PK]
  ---
  biochar_id: UUID [FK]
  storage_location: String
  mixed_with_fertilizer: Boolean
  organic_fertilizer_used: String
  application_details: String
}

Farmer ||--o{ LandParcel
Farmer ||--o{ FertilizerInventory
LandParcel ||--o{ Biomass
Farmer ||--o{ BiocharProduction
```

```
BiocharProduction ||--|| BiocharStorage
@enduml
```

NoSQL Structure Example (MongoDB)

Sensor Data Collection (NoSQL)
json
Copy code

```
{
  "_id": "sensor_data_001",
  "furnace_id": "furnace_001",
  "biomass_weight": 500.0,
  "combustion_temp": [350.5, 400.7, 450.3],  // temperature over time
  "water_usage": 100.0,
  "biochar_weight": 50.0,
  "timestamp": "2024-09-15T10:45:00Z"
}
```

1.

Fertilizer Documents Collection (NoSQL)
json
Copy code

```
{
  "_id": "fertilizer_doc_001",
  "farmer_id": "farmer_001",
  "document_type": "purchase_receipt",
  "file_url":
"https://storage.example.com/fertilizer/receipt_001.jpg",
  "uploaded_at": "2024-09-15T12:00:00Z"
}
```

2.

Storage Documents Collection (NoSQL)
json
Copy code

```
{
  "_id": "storage_doc_001",
  "biochar_id": "biochar_001",
```

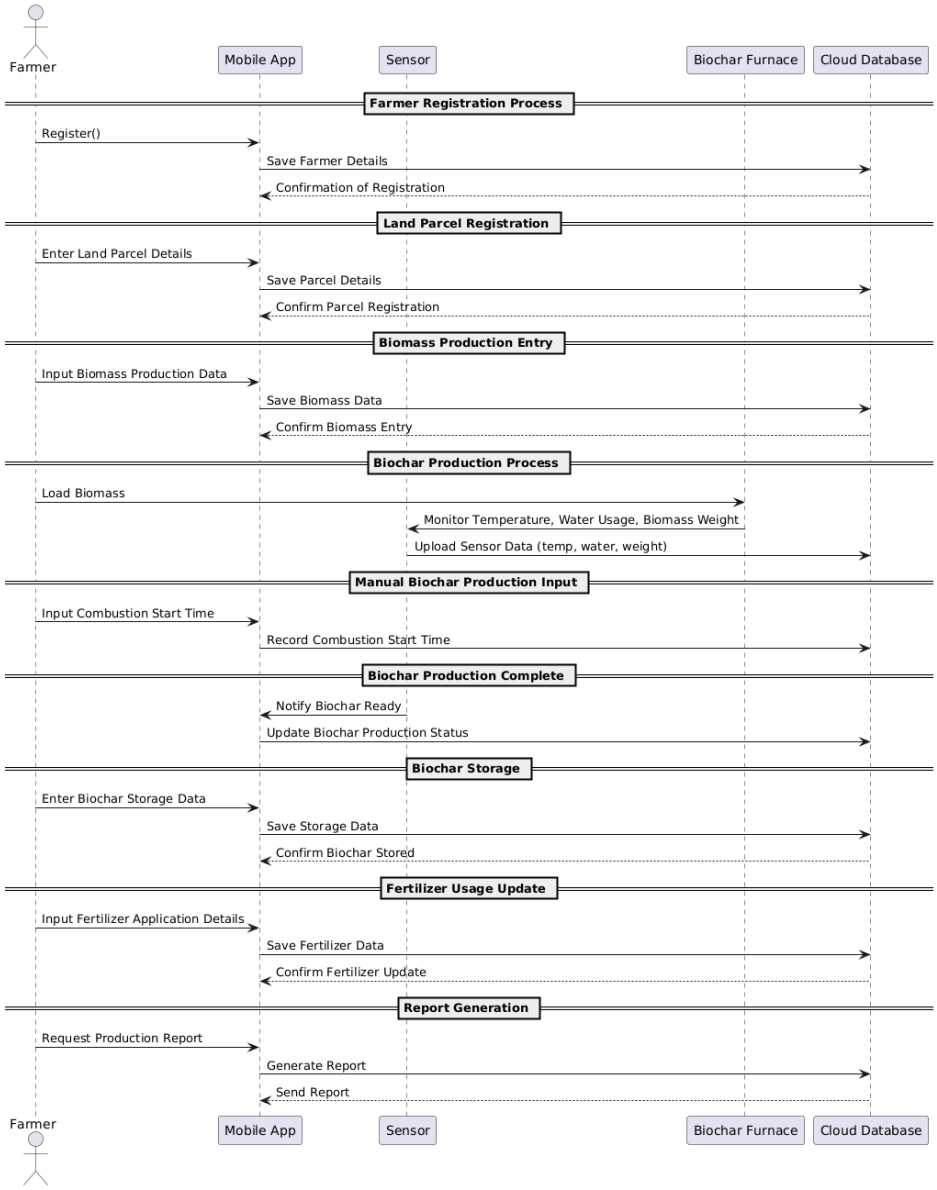
```
  "storage_location": "Warehouse A",
  "document_type": "storage_proof",
  "file_url": "https://storage.example.com/storage/proof_001.jpg",
  "uploaded_at": "2024-09-15T14:30:00Z"
}
```

Draft Sequence Diagram:

Key Elements of the Sequence Diagram:

1. **Farmer Registration Process**: Farmer registers through the mobile app, and the data is saved to the cloud.
2. **Land Parcel Registration**: Farmer enters land parcel information through the mobile app, and the system saves it.
3. **Biomass Production Entry**: After harvesting, the farmer records biomass production data.
4. **Biochar Production Process**: The furnace starts the biochar production, with sensors monitoring combustion details like temperature and water usage.
5. **Manual Biochar Input**: Some details, like combustion start time, are entered manually by the farmer.
6. **Biochar Storage**: After production, the farmer records storage data for the biochar.
7. **Fertilizer Usage Update**: Farmers input fertilizer application details, and the system tracks this data.
8. **Report Generation**: The system generates reports upon request based on data stored in the cloud database.

This sequence diagram captures the flow of interactions and data transfers among the farmer, mobile app, sensors, furnace, and cloud database.


Farmer
Mobile App
Sensor
Biochar Furnace
Cloud Database
Farmer Registration Process
Register()
Save Farmer Details
Confirmation of Registration
Land Parcel Registration
Enter Land Parcel Details
Save Parcel Details
Confirm Parcel Registration
Biomass Production Entry
Input Biomass Production Data
Save Biomass Data
Confirm Biomass Entry
Biochar Production Process
Load Biomass
Monitor Temperature, Water Usage, Biomass Weight
Upload Sensor Data (temp, water, weight)
Manual Biochar Production Input
Input Combustion Start Time
Record Combustion Start Time
Biochar Production Complete
Notify Biochar Ready
Update Biochar Production Status
Biochar Storage
Enter Biochar Storage Data
Save Storage Data
Confirm Biochar Stored
Fertilizer Usage Update
Input Fertilizer Application Details
Save Fertilizer Data
Confirm Fertilizer Update
Report Generation
Request Production Report
Generate Report
Send Report
Farmer
Mobile App
Sensor
Biochar Furnace
Cloud Database

Plantuml

```
@startuml
actor Farmer
participant "Mobile App" as MobileApp
participant "Sensor" as Sensor
participant "Biochar Furnace" as Furnace
participant "Cloud Database" as CloudDB

== Farmer Registration Process ==
Farmer -> MobileApp : Register()
MobileApp -> CloudDB : Save Farmer Details
CloudDB --> MobileApp : Confirmation of Registration

== Land Parcel Registration ==
Farmer -> MobileApp : Enter Land Parcel Details
MobileApp -> CloudDB : Save Parcel Details
CloudDB --> MobileApp : Confirm Parcel Registration

== Biomass Production Entry ==
Farmer -> MobileApp : Input Biomass Production Data
MobileApp -> CloudDB : Save Biomass Data
CloudDB --> MobileApp : Confirm Biomass Entry

== Biochar Production Process ==
Farmer -> Furnace : Load Biomass
Furnace -> Sensor : Monitor Temperature, Water Usage, Biomass Weight
Sensor -> CloudDB : Upload Sensor Data (temp, water, weight)

== Manual Biochar Production Input ==
Farmer -> MobileApp : Input Combustion Start Time
MobileApp -> CloudDB : Record Combustion Start Time

== Biochar Production Complete ==
Sensor -> MobileApp : Notify Biochar Ready
MobileApp -> CloudDB : Update Biochar Production Status
```

```
== Biochar Storage ==
Farmer -> MobileApp : Enter Biochar Storage Data
MobileApp -> CloudDB : Save Storage Data
CloudDB --> MobileApp : Confirm Biochar Stored

== Fertilizer Usage Update ==
Farmer -> MobileApp : Input Fertilizer Application Details
MobileApp -> CloudDB : Save Fertilizer Data
CloudDB --> MobileApp : Confirm Fertilizer Update

== Report Generation ==
Farmer -> MobileApp : Request Production Report
MobileApp -> CloudDB : Generate Report
CloudDB --> MobileApp : Send Report

@enduml
```