‘S.R.M’

A blockchain solution for a perishable supply chain

ARCHITECTURE DOCUMENT

Design Patterns for Blockchain – BCDV 1011

George Brown College

Professor Dave Mckay & Professor Paul Chafe

June 2020

CHERUKKATIL NASEER – 101280592
HENRY ERIKO MWENGE – 101286818
LORRAN SUTTER – 101276593
WANJA MASCARENHAS - 101280022
ZAKARIYA JASAT – 101092428

SupplyBlock Risk Management Inc.

A PERISHIBLE SUPPLY CHAIN START-UP.

We are leveraging *Distributed Ledger Technology* by integrating *Blockchain* infrastructure and *Ethereum Smart Contracts* in our business model.

We strive to streamline how perishable supply chains work for manufacturers, transporters, and distributors.

Our business will facilitate distributers in monitoring commodity status and pinpoint the specific area of failure to minimise costs related to product recalls.

Table of Contents

Goals of the Architecture

Smart Contract

We want to create a smart contract leveraging Ethereum's ERC721 Token to facilitate how the different stakeholders in a perishable supply chain (Manufacturers, Transporters, Distributors) interact. In our smart contracts we will be able to store information like:

- The serial number which will help us identify each battery and correlate it to an ERC721 token.
- The temperature to monitor the environmental conditions while transporting.
- The location of the asset to continuously track the movement and monitor the stage of the supply chain it has reached.
- The manufacturer's information to trace the source of each battery.
- The requesting distributors to associate each transportation batch from manufacturer to distributor.

Our smart contract will streamline the supply chain process between Manufacturer, Transporter and Distributor. The smart contract will be able to create new instances of battery tokens and associates them to each unique battery, creates new orders from distributors, and facilitates a smooth agreement between the stakeholders. We will always be able to update the location and thermal activity and get an update on the status of the battery any time.

IoT

We are leveraging *Internet of Things* to work hand-in-hand with our backend infrastructure, to provide detailed and accurate updates of the battery's condition.

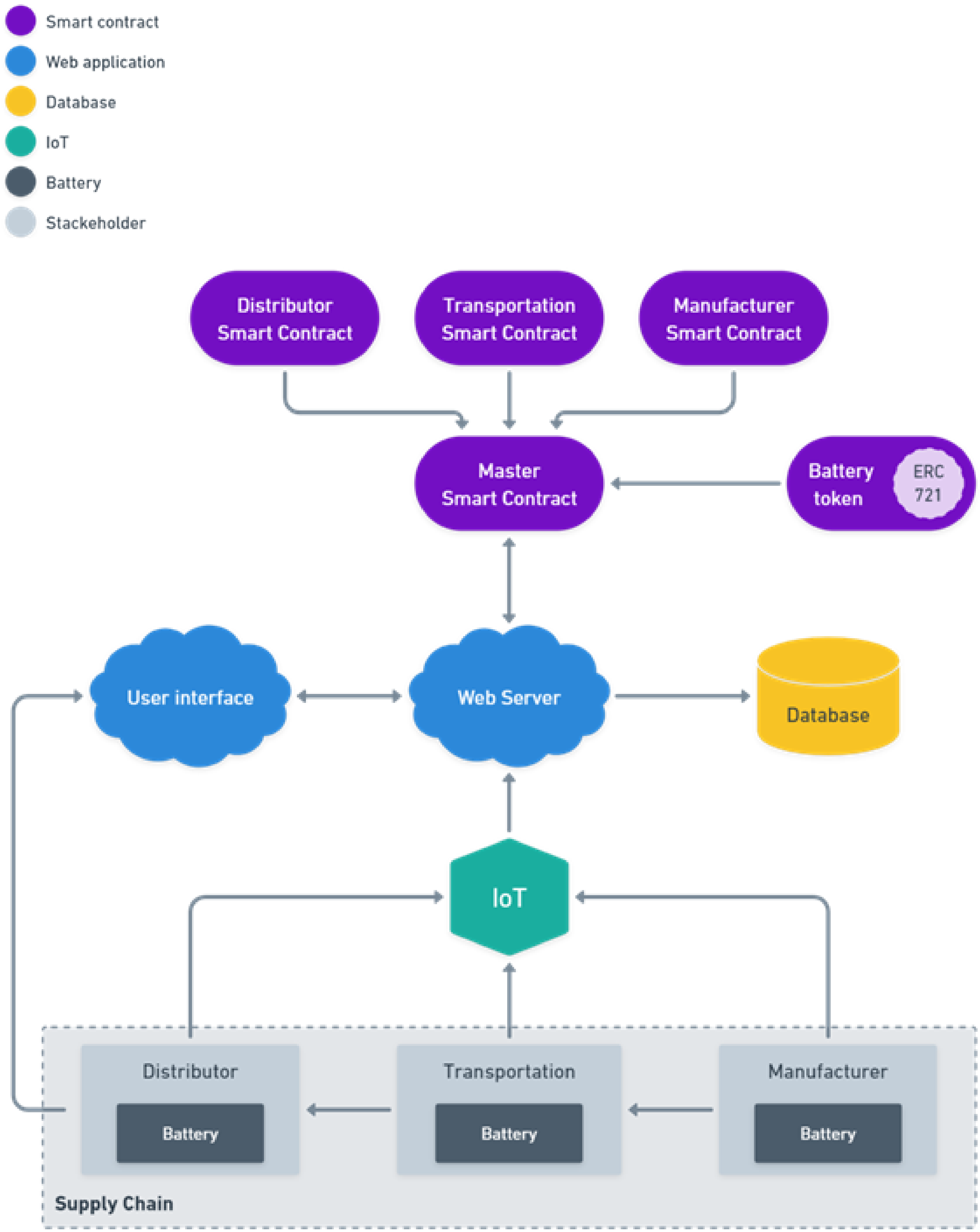
Frontend

Our frontend solution leverages the information provided from our web servers, to create a friendly user interface specifically designed for the distributors in the supply chain. Distributors will be able to utilize our services through a unique login to monitor the conditions of the assets.

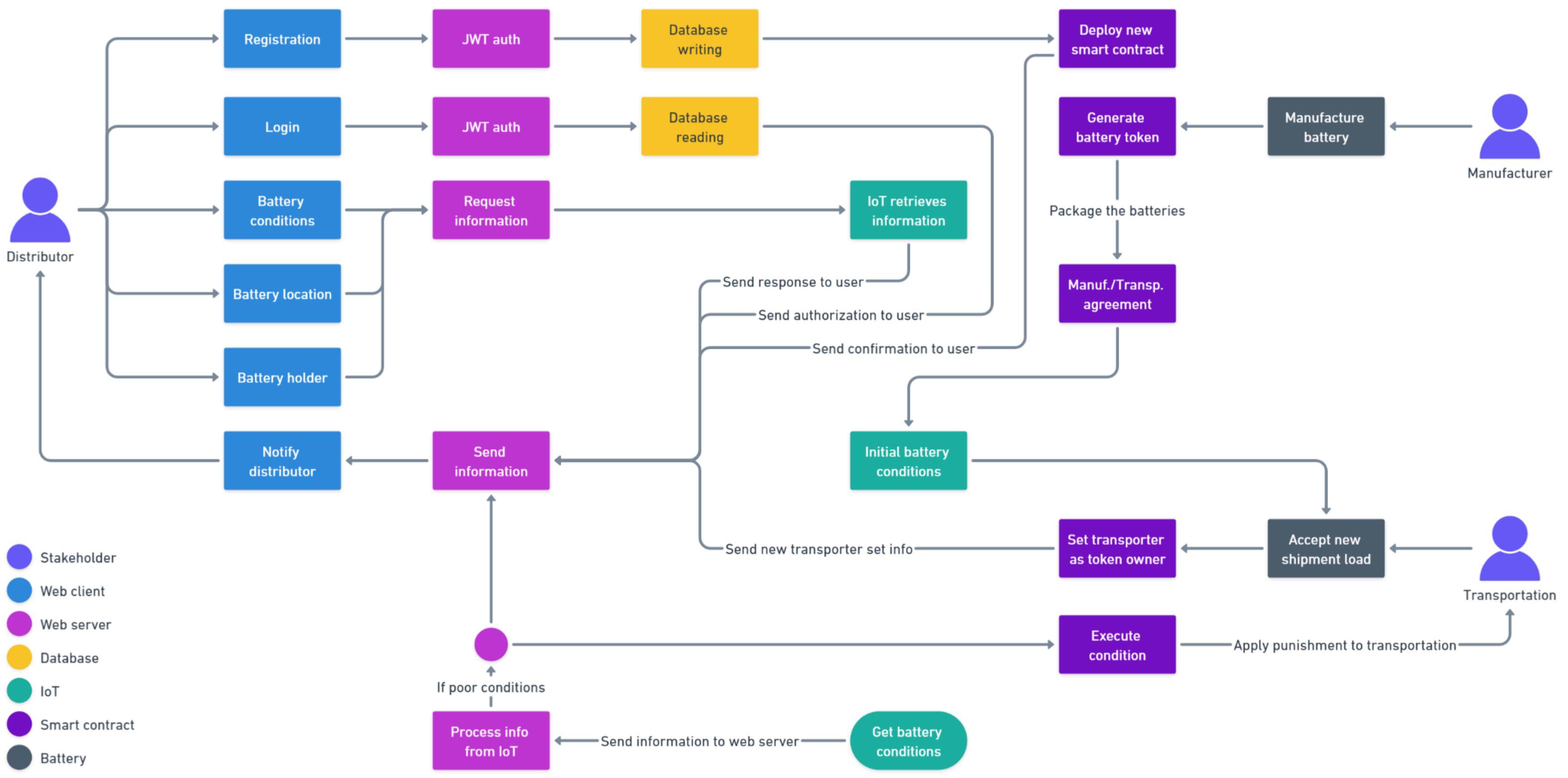
Backend

The backend creates the bigger picture by connecting the dots between the smart contracts, the IoT, and the front end. Our backend will be built as an API which creates more efficient and flexible communication between the frontend, backend, smart contracts, and the IoT.

Architecture Diagram



Flow Diagram



Data Definition

Database

1. Data

Name	Type
name	String
password	String
address	String
registration_date	Date

Smart contract

1. Data

Battery struct

uint256 id

string manufacturer

bytes6 serialno

int16 thermal

bytes26 location

Byte array to store latitude and longitude in the form

<lat>;<long> Example: -89.12345678;-168.12345678

address requestingDistributor

mapping(uint256 => Battery) private batteries

Maps token ids to battery info

uint256 private nextId

Next token id to be minted

2. Functions

Name	Input	Description
addManufacturer	(address manufacturer) public onlyAdmin()	Add address as manufacturer role
addTransporter	(address transporter) public onlyAdmin()	Add address as transporter role
addDistributor	(address distributor) public onlyAdmin()	Add address as distributor role
makeBattery	(string memory _manufacturer, bytes6 _serialno, int16 _thermal, bytes26 _location) public onlyManufacturer()	Mints a new battery token
orderBattery	(uint256 _id) public onlyDistributor()	Set requesting Distributor in battery token
thermalMonitor	(uint256 _id, int16 _thermal, bytes26 _location) public onlyTransporter() onlyOwner(_id)	Update current thermal and location info's
transferBattery	(uint256 _id, address _to, int16 _thermal, bytes26 _location) public onlyOwner(_id)	Transfer battery
getBatteryTrackingInfo	(uint256 _id) public view onlyRequestingDistributor(_id) returns (int16, bytes26, address)	Get current battery info. Returns thermal, location and current owner

Backend

1. Endpoints

ADMINISTRATOR

EndPoint	Description
/admin/addManufacturer	This endpoint calls addManufacturerfrom smart contract
/admin/addTransporter	This endpoint calls addTransporterfrom smart contract
/admin/addDistributor	This endpoint calls addDistributorfrom smart contract

DISTRIBUTOR

EndPoint	Description
/distributors/index	get list of registered distributors in db
/distributors/new	register new distributor
/distributors/login	distributor login
/distibutors/batteryInfo/:tokenId	Get current info of battery. This endpoint calls getBatteryTrackingInfo from smart contract
/distributors/orderBattery/:tokenId	This endpoint calls orderBattery from smart contract

MANUFACTURER

EndPoint	Description
/manufacturers/transferBattery	This endpoint calls transferBattery from smart contract
/manufacturers/makeBattery	This endpoint calls makeBattery smart contract

TRANSPORTER

EndPoint	Description
/transporters/transferBattery	This endpoint calls transferBattery from smart contract

ORACLE (IoT)

EndPoint	Description
/transporters/thermalMonitor	This endpoint calls thermalMonitor from smart contract

Tech Stack

Smart Contract

- Solidity - smart contract programming language
- Truffle - dApp environment
- Ethereumjs-util - utility functions for Ethereum
- Truffle-assertions - additional assertions for truffle
- Bignumber.js - library to handle big numbers

Backend

- Express.js - web application framework
- MongoDB - NoSQL database
- Mongoose - object data modeling (ODM) library for MongoDB and Node.js
- Async - library to perform asynchronous operations
- Express validator - middleware to validate data
- Bcryptjs - library to perform cryptography
- JWT.IO - JSON Web Tokens to allow, decode, verify, and generate JWT
- Jest - library for tests
- Web3.js - interact with smart contracts
- Dotenv - loads environment variables from a .env file

Frontend

- Rimble - design system
- ReactJS - frontend library
- Axios - HTTP requests

Team Bio

Cherukkatil Naseer

Nas is a Blockchain Development student at George Brown College and Director of NLM UA-Systems. He believes that rapid development in AI, IoT and Blockchain would create better efficiency and security in most of the industries and would open more jobs/business opportunities in new sectors. Nas has been well-trained on avionics products of Boeing and Airbus commercial Aircrafts. He holds UA (Unmanned Aircraft) Pilots certificate approved by CAAS. He served Red-cross, Singapore as a volunteer for two years where he trained on First Aid and CPR. He spends his free time flying, building, and modifying drones.

Henry Eriko Mwenge

Eliko 岳飞 is the co-founder of The Briefcase Trader and is responsible for producing the company's equity research reports. He is the chief editor of The Briefcase Trader Soft Commodity and Livestock Report. Eliko is passionate about commodities and developing a fully integrated Structured Commodity Trade Finance network specific to emerging markets. He enjoys playing rugby, birdwatching, and beekeeping.

Lorran Sutter

Lorran is a Computational Engineer that studied a couple of different fields, such as computational modeling, numerical methods, computer graphics and machine learning. He has work experience in web development mainly employing C# in developing web crawlers and automated tests for large legal management systems. Engineer by background and programmer by heart, in love with Python and JavaScript. Always looking for innovative and disruptive technologies, which led him to start studying Blockchain Development in George Brown College. He believes that humanity must rely on technology to build a better future.

Wanja Mascarenhas

Wanja Mascarenhas is an IT professional and educator. As a developer, Wanja worked in the banking industry. As an educator, she taught computational algorithms, programming, data structures, and operational research. She spends her free time volunteering. She loves talking with friends and sharing experiences. Wanja is currently studying Blockchain Development at George Brown College.

Zakariya Jasat

Zak has lived and worked in nations across the globe, making him an empowered multicultural business administrator. He is exceptional at adapting to change, managing human resources, and viewing situations through several lenses. He has helped set up businesses from the ground up and lead multiple keystone projects. He truly appreciates the importance of owning strategic objectives by improving organizational performance through boosting employee engagement and creating a healthy organizational culture through fostering diversity and inclusion. Zak is a calculated risk taker, motivated self starter and is always committed to leaving a legacy that inspires positive change everywhere he goes. He is the strong open-hearted wind that ignites powerful contribution.