# Syrus'23

**Problem Statement:** 3PS-5

Team Name: Cookie Bytes


Team Leader: Tanay Phatak

Team Members: Tanmay Damle

Sakshi Bhojwani

Ayush Jain


Track: (1) Polygon + Ethereum

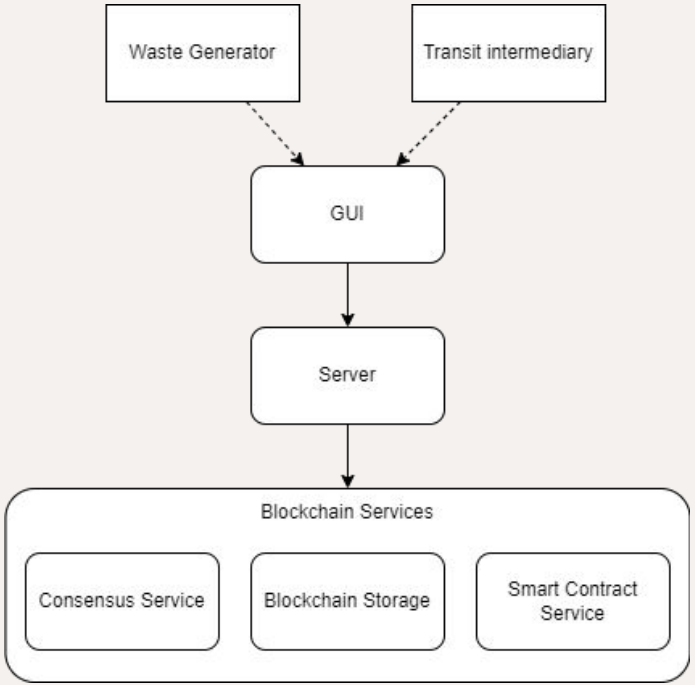
# Introduction

- Tons of waste is generated on daily basis from various sources. Some of it is recyclable.
- Everyone is curious about where this waste really goes.
- Does it actually reach the intended waste recycling plants?
- To answer these questions we are building a system based on blockchain which tracks the waste after it is collected from the source till it is delivered to the authentic recycling plants.
- By using blockchain for waste tracking, it becomes possible to ensure the traceability and accountability of the waste, which can help in reducing environmental pollution and improving waste management practices.

# Use case

- Waste transit manager loads the data of waste package into the system.
- The waste package is assigned a unique id that will be used to track it.
- It’s location/status will be updated as it passes certain check points.
- Finally as it enters the designated recycle centre transaction will be successfully terminated.
- Incentives will be provided based on the weight of the effectively recyclable waste.

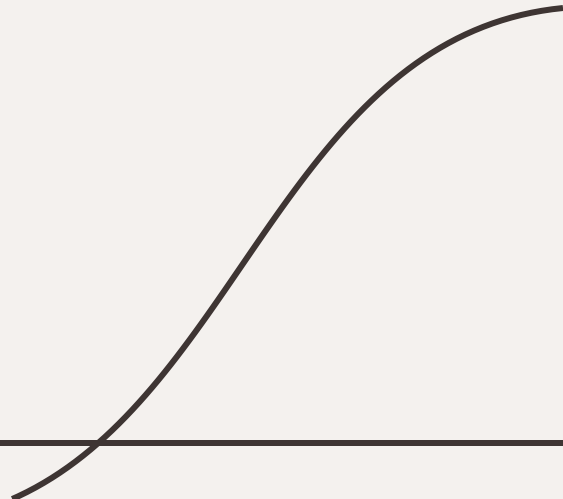
# Block Diagram



# Implementation

Tech Stack Used:

- HTML/CSS with Bootstrap.
- Solidity
- MongoDB
- Node.js - Express.js
- Polygon



- The end users i.e. the waste generators and intermediaries access the system by the GUI wherein they will be provided with basic login/sign-up service.
- The GUI is linked to the server/backend built using Node.js
- Finally, the server engages in many request-response cycles with the underlying blockchain architecture.
- Key blockchain components including "Consensus Service," "Blockchain Storage," and "Smart Contract Service" are all housed within the blockchain architecture.
- The waste generator(Main Source) and the recycling plant can only track the waste. The transporter has access to add new waste packages apart from tracking it.
- Additionally, the recycling plant will have an option to update the plant status(functioning/ not function) and be able to see future inbound packages to plan for resource allocation accordingly.

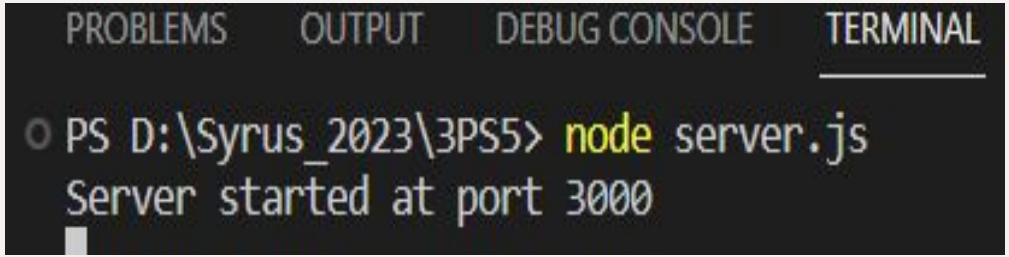
# Conclusion

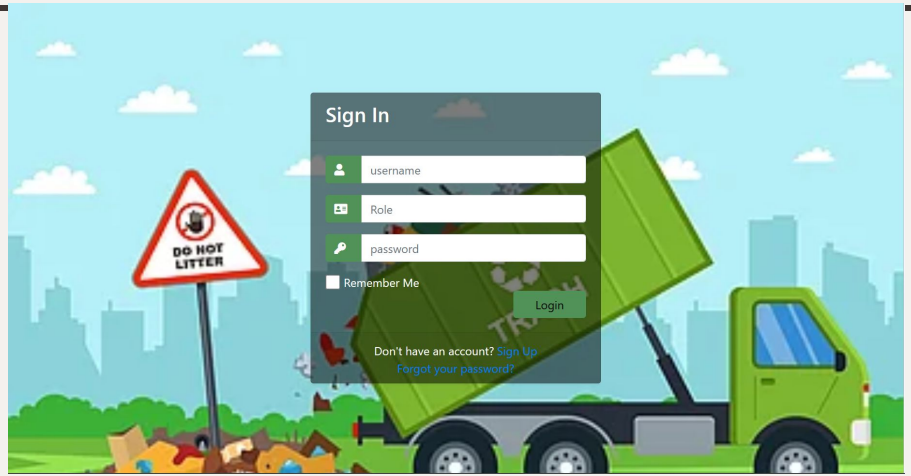
All in all a waste tracking system using blockchain technology has the potential to revolutionize the waste management industry by providing a secure and transparent way to track and manage waste. With its ability to automate waste tracking, disposal, and recycling procedures, this system can help reduce waste, improve efficiency, and ensure environmental sustainability. By using smart contracts and blockchain technology, the system can eliminate fraud, increase accountability, and promote transparency, thereby creating a more reliable and trustworthy waste management system. Overall, the proposed waste tracking system using blockchain technology has the potential to transform the waste management industry and create a cleaner and more sustainable future.

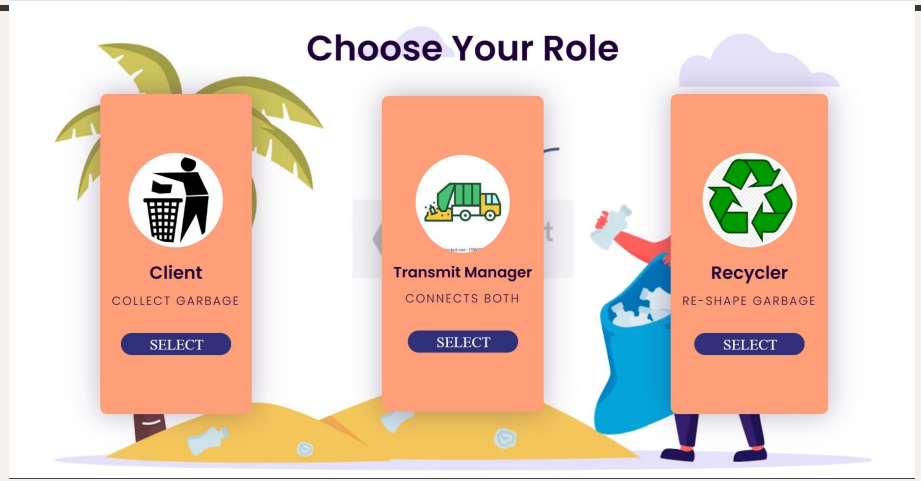
# After 1st coding round(Updates)

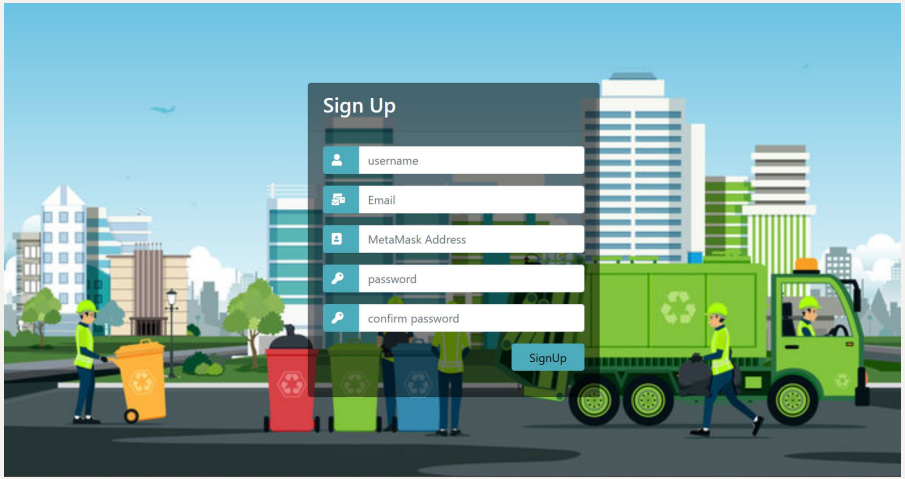
- Designed the UI using HTML/CSS.
- Created server using Node.js - Express.js
- Started with the development and testing of Smart Contracts using Solidity.

Server Running:



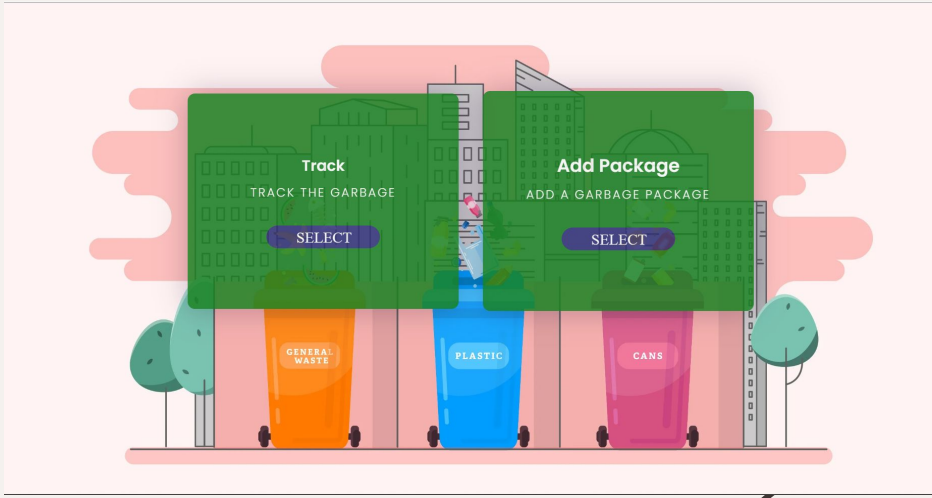

Sign In
username
Role
password
Remember Me
Login
Don't have an account? Sign Up
Forgot your password?
DO NOT LITTER


Choose Your Role
Client
COLLECT GARBAGE
SELECT
Transmit Manager
CONNECTS BOTH
SELECT
Recycler
RE-SHAPE GARBAGE
SELECT

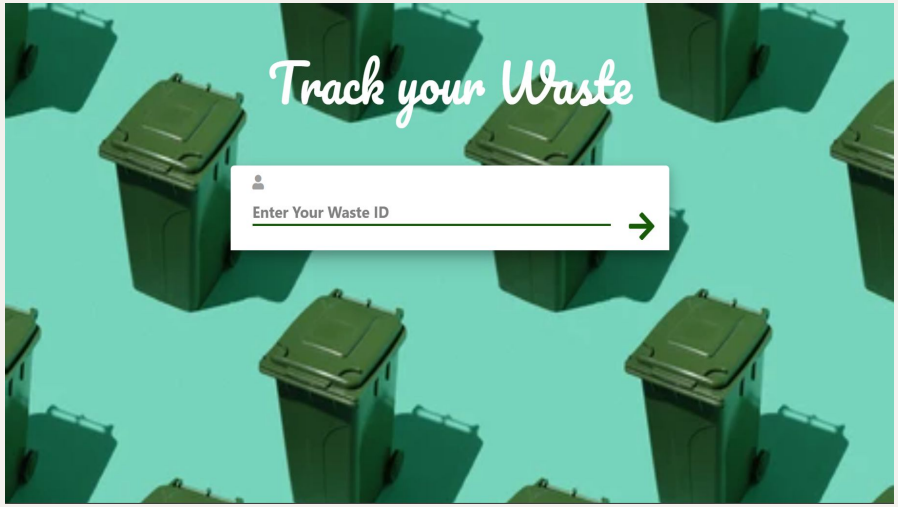

Sign Up
username
Email
MetaMask Address
password
confirm password
SignUp

# Updates after 1st Review

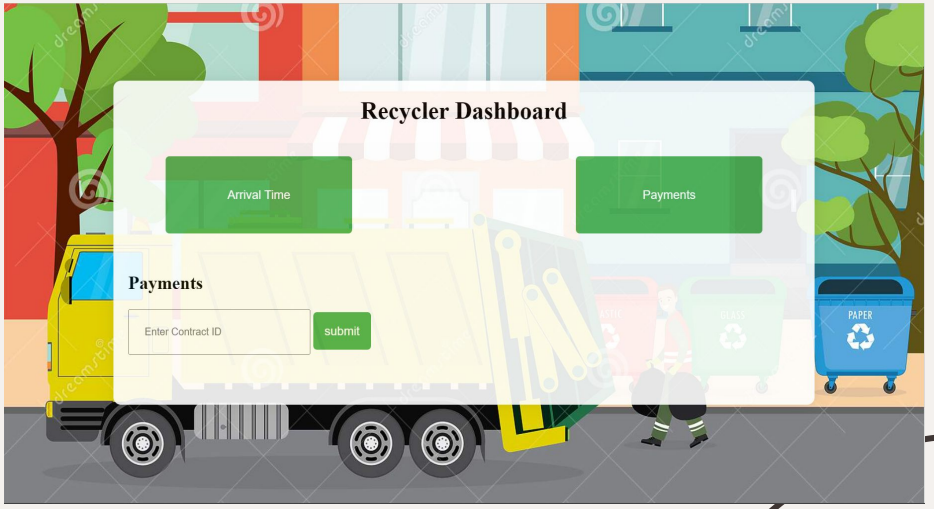

Dashboard page for every logged in user i.e. the source of the waste.




Transporter dashboard which allows for addition of new package and tracking of existing package.




Waste tracker page for for tracking the waste package.




Recycler Dashboard, where arrival time of package is logged and incentives are calculated in proportion to the weight of effectively recyclable waste.

```
in create contract
Contract deployed to 0xA2848608cFa8AE52928cae891c2fCb363b91b1Bc
[
  { inputs: [], stateMutability: 'nonpayable', type: 'constructor' },
  {
    inputs: [ [Object], [Object], [Object], [Object] ],
    name: 'addTransit',
    outputs: [],
    stateMutability: 'nonpayable',
    type: 'function'
  },
```

Smart contract deployed.

```
[
  {
    source: 'mumbai',
    destination: 'delhi',
    departureTime: BigNumber { value: "1678517334083" },
    arrivalTime: BigNumber { value: "0" }
  }
]
```

Internal contract details