**Diamante Network Operations**

**This project demonstrates the use of the Diamante SDK to perform various operations on the Diamante blockchain network, including creating accounts, making payments, managing data, and setting options.**

**Table of Contents**

* [**Installation**](#installation)
* [**Running the Project**](#running-the-project)
* [**Services**](#services)
  + [**Create Account**](#create-account)
  + [**Payments**](#payments)
  + [**Manage Data**](#manage-data)
  + [**Set Options**](#set-options)

**Installation**

**To set up the project, follow these steps:**

1. **Clone the repository:**

**bash**

**Copy code**

**git clone https://github.com/your-repo/diamante-network-operations.git**

1. **Navigate to the backend directory and install dependencies:**

**bash**

**Copy code**

**cd diamante-network-operations/diamante-demo**

**npm install**

1. **Navigate to the frontend directory and install dependencies:**

**bash**

**Copy code**

**cd ../diamante-frontend**

**npm install**

**Running the Project**

1. **Start the backend server:**

**bash**

**Copy code**

**cd ../diamante-demo**

**node index.js**

1. **Start the frontend development server:**

**bash**

**Copy code**

**cd ../diamante-frontend**

**npm start**

1. **Open your browser and navigate to http://localhost:3000.**

**Services**

**Create Account**

**The Create Account service generates a new keypair and funds the account using Friendbot.**

**Frontend**

**In the frontend, navigate to the "Create Account" page by clicking the "Create Account" button on the home page. Then, click "Generate Keypair" to create a new keypair and "Get Test DIAM" to fund the account.**

**Backend**

**The backend provides the following endpoints:**

* **Create Keypair**
  + **Endpoint: /create-keypair**
  + **Method: POST**
  + **Description: Generates a new keypair.**
  + **Response:**

**json**

**Copy code**

**{**

**"publicKey": "GCPH3KD3HEEWVBH4TH4RCV7RM5SRSKRTTX0FQQF6EK3QDFSWM73LYXUD",**

**"secret": "SDYIM7HJCJVTRWFC7W3YN2WTIIY7FOZZXM7UCNGGIG47JAAIKV6EF3Y"**

**}**

* **Fund Account**
  + **Endpoint: /fund-account**
  + **Method: POST**
  + **Description: Funds the account using Friendbot.**
  + **Request Body:**

**json**

**Copy code**

**{**

**"publicKey": "GCPH3KD3HEEWVBH4TH4RCV7RM5SRSKRTTX0FQQF6EK3QDFSWM73LYXUD"**

**}**

* + **Response:**

**json**

**Copy code**

**{**

**"message": "Account GCPH3KD3HEEWVBH4TH4RCV7RM5SRSKRTTX0FQQF6EK3QDFSWM73LYXUD funded successfully"**

**}**

**Payments**

**The Payments service allows you to send DIAM from one account to another.**

**Frontend**

**In the frontend, navigate to the "Payments" page by clicking the "Payments" button on the home page. Enter the sender's secret key, receiver's public key, and amount, then click "Make Payment" to initiate the transaction.**

**Backend**

**The backend provides the following endpoint:**

* **Make Payment**
  + **Endpoint: /make-payment**
  + **Method: POST**
  + **Description: Makes a payment from one account to another.**
  + **Request Body:**

**json**

**Copy code**

**{**

**"senderSecret": "SDYIM7HJCJVTRWFC7W3YN2WTIIY7FOZZXM7UCNGGIG47JAAIKV6EF3Y",**

**"receiverPublicKey": "GCPH3KD3HEEWVBH4TH4RCV7RM5SRSKRTTX0FQQF6EK3QDFSWM73LYXUD",**

**"amount": "10"**

**}**

* + **Response:**

**json**

**Copy code**

**{**

**"message": "Payment of 10 DIAM made to GCPH3KD3HEEWVBH4TH4RCV7RM5SRSKRTTX0FQQF6EK3QDFSWM73LYXUD successfully"**

**}**

**Manage Data**

**The Manage Data service allows you to set or delete key-value pairs on an account.**

**Frontend**

**In the frontend, navigate to the "Manage Data" page by clicking the "Manage Data" button on the home page. Enter the sender's secret key, the key, and the value, then click "Manage Data" to set or delete the key-value pair.**

**Backend**

**The backend provides the following endpoint:**

* **Manage Data**
  + **Endpoint: /manage-data**
  + **Method: POST**
  + **Description: Sets or deletes a key-value pair on an account.**
  + **Request Body:**

**json**

**Copy code**

**{**

**"senderSecret": "SDYIM7HJCJVTRWFC7W3YN2WTIIY7FOZZXM7UCNGGIG47JAAIKV6EF3Y",**

**"key": "example\_key",**

**"value": "example\_value"**

**}**

* + **Response:**

**json**

**Copy code**

**{**

**"message": "Data for key example\_key managed successfully"**

**}**

**Set Options**

**The Set Options service allows you to set various account options such as inflation destination, home domain, and thresholds.**

**Frontend**

**In the frontend, navigate to the "Set Options" page by clicking the "Set Options" button on the home page. Enter the sender's secret key, inflation destination, home domain, and thresholds, then click "Set Options" to update the account options.**

**Backend**

**The backend provides the following endpoint:**

* **Set Options**
  + **Endpoint: /set-options**
  + **Method: POST**
  + **Description: Sets various account options.**
  + **Request Body:**

**json**

**Copy code**

**{**

**"senderSecret": "SDYIM7HJCJVTRWFC7W3YN2WTIIY7FOZZXM7UCNGGIG47JAAIKV6EF3Y",**

**"inflationDest": "GCPH3KD3HEEWVBH4TH4RCV7RM5SRSKRTTX0FQQF6EK3QDFSWM73LYXUD",**

**"homeDomain": "example.com",**

**"lowThreshold": "1",**

**"medThreshold": "2",**

**"highThreshold": "3"**

**}**

* + **Response:**

**json**

**Copy code**

**{**

**"message": "Options set successfully"**

**}**