

API Documentation

This API documentation provides an overview of the endpoints and their functionalities for the StarkInfra client management system.

Sure! Here's the API documentation for the provided code:

Base URL

The base URL for all endpoints is: `https://divpix-backend.onrender.com/api`

Authentication

All endpoints require authentication by adding a header called `Authorization` with a Password (ask the development team if you need access to the online API). The authentication process ensures that only authorized users can access the API.

To authenticate, include the necessary authentication credentials in the request headers.

Error Handling

If an error occurs during the API request, the response will contain an error message and an appropriate status code.

The following status codes are used:

- 200: Success
- 201: Resource created successfully
- 400: Bad request, missing or invalid parameters
- 401: Unauthorized, authentication failed
- 404: Resource not found
- 500: Internal server error

Get Payment Options

GET `/api/get-payment-options`
POST `/api/get-payment-options`

Retrieve eligible payment options based on the provided data.

Request Methods

- GET: Retrieve eligible payment options.
- POST: Retrieve eligible payment options.

Parameters

- `final_user_id` (string): ID of the final user.
- `purchase_amount` (float): Amount of the purchase.
- `down_payment` (float): Amount of the down payment.
- `final_user_document` (string): Document of the final user.

Response

- 200 OK: JSON response containing the eligible payment options:

```
{
  "eligible_options": [
    {
      "number_splits": (int) Number of splits,
      "interest_rate": (float) Interest rate,
      "monthly_payment": (float) Monthly payment amount,
      "purchase_amount": (float) Purchase amount,
      "total_amount": (float) Total amount including interest
    }
  ]
}
```

- 400 Bad Request: If there are missing or invalid parameters or an invalid credit score. JSON response:

```
{
  "error": (str) Error message
}
```

Example Usage

Request

GET /api/get-payment-options?final_user_id=12345&purchase_amount=1000&down_payment=200&final_user_document=1234567890

Response

```
{
  "eligible_options": [
    {
      "number_splits": 1,
      "interest_rate": 0,
      "monthly_payment": 1000,
      "purchase_amount": 1000,
      "total_amount": 1000
    },
    {
      "number_splits": 3,
      "interest_rate": 0.05,
      "monthly_payment": 339.15,
      "purchase_amount": 1000,
      "total_amount": 1017.45
    },
    {
      "number_splits": 6,
      "interest_rate": 0.1,
      "monthly_payment": 175.53,
      "purchase_amount": 1000,
      "total_amount": 1053.18
    }
  ]
}
```

Please note that this documentation is based on the provided code snippet, and the actual behavior of the API endpoints may vary depending on the implementation and integration with other modules or services.

Sure! Here's the generated API documentation for the provided code:

API Documentation

Create Payment

API endpoint to create a payment and generate payment details.

Method: POST

Endpoint: /api/create-payment

Parameters

- `final_user_id` (str): ID of the final user.
- `purchase_amount` (float): Amount of the purchase.
- `number_splits` (int): Number of splits for the payment.
- `down_payment` (float, optional): Amount of the down payment.
- `interest_rate` (float, optional): Interest rate for the payment.
- `monthly_payment` (float, optional): Monthly payment amount.

Response

- **Success:** JSON response containing the payment details.

```
{
  "message": "Payment created successfully",
  "qr_code_copy": (str) QR code copy for down payment,
  "qr_code_img_link": (str) QR code image link for down payment,
  "due_dates": [(str) Due dates for split payments],
  "monthly_payment": (float) Monthly payment amount,
}
```

- **Error:** JSON response with an error message.

```
{
  "error": (str) Error message
}
```

Status code 400 is returned for missing fields or invalid values, and status code 500 is returned for an error during payment creation.

Please note that this documentation assumes the presence of additional imported modules and external functions (`authenticate`, `get_credit_score`, `api_split_payments.save_split_payments_to_database`, `api_payment_transactions.save_payment_transaction_to_database`). The documentation focuses on the API endpoints and parameters.

Create a New Client

POST /api/clients

Create a new client in the StarkInfra system.

Request Body

- `name` (string, required): Name of the user.
- `email` (string, required): Email address of the user.
- `role` (string, required): Role or position of the user within StarkInfra (e.g., admin, customer support).
- `created_at` (string): Timestamp indicating when the user account was created.
- `password` (string): Password used for login.

Response

- 201 Created: If the client is created successfully.
- 400 Bad Request: If the request body is missing or invalid.
- 500 Internal Server Error: If there's an error during the client creation process.

Get Information about a Single Client

GET /api/clients/{id}

Retrieve information about a single client in the StarkInfra system.

Path Parameters

- `id` (string, required): The unique identifier of the client.

Response

- 200 OK: JSON object containing the client's information if found.
- 404 Not Found: If the client is not found.
- 500 Internal Server Error: If there's an error during the retrieval process.

Delete a Single Client

DELETE /api/clients/{id}

Delete a single client from the StarkInfra system.

Path Parameters

- `id` (string, required): The unique identifier of the client.

Response

- 200 OK: If the client is deleted successfully.
- 404 Not Found: If the client is not found.
- 500 Internal Server Error: If there's an error during the deletion process.

Update Information of a Single Client

PUT /api/clients/{id}

Update information of a single client in the StarkInfra system.

Path Parameters

- `id` (string, required): The unique identifier of the client.

Request Body

- `client_id` (string): Unique identifier for each StarkInfra user.
- `name` (string): Name of the user.
- `email` (string): Email address of the user.
- `role` (string): Role or position of the user within StarkInfra (e.g., admin, customer support).
- `created_at` (string): Timestamp indicating when the user account was created.
- `password` (string): Password used for login.

Response

- 200 OK: If the client is updated successfully.
- 400 Bad Request: If the request body is missing or invalid.

- 404 Not Found: If the client is not found.
- 500 Internal Server Error: If there's an error during the update process.

Create a New Final User

POST /api/final_users

Create a new final user in the StarkInfra system.

Request Body

- `client_id` (string, required): Unique identifier for each StarkInfra user.
- `name` (string, required): Name of the user.
- `email` (string, required): Email address of the user.
- `password` (string, required): User password.
- `role` (string, required): Role or position of the user within StarkInfra (e.g., admin, customer support).
- `created_at` (string): Timestamp indicating when the user account was created.

Response

- 201 Created: If the user is created successfully.
- 400 Bad Request: If the request body is missing or invalid.
- 500 Internal Server Error: If there's an error during the user creation process.

Get Information about a Single Final User

GET /api/final_users/{id}

Retrieve information about a single final user in the StarkInfra system.

Path Parameters

- `id` (string, required): The unique identifier of the final user.

Response

- 200 OK: JSON object containing the user's information if found.
- 404 Not Found: If the user is not found.
- 500 Internal Server Error: If there's an error during the retrieval process.

Delete a Single Final User

DELETE /api/final_users/{id}

Delete a single final user from the StarkInfra system.

Path Parameters

- `id` (string, required): The unique identifier of the final user.

Response

- 200 OK: If the user is deleted successfully.
- 404 Not Found: If the user is not found.
- 500 Internal Server Error: If there's an error during the deletion process.

Create a New Split Payment

POST /api/split_payments

Create a new split payment in the StarkInfra system.

Request Body

- `split_payment_id` (string, required): Unique identifier for each split payment.
- `original_amount` (number, required): The total original amount of the split payment.
- `interest_rate` (number, required): The interest rate charged on the split payment.
- `due_date` (string, required): The due date for the split payment.
- `status` (string, required): Current status of the split payment (e.g., pending, paid, overdue).
- `payment_method` (string, required): The payment method used for the split payment (e.g., Pix).
- `final_user_id` (string, required): ID of the customer making the split payment.
- `client_id` (string, required): ID of the StarkInfra user managing the split payment.
- `created_at` (string): Timestamp indicating when the split payment was created.

- `total_amount` (number, required): The total amount of the split payment after applying interest.

Response

- 201 Created: If the split payment is created successfully.
- 400 Bad Request: If the request body is missing or invalid.
- 500 Internal Server Error: If there's an error during the split payment creation process.

Get Information about a Single Split Payment

GET `/api/split_payments/{id}`

Retrieve information about a single split payment in the StarkInfra system.

Path Parameters

- `id` (string, required): The unique identifier of the split payment.

Response

- 200 OK: JSON object containing the split payment's information if found.
- 404 Not Found: If the split payment is not found.
- 500 Internal Server Error: If there's an error during the retrieval process.

Delete a Single Split Payment

DELETE `/api/split_payments/{id}`

Delete a single split payment from the StarkInfra system.

Path Parameters

- `id` (string, required): The unique identifier of the split payment.

Response

- 200 OK: If the split payment is deleted successfully.
- 404 Not Found: If the split payment is not found.
- 500 Internal Server Error: If there's an error during the deletion process.

Update Information of a Single Split Payment

PUT `/api/split_payments/{id}`

Update information of a single split payment in the StarkInfra system.

Path Parameters

- `id` (string, required): The unique identifier of the split payment.

Request Body

- `split_payment_id` (string, required): Unique identifier for each split payment.
- `original_amount` (number): The total original amount of the split payment.
- `interest_rate` (number): The interest rate charged on the split payment.
- `due_date` (

string): The due date for the split payment.

- `status` (string): Current status of the split payment (e.g., pending, paid, overdue).
- `payment_method` (string): The payment method used for the split payment (e.g., Pix).
- `final_user_id` (string): ID of the customer making the split payment.
- `client_id` (string): ID of the StarkInfra user managing the split payment.
- `created_at` (string): Timestamp indicating when the split payment was created.
- `total_amount` (number): The total amount of the split payment after applying interest.

Response

- 200 OK: If the split payment is updated successfully.
- 400 Bad Request: If the request body is missing or invalid.
- 404 Not Found: If the split payment is not found.
- 500 Internal Server Error: If there's an error during the update process.

Get All Split Payments

GET /api/split_payments

Retrieve information about all split payments in the StarkInfra system.

Response

- 200 OK: JSON object containing the information of all split payments.
- 500 Internal Server Error: If there's an error during the retrieval process.

Get Split Payments by Due Date

GET /api/split_payments/due_date/{due_date}

Retrieve information about split payments with a specific due date in the StarkInfra system.

Path Parameters

- `due_date` (string, required): The due date of the split payments.

Response

- 200 OK: JSON object containing the information of split payments with the specified due date.
- 500 Internal Server Error: If there's an error during the retrieval process.

Create a New Payment Transaction

POST /api/payment_transactions

Create a new payment transaction in the StarkInfra system.

Request Body

- `transaction_id` (string): Unique identifier for each payment transaction.
- `split_payment_id` (string, required): ID of the split payment associated with the transaction.
- `amount` (number, required): The amount of the payment transaction.
- `status` (string, required): Current status of the payment transaction (e.g., processing, completed, failed).
- `transaction_date` (string, required): Timestamp indicating when the transaction took place.
- `payment_method` (string, required): The payment method used for the transaction (e.g., Pix).
- `client_id` (string, required): ID of the StarkInfra user initiating the transaction.
- `final_user_id` (string, required): ID of the customer involved in the transaction.
- `type` (string, required): Specifies if it's a down_payment or one of the split portions.
- `due_date` (string, required): Due date for the transaction to be effective.
- `qr_code_copy` (string, required): Code for making transactions "Pix Cópia e Cola".
- `qr_code_img_link` (string, required): Link for QR Code image generated.
- `stark_uuid` (string, required): UUID for the transaction.

Response

- 201 Created: If the payment transaction is created successfully.
- 400 Bad Request: If the request body is missing or invalid.
- 500 Internal Server Error: If there's an error during the payment transaction creation process.

Get Information about a Single Payment Transaction

GET /api/payment_transactions/{transaction_id}

Retrieve information about a single payment transaction in the StarkInfra system.

Path Parameters

- `transaction_id` (string, required): The unique identifier of the payment transaction.

Response

- 200 OK: JSON object containing the payment transaction's information if found.
- 404 Not Found: If the payment transaction is not found.
- 500 Internal Server Error: If there's an error during the retrieval process.

Delete a Single Payment Transaction

DELETE /api/payment_transactions/{transaction_id}

Delete a single payment transaction from the StarkInfra system.

Path Parameters

- `transaction_id` (string, required): The unique identifier of the payment transaction.

Response

- 200 OK: If the payment transaction is deleted successfully.
- 404 Not Found: If the payment transaction is not found.
- 500 Internal Server Error: If there's an error during the deletion process.

Update Information of a Single Payment Transaction

PUT /api/payment_transactions/{transaction_id}

Update information of a single payment transaction in the StarkInfra system.

Path Parameters

- `transaction_id` (string, required): The unique identifier of the payment transaction.

Request Body

- `split_payment_id` (string): ID of the split payment associated with

the transaction.

- `amount` (number): The amount of the payment transaction.
- `status` (string): Current status of the payment transaction (e.g., processing, completed, failed).
- `transaction_date` (string): Timestamp indicating when the transaction took place.
- `payment_method` (string): The payment method used for the transaction (e.g., Pix).
- `client_id` (string): ID of the StarkInfra user initiating the transaction.
- `final_user_id` (string): ID of the customer involved in the transaction.
- `type` (string): Specifies if it's a `down_payment` or one of the split portions.
- `due_date` (string): Due date for the transaction to be effective.
- `qr_code_copy` (string): Code for making transactions "Pix Cópia e Cola".
- `qr_code_img_link` (string): Link for QR Code image generated.
- `stark_uuid` (string): UUID for the transaction.

Response

- 200 OK: If the payment transaction is updated successfully.
- 400 Bad Request: If the request body is missing or invalid.
- 404 Not Found: If the payment transaction is not found.
- 500 Internal Server Error: If there's an error during the update process.