**Residential Mortgage Securities (RMBS) Credit Rating - Documentation:**

1. **Overview:**

This application provides a web interface for managing and evaluating residential mortgages. Users can input mortgage data, calculate credit ratings, and perform CRUD operations on mortgage records.

1. **Architecture:**

The application follows a three-tier architecture:

**Frontend**: A React.js application providing the user interface.

**Backend**: A Django REST Framework API handling data processing and logic.

**Database**: A MySQL database for data storage.

1. **Database:**

**Database**: MySQL

**Database Location**: database/mortgage\_db

**Schema Details**:

The database includes a mortgages table with the following schema:

id (auto-increment, primary key), credit\_score, loan\_amount, property\_value, annual\_income, debt\_amount, loan\_type, property\_type, credit\_rating

1. **Backend Setup:**

**Language**: Python (Django REST Framework)

**Dependencies**:

Install the required Python packages:

pip install -r requirements.txt

**API Endpoints**:

Add Mortgage:

POST: http://127.0.0.1:8000/api/mortgage/add/

Calculates the credit rating and stores the mortgage record in the database.

List Mortgages:

GET: http://127.0.0.1:8000/api/mortgage/list/

Retrieves all mortgage records from the database.

Edit Mortgage:

PUT: http://127.0.0.1:8000/api/mortgage/edit/<id>/

Updates the mortgage record using specified ID and recalculates the credit rating.

Delete Mortgage:

DELETE: http://127.0.0.1:8000/api/mortgage/delete/<id>/

Deletes the mortgage record using specified ID.

Logging:

Logs are stored in database/logs/mortgage.log.

Logging configuration is managed through environment variables.

Running the Backend:

Run the Django development server:

python manage.py runserver

1. **Frontend Setup:**

**Language**: React.js

**Dependencies**:

Install the required packages:

npm install

**Routes:**

Mortgage List: /

Displays the mortgage list with edit and delete button.

Add Mortgage: /add

Display a form for entering new mortgage data and submit to backend API.

Edit Mortgage: /edit/:id

Provides a form for editing using id and display the existing data using backend API.

**Running the Frontend:**

Go to the frontend directory.

Start the React server:

npm start

1. **Application Usage:**

**Add Mortgage:**

Link: http://localhost:3000/add.

**View Mortgages:**

Link: http://localhost:3000/.

**Edit Mortgage:**

Click on Edit button in mortgage list.

Link: http://localhost:3000/edit/:id

**Delete Mortgage:**

Click on Delete button in mortgage list.