University: San Jose State University
Course: Enterprise Software - CMPE172/Spring 2020
Team Members:
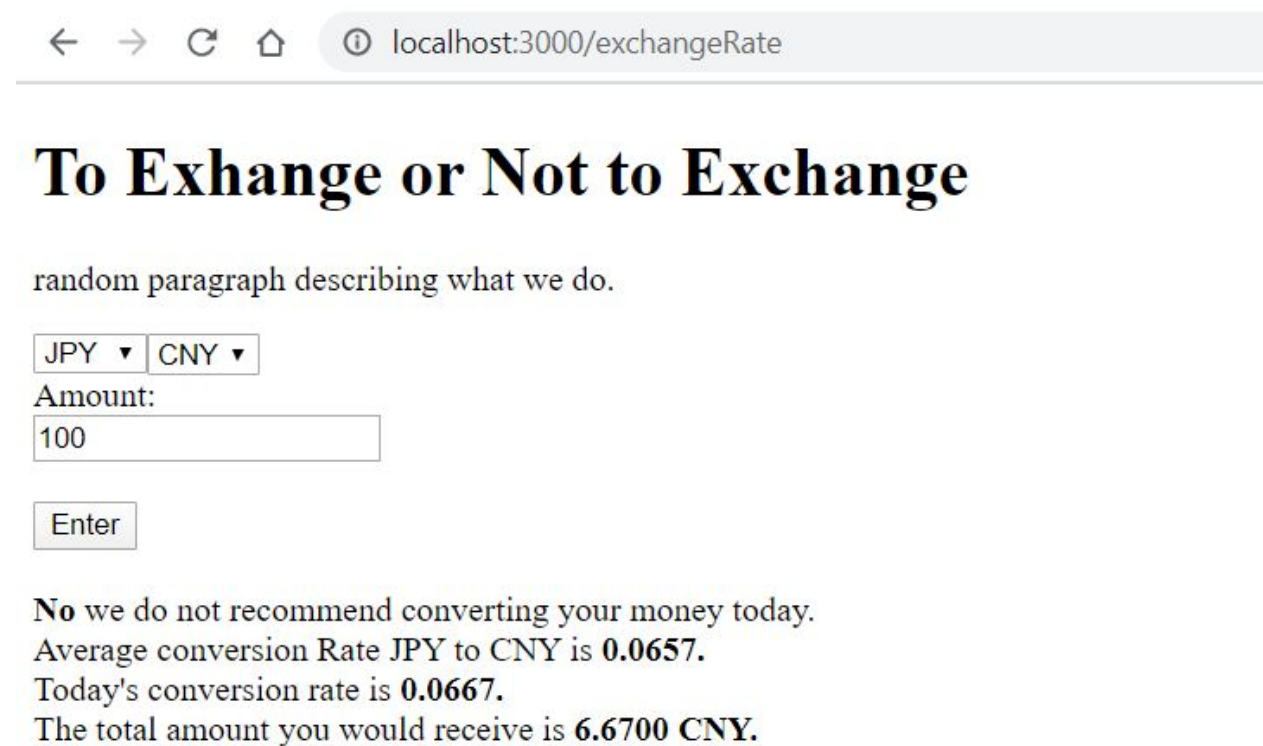   Abhinav Sarma
        Ishant Sharma
        Keshuv Bagaria

Project Introduction:

        In order to help our users to make a decision on whether or not they should exchange money, our application takes an average of the exchange rates between the two currencies for the past 30 days and compares that to the current exchange value. We display this data to the customer and also give our recommendations based on these numbers.

Sample Demo Screenshots:

To Exhange or Not to Exchange

random paragraph describing what we do.

CAD ▾ USD ▾
Amount:
100

Enter

Yes we recommend converting your money today.
Average conversion Rate CAD to USD is **0.7112.**
Today's conversion rate is **0.7099.**
The total amount you would receive is **70.9900 USD.**

To Exhange or Not to Exchange

random paragraph describing what we do.

CAD ▼ INR ▼
Amount:
50

Enter

Yes we recommend converting your money today.
Average conversion Rate CAD to INR is **54.0813.**
Today's conversion rate is **53.8283.**
The total amount you would receive is **2691.4150 INR.**

Prerequisites for setup:

  List of required software to download:
  1. Docker
  2. Node.js

  Setup Steps/Build Scripts:
  1. Node.js: run "npm install" in the folder where you save the node.js file. The set up will download the relevant dependencies to run the middleware.
  2. Maven Build for the Java code: Use command: "mvnw clean package" to create the jar file that would be used inside the Docker. This jar file will contain the java code to connect to the API and send data into the RDS database. The jar file is also attached to the project implementation canvas page.
  3. Docker:
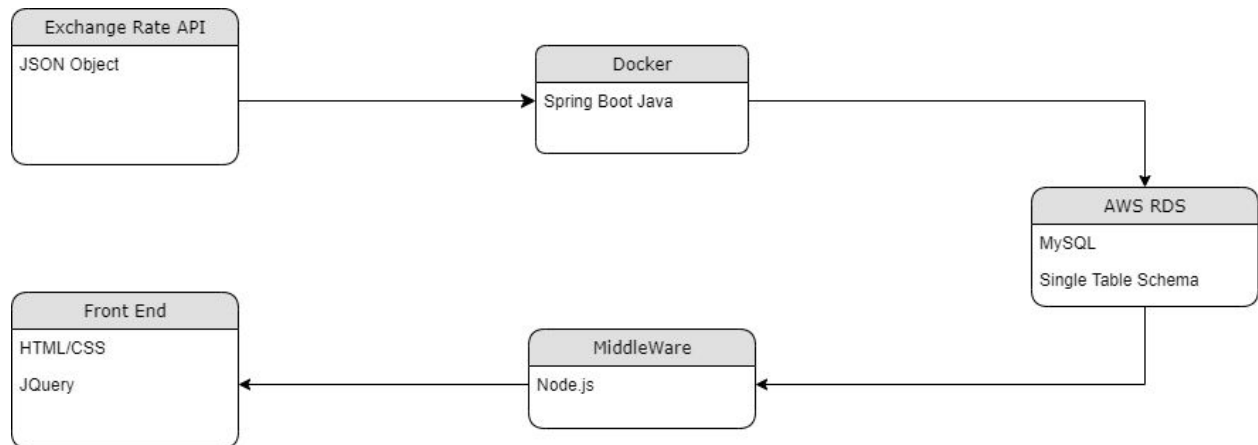   https://medium.com/@wkrzywiec/how-to-put-your-java-application-into-docker-container-5e0a02acdd6b
     Use the following link to create a dockerfile and copy the jar file created in the previous step into the dockerfile.

Instructions on how to run the project locally:

  Refer to the DBConnection.json file that was attached in the Project Report. Save that file in the same folder as your 'index.js' (node module). Save the html file in the same folder as

well. Then simply run that node file. "localhost:3000/exchangeRate" URL would display the webpage.

Sequence Diagram:

Exchange Rate API
JSON Object

Docker
Spring Boot Java

AWS RDS
MySQL
Single Table Schema

Front End
HTML/CSS
JQuery

MiddleWare
Node.js

Schema: