

Documentation

Introduction

Our currency exchange system has added multiple functionalities to what we have done in the labs so far. The features that we added are:

- plots to help visualize exchange rate formation
- user-specific plots to show each user his/her contributions
- tables to show transactions with tellers and transactions between users
- insights on how each transaction affects the exchange rate
- inter-user transactions where users can request certain amounts of money at their own rates and other users can accept such exchanges (timeline)
- user balance (which is ideally linked to each user's bank account) to indicate the amount of money (both USD and LBP) that the user has and which plays a role in the requests posted/accepted and transactions recorded
- smarter calculator
- navigation bar

Use Cases

As a ...	I want to ...	So that ...
User	View current exchange rates based on the past 3 days	I know what exchange rates to expect
User	Record a transaction	I contribute to the rates' accuracy
User	View my transactions with tellers and with other users	I keep track of my exchanges
User	Create an account	I can make and accept requests linked to my balance
User	View my balance	I can know the exact amount of money I have before posting or accepting an exchange request
User	Post an exchange request	I can make a request at a better rate than teller ones
User	Delete an exchange request that I previously posted	I remove my money from being on hold if I no longer need the transaction

User	View posted exchange requests	I can accept a request if it benefits me
User	Accept another user's request	I exchange money with the user and have my balance updated
User	View insights about the exchange rates	I understand better the formation of exchange rates
User	View plots of the exchange rates	I can visualize the fluctuations
User	View plots of my transactions	I can visualize my exchanges
User	Calculate LBP value from USD or USD value from LBP	I know the exact equivalents of LBP/USD inputs based
Developer	Implement registration checks	I guarantee that user credentials are secure
Developer	Implement login checks	I make sure that only authenticated users can access their private information
Developer	Disallow negative numbers and zeros in all inputs	I reduce false information and reduce inaccuracies in the rates
Developer	Disallow users to post requests with amounts of money they do not have	I prevent errors in exchanges where the user does not have the money he/she needs to exchange
Developer	Disallow users to accept requests with amounts of money they do not have	I prevent errors in exchanges where the user does not have the money he/she is accepting to exchange
Developer	Hold user's money that he/she has placed requests for	They do not use money they do not have (user should delete request so that his/her money is not on hold anymore)

Activity Diagram (High-Level Architecture)

