Testudo Bank Certificate of Deposit Program

Owner: Justin Lumpkin (jl1@umd.edu)
Last updated: 22 April 2024

Problem Statement

A **Certificate of Deposit (CD)** program will excite customers with a new way to earn interest and combat inflation. Simultaneously, it provides TestudoBank with additional capital to use for issuing loans and investing in securities. This diversifies our revenue stream, making us less reliant on fees, which generate revenue at the expense of customer satisfaction.

Solutions Requirements

- Customers can purchase a CD and earn interest in exchange for losing access to their funds for a specified term
- Customer can view the status of any CDs they hold
- Once a CD matures, customer can redeem their funds and interest
- We should strive for risk-aversion
    - Even if we cancel the program, we still need to honor the CDs that have already been purchased!
    - It is better to have slow uptake, which we can increase with better rates, than quick uptake that costs TestudoBank an excessive amount of money.

Solutions Considered

The solution needs to

- set the parameters of the CD (yield, term, minimum deposit) and determine whether multiple parameter combinations will be available.
- specify the technical details of how CDs are stored with in the TestudoBank Database.

I consider these aspects of the solution separately because solutions to each part can be seamlessly mixed and matched.

Yield and Term Parameters

Option 1: Single yield/term/minimum deposit option

I only implement one CD yield and term combination.

Pros:

- A single CD option maximizes development speed.
- We can choose a rate based on the industry standard, which is likely to be low-risk.

Cons:

- Due to a lack of options, customers may not find this incentive enticing enough to invest at TestudoBank.
- In a future transition to multiple yield terms, we will need to refactor code to enable customer choices

Option 2: Multiple yield/term/minimum deposit options

This is the inverse of the above case
Pros:

- Customers have more choices, and are more likely to join or stay at TestudoBank in order to take advantage of it.
- We can learn more about what priorities our customers have to better serve their needs.
- Once TestudoBank gains more experience in this field, we can build fully customizable CDs in which customers choose DepositAmount and Term and an internal formula will offer an interest rate

Cons:

- This presents a bit of a **1-way door** in that TestudoBank is stuck paying out whatever rates it chooses once a customer purchases the CD. In other words, this exposes TestudoBank to more risk than Option 1.

CD Representation in the TestudoBank Database

Option 1: Track Customer Holdings

In this approach I maintain a table with customers as the primary key which maps to all of the holdings the customer has.

Pros:

- Most of the time we process this table, it will probably be to access one customer at a time. This design makes that operation intuitive. You simply select the row corresponding to a customer. This matches the precedent of maintaining `CryptoHoldings`

Cons:

- Customers can have multiple holdings. SQL doesn't support a list data type natively, so the standard approach would be to maintain another table which stores one CD per row and attaches the necessary metadata.
    - This means we need to build almost all of the logic of **Option 2: Track CD History** in addition to building a holdings table.

Option 2: Track CD History

In this approach we maintain only the `CertificateOfDepositLogs` table. Then to get records associated with a customer, we must run a `SELECT` query on the customer ID.

Pros:

- The `CertificateOfDepositHoldings` table doesn't seem useful because there is too much metadata that goes with each CD (time of purchase, time of maturing, deposit amount, interest rate, etc). In contrast, `CryptoHoldings` only needs to store a single float representing how much of the cryptocurrency a customer has.
    - This means that diverging from the precedent is probably justified

Cons:

- This diverges from the `CryptoHoldings` and `CryptoHistory` precedent

Proposed Solution

Yield and Term Parameters

I will implement a single yield, term, and minimum deposit setting (**Option 1**). This will limit risk to TestudoBank by avoiding a 1-way door and simplify the development within the `MVCController` class.

Term: I fix the term to be 1-year, meaning we do not need to consider conversion between APY and other units of time.

Minimum Deposit: I will set the minimum deposit at $100 to ensure it is enough money for TestudoBank to purchase an interest-bearing asset.

Yield: I propose a rate of 1.75% to make us slightly more competitive than the average bank, while limiting risk.

Database Representation

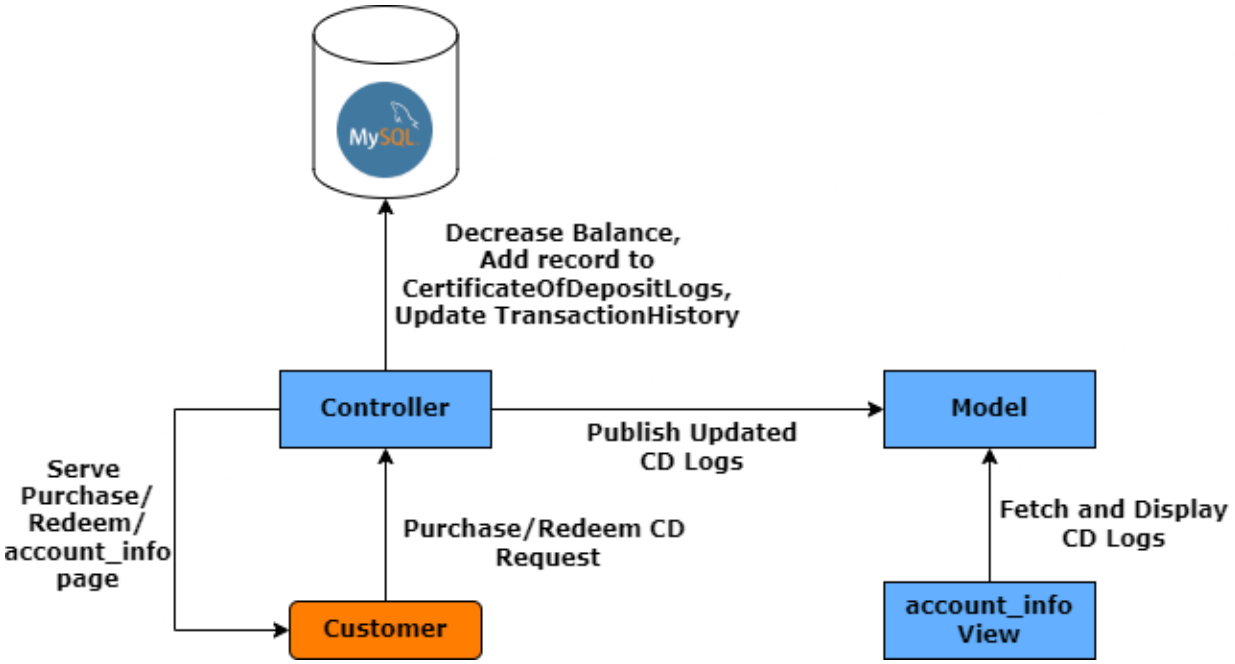
I store only the `CertificateOfDepositLogs` (**Option2**). This simplifies the implementation by only requiring one additional table to maintain.

Customer View

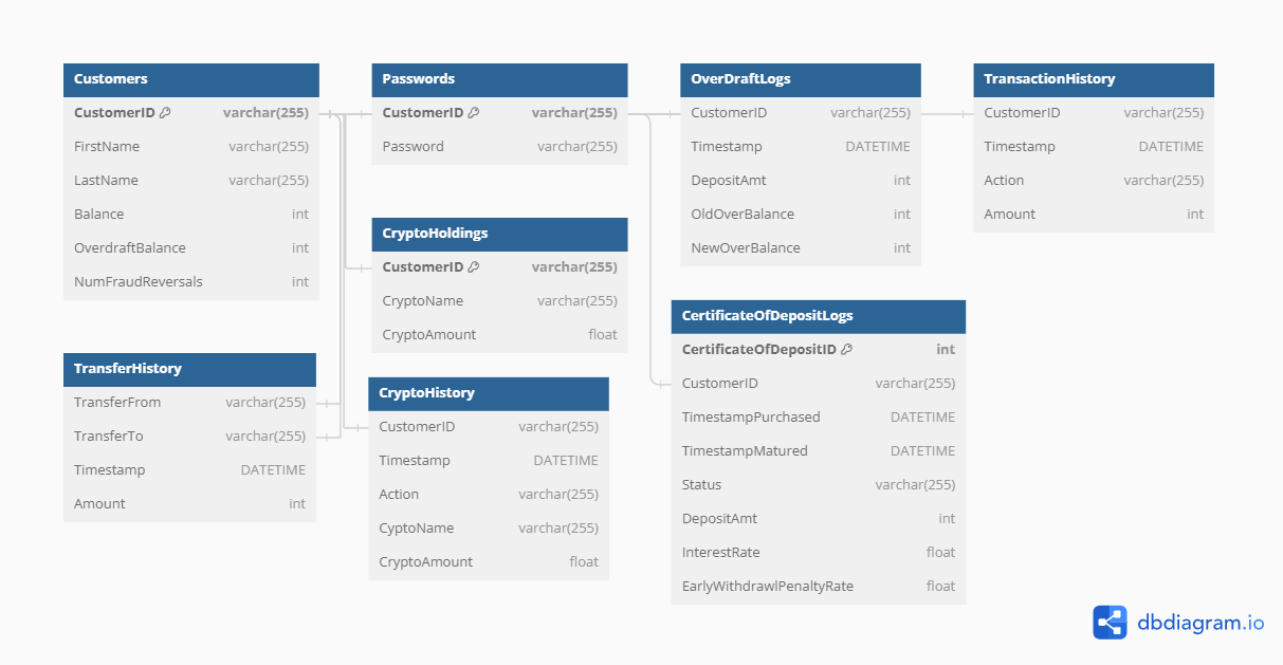
I will use separate pages for purchasing and redeeming CDs. The customer can view CDs on the `Account Info` page.

- To match precedent, I will also add `PurchaseCD` and `RedeemCD` as actions in the TransactionHistory table.

MVC Diagram



MySQL DB Schema



DB Notes

I add the `CertificateOfDepositLogs` table.

- The `CertificateOfDepositID` gives users a reference when they decide which CD to redeem
- I store `TimestampPurchased` and `TimestampMatured` in lieu of a CD term because SQL has a native `DATETIME` type but no native `TIMEDELTA`.
    - This also enable quick implementation of new CD terms.
- `Status` will indicate whether a CD is still held by the customer or if it has been redeemed
- `DepositAmt` the amount of money the customer put into the CD
- `InterestRate` and `EarlyWithdrawlPenaltyRate` are attributes of the CD. Including them in the database will make it easy to add more options in the future

Testing

Test Cases:

- Customers can purchase a CD with $100 or more
- Customers cannot purchase a CD with $99.99
- CD Redemption after the maturity deadline will result in the initial deposit being repayed plus earned interest of 1.75%
- CD Redemption before the maturity deadline will result in the initial deposit being repayed less an early withdrawl penalty of 10%

Appendix

- The current average CD rate in the US for a 1-year CD is 1.73% APY.
- Customers can overdraft for a fee of of 2%. Keeping CD interest rates lower than this amount ensures that customers cannot net money by purchasing a CD on overdraft funds.