FRM
Functional Specification and Software Design

Short Description:	Project 1 ALM Model
Requirement Number:	
Functional Specification Owner:	HC Inc.
Software Design Owner:	
Test Plan Owner:	
Applicable Releases:	1.0
Clients	ABC Bank
Attachments	Model (Excel file), User Manual

1. Document Revision History

Revision	Date	Name	Description of Change
0.0	Feb 4, 2018	John Gary	Initial Version

2. Description of Request

ABC Bank requires a model to manage the interest rate risk (IRR) and liquidity risk (LR) of its Balance Sheet while maintaining a Cash Account.

3. Financial Model Specification

The goal is to simulate a model that helps the bank manage liquidity and interest rate risk to reduce liquidity risk exposure in ABC's Bank Balance Sheet within a one-month period. In ALM, Liquidity Coverage Ratio (LCR) is a stock-based approach to measuring liquidity risk. The model must ensure that at the end of each simulation time t_{sim}, LCR(t_{sim}) >= 1 + *e*. If this restriction is violated then the Bank must employ risk management strategies by selling low rated, low yield assets to finance the purchase of high quality liquid assets.

Additionally, the bank has a Cash Account in which any net funding surplus (Net cash outflows are negative) is allocated to. If there are outstanding net cash outflows, the Cash Account balance is used to pay them out, followed by the sale of high quality liquid assets with the lowest yield. At the end of t_{sim}, after carrying out the proposed risk management strategy, excess cash in

the Cash Account greater than buffer B is reinvested.

Moreover, the model will analyze the durations of assets and liabilities on the Balance Sheet to ensure the Duration Gap, DGap(t_{sim}) is in the range, [-c, h]. If DGap(t_{sim}) is out of range, the model will recommend a strategy to keep high yield assets and trade short duration assets for long duration ones or vice versa to minimize |DGap(t_{sim})|.

Finally, the model will simulate step-by-step (in one month increments) from tsim, where at the beginning and the end of each month a Risk Analysis report is presented. This report will analyze the LCR, Cash Account and Duration Gap. If a problem is detected (values not falling within limits) the model will notify the user and provide a resolution to the problem. The user will then agree to execute one of the resolutions (balance sheet and cash account will be updated and risk measures will be again reported) or chose to do nothing. The model will then move to the next tsim (tsim + 1 month).

Functions needed:

Sell an asset from the Balance Sheet
Buy an asset from the bond market
Calculate cash flows of a bond for the time [*tsim*, *tsim* + 1 month)
Aggregate cash outflows and inflows of the Balance Sheet for the time [*tsim*, *tsim* + 1 month)
Calculate Net Cash Outflows (NCF) for the time [*tsim*, *tsim* + 1 month)
Calculate SHQLA = PV of high quality liquid assets (including cash) at time *tsim*
Calculate LCR = SHQLA / NCF at time *tsim*
Sell low rated-low yield assets and buy high quality liquid assets until LCR >= 1 + e
Calculate cash at time *tsim* + 1 month
While cash at time *tsim* + 1 month < 0, sell highest quality assets with lowest yield
Check if cash at time *tsim* < B and buy high quality assets until Cash at time *tsim* < B
Calculate value of assets
Calculate value of liabilities
Calculate duration of assets
Calculate duration of liabilities
Calculate Duration Gap = $D_A - (V_L / V_A) D_L$ at time tsim
Sells expired bond
Creates an array of all yield in bond market
Creates an array of all PV in bond market
Creates an array of all duration in bond market
Returns bond yield based on bond id
Returns PV based on bond id
Returns duration based on bond id
Calculates the threshold for high yield assets
Buy short duration assets and sell long duration assets until DGAP at *tsim* <= h
Buy long duration assets and sell short duration assets until DGAP at *tsim* >= -c
Report LCR, cash in Cash Account, and Duration Gap at the beginning and end of each time interval and apply necessary risk management strategies (including updating cash account and Balance Sheet) if user so chooses
Proceed to next *tsim*
Initiate simulation and provide Risk Analysis Report of risk measures LCR, cash, and DGAP

4. Functional Specification

4.1 Model Function Design

4.1.1 Add a VBA Function that when asked to sell an asset from the Balance Sheet first checks if the asset is in the array asset() and then if the number of units of that asset is > 0. If so, then adjust cash by:

$$\text{Cash}(tsim)_{new} = \text{Cash}(tsim)_{old} + \text{PV(bond)}$$

In addition, decrease the number of units of that asset in the asset() array by 1. If the number of units is equal to 1, remove the asset from asset().

4.1.2 Add a VBA Function that when asked to buy an asset from the bond market adjust cash by:

$$\text{Cash}(tsim)_{new} = \text{Cash}(tsim)_{old} - \text{PV(bond)}$$

In addition, if the bond is in the asset() array, then increase the number of units of that asset by 1. Otherwise, add the bond to asset() and set number of units as 1.

4.1.3 Add a VBA Function that calculates cash flows of a bond that occur in the time range [*tsim*, *tsim* + 1 month). After generating all of a bonds cash flows, at *tsim*, the current period cash flows of a bond should be:

Cash flows(*i*, *tsim*) = sum of cash flows for bond *i* which have associated date in the range [*tsim*, *tsim* + 1 month).

4.1.4 Add a VBA Function that aggregates cash inflows and cash outflows in the time range [*tsim*, *tsim* + 1 month). After loading the Balance Sheet, cash inflows are calculated from assets as:

Cash inflows(*tsim*) = sum of Cash flows(*i*, *tsim*) for all assets *i* in the Balance Sheet
Cash outflows(*tsim*) = sum of Cash flows(*i*, *tsim*) for all liabilities *i* in the Balance Sheet

4.1.5 Add VBA Function that calculates net cash outflows:

NCF(*tsim*) = Cash outflows(*tsim*) – Cash inflows(*tsim*)

4.1.6 Add VBA Function that calculates present value of high quality assets (with AAA rating and maturity less than 1 year) in Balance Sheet and adds cash to a variable SHQLA:

SHQLA($tsim$) = present value at $tsim$ of high quality assets + cash

4.1.7 Add VBA Function that calculates Liquidity Coverage Ratio at simulation date $tsim$.

LCR($tsim$) = SHQLA($tsim$) / NCF($tsim$)

4.1.8 Add a VBA Function that finds the bond in the bond market that has an AAA rating and maturity less than 1 year and the highest PV and adds it to some variable TOBUY.

4.1.9 Add a VBA Function that finds the assets with the lowest ratings in the Balance Sheet and sorts them by yield. Then add the present value of the asset with the lowest rating and lowest yield to some variable TOSELL.

4.1.10 Add a VBA Function that performs while LCR is < 1+e. Based on choices of assets TOBUY and TOSELL, execute transactions. If cash < 0 continue selling the lowest rated, lowest yield assets until cash is >=0

4.1.11 Add a VBA Function that checks if Cash($tsim$) >= 0. If not then while Cash($tsim$) < 0, sell the lowest yield, high quality asset in the Balance Sheet. Adjust the array asset() and cash accordingly.

4.1.12 Add a VBA Function that performs while Cash >= B. Choose the bond with the highest yield among all available high quality bonds in the bond market. Buy the bond until Cash < B and add it to asset() and update cash accordingly. Note that because we are buying high quality assets, the value SHQLA will remain unchanged.

4.1.13 Add a VBA Function that calculates market value of assets where:

$V_A(tsim)$ = sum of present value of all assets

4.1.14 Add a VBA Function that calculates market value of liabilities where:

$V_L(tsim)$ = sum of present value of all liabilities

4.1.15 Add a VBA Function that calculates total modified duration of assets where:

$D_A(tsim)$ = weighted average of asset durations (weighted by present value)

4.1.16 Add a VBA Function that calculates total modified duration of liabilities where:

$D_L(tsim)$ = weighted average of liability durations (weighted by present value)

4.1.17 Add a VBA Function that calculates the duration gap at time ($tsim$). We can find

the duration gap by calculating:

$DGap(tsim) = D_A(tsim) - (V_L(tsim)/V_A(tsim))\ D_A(tsim)$

4.1.18 Add a VBA Function that performs while DGAP(*tsim*) is > h. Of the non-high yield assets, this function chooses the bond with the highest duration TOSELL.

Conditional on the bond having a smaller present value than the bond being sold, it chooses lowest duration bond TOBUY. Execute transactions and adjust asset() and cash accordingly.

4.1.19 Add a VBA Function that performs while DGAP(*tsim*) is < -c. Excluding high yield assets, choose the bond with the lowest duration TOSELL.

Conditional on the bond having a larger present value than the bond being sold, choose the highest duration bond TOBUY. Execute transactions and adjust asset() and cash accordingly.

4.1.20 Add a VBA Function that at *tsim*, sells any assets that have maturity <= *tsim*.

4.1.21 Add a VBA Function that generates an array that calculates the yield of every bond in the market.

4.1.22 Add a VBA Function that generates an array that calculates the PV of every bond in the market.

4.1.23 Add a VBA Function that generates an array that calculates the duration of every bond in the market.

4.1.24 Add a VBA Function that given a Bond ID, returns the yield of that bond.

4.1.25 G Add a VBA Function that given a Bond ID, returns the PV of that bond.

4.1.26 Add a VBA Function that given a Bond ID, returns the yield of that bond.

4.1.27 Add a VBA Function that calculates the max yield and the min yield for all assets in the balance sheet and returns a value equal to:

High Yield Boundary = (Max yield + Min Yield) * 0.75

This value is the threshold for high yield assets.

4.1.28 Add a VBA Function that reports risk measures for LCR, DGAP, and cash, and proposes the above risk management strategies if the risk measures do not fall within their limits and runs the functions until all of the constraints are satisfied, if the user does agrees to proposed strategy. Once complete, report the new risk

return the new asset() array.

4.1.29 Add a VBA Function that, when prompted to go to next time-step, calculates new *tsim* as old *tsim* + 1 month (default time-step) and calculates new cash as:

Cash(*tsim*) = Cash(*tsim* – 1 month) – NCF(*tsim* – 1 month)

Once the time horizon set by the user is reached (default is 6 months/time-steps), it should be recommended that the user terminate the simulation.

4.1.30 Add a VBA main Macro that is linked to a button in the excel file. When clicked, it should initiate the simulation by creating yield, present value, and duration arrays, running the risk management procedure, writing risk measures to the Risk Analysis Report sheet and when finished, prompt the user to move to the next *tsim*. This is accomplished by using the 29s functions outlined above.

4.2 GUI Design

4.2.1 Input GUI: Balance Sheet in spread sheet: “Balance Sheet”, Bond Market in spread sheet: “bonmkt”, Exchange Rates in spread sheet: “SpotFX”, Simulation Time, Time Horizon, Time Step, Liquidity Tolerance e, Cash Buffer B, Duration Gap range [-c, h]

4.2.2 Output GUI: Risk Analysis Report in spread sheet: “Risk Measures Report”, risk measures: LCR, DGAP, and cash, and current/end dates

5. Validation

Testing special case of DGap(*tsim*) when V_A(*tsim*) = 0
Testing special case of LCR(*tsim*) when NCF(*tsim*) = 0
Testing special case when all risk measures fall within required constraints
Testing special case when user inputs are not in acceptable range
Testing special case when there are many assets with the same rating

6. References

Alan Yang, “ALM: Interest Rate Risk Course Note,” for ECO2508, Jan 2018.
Alan Yang, “ALM: Liquidity Risk Course Note,” for ECO2508, Jan 2018.