

**Title: ABABIL.INV.04.02 - REACTIVATE FINANCING ACCOUNT**

**Scope:** The purpose of this test is to verify the functionality of reactivating a financing account.

**Test Environment**

**Name Of The Web Based Application Under Test:** ABABIL

**Nature Of The Web Based Application Under Test:** Core Islamic Banking Software Solutions

**Name Of The Web Browser:** CHROME (Current Version), MOZILLA FIREFOX (Current Version)

**Name Of The OS:** Windows 10

**Test Procedure and Verification**

**Explanation of terms:**

**S<number>:** stands for identification of a test procedure step.

**V<number>:** stands for identification for the corresponding verification(s).

**V<number>:** N/A stands for verification is not required for this step.

**Input/Output Dataset Information:**

The input/output datasets referenced in this test case are stored, by the test case name, in [Storage Name], under "[Storage Location Path]".

The tools referenced in this test case are stored, by tool classification, in [Storage Name], under "[Storage Location Path]".

[ Detail Run ]

S1: Launch, Login And Navigate To Ababil → **Finance** Module:

Logon to Ababil application. Once the Ababil home page appears, click on the icon of the "**Finance**" module, then dropdown the Finance main menu, click on the Investment menu item and then click **Account**.

V1: Verify that the **Account** UI appears as per expectation. The "**Looking for Financing account? Search here**" label will be present on the page. Take a screenshot of the page and save to the designated storage for record keeping. [Screenshot 01]

S2: Select a closed account of same branch

V2: Verify that a list of closed accounts will appear if searched by own branch and status as **Closed**.

Take a screenshot of the page and save to the designated storage for record keeping. [Screenshot 02]

S3: **Reactivate** Account

V3: Verify that the **“Financing Account Details”** will appear at the top of the page and **Reactivate** will appear at the right top of the interface.

Also verify that, **“Financing account reactivated successfully”** appears at the pop up.

Also verify that **“Looking for Financing account? Search here”** label will appear at the top of the page.

Take a screenshot of the page and save to the designated storage for record keeping. [Screenshot 03]

S4: **Reactivate** Account

V4: Verify that a pop up will appear as **“Task sent for verification. Account no: \*\*\*\*\*”**.

Also verify that the number showing in the pop up is the same account number that is about to reactivate.

Take a screenshot of the page and save to the designated storage for record keeping. [Screenshot 04]

S5: Verify and Accept From My Task:

If the task is sent to verify then login as a verified user and verify the required details. Then click on the accept button.

V5: Verify that the success popup message will be displayed **“Financing account reactivated successfully. Account Number: \*\*\*\*\*”**.

Also verify that the number showing in the pop up is the same account number that is reactivated.

Save the Voucher ID.

Take a screenshot of the page and save to the designated storage for record keeping. [Screenshot 05]

S6: Risk and Hazard:

Throughout this testing, make sure all labeling, including messages, icons and messages of operation

guidelines are accurate, written in short concise sentences, and written in simple and familiar words.

V6: Verify following items wherever appropriate:

Throughout this testing verify the AUT based on the following viewpoints:

- i) Make sure that the user interface is simple, easy to understand and screen designs are clear, concise, consistent, complete and unambiguous.
- ii) Make sure that the abbreviations, symbols, text and acronyms placed on or displayed by the AUT are consistent and unambiguous.
- iii) Make sure that the AUT provides immediate and clear feedback following user entries, whenever necessary.
- iv) Make sure that the operation steps are easy-to-remember.
- v) Make sure that the prompts, menus, etc. are used to cue the user regarding important steps.
- vi) Make sure that the AUT does not hang during run time or "strand" the user.
- vii) Make sure that the AUT provides the users useful information in the case of an error. Make sure that the AUT provides conspicuous mechanisms for correction and troubleshooting guidance.
- viii) Make sure that the AUT does not overload or confuse the users with information that is unformatted, densely packed or presented too briefly.
- ix) Make sure that the use of symbols, icons, colors and abbreviations are acceptable to convey information reliably, precisely and quickly.
- x) Make sure that dedicated display mechanisms are used for highly critical and time sensitive information.