|  |  |
| --- | --- |
| File Name | UC\_TransactionManager\_DeleteTransaction\_.docx |
| Use Case Name | Transaction Manager Deletes Transaction |
| Project | Account Project v03 |
| Author | Bob Trapp |
| Date | 2017-11-03 |
| Specification Document | NA (See Assignment for Project Version 3) |
| Parent Document |  |
| Use Case Diagram | TransactionManagementUseCaseDiagram.png |

|  |  |  |
| --- | --- | --- |
| **Mod Date** | **Mod By** | **Mod Description** |
| 2017-11-02 | Bob Trapp | Initial Creation |
|  |  |  |

|  |  |
| --- | --- |
| Goals | The Transaction Manager wishes to completely delete an Transaction record. |
| Primary Actor | * Transaction Manager |
| Secondary Actor(s) | * DBMS |
| Pre-Conditions | * The Transaction to be deleted exists. * The Transaction Manager is at the Transaction Management Menu |
| Post Conditions | * The deleted Transaction record has been stored to the Deleted Transaction storage along with the date of deletion, the reason for deletion and the user that deleted the Transaction. * The Transaction Manager is back to the Transaction Management Menu |
| Basic Flow | 1. The Transaction Manager selects Delete Transaction from the Transaction Management Menu 2. The System shows the available Transactions 3. The System prompts the Transaction Manager to select an Transaction for deletion 4. The Transaction Manager specifies an Transaction to delete and submits 5. The System verifies the Transaction 6. The System requests the reason for the deletion 7. The Transaction Manager enters the reason for the deletion and submits 8. The System verifies the reason 9. The System sends the delete call to the DBMS, including the reason 10. The DBMS copies the Transaction to the Deleted Transactions store, along with the reason and the user. 11. The DBMS deletes the Transaction from the Transactions store 12. The DBMS returns a result with the number of records affected 13. The System displays a success message 14. The Transaction Manager acknowledges the success 15. The System displays the Transaction Management Menu |
| Alternate Flow(s) | 1. If there are no Transactions available for deletion (Extension Point: Step 2)    1. The System displays a message stating that there are no Transactions available    2. The Transaction Manager acknowledges the message    3. The System goes to Step 15 2. If the Transaction Manager specifies an Transaction that is not available (Extension Point: Step 5)    1. The System displays an error message    2. The Transaction Manager acknowledges the error message    3. The System returns to Step 2 3. If the entered reason for deletion is not valid (Extension Point: Step 8)    1. The System displays an error message    2. The Transaction Manager acknowledges the error message    3. The System returns to Step 6 4. If the DBMS cannot delete the Transaction (Extension Point: Step 11)    1. The System displays an error message    2. The Transaction Manager acknowledges the error message    3. The System goes to Step 15 5. If the Transaction Manager chooses Cancel    1. The System asks the Transaction Manager to confirm the Cancel    2. If the Transaction Manager confirms the Cancel, the System goes to Step 15    3. If the Transaction Manager de-confirms the Cancel, the System returns to the step where the Cancel was originally chosen |
| Open Issues | 1. There are no open issues |
| Remarks | 1. The Transaction Manager may choose to Cancel until the reason for deletion has been submitted successfully and the System is sending data to the DBMS. 2. To enforce substantial entries, the reason for deletion must be at least ten characters long, but can be appreciably longer. 3. If the Transaction cannot be deleted from the Transaction store, then the deletion record must be converted to an Attempted Deletion in the Deleted Transaction store. 4. All error messages displayed to the Transaction Manager must be informative and explain what happened in non-technical terms. |