|  |  |
| --- | --- |
| File Name | UC\_TransactionManager\_InvalidateTransaction.docx |
| Use Case Name | Transaction Manager Invalidates 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 mark an Transaction as being no longer usable (invalid). |
| Primary Actor | * Transaction Manager |
| Secondary Actor(s) | * DBMS |
| Pre-Conditions | * The Transaction is stored in the data store and is currently marked as valid * The Transaction Manager is currently at the Transaction Management Menu |
| Post Conditions | * A new Transaction record exists in the data store that undoes the effect of the invalidated Transaction * The specified Transaction has been marked as invalid in the data store with the Transaction ID of the new Transaction that invalidated it * The invalidation date, user, and reason are stored in the data store * The Transaction Manager is back to the Transaction Management Menu |
| Basic Flow | 1. The Transaction Manager chooses Invalidate Transaction from the Transaction Management Menu 2. The System displays the available Transactions 3. The System prompts the Transaction Manager to select an Transaction for invalidation 4. The Transaction Manager specifies the Transaction and submits 5. The System verifies the Transaction 6. The System prompts the Transaction Manager for the reason for the invalidation 7. The Transaction Manager enters the reason and submits 8. The System verifies the reason 9. The System sends the invalidate call to the DBMS along with the reason. 10. The DBMS creates a new Transaction that reverses the action of the selected Transaction 11. The DBMS marks the Transaction as invalid with the ID of the new Transaction record 12. The DBMS records the date, user and reason for the invalidation 13. The DBMS returns a result with the Transaction ID and date for the new Transaction 14. The System displays a success message 15. The Transaction Manager acknowledges the message 16. The System displays the Transaction Management Menu |
| Alternate Flow(s) | 1. If there are no valid Transactions to display (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 16 2. If the Transaction Manager does not enter a good Transaction (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 Transaction Manager does not enter a good reason (Extension Point: Step 8)    1. The System displays an error message    2. The Transaction Manager acknowledges the error    3. The System returns to Step 6 4. If the DBMS cannot update the record (Extension Point: Step 13)    1. The System displays an error message    2. The Transaction Manager acknowledges the error    3. The System goes to Step 16 5. If the Transaction Manager chooses Cancel    1. The System prompts the Transaction Manager to confirm the Cancel    2. If the Transaction Manager confirms the Cancel, then the System goes to Step 16    3. If the Transaction Manager de-confirms the Cancel, the System resumes at the step where the Cancel occurred |
| Open Issues | 1. There are no open issues |
| Remarks | 1. The Transaction Manager may choose to Cancel until the reason for invalidation has been submitted successfully and the System is sending the data to the DBMS. 2. To enforce substantial entries, the reason for invalidation must be at least ten characters long, but can be appreciably longer. 3. All error messages displayed to the Transaction Manager must be informative and explain what happened in on-technical terms. 4. Invalidating a Transaction record is called, “Making a Correction”. |