Mini Project Report

Linux OS and programming

Test Plan and Output

**Test Plan:**

**1. High Level Test Plan (Integration Test Plan)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test ID** | **Description** | **Precondition** | **Expected IO** | **Expected OP** | **Actual OP** |
| TL1 | Displaying the correct name, account no and the balance. | Account Number, Name and Balance set using memory allocation | Set using the GTest Framework | Build Success  Output Passing. | Build Success  Output Passing. |
| TL2 | Debit any amount in the account number | Existing Bank account customer | Set using the GTest Framework | Build Success  Output Passing. | Build Success  Output Passing. |
| TL3 | Credit any amount in the account number | Existing Bank account customer | Set using the GTest Framework | Build Success  Output Passing. | Build Success  Output Passing. |
| TL4 | Count the number of customers in list. | Add the customers at the start of the code | Set using the GTest Framework | No of Customers in list | No of Customers in list |
| TL5 | Get the Balance of a customer | Existing Bank account customer | Set using the GTest Framework | Balance of the customer | Balance of the customer |

**2. Low Level Test Plan (Unit Test Plan)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test ID** | **Description** | **Precondition** | **Expected IO** | **Expected OP** | **Actual OP** |
| TH1 | Default Constructor Check | Set the name, account and balance to zero. | Set using the GTest Framework | Build Success  Output Passing. | Build Success  Output Passing. |
| TH2 | Parameterized Constructor Check | Set a name, account and balance. | Set using the GTest Framework | Build Success  Output Passing. | Build Success  Output Passing. |
| TH3 | Computation Test  Compute credit and debit and check balance | Set a name, account and balance. | Set using the GTest Framework | Build Success  Output Passing. | Build Success  Output Passing. |
| TH4 | With Dynamic Cast Test | Set a name, account and balance. | Set using the GTest Framework | Build Success  Output Passing. | Build Success  Output Passing. |
| TH5 | Add customers to a list of savings account and credit account | Create a list of accounts and use Mutex and semaphore for synchronization | Set using the GTest Framework | Build Success  Output Passing. | Build Success  Output Passing. |
| TH6 | Remove a customer from a list of savings account and credit account | Preexisting list of accounts and use Mutex and semaphore for synchronization | Set using the GTest Framework | Build Success  Output Passing. | Build Success  Output Passing. |
| TH7 | Compute Maximum Balance from the list of customers | Preexisting list of accounts and use Mutex and semaphore for synchronization | Set using the GTest Framework | Build Success  Output Passing.  Display the maximum balance | Build Success  Output Passing. Display the maximum balance |
| TH8 | Count the number of customers in list. | Preexisting list of accounts and use Mutex and semaphore for synchronization | Set using the GTest Framework | Build Success  Output Passing.  Display the number of customers. | Build Success  Output Passing. Display the number of customers |