| **Test Name** | | | Replicate Balance Limit Error | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Tested:** | | | Play Game | | | |
| **Test Description:** | | | Replicating an error where the balance limit is never reached e.g. the limit is 0 but the game ends when player balance reaches 5. | | | |
| **Pre-conditions** | | | Player balance is 100. Player limit is 0. Player bet amount is 5. Player Name is Fred. Amount Betted is 5. | | | |
| **Post-conditions** | | | Game results are output in console. | | | |
| **Notes:** | | When the player loses a game through the decreasing of their balance to the limit, the limit is never quite reached when it should be. | | | | |
| **Result (Pass/Fail/Warning/Incomplete)** | |  | | | | |
|  | **TEST STEP** | | | **EXPECTED TEST RESULTS** | P | F |
|  | Program is executed with designated pre-conditions. | | | Game results are displayed in console. Results show player balance increasing with a win and decreasing with a loss. After enough losses the player balance approaches the limit and the game halts when the limit is reached. Limit is not exceeded. |  | F |
|  | Q is input through console | | | Execution stops. | P |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Data Table** | | | | | |
|  | **1** | **2** | **3** | **4** | **5** |
| Balance | 100 |  |  |  |  |
| Expected end Balance | 0 |  |  |  |  |
| Limit | 0 |  |  |  |  |
| Name | “Fred” |  |  |  |  |
| Bet | 5 |  |  |  |  |
| Q | “q” |  |  |  |  |