Validation rule and its implementation on basis macros

What must be done?

Software have to check validity of the Event list property such as “COLLECTIVELY EXHAUSTIVE”, namely, a sum of event’s probabilities of the Event List is 1?

For this purpose, system must be able to implement the functional requirement SRS4.11:

SRS4.11: Software must start to solve a problem only if the property of the Event List is the collectively exhaustive. It means that sum of the event’s probabilities of the set of events is satisfying the constraint:

How would it be made?



Figure 4.3.1: Flow-chart of the algorithm to validate collectively exhaustive property of the Event List.

**Note:** The algorithm on Figure 4.3.1 can be implemented by using two approaches. ***First approach*** is submitted in the file “DMT(ValidationRuleWithVBA\_Ver-3\_3pages)\_2017.01.20.docx” on the base of ***VBA-codes***. ***Second approach*** is submitted file “DMT(ValidationRuleWithMacros\_Ver-1\_7pages)\_2017.01.20.docx” on the base of ***macros***. Since both approaches implement the same algorithm the results will be same. There is one of them required to implement in the system. It is impossible to implement both approaches in the single system simultaneously!

**Create a macro to carry out automatically a sequence of actions corresponding to Algorithm 4.3.1, these are:**

1. Enter following data {ActionList; EventList; m; n; Tolerance};
2. Determine the SumOfProbabilities of the entered list of events;
3. Compute UFL=1–Tolerance;
4. Compute OFL=1+Tolerance;
5. IF UFL<SumOfProbabilities<OFL THEN
   1. Delete the out-of-date rows of the table “tbl\_Act–EventCombinations”;
   2. Generate Act-Event combinations for given set of actions and events

;

* 1. Append the new set of m\*n rows into the table “tbl\_Act–EventCombinations”;
  2. Close form of which the macro is invoked;
  3. Open the form of which the macro has been invoked;

1. ELSE
   1. Submit the figure of the SumOfProbabilities as VDU messages:
   2. Message-1 is “SumOfProbabilities is outside of the given Tolerance”
   3. Message-2 is “It means–the list of events isn’t collectively exhaustive!”
   4. Message-3 is “Check the event’s probabilities of the list of events”;
2. ENDIF;

As a result of these actions the table “tbl\_Act–EventCombinations” will be filled by the up-to-date rows as it is shown below as an example for the Task no.1:

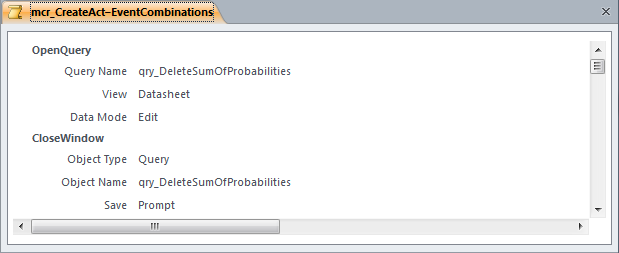
|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Action** | **Event** | **Probability** |
| 1 | Click on Strong rotation button | The wheel is stopped in the White region | 0.3 |
| 2 | Click on Strong rotation button | The wheel is stopped in the Red region | 0.7 |
| 3 | Click on Weak rotation button | The wheel is stopped in the White region | 0.3 |
| 4 | Click on Weak rotation button | The wheel is stopped in the Red region | 0.7 |

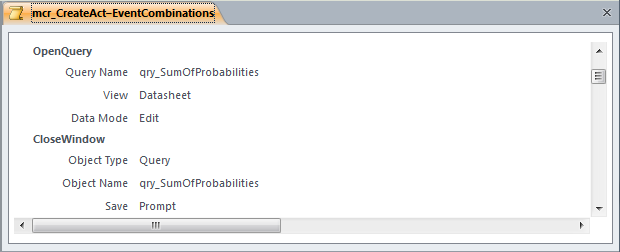
**How does it to do?**

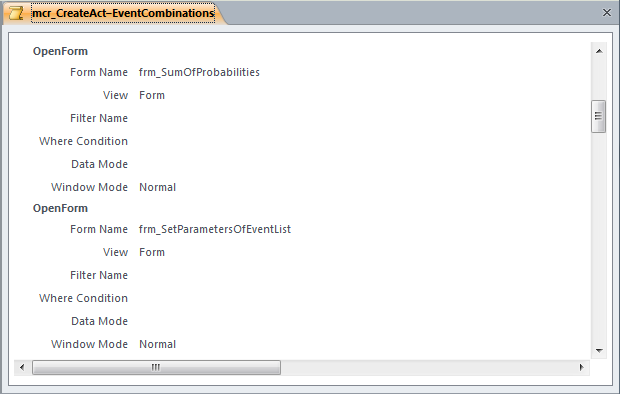
Create the macro “mcr\_CreateAct–EventCombinations” consisting of several sequential actions:

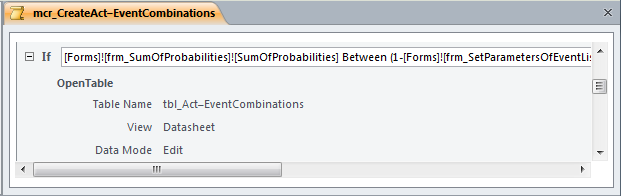
1. Open query “qry\_DeleteSumOfProbabilities”;
2. Close query “qry\_DeleteSumOfProbabilities”;
3. Open query “qry\_SumOfProbabilities”;
4. Close query “qry\_SumOfProbabilities”;
5. Open form “frm\_SumOfProbabilities”;
6. Open form “frm\_SetParametersOfEventList”;
7. IF [Forms]![frm\_SumOfProbabilities]![SumOfProbabilities] Between (1-[Forms]![frm\_SetParametersOfEventList]![ToleranceOfSumProbabilities]) And (1+[Forms]![frm\_SetParametersOfEventList]![ToleranceOfSumProbabilities]) THEN DO;
   1. Open table “tbl\_Act–EventCombinations”;
   2. Open query “qry\_DeleteCombinations”;
   3. Open query “qry\_CreateAct–EventCombinations”;
   4. Close query “qry\_CreateAct–EventCombinations”;
   5. Close form “frm\_CreateAct–EventCombinations”;
   6. Open form “frm\_CreateAct–EventCombinations”;
   7. Close table “tbl\_Act–EventCombinations”;
   8. Close form “frm\_SetParametersOfEventList”;
   9. Close form “frm\_SumOfProbabilities”;
   10. END DO;
8. ELSE DO;
   1. Close form “frm\_SumOfProbabilities”;
   2. Close form “frm\_SetParametersOfEventList”;
   3. MessageBox “SumOfProbabilities is outside of the tolerance”;
   4. MessageBox “It means – the list of events isn’t collectively exhaustive!”;
   5. MessageBox “Check the event’s probabilities figures of the list of events”;
   6. Open form “frm\_SumOfProbabilities”;
   7. Open query “qry\_DeleteCombinations”;
   8. Close form “frm\_CreateAct–EventCombinations”;
   9. END DO;
9. END IF;

This sequential of actions will automatically generate a set of the complete act-event combinations of the current task as it is shown on screenshot below:









Complete view of the logical expression is shown in the Expression Builder window below:

