

### Functional requirements

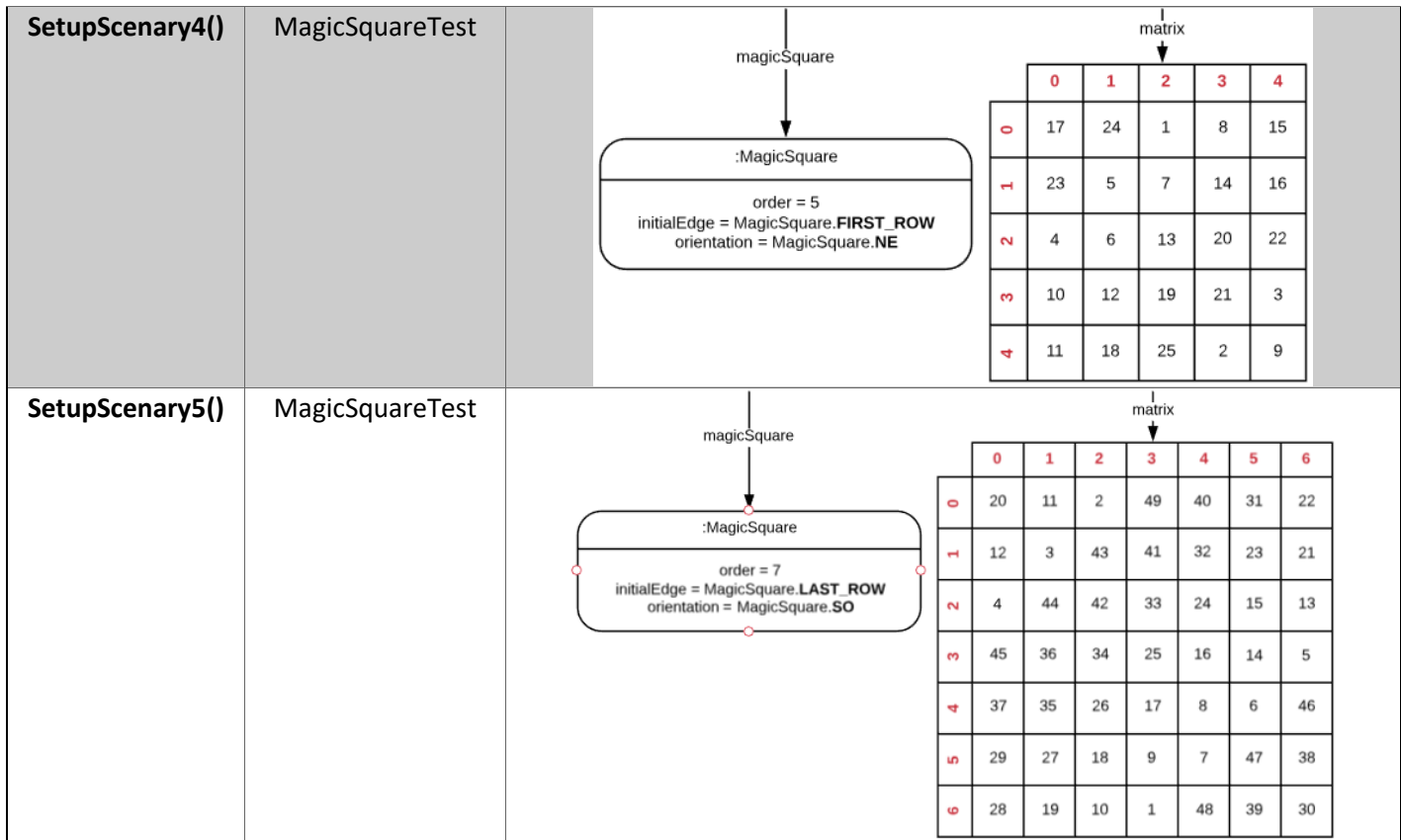
Name	FR#1	Generate magic square
Resume	Generate an integer matrix that represents a magic square of odd order with the characteristics received as parameters	
Input	<ul style="list-style-type: none"> <li>Integer that represents the matrix order</li> <li>Integer that represents the initial edge</li> <li>Integer array that indicates the orientation</li> </ul>	
Output	An integer matrix that represents the magic square	

Name	FR#2	Highlight the row and column of a selected box
Resume	Highlight the row and column of a selected box and show the magic constant at the end of each	
Input	<ul style="list-style-type: none"> <li>The row of the box</li> <li>The column of the box</li> </ul>	
Output	The highlighted row and column and the result at the end of each	

### Traceability

ID	Functional requirement	Method	Class
FR#1	Generate magic square	generateMagicSquare : int[][] correctIndex generateButtonPressed:void edgeSelected:void dirSelected:void	MagicSquare MagicSquare MagicSquareController MagicSquareController MagicSquareController
FR#2	Highlight the row and column of a selected box	handle:void	MagicSquareController.OnBoxSelected

Scenarios		
Name	Class	Scenario
setupScenary1()	MagicSquareTest	Empty
setupScenary2()	MagicSquareTest	<pre> sequenceDiagram     participant Test as :MagicSquareTest     participant MS as :MagicSquare     Test-&gt;&gt;MS: magicSquare     activate MS     MS: order = 3 initialEdge = MagicSquare.LAST_ROW orientation = MagicSquare.SO     deactivate MS           </pre>
setupScenary3()	MagicSquareTest	<pre> sequenceDiagram     participant Test as :MagicSquareTest     participant MS as :MagicSquare     Test-&gt;&gt;MS: magicSquare     activate MS     MS: order = 11 initialEdge = MagicSquare.FIRST_COLUMN orientation = MagicSquare.SE     deactivate MS           </pre>



### Design of test cases

<b>Objective of the test:</b>		Verify that a MagicSquare is created if the parameters given to the constructor are in the domain		
Class	Method	Scenary	Input	Result
MagicSquare	MagicSquare	setupScenary1	<ul style="list-style-type: none"> <li>order = 13</li> <li>initialEdge = MagicSquare.FIRST_COLUMN</li> <li>orientation = MagicSquare.SE.clone()</li> </ul>	A new MagicSquare has been successfully created. Each of the attributes of the new MagicSquare has correctly assigned the information passed by parameter.

<b>Objective of the test:</b>		Verify that a MagicSquare is not created if the parameters given to the constructor are not in the domain		
Class	Method	Scenary	Input	Result
MagicSquare	MagicSquare	setupScenary1	<ul style="list-style-type: none"> <li>order = -2</li> <li>initialEdge = MagicSquare.FIRST_COLUMN</li> <li>orientation = MagicSquare.NE</li> </ul>	The constructor does not create a new MagicSquare as the order is not in the domain
MagicSquare	MagicSquare	setupScenary1	<ul style="list-style-type: none"> <li>order = 3</li> <li>initialEdge = "valar morghulis"</li> <li>orientation = MagicSquare.NO</li> </ul>	The constructor does not create a new MagicSquare as the initialEdge is not in the domain

<b>MagicSquare</b>	MagicSquare	setupScenary1	<ul style="list-style-type: none"> <li>• order = 5</li> <li>• initialEdge = MagicSquare.LAST_COLUMN</li> <li>• orientation = new int[]{-365, 1999}</li> </ul>	The constructor does not create a new MagicSquare as the orientation is not in the domain
<b>MagicSquare</b>	MagicSquare	setupScenary1	<ul style="list-style-type: none"> <li>• order = -86</li> <li>• initialEdge = "valar dohaeris"</li> <li>• orientation = new int[]{5,5}</li> </ul>	The constructor does not create a new MagicSquare because none of the parameters is within the domain

<b>Objective of the test:</b>		Verify that the matrix that represents the magic square is generated correctly when the MagicSquare attributes comply with the conditions of the Loubere method (siame method) and that it is not generated if they do not		
Class	Method	Scenary	Input	Result
<b>MagicSquare</b>	generateMagicSquare	setupScenary2	<ul style="list-style-type: none"> <li>• none</li> </ul>	The matrix has been generated correctly
<b>MagicSquare</b>	generateMagicSquare	setupScenary3	<ul style="list-style-type: none"> <li>• none</li> </ul>	The matrix is not generated as this is not a valid magic square

<b>Objective of the test:</b>		Verify that all rows, columns and diagonals of the generated matrix add the magical sum		
Class	Method	Scenary	Input	Result
<b>MagicSquare</b>	getMagicalSum	setupScenary4	<ul style="list-style-type: none"> <li>• none</li> </ul>	The rows, columns and diagonals of the matrix add the same number
<b>MagicSquare</b>	getMagicalSum	setupScenary5	<ul style="list-style-type: none"> <li>• none</li> </ul>	The rows, columns and diagonals of the matrix add the same number

