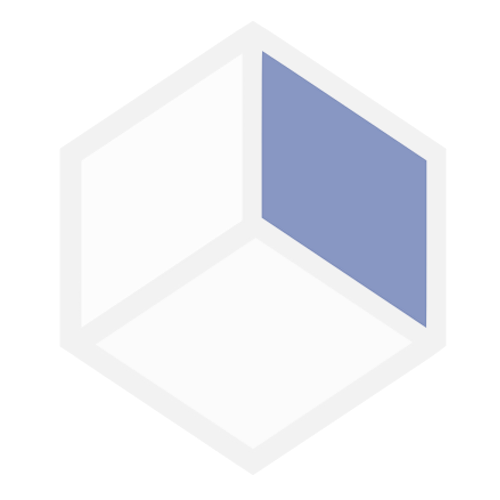
****

**URL to the project:** <https://github.com/MiguelRAvila/projectTOBARA>

**URL to the Library: <https://pypi.org/project/TOBARA/>**

**Description**

Our system is a Boolean function analyzer tool whose main task is to reduce the function in its simplest expression.

With this information we will create a library that allows any user to consulate the elements of any Boolean function they want to introduce.

**Process**

Our main goal is the analysis of the Boolean functions and the breakdown of their main components:

* Get the function
* Determinate their variables
* Determinate their terms
* Simplify the function

**Objective**

* Algorithms for reducing Boolean functions.

**Requirements**

1. **System actors**

User. A person who uses the system.

* May consult if the Boolean function is in its simplest expression (it cannot be simplified).
* May ingress a Boolean function to be reduced in its simplest expression.
* May get their Boolean function in its sum of product expression.

1. **User requirements**.

* If the user ingress an invalid input the system will return error and the user will have to change it.
* The users might ingress a Boolean function and get its simplest expression.
* The users might convert their Boolean function into its sum of product expressions.
* The users might install a library from the Pip service of Python.
* The users might create a truth table and minterms table for the Boolean functions.
* The users might create Kmaps for the Boolean functions.

**System requirements**

* 1. **Functionals**

|  |  |
| --- | --- |
| **FR001** | **Identification if the function is in its simplest expression** |
| Priority | High |
| Description | The system must be able to receive and determine the number of variables the function has. |

|  |  |
| --- | --- |
| **FR002** | **Minimization of the expression** |
| Priority | High |
| Description | The system must be able to receive and verify the number of terms the function has. |

|  |  |
| --- | --- |
| **FR003** | **Create the truth table** |
| Priority | High |
| Description | The system must be able to receive the Boolean function and read the minterms involved. |

|  |  |
| --- | --- |
| **FR004** | **Sum of products** |
| Priority | High |
| Description | The system must be able to receive a Boolean function and translate it in its sum of products expression. |

|  |  |
| --- | --- |
| **FR005** | **Reduce function** |
| Priority | High |
| Description | The system must be able to reduce the function in its simplest expression intoduccing the number of variables and the minterms that integrate the function.  The system must return a string with the literal terms of the simplified function. |

|  |  |
| --- | --- |
| **FR006** | **Installation** |
| Priority | High |
| Description | The system must be re-raised to the Pip system so it might be used from the Python pip installation. |

|  |  |
| --- | --- |
| **FR007** | **Truth tables and minterms tables** |
| Priority | Medium |
| Description | The system must have a function that allows the user to create a truth table or a minterms table introducing the number or variables and minterms that integrate the function. |

|  |  |
| --- | --- |
| **FR008** | **Creation of the Kmaps** |
| Priority | Low |
| Description | The system must have a function that allows the user to introduce a Boolean function putting the number of variables and minterms that integrate the function and then generate a table that represents their Karnaugh Map. |

* 1. **No functional**

|  |  |
| --- | --- |
| **NFR001** | **Inputs** |
| Description | The expression must be a Boolean function and contain non-repetitive variables. |

|  |  |
| --- | --- |
| **NFR002** | **Reduction algorithm** |
| Description | The system will be focused in the reduction of the function by the implementation of a reduction algorithm based in the K-maps. |

|  |  |
| --- | --- |
| **NFR003** | **Standardization of the matrix** |
| Description | The matrix will have an order for the creation of the truth tables. |

|  |  |
| --- | --- |
| **NFR004** | **Sum of Products** |
| Description | The expression will be given with the minterms of the Boolean function (2^n terms). |

|  |  |
| --- | --- |
| **NFR005** | **Installation (inclusion in the pip service)** |
| Description | The system must be registered y count with:  A presentation repository to the library.  Its presentation of functions and installer. |

|  |  |
| --- | --- |
| **NFR006** | **Maximum number of terms for a table** |
| Description | The system will have a limit of 32 terms for the creation of tables(this includes the truth tables and the minterms tables). |

|  |  |
| --- | --- |
| **NFR007** | **Maximum number of terms to be reduce** |
| Description | The system limit for the reduction of the Boolean functions will be 16 terms. |

|  |  |
| --- | --- |
| **NFR008** | **Kmaps limits** |
| Description | The limit for the creation of Kmaps will be of 4 variables and must consider the general notation of the Kmaps. |

**Use cases**

**UC001.** Simplest expression.

**Description.** Consult if the function is in its simplest expression.

**Sequence.**

1. Consult the tool Mini(funciónBool).
2. Receives a Boolean value (True or False).

**Alternative departures:** 1.1 If the user ingress an invalid input the system will return error and the user will have to change it.

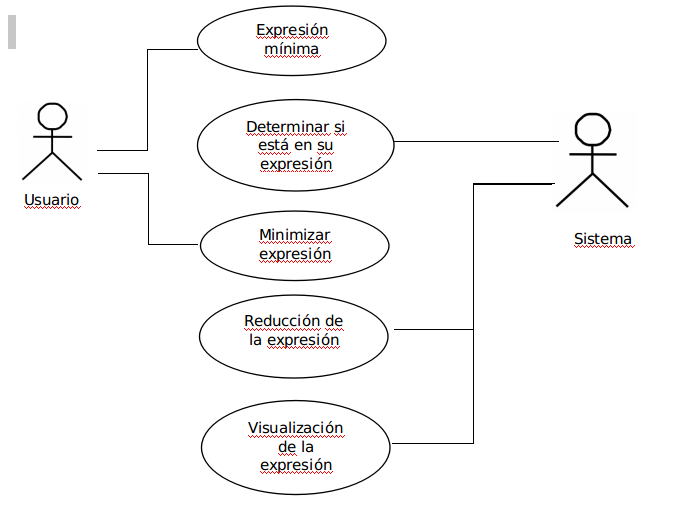
**UC002**. Simplify.

**Description.** Given a Boolean function, the system returns its simple expression.

1. Consult the tool reducc(funciónBool).
2. Receives an array with the simplified function.

**Alternative departures:** 1.1 If the user ingress an invalid character the system will return error and the user will have to change it.

**Use cases diagram**



Simplest expression

Determinate if the function is in its simplest expression

Minimize expression

Simplify expression

Visualization of the expression

User

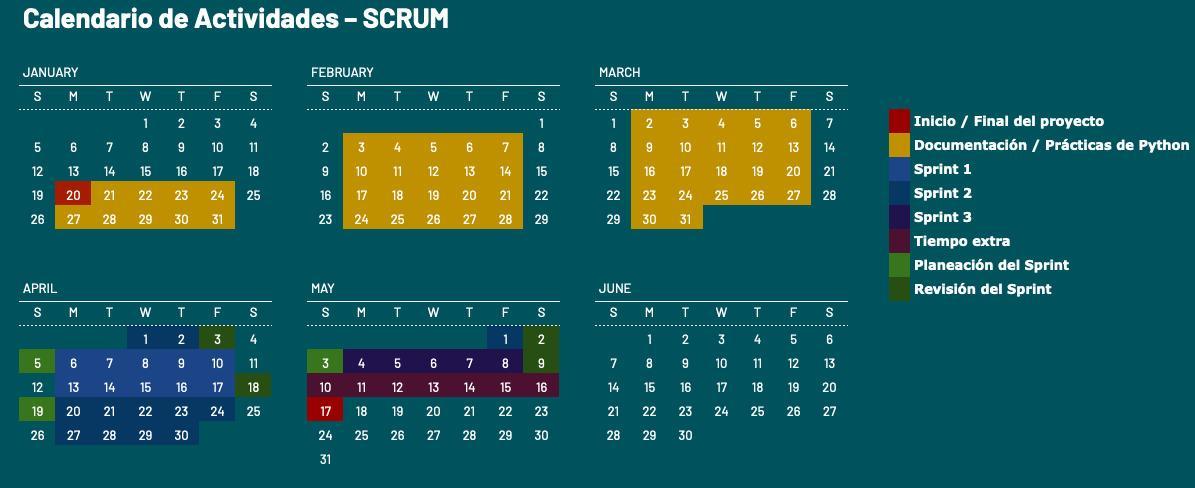
System

**Development process:**

Our activity calendars start from January 20th until June 15th. With this period, we establish some range and important dates that follow this way: From January 21st until June 8th we will realize the documentation and the Python tests. There will be 3 sprints established in this period:

* Sprint 1: We establish the objectives on April 19th and the sprint ends on May 2nd. The results of the sprint will be evaluated on May 4rd.
* Sprint 2: We establish the objectives on May 5th and the sprint ends on May 21st. The results of the sprint will be evaluated on May 22nd.
* Sprint 3: We establish the objectives on April 19th and the sprint ends on June 9th. The results of the sprint will be evaluated on June 10th.

Additionally, we establish an extra time in case the team faces some problems that might delay the date of delivery. This period is established for June 10th until June 15th.



**Individual contribution metric:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Task | Evidence | Percent of the task | Responsible | Date of delivery | Complete |
| Investigation | | | | | |
| Algorithm | Presentation in the reunion | 15% | Audny | April 19th | Check |
| Algorithm operation in Python | Repository | 4% | Pamela | May 2nd | Check |
| Project organization | Repository and PyPi | 10% | Miguel | May 2nd | Check |
| Codification (Functions) | | | | | |
| getBin | Code in the repository | 5% | Miguel | June 13rd | Check |
| getTable | Code in the repository | 14% | Jorge | June 13rd | Check |
| getTer | Code in the repository | 4% | Jorge | June 13rd | Check |
| getVar | Code in the repository | 4% | Pamela | June 13rd | Check |
| reduceFun | Code in the repository | 22% | Roberto | June 14th | Check |
| Library organization | In the repository | 5% | Miguel | June 13rd | Check |
| Organization | | | | | |
| First presentation | Presentation in the meeting | 2% | Miguel | March 6th | Check |
| Second presentation | Presentation in the meeting | 4% | Audny | May 1st | Check |
| Logs | Repository | 11% | Pamela | Per each activity | Check |

|  |  |  |  |
| --- | --- | --- | --- |
| Member | # Tasks mandated | # Tasks delivered complete. | Percentage |
| Audny | 2 | 2 | 19% |
| Jorge | 2 | 2 | 18% |
| Miguel | 3 | 3 | 22% |
| Pamela | 3 | 3 | 19% |
| Roberto | 1 | 1 | 22% |

Standards

* Respects the delivery date
* Presents the pertinent evidence