**URL del Repositorio:** <https://github.com/MiguelRAvila/projectTOBARA>

**URL de la Presentación:** <https://github.com/MiguelRAvila/projectTOBARA/blob/master/ProjectTOBARA.pptx>

**Descripción**

Nuestro sistema es una herramienta de análisis de funciones booleanas, cuya tarea principal es descomponer la función ingresada en sus componentes principales (tabla de verdad, Min-términos y suma de productos). Obtenido esto, podemos realizar la minimización de la función boolena mediante el algoritmo Quine–McCluskey.

Con estos datos realizaremos una API que permita a cualquier usuario consultar estos elementos de cualquier función booleana que quieran introducir.

**Proceso**

Nuestro sistema tendrá como objetivo principal el análisis de funciones booleanas y su descomposición en sus componentes principales:

• Tabla de verdad.

• Min-términos.

• Suma de productos.

Con dichos datos, podemos reducir la función booleana mediante el Algoritmo Quine–McCluskey (Escogido sobre el método K-map por ser más fácil de implementar).

**Objetivos**

* Algoritmos de reducción de funciones booleanas.
* Visualización de datos. (Secundario)
* Desarrollo de la interfaz. (Secundario)

**Requerimientos**

1. **Actores del sistema**

Usuario. Persona que usa el sistema.

* Puede consultar si una expresión booleana se encuentra en su expresión mínima (no se puede reducir más).
* Puede ingresar una función booleana para ser reducida a su expresión mínima.
* Puede recibir su función booleana en su expresión de suma de productos.

1. **Requerimientos del usuario**

* Los usuarios podrán preguntar si una función se encuentra en su expresión mínima.
* Los usuarios podrán obtener la tabla de verdad correspondiente a su función.
* Los usuarios podrán ingresar una función booleana y obtendrán su expresión mínima.
* Los usuarios podrán convertir su función booleana en su forma de suma de productos.

1. **Requerimientos del sistema**
   1. **Funcionales**

|  |  |
| --- | --- |
| **RF001** | **Identificación si es expresión mínima** |
| Prioridad | Alta |
| Descripción | El sistema deberá ser capaz de recibir y determinar si una expresión booleana se encuentra en su mínima expresión. Devolviendo una expresión booleana (True o False). |

|  |  |
| --- | --- |
| **RF002** | **Mínimización de la expresión** |
| Prioridad | Alta |
| Descripción | El sistema deberá ser capaz de recibir y reducir una expresión booleana hasta su mínima expresión. |

|  |  |
| --- | --- |
| **RF003** | **Generación de tabla de verdad** |
| Prioridad | Alta |
| Descripción | El sistema deberá ser capaz de recibir una función booleana y generar una matriz que representa su tabla de verdad. |

|  |  |
| --- | --- |
| **RF004** | **Suma de productos** |
| Prioridad | Alta |
| Descripción | El sistema deberá ser capaz de recibir una función booleana y convertirla su expresión de suma de productos. |

* 1. **No funcionales**

|  |  |
| --- | --- |
| **RFN001** | **Entradas** |
| Descripción | La expresión debe de ser una expresión boolena y contener variables no repetidas. |

|  |  |
| --- | --- |
| **RFN002** | **Algoritmo de Reducción** |
| Descripción | El sistema se centrara en la reducción mediante la implementación de un algoritmo de reducción mediante K-maps. |

|  |  |
| --- | --- |
| **RFN003** | **Estandarización de las matrices** |
| Descripción | La matriz estará ordenada de la manera estándar para la elaboración de tablas de verdad. Es decir, |

|  |  |
| --- | --- |
| **RFN004** | **Suma de Productos** |
| Descripción | La expresión será dada con los minterminos de la función booleana (2^n términos). |

**Casos de uso**

**CU001.** Expresión mínima.

**Descripción.** Consultar si una función ya esta en su mínima expresión.

**Secuencia.**

1. Consulta a la herramienta Mini(funciónBool).
2. Recibe un valor boolneano (True o False).

**Salidas alternativas:** 1.1 Si el usuario ingresa una entrada no válido, el sistema le marcará como error y el usuario tendrá que modificarlo.

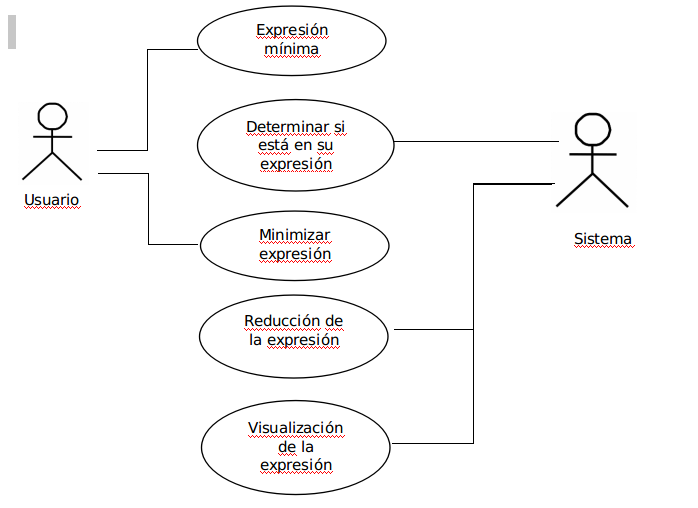
**CU002**. Minimizar.

**Descripción.** Dada una función booleana, regresa la expresión minimizada.

1. Consulta a la herramienta reducc(funciónBool).
2. Recibe un arreglo con la función minimizada.

**Salidas alternativas:** 1.1 Si el usuario ingresa un caracter no válido, el sistema le marcará como error y el usuario tendrá que modificarlo.

**Diagrama de casos de uso**



**Proceso de Desarrollo:**

Nuestro calendario de actividades abarca desde el día **20 de Enero** hasta el día **29 de Abril**.

Teniendo este rango en mente, establecimos otros rangos y fechas específicas de la siguiente manera:

Desde el **21 de Enero** hasta el **13 de Marzo** se realizará la Documentación y las Prácticas de Python

Se realizarán *3 sprints establecidos en estos rangos:*

*-* S*print 1:* se establecen los objetivos el **16 de Marzo**, y el sprint debe durar desdeese día hasta el **24 de Marzo** para que el **25 de Marzo** se revisen los resultados del sprint.

*-* S*print 2:* se establecen los objetivos el **26 de Marzo**, y el sprint debe durar desde día hasta el **2 de Abril** para que el **3 de Abril** se revisen los resultados del sprint.

*-* S*print 3:* se establecen los objetivos el **6 de Abril**, y el sprint debe durar desde ese día hasta el **14 de Abril** para que el **15 de Abril** se revisen los resultados del sprint.

Establecimos igual un *tiempo extra* para acomodarnos de ser necesario y para anteponernos a situaciones que nos puedan atrasar, la cual abarca desde el **16 de Abril** hasta el **28 de Abril**.

Imagen que contiene circuito

Descripción generada automáticamente

**Métrica de contribución individual:**

**Metrica de la primera entrega:**

La metrica se divide en tres tareas principales: investigación, diseño y organización. Estas a su vez se subdividen en tareas las cuales tienen un porcentaje total de 100 puntos. Cada integrante obtiene un puntaje con respecto a su participación en dicha sub tarea.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tarea: Investigación | | | | | |
| Subtarea | Miguel | Roberto | Audny | Pamela | Jorge |
| Del Algoritmo | 15 | 35 | 10 | 35 | 10 |
| Bases de datos (Django and Flask) | 10 | 40 | 10 | 30 | 10 |
| Desarrollo de APIs | 20 | 30 | 10 | 30 | 10 |
| Puntuación individual: | 45 | 105 | 30 | 95 | 30 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tarea: Diseño | | | | | |
| Subtarea | Miguel | Roberto | Audny | Pamela | Jorge |
| Presentación | 20 | 15 | 15 | 15 | 35 |
| Documento de entrega | 20 | 10 | 20 | 10 | 40 |
| Diagramas | 15 | 15 | 20 | 15 | 35 |
| Puntuación Indivudual: | 55 | 40 | 55 | 40 | 110 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Tarea: Organización | | | | | |
| Subtarea | Miguel | Roberto | Audny | Pamela | Jorge |
| General del equipo | 45 | 10 | 20 | 10 | 15 |
| Fechas y calendarios | 35 | 10 | 35 | 10 | 10 |
| Proceso de Desarrollo | 40 | 10 | 30 | 10 | 10 |
| Puntuación Individual | 120 | 30 | 85 | 30 | 45 |