

Universidad ORT Uruguay

Facultad de Ingeniería

Diseño de Aplicaciones I

Obligatorio I

Sebastian Gonzalez - 208938

<https://github.com/ORT-DA1/208938>

21/05/2020

Indice:

Descripción General	3
Diseño	4
Pruebas Unitarias	7

1. Descripción General

SentimentAnalysis es un sistema de análisis de sentimientos, cuyo objetivo es interpretar diferentes frases, y asignar a estas su correspondiente sentimiento (Positivo, Negativo, Neutro) y una Entidad a la cual hace referencia.

A su vez, el mismo permite generar alarmas, las cuales se deben asociar a un sentimiento y a una entidad, para que el sistema cuando analice las frases, actualice la alarma, pudiendo generar una correspondiente alerta en caso de que se cumpla el objetivo de la misma (Cantidad de Post en X cantidad de días).

El sistema brinda diferentes funcionalidades como:

Entidades:

- Alta
 - Permite ingresar una nueva Entidad.
- Baja
 - Se permite eliminar una entidad existente.
 - En caso de que la entidad se encuentre asociada a una Frase, a dicha Frase se le removerá la Entidad asignada.
 - Si existen Alarmas para la entidad a borrar, la esta también será eliminada.
- Modificacion
 - Se permite modificar los datos de una entidad existente.
 - Se volverán a analizar todas las frases que estaban asociadas a la misma.

Sentimientos:

- Alta
 - Se permite ingresar un nuevo Sentimiento.
- Baja
 - Se permite eliminar un sentimiento existente.
 - En caso de que el sentimiento se encuentre asociado a una Frase, a dicha Frase se le removerá el Sentimiento asignado.
- Modificacion
 - Se permite modificar un sentimiento existente.
 - Se volverán a analizar todas las frases que estaban asociadas al mismo.

Frases:

- Alta
 - Se permite ingresar una nueva Frase.
 - Una vez ingresada, automáticamente se analizará la misma, asignándole un Sentimiento y Entidad, en caso de ser posible.
 - Si la frase posee más de un Sentimiento, se le asignará sentimiento Neutro.
 - En caso de que posea más de una Entidad, se le asignará la primera que se encontró y como sentimiento Neutro.
- Baja
 - Se permite eliminar Frases existentes.
 - Al eliminar una frase, se actualizarán las alarmas.
- Modificacion
 - Se permite modificar una frase existente.
 - Al modificarla, se volverá a analizar y a su vez, se volverán a actualizar las alarmas.

Alarmas:

- Alta
 - Se permite crear una nueva Alarma.
 - Al ingresar la alarma, se analizaran las Frases ya ingresadas.
- Baja
 - Se permite eliminar Alarmas existentes.
- Modificacion
 - Se permite modificar una alarma existente.
 - Al modificar una Alarma, se volverán a analizar las Frases existentes con los nuevos datos de la Alarma.

Reportes:

- Alertas
 - Se exhibirán las Alertas generadas, las cuales son todas aquellas Alarmas que hayan cumplido con el objetivo de Posts.
- Frases
 - Se exhibirá información de las frases analizadas.

2. Diseño

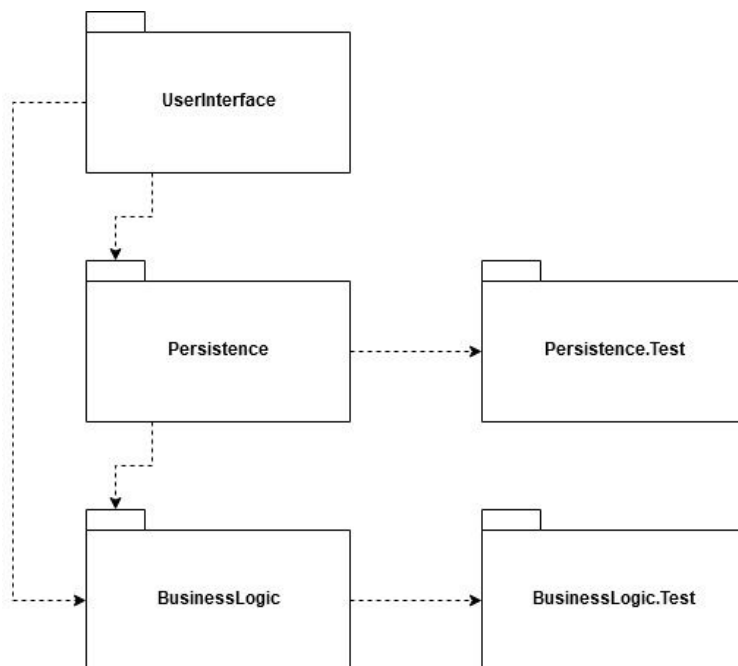
Nuestra selección del diseño a utilizar se basó en encapsular la lógica de las funcionalidades en clases según responsabilidades, las cuales se detallan a continuación:

- **UserInterface:** Se implementa la interfaz del usuario, la cual será utilizada por el usuario para la interacción con el sistema.
- **BusinessLogic:** Definición de las entidades utilizadas en el sistema, enumerados y excepciones. También se implementan las reglas de negocio.
- **Persistence:** Acceso y persistencia de los datos. En esta primera instancia de proyecto se guardarán los datos en listas en memoria.
- **Test:** Los paquetes “.Test” contendrán las pruebas unitarias correspondientes a su paquete.

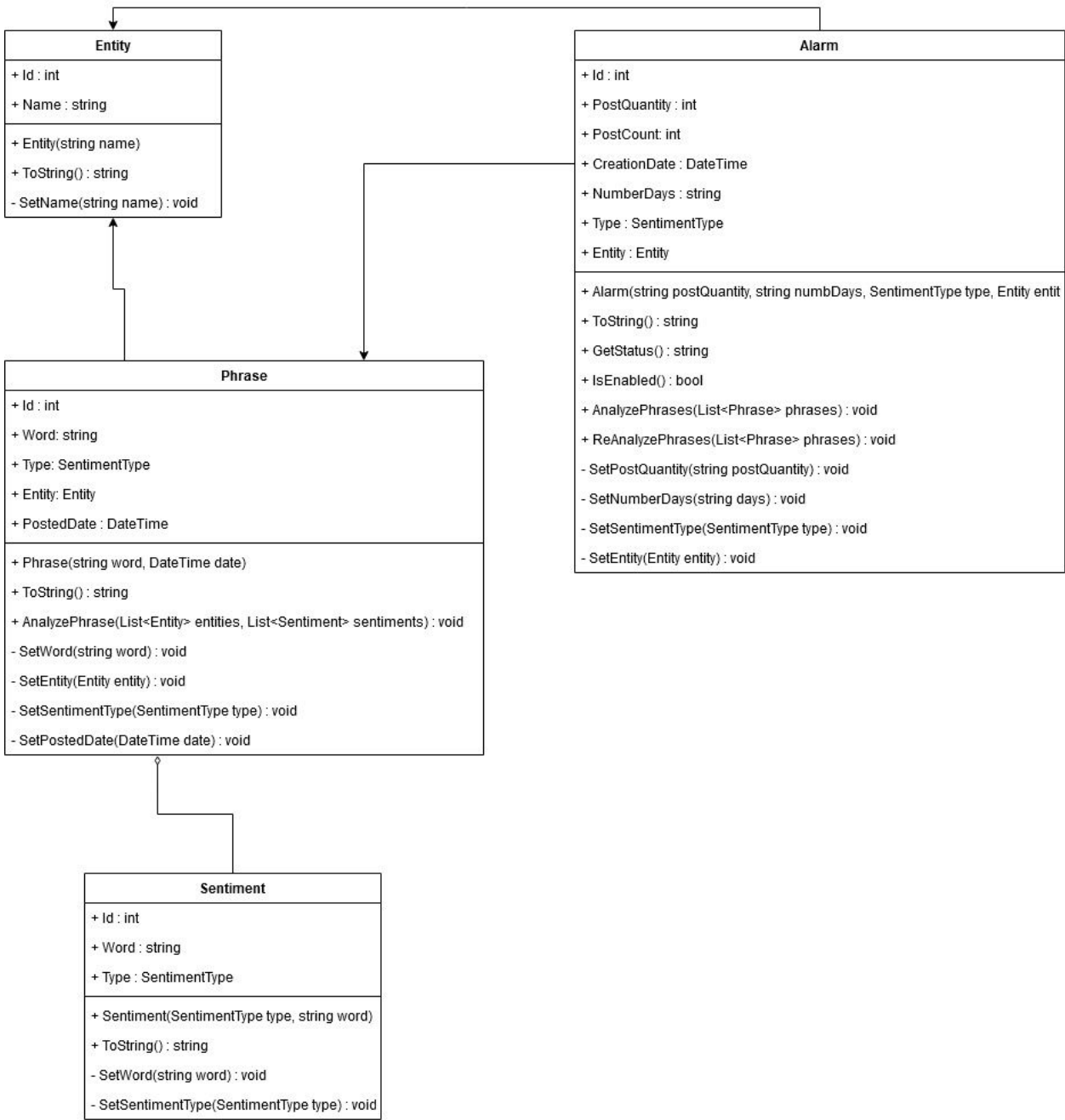
2.1. Diagrama de Paquetes

En el Diagrama se puede observar la dependencia entre los diferentes Paquetes. Dado que UserInterface hace uso de los paquetes Persistence y BusinessLogic. BusinessLogic no posee de ningún otro paquete y Persistence solo tiene dependencia de BusinessLogic.

En cuanto a los paquetes Test, dependen de sus correspondientes.



2.2. Diagrama de Clases

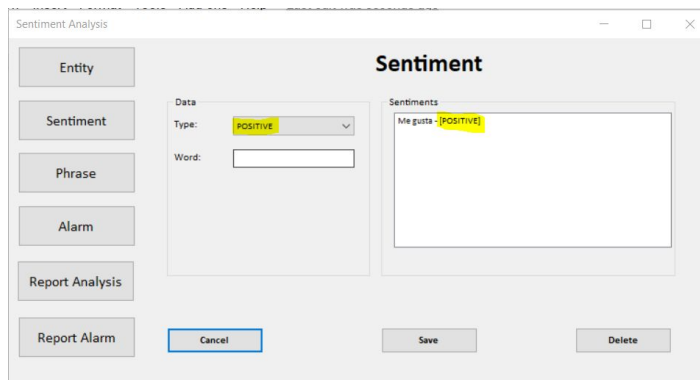


2.3. Mecanismos generales

Como se mencionó en el punto 2, la capa **UserInterface** se encarga de la interacción con el usuario, la misma recolecta la información ingresada por este y le informa a la capa de persistencia **Persistence**, la cual realiza dos funciones, la primera es tomar esta información y enviarla a la capa de lógica **BusinessLogic** donde será validada, en caso de que no se cumpla alguna validación, lanzará una excepción que será capturada en la capa **UserInterface** para exhibir el error al usuario. En caso de que la validación realizada en **BusinessLogic** sea exitosa, se realizarán las validaciones correspondientes a la capa **Persistence**, las cuales se comportan de la misma forma que las validaciones de **BusinessLogic**, en caso de fallar serán informadas al usuario y en caso contrario, se almacenará en memoria, y se exhibirá un mensaje exitoso al usuario.

Para el caso de las Frases, al dar de alta una nueva Frase, se analizará la misma en busca de un Sentimiento y una Entidad, en caso de que se encuentren más de un sentimiento o entidad, la Frase se almacenará de igual forma, pero exhibirá como error que se encontraron múltiples entidades / sentimientos. De igual forma si no se encuentra un sentimiento u entidad.

Luego en el caso de la pantalla de Sentimientos, la lista solo exhibirá los sentimientos correspondientes al seleccionado en el ComboBox.



The screenshot shows a window titled "Sentiment Analysis" with a sidebar on the left containing buttons: "Entity", "Sentiment", "Phrase", "Alarm", "Report Analysis", and "Report Alarm". The main area is titled "Sentiment" and contains a "Data" section with a "Type" dropdown menu set to "POSITIVE" and a "Word" input field. To the right is a "Sentiments" list box containing the text "Me gusta - (POSITIVE)". At the bottom of the window are three buttons: "Cancel", "Save", and "Delete".

3. Pruebas Unitarias

3.1. Cobertura Alcanzada

Para las pruebas unitarias se hizo uso de la herramienta MSTest, la cual nos permite programar pruebas unitarias y analizar la cobertura que las mismas poseen sobre el código.

Como se puede observar, se logró cubrir el 100% del código mediante la creación de 64 pruebas para la capa BusinessLogic:

{ }	BusinessLogic	0	0.00%	267	100.00%
▷	Alarm	0	0.00%	108	100.00%
▷	Entity	0	0.00%	20	100.00%
▷	Phrase	0	0.00%	107	100.00%
▷	Sentiment	0	0.00%	32	100.00%

Para el caso de la capa Persistence, se crearon 65 pruebas, las cuales logran cubrir un 99.14%, el 0.86% restante corresponde a pruebas que por lógica poseen un Try/Catch y no se toma como cubierto el ultimo bracket del try.

EntityRepository	2	1.72%	114	98.28%
Add(BusinessLogic.Entity, ...	1	5.00%	19	95.00%
Delete(BusinessLogic.Entit...	0	0.00%	40	100.00%
EntityRepository()	0	0.00%	3	100.00%
ExistsName(string)	0	0.00%	6	100.00%
GetAllEntities()	0	0.00%	3	100.00%
GetByld(int)	0	0.00%	7	100.00%
IsEmpty()	0	0.00%	3	100.00%
Modify(BusinessLogic.Enti...	1	2.94%	33	97.06%
AlarmRepository	0	0.00%	48	100.00%
Add(BusinessLogic.Alarm,...	0	0.00%	8	100.00%
AlarmRepository()	0	0.00%	3	100.00%
Delete(BusinessLogic.Alar...	0	0.00%	7	100.00%
ExistsAlarm(BusinessLogic...	0	0.00%	8	100.00%
GetAllAlarms()	0	0.00%	3	100.00%
GetByld(int)	0	0.00%	7	100.00%
IsEmpty()	0	0.00%	3	100.00%
Modify(BusinessLogic.Alar...	0	0.00%	9	100.00%
PhraseRepository	0	0.00%	75	100.00%
Add(BusinessLogic.Phrase,...	0	0.00%	21	100.00%
Delete(BusinessLogic.Phra...	0	0.00%	15	100.00%
ExistsWord(string)	0	0.00%	6	100.00%
GetAllPhrases()	0	0.00%	3	100.00%
GetByld(int)	0	0.00%	7	100.00%
IsEmpty()	0	0.00%	3	100.00%
Modify(BusinessLogic.Phr...	0	0.00%	17	100.00%
PhraseRepository()	0	0.00%	3	100.00%

SentimentRepository	2	1.69%	116	98.31%
Add(BusinessLogic.Sentim...	1	3.13%	31	96.88%
Delete(BusinessLogic.Sent...	0	0.00%	27	100.00%
ExistsSentiment(string)	0	0.00%	6	100.00%
GetAllSentiments()	0	0.00%	3	100.00%
GetById(int)	0	0.00%	7	100.00%
GetBySentimentType(Busi...	0	0.00%	5	100.00%
IsEmpty()	0	0.00%	3	100.00%
Modify(BusinessLogic.Sen...	1	3.13%	31	96.88%
SentimentRepository()	0	0.00%	3	100.00%

3.2. Análisis de Frases

Al comienzo del Sistema, no existen registros de Sentimientos, Entidades o Frases.

The image displays three screenshots of the 'Sentiment Analysis' application interface, showing the 'Entity', 'Sentiment', and 'Phrase' forms.

Entity Form: The 'Entity' tab is selected. The 'Data' section includes a 'Name' input field. The 'Entities' list is empty. The 'Report Analysis' button is visible.

Sentiment Form: The 'Sentiment' tab is selected. The 'Data' section includes a 'Type' dropdown menu (set to 'POSITIVE') and a 'Word' input field. The 'Sentiments' list is empty. The 'Report Analysis' button is visible.

Phrase Form: The 'Phrase' tab is selected. The 'Data' section includes a 'Word' input field and a 'Date' dropdown menu (set to '21/05/2020 13:16'). The 'Phrases' list is empty. The 'Report Analysis' button is visible.

Registro la Entidad: Google

Sentiment Analysis

Entity

Entity

Data

Name:

Entities

Google

Entity saved correctly.

Cancel Save Delete

Registro el Sentimiento: Me gusta

Sentiment Analysis

Sentiment

Data

Type:

Word:

Sentiments

Me gusta - [POSITIVE]

Sentiment saved correctly.

Cancel Save Delete

Registro la Frase: Me gusta Google

Sentiment Analysis

Phrase

Data

Word:

Date:

Type:

Entity:

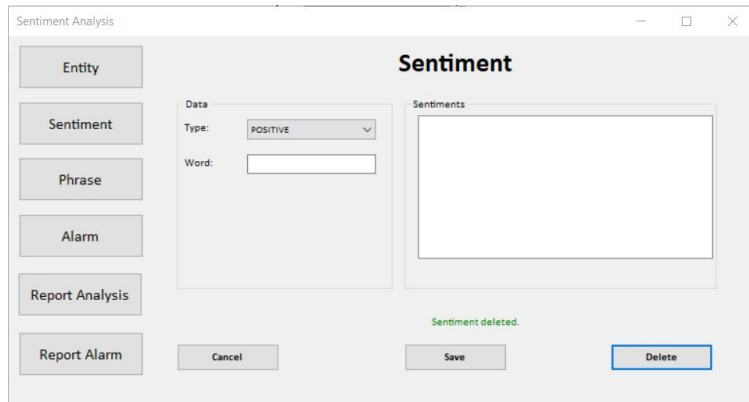
Phrases

Me gusta google [Type: POSITIVE] [Entity: Google]

Cancel Save Delete

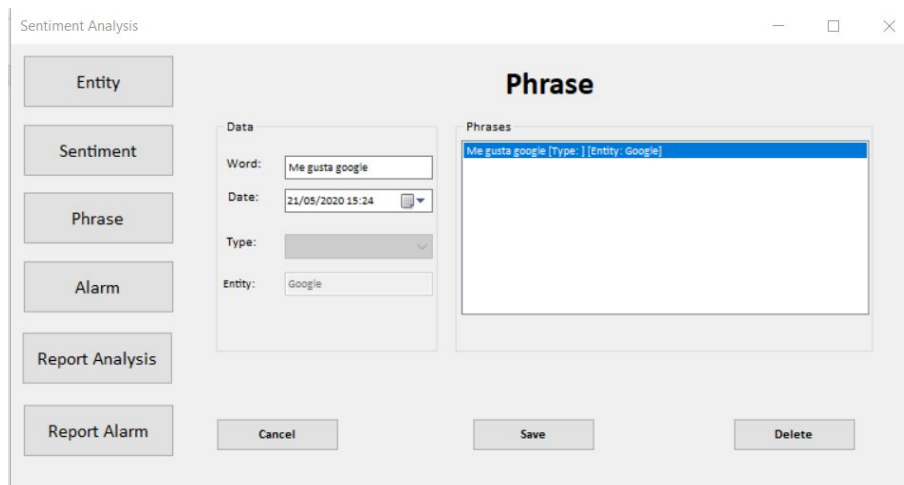
Se puede observar que se le asignó el sentimiento Positivo (Me gusta) y la entidad Google.

Eliminar un sentimiento, elimina el sentimiento de la Frase.
Se elimina el sentimiento Me Gusta.



The screenshot shows the 'Sentiment Analysis' application window with the 'Sentiment' tab selected. On the left, there is a sidebar with buttons: 'Entity', 'Sentiment', 'Phrase', 'Alarm', 'Report Analysis', and 'Report Alarm'. The main area is titled 'Sentiment' and contains a 'Data' section with 'Type' set to 'POSITIVE' and an empty 'Word' field. To the right is a 'Sentiments' list box, which is currently empty. Below the list box, a green message 'Sentiment deleted.' is displayed. At the bottom, there are 'Cancel', 'Save', and 'Delete' buttons.

Observamos que la Frase no tiene sentimiento:



The screenshot shows the 'Sentiment Analysis' application window with the 'Phrase' tab selected. The sidebar is the same as in the previous screenshot. The main area is titled 'Phrase' and contains a 'Data' section with 'Word' set to 'Me gusta google', 'Date' set to '21/05/2020 15:24', 'Type' set to an empty dropdown, and 'Entity' set to 'Google'. To the right is a 'Phrases' list box containing one entry: 'Me gusta google [Type:] [Entity: Google]'. At the bottom, there are 'Cancel', 'Save', and 'Delete' buttons.

Ahora si vuelvo a ingresar el Sentimiento, la frase debe volver a analizarse y nuevamente le asignará el sentimiento positivo.

Sentiment Analysis

Sentiment

Entity

Sentiment

Phrase

Alarm

Report Analysis

Report Alarm

Data

Type: POSITIVE

Word:

Sentiments

ME GUSTA - [POSITIVE]

Sentiment saved correctly.

Cancel Save Delete

Sentiment Analysis

Phrase

Entity

Sentiment

Phrase

Alarm

Report Analysis

Report Alarm

Data

Word: Me gusta google

Date: 21/05/2020 15:24

Type: POSITIVE

Entity: Google

Phrases

Me gusta google [Type: POSITIVE] [Entity: Google]

Cancel Save Delete

3.3. Generación de Alertas

Al comienzo del Sistema no existen Entidades, Sentimientos, Frases o Alarmas.

Sentiment Analysis

Entity

Entity

Sentiment

Phrase

Alarm

Report Analysis

Report Alarm

Data

Name:

Entities

Cancel Save Delete

Sentiment Analysis

Sentiment

Entity

Sentiment

Phrase

Alarm

Report Analysis

Report Alarm

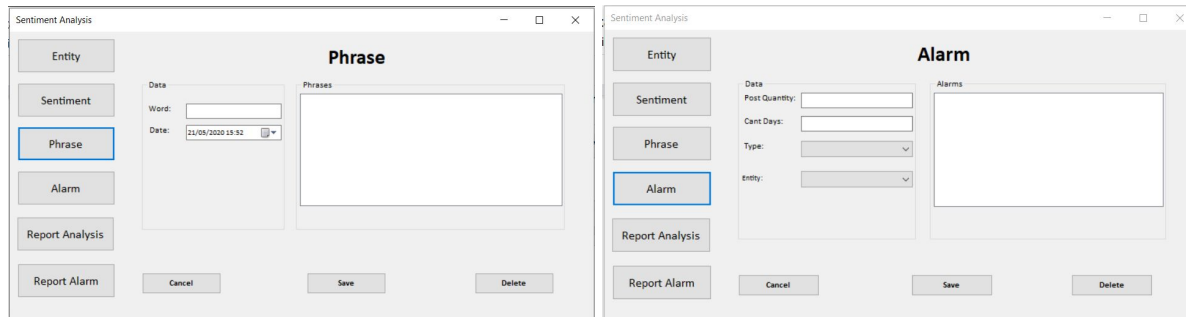
Data

Type: POSITIVE

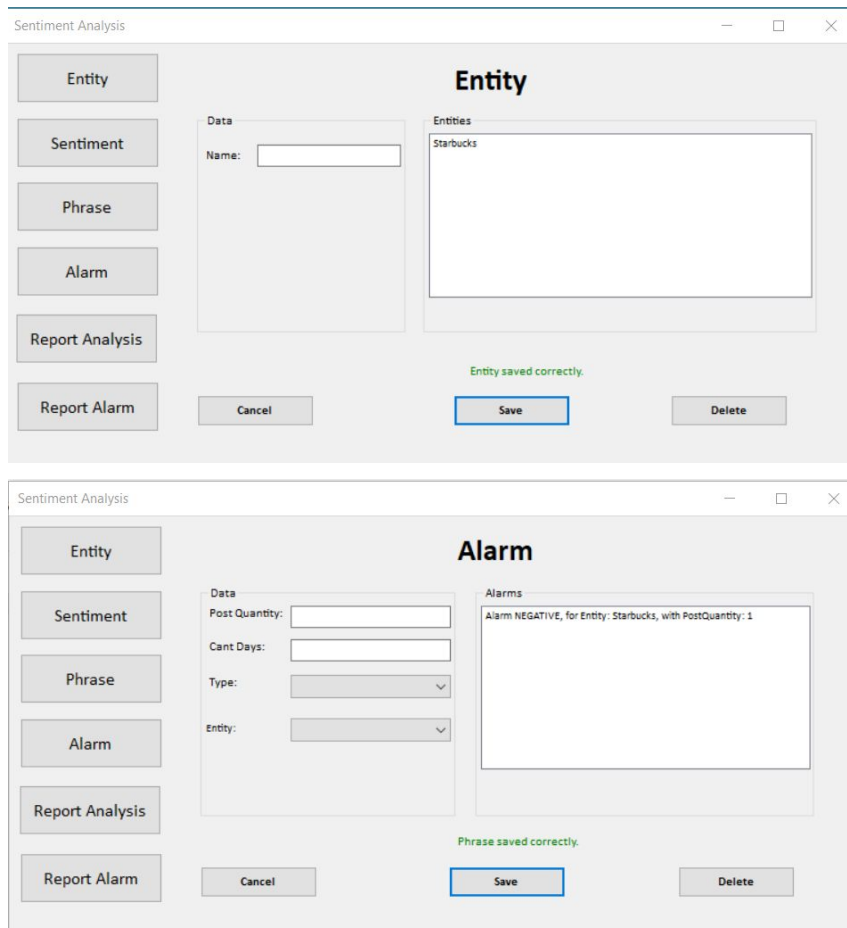
Word:

Sentiments

Cancel Save Delete



Doy de alta una nueva entidad y luego creó una Alarma para esa entidad.



Ahora al crear una Frase negativa, que haya sido ingresada el día 19/05/2020, la alarma poseería la cantidad de comentarios necesarios para crear una alarma.

Para esto creó un sentimiento negativo y luego la frase.

Sentiment Analysis

Sentiment

Entity

Sentiment

Phrase

Alarm

Report Analysis

Report Alarm

Data

Type:

Word:

Sentiments

no quiero - [NEGATIVE]

Sentiment saved correctly.

Cancel Save Delete

Sentiment Analysis

Phrase

Entity

Sentiment

Phrase

Alarm

Report Analysis

Report Alarm

Data

Word:

Date:

Phrases

no quiero ir a starbucks [Type: NEGATIVE] [Entity: Starbucks]

Phrase saved correctly.

Cancel Save Delete

Al consultar la alarma vemos que la misma se activó, por ende generó alarma y tiene un post en su contador.

Sentiment Analysis

Alarm

Entity

Sentiment

Phrase

Alarm

Report Analysis

Report Alarm

Data

Post Quantity:

Cant Days:

Type:

Entity:

Post Count:

☒ Is Enabled?

Alarms

Alarm NEGATIVE, for Entity: Starbucks, with PostQuantity: 1

Cancel Save Delete

Alerts Report

	Entity	InitDate	NumberDays	SentimentType	CountPost
▶	Starbucks	21/05/2020 15:59	2	NEGATIVE	1

Alarm

Report Alarm