



Departamento de Ingeniería Informática  
Paradigmas de la programación

Universidad de Santiago de Chile  
2021



Universidad de Santiago de Chile  
Facultad de Ingeniería  
Departamento de Ingeniería Informática  
Laboratorio Paradigmas de la programación  
Paradigma Funcional  
Catalina Isidora Riquelme Zamora

Profesor: Víctor Flores  
Sección: B-2

Agosto 2021  
Santiago - Chile



## Contenido

Introducción .....	2
Descripción del problema .....	2
Descripción breve del paradigma.....	2
Análisis del problema .....	3
Diseño de la solución.....	4
Aspectos de implementación.....	4
Instrucciones de uso.....	6
Resultados y autoevaluación.....	6
Conclusión .....	7
Referencias.....	9
Anexo .....	10



## Introducción

Para un programador que desea desarrollarse de manera óptima es imprescindible conocer y adquirir diversos paradigmas de la programación, siendo este un marco de referencia que indica la forma de programar, enfrentar un problema, analizarlo, diseñar una solución e implementación, estos son constituidos por principios y un grupo acotado de conceptos sobre cómo se percibe el mundo. Los paradigmas de programación a diferencia de los lenguajes de programación no son tan variados siendo esto una ventaja para quien desea aprenderlos, dentro de estos se encuentra el paradigma lógico el cual destaca por centrarse en el qué y no en el cómo, en base a esto el proyecto a trabajar se verá totalmente ligado a este paradigma.

## Descripción del problema

El problema por abordar en este laboratorio consta de la simulación e implementación a través del paradigma orientado a objetos de una red social para llevar a cabo esto es necesario contar con un uso correcto del paradigma, es decir, cumplir con la utilización de clases, atributos, métodos y objetos de las clases.

## Descripción breve del paradigma

El paradigma orientado a objetos, POO, tiene como objetivo el desarrollo computacional basado en una colección de objetos que están interrelacionados y trabajan conjuntamente para resolver un problema, este paradigma se caracteriza por ser más cercano a la manera en la que se modela la mayoría de las cosas en la vida real, diferenciándolo de otros paradigmas de la programación.

Para la elaboración de este laboratorio se utiliza el concepto de Objeto, el cual hace referencia a unidades las cuales pueden tener diversos atributos (características) y métodos (comportamientos). En POO el objetivo principal es realizar programas los cuales son diseñados con la finalidad de crear objetos, los cuales si es necesario interactúen entre sí.

Los pilares de este paradigma son la abstracción, clases, encapsulación, herencia y polimorfismo, en donde la abstracción consiste en permite la representación de las diversas características esenciales de un objeto. Por otra parte, las clases son una plantilla las cuales definen de manera genérica como van a ser los objetos de un determinado tipo. El encapsulamiento permite asegurar que el contenido de la información de un objeto esta oculta al mundo exterior, es decir, un objeto A no conoce lo que hace el objeto B y viceversa. Por otro lado, la herencia define relaciones jerárquicas entre clases, de forma que los atributos y métodos comunes puedan ser reutilizados, facilitando el trabajo del programador. Finalmente, el polimorfismo consiste en diseñar objetos para compartir comportamientos, lo que nos permite procesar objetos de diferentes maneras. Un aspecto esencial para el diseño de la solución es tomar en cuenta los conceptos de acoplamiento y cohesión, los cuales permiten un nivel elevado de calidad del diseño.



## Análisis del problema

Para lograr implementar de manera correcta y optima la red social se aborda el primer requisito funcional el cual es implementar diversas clases y estructuras que le darán forma al programa, para lograr esto se implementaron diversas clases, incluyendo sus respectivos constructores, getter y setters. Dentro de esta plataforma existirán un total de cinco clases user, post, social, reaction y comment.

Una vez definidas las clases con las que se trabajará se realiza la implementación del resto de funcionalidades que tendrá esta plataforma, como lo es el menú interactivo por consola, el cual permite la interacción del usuario con la solución, logrando así una simulación de una red social.

Por otra parte, la funcionalidad authentication permite el registro de nuevos usuarios dentro de la plataforma, donde se almacena el nuevo usuario en la red social junto con su fecha de registro e identificador, esta función recibe el nombre de Register dentro de la plataforma. Una función por destacar dentro de la funcionalidad authentication es Login, la cual permite el ingreso de un usuario a la plataforma, validando las credenciales del usuario antes de permitir su ingreso. Finalmente, la funcionalidad Logout permite la salida de un usuario de la plataforma, donde debe existir una sesión iniciada previamente para lograr salir de ella de manera correcta.

La siguiente funcionalidad dentro de la plataforma es post, la cual permite a un usuario con sesión activa dentro de la red social realizar una nueva publicación, la cual puede ser dirigida a otros usuarios que se encuentren registrados dentro de la misma red social, esta publicación debe ser almacenada dentro de la red social logrando rellenar los diversos atributos con los cuales cuenta la clase Post.

Otra funcionalidad requerida es follow la cual permite a un usuario con sesión activa dentro de la red social poder seguir a otro usuario que se encuentre registrado dentro de la plataforma, en donde el contenido debe quedar vinculado de alguna manera al usuario autor, además es posible compartir esta publicación con más de un usuario, con el único requisito que estos se encuentren registrados.

La funcionalidad visualize permite obtener una representación visual y comprensible por el usuario de la plataforma, esta funcionalidad se divide en dos funciones, la primera SocialNetworkToString la cual transforma la red social a un String, esta puede ser ejecutada tanto con sesión activa, como sin ella, en el caso de ser ejecutada con una sesión activa se debe entregar todos los elementos que contenga este usuario activo junto con los detalles de este, por el contrario en caso de ser ejecutada sin sesión activa se debe entregar todo lo que se encuentre registrado dentro de la red social. La segunda función de visualize es PrintSocialNetwork la cual permite imprimir en pantalla el string obtenido gracias a SocialNetwork en un formato comprensible por el usuario, evitando mostrar datos sensibles.

Finalmente, la funcionalidad comment permite a un usuario con sesión iniciada dentro de la plataforma comentar tanto publicaciones como comentarios, este comentario debe ser almacenado dentro de la red social logrando rellenar los diversos atributos de la clase Comment.

## Diseño de la solución

El enfoque para la resolución de este laboratorio es la representación de la red social a través de seis clases las cuales representan las diversas relaciones en esta plataforma. La clase Social presenta cuatro atributos, `userArrayList`, arreglo de User el cual almacena los usuarios registrados en la red social; `postArrayList`, arreglo de Post el cual representa las publicaciones realizadas en la red social; `commentArrayList`, arreglo de Comment el cual almacena los comentarios realizados en la red social; `activeUser`, arreglo de User el cual almacena al usuario activo en caso de que exista una sesión activa.

Por otra parte, la clase Usuario, la cual tiene como atributos, `name`, String el cual almacena el nombre de registro del usuario; `password`, String que almacena la contraseña escogida por el usuario; `dateUser`, String el cual indica la fecha de registro del usuario en la plataforma; `idUser`, entero que representa el identificador único del usuario; `postListUser`, arreglo de Posts donde se almacenaran las publicaciones realizadas por el usuario; `followersArrayList`, arreglo de Strings donde se almacenan los seguidores del usuario; `feedUser`, arreglo de Posts donde se almacenaran las publicaciones compartidas y menciones que tenga el usuario dentro de la plataforma.

La clase Post cuenta con seis atributos los cuales son `idPost`, entero el cual representa el identificador único de la publicación; `contentPost`, String que almacena el contenido de esta; `authorPost`, String que almacena el autor de contenido; `datePost`, String que almacena la fecha de creación del post; `typePost`, String que almacena el tipo de contenido (para este laboratorio únicamente se utilizara texto); `commentListPost`, arreglo de Comment el cual almacena los diversos comentarios referentes a la publicación.

Finalmente, la clase Reaction es una clase abstracta la cual tiene los siguientes atributos `idReaction`, entero el cual representa el identificador único de la reacción; `authorReaction`, String el cual almacena al autor de la reacción; `dateReaction`, String que almacena la fecha de creación de la reacción; `contentReaction`, String el cual el contenido de esta; `typeReaction`, String que almacena el tipo de contenido (para este laboratorio únicamente se utilizara texto). La clase comment es la clase hija de Reaction, por lo tanto, hereda todos sus atributos, además de estos contiene el atributo `commentListComment`, arreglo de Comment el cual almacena los comentarios realizados al comentario. Cabe destacar que la última clase Main no cuenta con atributos propios y contiene todo lo relacionado al menú interactivo con el usuario por consola, además de funciones esenciales para que el programa funcione de manera correcta.

Tras la definición de las diversas clases para el correcto funcionamiento del programa se implementan las diversas funciones de la red social. En el caso de la función Register se busca incorporar al arreglo contenedor de los usuarios de la red un nuevo usuario, en el caso de que este usuario ya se encuentre registrado la función se muestra un aviso avisando del error y no se registra de manera exitosa el usuario, en caso contrario se incorpora a la lista de usuarios pertenecientes a la plataforma.



Para implementar la función Login se busca dentro del arreglo contenedor de usuarios registrados las credenciales ingresadas por el usuario (usuario y contraseña), en el caso de encontrar alguna coincidencia se permite el ingreso del usuario a la red social, sacando el usuario de la lista contenedora con el fin de trasladarlo al arreglo contenedor del usuario activo. En el caso contrario se muestra un mensaje explicando el error. Por otro lado, para implementar la función Logout se realiza el proceso contrario, es decir, se retira el usuario desde la lista contenedora del usuario activo con la finalidad de regresarlo al arreglo que contiene el resto de usuario registrados en la plataforma.

Para implementar la función Post se define un identificador único para la nueva publicación a través de obtener el largo del arreglo contenedor de las publicaciones dentro de la red social adicionándole uno, además de obtener la fecha actual del sistema con la finalidad de completar los diversos atributos iniciales de la clase, en el caso de que el usuario desee dirigir la publicación creada una vez creado el post se agregara al atributo feed de cada usuario.

Para implementar la función Follow se busca dentro del arreglo contenedor de usuarios registrados alguna coincidencia con el usuario a seguir por el usuario activo, en el caso que exista alguna coincidencia se agrega a la lista de seguidores el nombre del usuario activo, registrándolo como seguidor del nombre ingresado. En el caso que no exista coincidencia al momento de buscar en los usuarios registrados, se retorna un error avisándole al usuario que el nombre ingresado no se encuentra dentro de la plataforma.

Para implementar la función Share se revisa en primera instancia que tanto como el usuario ingresado como el id de publicación sean válidos, en el caso contrario se le avisa al usuario a través de un aviso de error solicitando los datos nuevamente, una vez revisado se agrega a cada uno de los usuarios ingresados la publicación dentro del arreglo feed, registrando la publicación compartida.

Finalmente para implementar la función Comment se le ofrece al usuario dos posibilidades una de ellas es comentar una publicación, en donde se busca dentro del arreglo contenedor de publicaciones registradas en la red social el identificador ingresado por el usuario, en el caso de encontrar una coincidencia se registra un nuevo comentario dentro del arreglo en donde se almacenan los comentarios realizados a la publicación, en el caso contrario se le avisa al usuario a través de un aviso de error solicitando los datos nuevamente. La segunda opción que puede escoger el usuario es comentar un comentario, en donde se busca dentro del arreglo contenedor de comentarios registrados en la red social el identificador ingresado por el usuario, en el caso de encontrar una coincidencia se registra un nuevo comentario dentro del arreglo en donde se almacenan los comentarios realizados al comentario, en el caso contrario se le avisa al usuario a través de un aviso de error solicitando los datos nuevamente.

## Aspectos de implementación

Del mismo en que fue realizado el primer laboratorio, referente al paradigma funcional, se estructuró el código de manera que cada archivo contiene una clase diferente. Para lograr una correcta implementación del proyecto se utilizó el lenguaje Java donde se aplicó la versión libre de la plataforma de desarrollo Java bajo concepto de lenguaje orientado a objetos, es decir, OpenJDK versión 11, con el IDE IntelliJ IDEA la herramienta de compilación para el proyecto fue Gradle, el ambiente de desarrollo donde se ejecutó el entregable fue el sistema operativo Windows 10.

## Instrucciones de uso

Para ejecutar el proyecto Java de manera correcta con Gradle, basta con abrir una terminal/console en el directorio principal del proyecto y ejecutar el siguiente comando:

`gradlew run` (en GNU/Linux)

`gradlew.bat run` (en Windows)

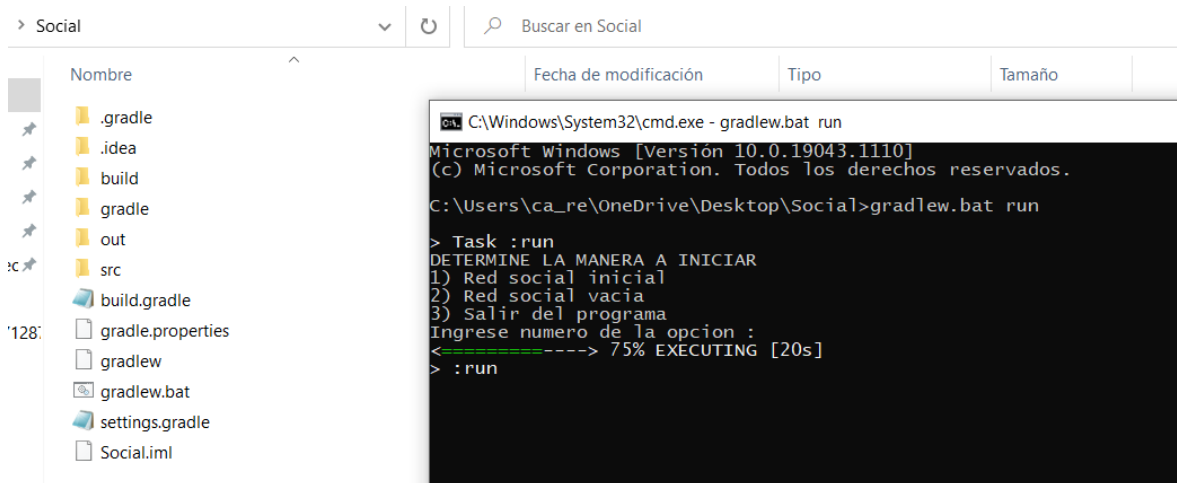


Figura 1: Ejemplo ejecución programa

Por otro lado, al momento de iniciar el programa se desplegará un menú interactivo con el cual el usuario puede interactuar de manera amigable, donde en primera instancia se determinará si se comenzará con una red social ya creada, la cual ya cuenta con usuarios y publicaciones registradas, por el contrario, una red social vacía. Cualquiera sea la opción elegida por el usuario lo llevará a un nuevo menú el cual le mostrará las diversas interacciones que puede realizar con la plataforma. Cabe destacar que algunas funcionalidades requieren una sesión activa para lograr esto se puede registrar como usuario nuevo en la plataforma o de lo contrario ingresar a la plataforma a través de credenciales ya registradas anteriormente. Una vez iniciada la sesión se despliegan las siguientes funciones: realizar publicación, seguir a un usuario, compartir publicación, visualizar red social, comentar publicación/comentario y cerrar sesión. Es importante destacar que si el usuario lo desea puede interactuar con la red social de diversas maneras para posteriormente cerrar sesión e iniciar con credenciales diferentes, cambiando así el usuario activo.

La mayoría de los posibles errores en los cuales puede caer el programa es el ingreso por teclado de entradas, por ejemplo, al ingresar un string en vez de un número entero o ingresar un número fuera del rango al interactuar con los menus. Un posible error importante por destacar es al momento de seguir a una persona, esta funcionalidad no contiene un verificador al momento de volver a seguir a un determinado usuario, es decir, se puede seguir mas de una vez a un usuario sin que el programa avise que existe un error, es importante evitar esta integración de la manera equivocada ya que puede provocar errores en futuras interacciones y no será avisado al usuario.

Los ejemplos para probar el programa se encuentran anexos al documento.

## Resultados y autoevaluación

Para la realización de este proyecto hubo varios requerimientos que debían cumplirse para la posterior evaluación, entre estos requerimientos se pueden apreciar los funcionales y los NO funcionales. A continuación, se presentarán dos tablas indicando los requerimientos y el grado de logro de cada uno, es decir, la autoevaluación.

Estado	Puntaje
No realizado	0
Funciona el 25% de las veces o no funciona	0.25
Funciona el 50% de las veces	0.5
Funciona el 75% de las veces	0.75
Funciona el 100% de las veces	1

Requerimientos funcionales	Evaluación
Clases e interfaces	1
Menú interactivo por terminal	1
Authentication	1
Post	1
Follow	1
Share	1
Visualize	1
Comment	1
Like	0
Is Viral	0
Search	0
Bot Post	0

Requerimientos NO funcionales	Evaluación
Autoevaluación	1
Lenguaje y herramientas de trabajo	1
Interacciones con el programa	1
Uso del paradigma	1
Prerrequisitos	1
Documentación	1
Organización del código	1
Diagrama de análisis	1
Diagrama de diseño	1
Uso de git	1

Se realizaron un total de 14 pruebas, realizándose 2 pruebas por cada requerimiento funcional, exceptuando la funcionalidad clases e interfaces, estas funcionaron en su totalidad, logrando un 100% en su funcionalidad. No se lograron realizar dos requerimientos funcionales, Like, Is Viral, Search y Bot Post, correspondientes a los predicados a implementar de manera libre, estos no





fueron implantados ya que no se logró llegar a una correcta implementación de estos por lo que se decidió dar preferencia a los requerimientos anteriores, para que funcionaran de manera óptima.

## Conclusión

Al realizar el proyecto, se presentaron diversas complicaciones en su elaboración debido principalmente a la limitación del tiempo, perjudicando la elaboración de algunos requerimientos funcionales. Un aspecto positivo al momento de realizar este proyecto fue contar con información desarrollada durante el laboratorio de lenguaje funcional y lógico lo cual ayudo en la definir las bases del actual proyecto. Además, se puede destacar de manera positiva que a diferencia de los laboratorios pasados tanto funcional como lógico, este si cuenta con una ejecución amigable con el usuario, ya que cuenta con un menú interactivo el cual permite al usuario navegar a través de la red social de manera fácil y rápida, en comparación a los proyectos pasados.

Los resultados alcanzados en la elaboración de este proyecto utilizando el paradigma orientado a objetos fue superior respecto al paradigma funcional y lógico, debido a que se realizo un número de funciones con mayor integridad, incorporado un número mayor de verificador con la finalidad de evitar posibles caídas al interactuar con el usuario. Por otro lado, comparado al trabajo realizado con el paradigma lógico este fue menor, ya que se alcanzo un número menor de requerimientos funcionales, debido principalmente al menor tiempo en comparación al paradigma anterior.



## Referencias

Canelo, M. M. (2020, 14 diciembre). *¿Qué es la Programación Orientada a Objetos?* Profile Software Services. Recuperado de: <https://profile.es/blog/que-es-la-programacion-orientada-a-objetos/>

JavaTpoint(s.f.) *"Java toString() method"*. Recuperado de: [https://www.javatpoint.com/understanding-toString\(\)-method](https://www.javatpoint.com/understanding-toString()-method)

## Anexo

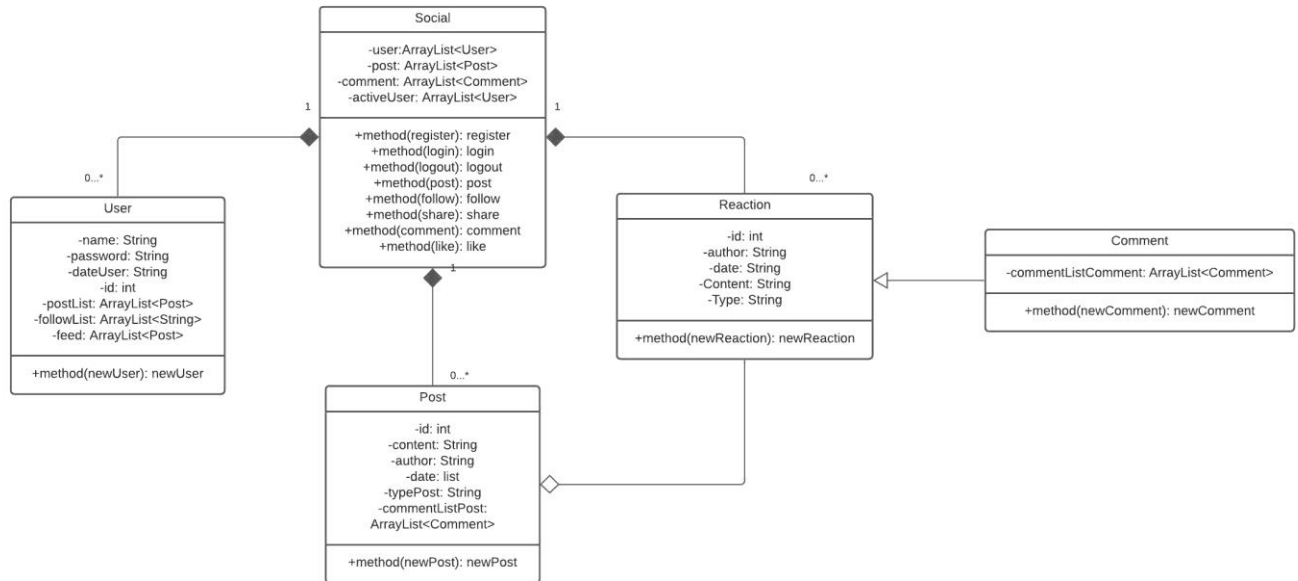


Figura 2: Diagrama de análisis

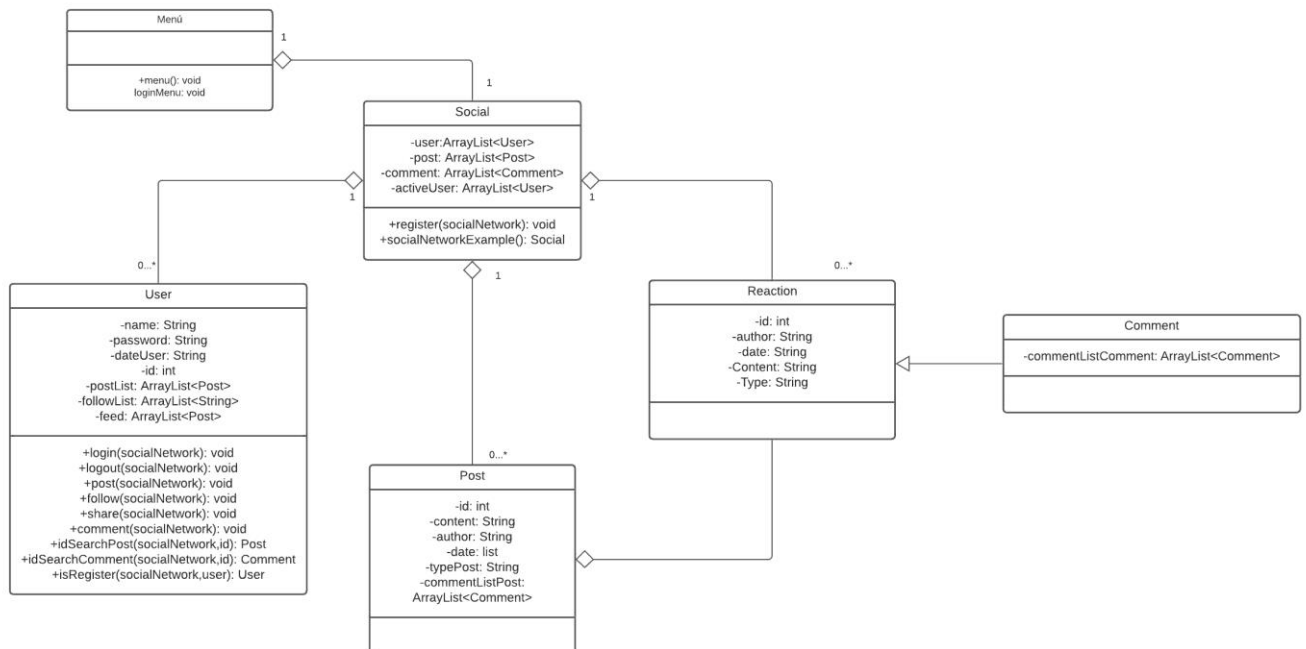


Figura 3: Diagrama de diseño



## **Red social inicial**

### Usuarios registrados

User 1 Usuario:cas Contraseña:123 Seguidores:[lizzy]

User 2 Usuario:lizzy Contraseña:pass Seguidores:[cas, maria, isidora]

User 3 Usuario:isidora Contraseña:pass123 Seguidores:[elizabeth]

User 4 Usuario:maria Contraseña:123 Seguidores:[isidora]

User 5 Usuario:elizabeth Contraseña:pass Seguidores:[cas]

### Publicaciones registradas

Post 1 Contenido: Hola mundo Autor: 'lizzy Fecha: 20/nov/2020

Post 2 Contenido: ¿Cual sera la temperatura de hoy? Autor: 'isidora Fecha: 18/nov/2021

Post 3 Contenido: Hoy tenemos prueba Autor: 'cas Fecha: 27/dic/2021

Post 4 Contenido: Yo creo que la tierra es plana Autor: 'maria Fecha: 14/dic/2021

Post 5 Contenido: ¿Cuándo empiezan las vacaciones? Autor: 'isidora Fecha: 12/nov/2020

Post 6 Contenido: ¿Ya dije Hola Mundo? Autor: 'lizzy Fecha: 27/dic/2021

Post 7 Contenido: Me encantan los caballos Autor: 'maria Fecha: 23/dic/2021

Post 8 Contenido: Hoy comenzare a comer mejor Autor: 'cas Fecha: 21/nov/2020

Post 9 Contenido: ¿Cuándo acaba el semestre? Autor: 'elizabeth Fecha: 3/dic/2021,

Post 10 Contenido: Adiós mundo Autor: 'isidora Fecha: 13/dic/2021