

Algoritmos y Programación 2
Vistara

Planteamiento del problema y diseño de la solución

Integrantes:

Juan Felipe Blanco Tigreros A00365164

Julian Camilo Bolaños Arias A00365918

Docente:

Juan Manuel Reyes

Monitor:

Gustavo Adolfo Villada

Universidad ICESI
Valle del Cauca, Cali
Mayo 15 de 2021

Enunciado

Tras las marchas y manifestaciones realizadas en Colombia entre la última semana de abril y primeras semanas de mayo, el flujo de información sobre lo que acontecía en nuestro entorno incrementó de manera abrupta tras las múltiples problemáticas asociadas a estas movilizaciones. En dicho flujo de información, el número de fuentes ha abundado, sin embargo la veracidad de dichas noticias y publicaciones es algo que ha sido juzgado por la sociedad en general. A este hecho se suma que no necesariamente los medios de comunicación tradicionales, como es el caso de RCN, han llegado a difundir noticias falsas o tergiversadas. Con lo anterior, encontrar información verídica se ha vuelto un problema del día a día de los colombianos, problema que se puede solventar por medio de nuestro programa, **Vistara**.

Vistara busca generar un espacio en el cual sus usuarios dispongan de manera rápida, fácil y sencilla de toda la información y noticias que están sucediendo en torno a un tema en particular. A su vez, se busca diferenciar las noticias falsas de las verídicas, esto con la finalidad de disminuir el alto grado de desinformación a la que está expuesta la población. Nuestro objetivo es difundir, conectar e informar.

Vistara estará compuesto por usuarios y publicaciones. Todo usuario tendrá de manera visible un nickname, una foto de perfil, una breve descripción, las publicaciones que haya hecho, publicaciones verificadas, un listado de usuarios que lo sigue y un listado de usuarios a los que sigue. Para crear dicho usuario se requerirá de un correo, una contraseña y el nickname antes mencionado. En la aplicación también existen los moderadores; son los usuarios encargados de definir qué publicaciones son veraces o falsas, crear categorías para las distintas publicaciones y eliminar publicaciones inapropiadas de la aplicación. No todos los usuarios son moderadores, un moderador tiene un alto grado de responsabilidad social y únicamente un usuario puede ascender a moderador cuando otro moderador lo indique y, a la vez, dicho usuario cumpla con los siguientes requisitos:

1. Tener al menos un mes de antigüedad en la aplicación (30 días).
2. Haber realizado al menos 20 publicaciones y que dichas publicaciones todas se encuentren verificadas.

Un moderador contará con un listado de publicaciones por verificar, él/ella tendrá un apartado específico donde encontrará todas las publicaciones que le fueron asignadas para verificar. En dicha sección habrán tres opciones: Verificar, marcar como falsa o eliminar, esta última ayuda a que los moderadores eliminen publicaciones que no estén relacionadas con el objetivo de la aplicación. Los moderadores de la aplicación estarán ordenados en un árbol binario de búsqueda, ordenado acorde a la cantidad de publicaciones que tengan por verificar,

esto con el objetivo de que una vez una publicación sea creada, se le asigne al moderador con menor número de publicaciones por verificar para no sobrecargar a algunos moderadores y distribuir adecuadamente el trabajo entre los moderadores de la aplicación.

Los moderadores también podrán crear categorías/temas con un nombre y un diseño (imagen) para permitir a los usuarios filtrar las noticias por tema.

Las publicaciones pueden ser de distintos tipos, sin embargo, todas ellas tendrán en común una referencia al usuario que las creó, un título, un texto como contenido, una fecha, un listado de los usuarios que han reaccionado, una serie de comentarios realizados por usuarios, un enlace a la noticia completa, una categoría y un estado, que puede ser: Verificada, Sin Verificar o Noticia Falsa.

Los distintos post que pueden existir son aquellos que contengan únicamente texto o una imagen. Por otro lado. Todo contenido (Publicación o comentario) se deberá visualizar junto con la foto de perfil del usuario que lo creó, al igual que todo usuario puede crear contenido.

Con el fin de que cada usuario pueda escoger qué noticias o temas prefiere leer de manera rápida, los usuarios contarán con un *feed* de publicaciones y un apartado de *trending*. El *feed* estará compuesto por las publicaciones de los usuarios que sigue, mientras que el *trending* tendrá las publicaciones más recientes, más comentadas y más reaccionadas en la aplicación.

Para que la publicación de un usuario aparezca en el *feed* de uno de sus seguidores, el seguidor lo deberá haber empezado a seguir antes de que este realizara la publicación, en caso contrario, un usuario no va a ver en su *feed* publicaciones previas al instante en el que comenzó a seguir al otro usuario.

Para que una publicación aparezca en *trending* dependerá de un rating que calculará la aplicación. El *trending* está ordenado de mayor a menor rating, los post con mayor rating aparecerán primero en este apartado, mientras que los de menor rating se encontrarán al final de este. El *trending* estará compuesto por publicaciones de al menos un rating superior a 5 y que no tengan más de un día de publicadas.

El rating es calculado por medio de la suma de la cantidad de reacciones y comentarios de la publicación, dividido entre los minutos transcurridos desde su publicación.

Siguiendo la idea de que **Vistara** busca conectar a las personas por medio de la información, un usuario podrá buscar a otro en la lista ordenada (A - Z) de todos los

usuarios en la aplicación y comenzar a seguirlo si así lo desea. Al encontrarlo se desplegarán:

1. Las noticias o publicaciones que ha hecho.
2. La cantidad de seguidores que tiene.
3. La cantidad de gente a la que sigue.
4. La cantidad de publicaciones que tiene.
5. La cantidad de publicaciones verificadas que ha hecho.
6. La cantidad de publicaciones falsas que ha publicado.

Los usuarios también van a poder filtrar publicaciones o noticias según un tema de su interés (categoría). Una vez buscado el tema, se usará la misma metodología que con el *trending* solo que para este caso se mostrarán únicamente las publicaciones de dicho tema.

Un usuario podrá visualizar constantemente su foto de perfil, su breve descripción, número de seguidores y gente a la que sigue, junto con las opciones para ver su *feed*, *trending* y la opción para crear una nueva publicación. Igualmente, tendrá la opción de dirigirse a un apartado para modificar sus datos personales.

Con el objetivo de que los usuarios que divulgan noticias entiendan qué tan atractivas son las noticias que generan y qué tanta gente interactúa con estas, se podrán generar reportes estadísticos de publicaciones en específico o de un usuario.

En los reportes de publicaciones, se encontrará cómo han evolucionado minuto a minuto las reacciones, comentarios y el rating de la noticia desde su publicación. Por su parte, los reportes de perfil o de usuario, contendrán el número de seguidores y personas seguidas, el total de publicaciones verificadas, falsas y pendientes por verificar.

En todos los procesos antes mencionados, como crear publicaciones, buscar usuarios, entre otros, en caso de presentarse un error, se le debe informar al usuario del error que se ha producido. Los principales errores de los que se debe notificar al usuario son:

1. Cuando se crea el usuario, en caso de que ya exista un usuario con el nickname escogido, se alertará y se solicitará ingresar un nuevo nickname.
2. Cuando se crea el usuario, en caso de que algún campo se encuentre vacío, se alertará y se solicitará revisar los datos faltantes.
3. Cuando se ingresa a la aplicación y se digita una contraseña incorrecta o un usuario inexistente, se alertará y se solicitará nuevamente los datos.
4. Cuando se está buscando un usuario, en caso de que no exista un usuario con el nombre indicado se alertará de esto.

En términos generales el programa se encargará de guardar sus propios datos de manera automática para brindar la mejor experiencia a los usuarios.

Requerimientos funcionales

El programa debe estar en capacidad de:

R1. Gestionar un usuario. Esto implica su creación, inicio de sesión y edición:

R1.1. Crear una cuenta de usuario acorde a los siguientes datos suministrados: Un correo electrónico, una contraseña y un nickname. El nickname deberá ser único y se deberá informar en caso de que ya exista un usuario con el nickname ingresado para cambiarlo. A su vez, ninguno de los parámetros mencionados puede ser una cadena de texto vacía. Una vez creado el usuario este se agregará al árbol binario de usuarios de la aplicación.

R1.2. Iniciar sesión con un usuario, solicitando el nickname del usuario, junto con su respectiva contraseña. En caso de que los datos ingresados coincidan con los de algún usuario en la aplicación, este pasará al menú principal de esta. Mientras que si alguno de los datos ingresados es una cadena vacía, el sistema lanzará una excepción y alertará al usuario. En caso de que los datos no coincidan con los de algún usuario, ya sea la clave, o un nombre de usuario inexistente, se lanzará una excepción y alertará de que los datos ingresados no son correctos.

R1.3. Permitir a un usuario modificar sus datos, brindándole la oportunidad de cambiar su nickname, contraseña, correo, descripción, foto de perfil y su enlace de contacto. Para esto el sistema le presentará al usuario los campos mencionados rellenos con la información actual y el usuario decidirá qué información cambiará. Una vez el usuario cambie los datos que desee, se guardarán los cambios y se modificará la información del usuario en el árbol binario de usuarios de la aplicación.

R2. Permitir al usuario buscar perfiles o noticias:

R2.1. Permitir al usuario buscar perfiles por un nickname en el buscador proporcionado. El usuario deberá ingresar el nickname del usuario a buscar y presionar la opción de buscar. En caso de que no exista ningún usuario con el nombre buscado, el sistema lanzará una excepción y notificará al usuario

de que no se encontraron usuarios con el nombre ingresado. Si el usuario sí existe, se cargará en pantalla su foto perfil, nombre de usuario, cantidad de publicaciones que tiene en general, número de publicaciones verificadas, falsas, su número de seguidores y de personas a las que sigue. A su vez se presentará un botón para empezar a seguir al usuario buscado y en caso de ser moderador, un botón para ascender al usuario a moderador.

R2.2. Permitir al usuario filtrar noticias por tema, para esto el usuario deberá escoger uno de los temas de la aplicación, los cuales se encuentran ordenados en un árbol binario por nombre y son presentados en orden descendente (A - Z). Una vez escogido, presionará en buscar y se cargará un *feed*, una pantalla con publicaciones cuya categoría sea la seleccionada.

R3. Permitir a un usuario interactuar con otros perfiles o noticias:

R3.1. Permitir a un usuario seguir a otro usuario presionando el botón de seguir en el perfil del usuario a seguir. Tras haber presionado el botón, primero se añadirá al usuario al listado de seguidores del usuario buscado. Con lo anterior, se incrementará en una unidad el número de seguidores del usuario buscado. Por su parte, el usuario que empezó a seguir al otro, añadirá a este último a su listado de cuentas seguidas e igualmente incrementará en una unidad esta misma cifra.

R3.2. Permitir a un usuario reaccionar a una publicación con un “me interesa”. Una vez el usuario haga click en esta reacción, el contador de reacciones de la publicación incrementará en una unidad.

R3.3. Permitir a un usuario comentar una publicación ingresando un texto como comentario. Una vez el usuario decida comentar, se creará un comentario con el contenido mencionado y una referencia al usuario que lo acaba de crear. Finalmente, se añadirá el comentario a la lista circular enlazada de comentarios de la publicación. Una vez el comentario sea añadido, este se podrá visualizar junto con todos los comentarios de la publicación.

R4. Permitir a un usuario gestionar su información y publicaciones:

R4.1. Permitir que un usuario cree publicaciones de distintos tipos. Siempre se ingresará un texto como contenido y esto generará una publicación de únicamente texto. En caso de que se añada una imagen a la publicación, se solicitará la ruta de la imagen. Una vez suministrados estos datos, el usuario creará la publicación y a esta se le asignará por defecto el estado de “sin verificar”. Posteriormente se enviará la publicación a que un moderador la revise.

R4.2. Visualizar estadísticas de una publicación, en las que se presente el número total de reacciones, comentarios y rating por minuto.

R4.3. Exportar estadísticas de una publicación en un archivo csv, en donde se tenga por cada minuto desde su publicación el total de reacciones, comentarios y rating en ese minuto.

R4.4. Visualizar estadísticas de un usuario, en las que se presente el número de seguidores, personas seguidas, total de publicaciones que haya hecho y estén verificadas, marcadas como falsas y pendientes por verificar.

R4.5. Exportar estadísticas de un usuario en un archivo csv, en el cual se tenga el número de seguidores, personas seguidas, total de publicaciones verificadas, falsas y pendientes por verificar.

R5. Mostrar en pantalla información relevante para el usuario:

R5.1. Desplegarle a un usuario una lista de post recientes (*feed*) basada en los usuarios a los que sigue. Para esto se recorre todo el listado de usuarios seguidos, revisando cuáles son las publicaciones más recientes, ordenando estas en una lista ordenados del más reciente al más viejo. Ninguna publicación de ningún usuario en el *feed* puede tener fecha de

publicación anterior a la fecha en la que se comenzó a seguir al usuario que la creó. En caso de que el usuario no siga a nadie o ninguno de los usuarios que sigue ha publicado algo después de que este los empezara a seguir, se cargará una pantalla en blanco con el anuncio de que no hay noticias recientes.

R5.3. Desplegar una pestaña donde se visualicen las publicaciones más recientes, con más reacciones y comentarios (*trending*), en este apartado se visualizarán las publicaciones con mayor rating en un día. Una publicación debe tener como valor mínimo de rating 5 para aparecer en este apartado.

R6. Permitir a los usuarios de tipo moderador realizar sus tareas, como la creación de otro moderador, creación de categorías y verificación de publicaciones:

R6.1. Permitir a un moderador marcar una publicación como verificada o falsa. Para esto, el moderador accederá a un apartado donde encontrará el listado de publicaciones por revisar, por cada publicación únicamente no tendrá comentarios ni reacciones, mientras que sí tendrá tres opciones: Verificar, Falsa o Eliminar. El moderador seleccionará una y confirmará su decisión. Una vez hecho esto, se actualizará el estado de la publicación y se procede a eliminar la publicación del listado de publicaciones por verificar del moderador.

R6.2. Permitir a un moderador crear una categoría o tema. Para esto el moderador deberá suministrar un nombre para la nueva categoría o tema y la ruta de una imagen opcional. Con esto, se añadirá la categoría o tema al árbol binario de categorías y temas de la aplicación. En caso de que el nombre escogido por el moderador ya exista, se le notificará y se le solicitará cambiarlo.

R6.3. Permitir a un usuario básico convertirse en moderador, al un moderador ubicarse en el perfil del usuario a convertir, le aparecerá la opción de “hacer moderador”. Una vez presionada esta opción, el sistema buscará el usuario por su nombre, verificará que cumpla con los requisitos mínimos (20 posts verificados y al menos un mes de antigüedad). En caso de que cumpla con los requisitos, se añadirá al nuevo moderador a la lista de moderadores en la primera posición, se cambiará/modificará el objeto de tipo usuario en la lista de usuarios de la aplicación, cambiándolo por uno de tipo moderador para que el nuevo moderador adquiriera la vista de moderador.

R6.4. Permitir a un moderador eliminar una publicación, al presionar el respectivo botón en el listado de publicaciones por revisar, se preguntará si desea confirmar el proceso, en caso de ser afirmativa la respuesta, se eliminará la publicación de la aplicación, lo que implica, eliminarla del listado

de publicaciones por revisar y de las publicaciones hechas por el usuario. Finalmente se reordena el listado de moderadores acorde a la cantidad de publicaciones por revisar.

R7. Gestionar publicaciones moderadas o a la espera de ser moderadas:

R7.1. Añadir un post a la lista de publicaciones para verificar de un moderador: Se recorrerá el listado de moderadores y se añadirá la noticia al listado de noticias por verificar del moderador en la primera posición de la lista. Con el moderador escogido, se añadirá al listado de publicaciones que este tiene por revisar. Una vez hecho esto, la lista de moderadores se reordenará desde los que menos publicaciones tienen por verificar hasta los que más publicaciones tienen por verificar.

R7.2. Eliminar un post a la lista de publicaciones para verificar de un moderador. Una vez el moderador ha verificado o marcado como falsa la noticia, este deberá confirmar su decisión y se modificará el estado de la publicación revisada, se eliminará la publicación de listado de publicaciones por revisar, se modificará el contador de noticias falsas o verificadas y el de noticias por verificar del usuario que creó la publicación. Finalmente se reorganizará la lista de moderadores acorde a la cantidad de publicaciones por revisar de cada uno de ellos.

R8. Manejar la información de la aplicación:

R8.1. Guardar información de manera serializada en distintos archivos, uno por cada elemento de la aplicación. El programa deberá serializar la información de todos los usuarios que se registren, junto con sus publicaciones y eventos. Esto, con la finalidad de que cada vez que se inicie el programa, este cargue la información serializada y no presente pérdida de información, aplicando así persistencia. Cabe destacar que todos estos procedimientos deberán hacerse de manera automática; no requiere de ninguna interacción con el usuario.

R8.2. Importar información de archivos de texto plano csv, el cual contendrá usuarios a cargar en el programa en cada fila. Por cada fila se tendrá el nickname, contraseña, correo, fecha en la que se creó el usuario y si es moderador o no. Este proceso únicamente lo podrá hacer un moderador y se notificará cuando el proceso se complete exitosamente o suceda un error en el proceso.

R9. Notificar los posibles errores en ejecución, por cada excepción lanzada en el sistema, se mostrará un recuadro informando al usuario del error que acaba de suceder y si es el caso, se le indicará el por qué del error.

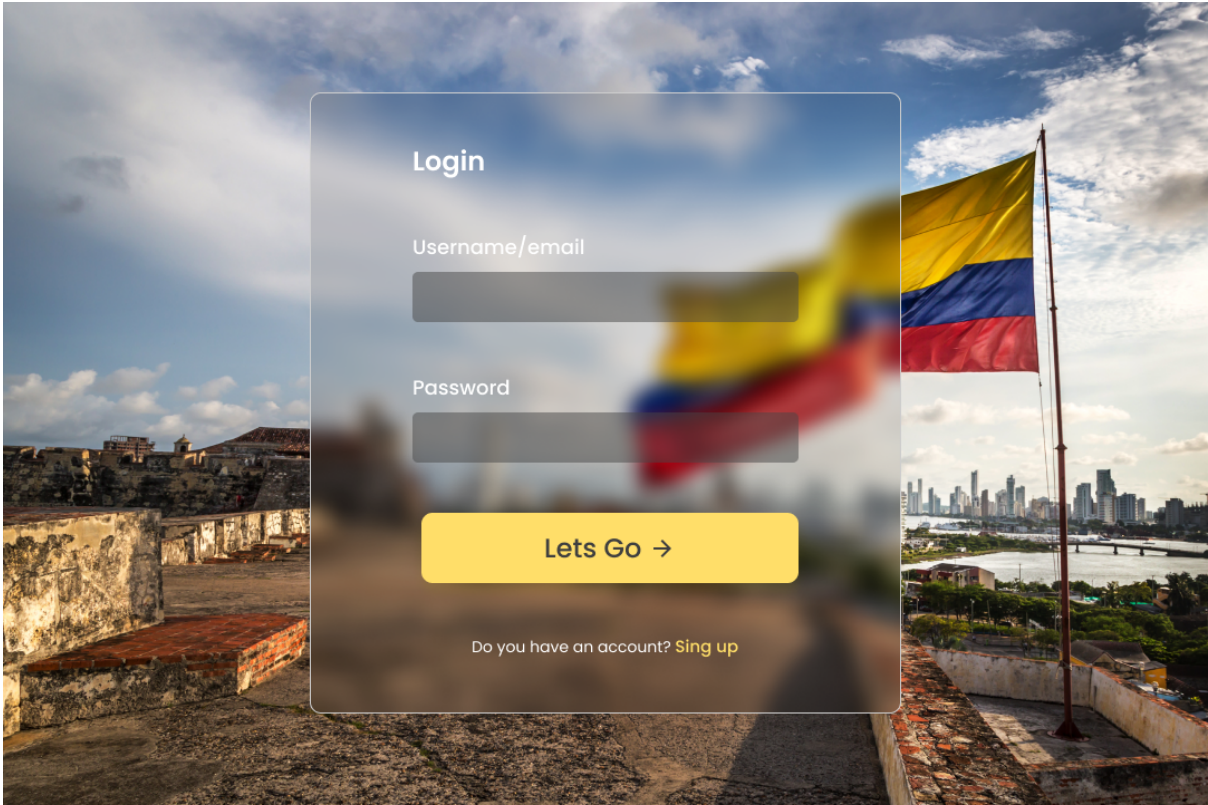
Justificación

Para el desarrollo de **Vistara** en los plazos establecidos, se requiere de mínimo dos integrantes en el proyecto debido a que es un programa con:

1. Una interfaz gráfica extensa y que hemos buscado hacer amigable con el usuario. Dicha interfaz requiere de tiempo de desarrollo para que se acople de la manera más fiel posible a los wireframes presentados, con esto, consideramos que entre 2 integrantes el desarrollo de esta interfaz será más rápido, se podrán analizar y corregir errores de manera más óptima y se podrán rediseñar adecuadamente los apartados que requieran de cambios o mejoras. Consideramos que este es un aspecto crucial en nuestro proyecto y que buscaremos tenerlo implementado lo antes posible para la segunda entrega del proyecto, acorde a los establecido.
2. Múltiples funcionalidades, partiendo de la base de que se tienen 27 requerimientos funcionales, los cuales implican una gran cantidad de procesos a administrar y codificar por parte de los integrantes. Adicionalmente, debido a que el programa ejecutará ciertos procesos por medio de hilos y de manera concurrente, esto hará del programa algo mucho más extenso y a su vez esto incrementa el grado de dificultad de este. Entre 2 integrantes estos procesos y tareas de desarrollo se pueden distribuir adecuadamente, de manera que entre los 2 podemos avanzar rápidamente y dar cumplimiento a todas las funcionalidades de la aplicación.
3. Dado el punto anterior, también aparecen el diseño y elaboración de casos de prueba, los cuales serán un factor muy útil en el proceso de desarrollo. Sin embargo, dada la extensión del proyecto, elaborar un caso de prueba para cada método implicará analizar detenidamente cada método en cada procedimiento y cómo se espera que la información se use en cada punto de la aplicación. A esto se suma el factor de que tanto la interfaz gráfica como las pruebas unitarias y su diseño se deberán presentar en una misma entrega, lo que hace que el flujo de trabajo distribuido entre ambos integrantes sea más soportable entre dos personas de lo que sería si el proyecto fuera desarrollado por un único integrante.

Wireframes

Login



Sign up



Join the **pieces**



Sign Up

Username

Email

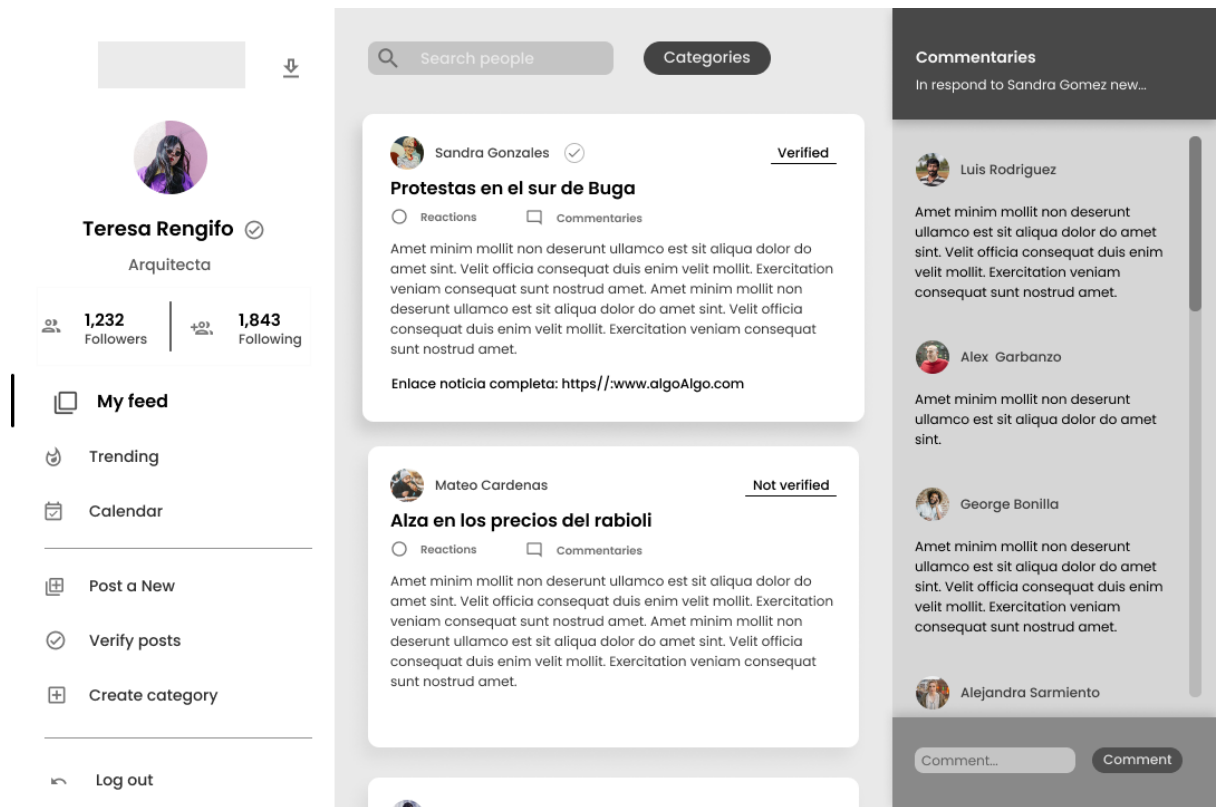
Password

Create account

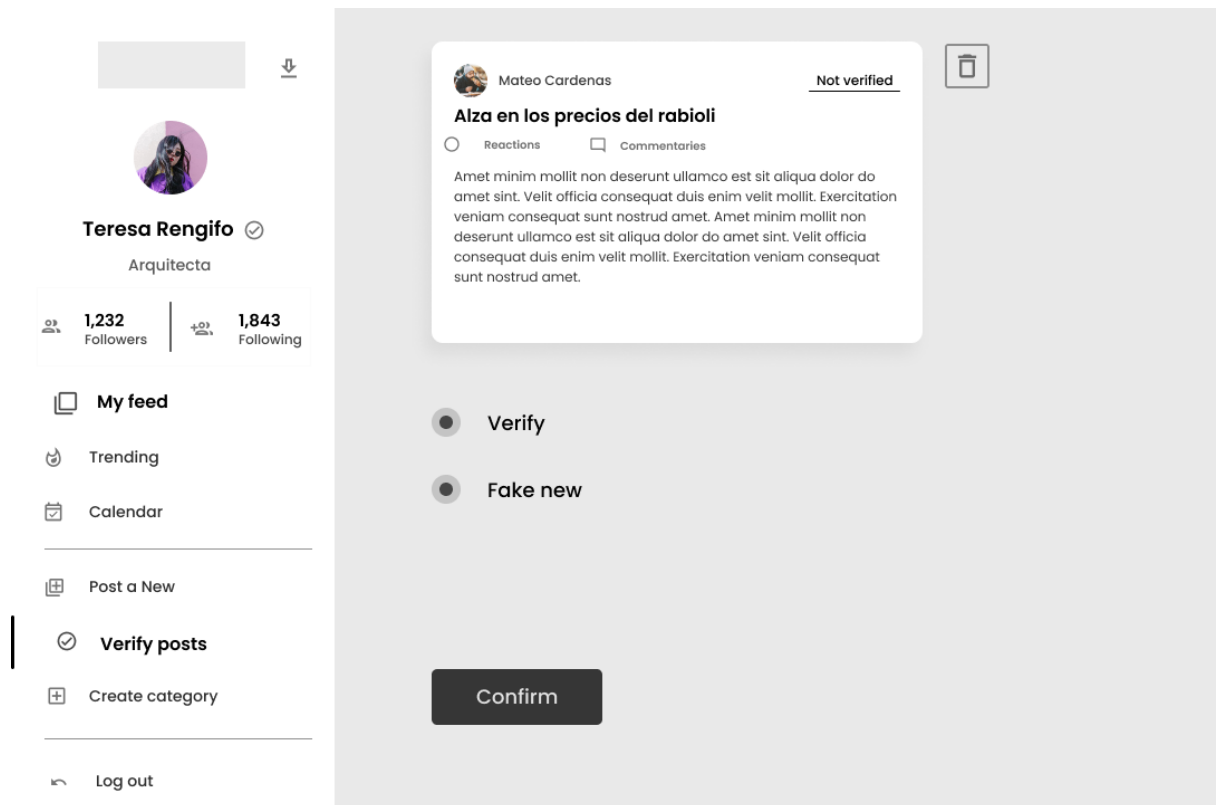
Already have an account? [Log in](#)

Moderador:


Menú inicial de moderador



Verificación de publicaciones



Creación de categoría



Teresa Rengifo ✓

Arquitecta

1,232

Followers

1,843

Following

My feed

Trending

Calendar

Post a New

Verify posts

Create category

Log out

New category name

A text with less than 20 characters

Example...

Image

Select an image from your files to upload


Add file

Path

.../access.log-2019-07-19.google_only.png

Create new category

Usuario normal: Menú inicial



Eleanor Pena

Art student

1,232

Followers

1,843

Following

My feed

Trending

Calendar

Post a New

Log out

Search people

Categories

Sandra Gonzales

✓


Verified

Protestas en el centro de Cali

Reactions

Commentaries

Social/Política



Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet.

Enlace noticia completa: <https://www.algoAlgo.com>

Mateo Cardenas

Not verified

Alza en los precios del rabioli

Reactions

Commentaries

Economía

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat

Commentaries

In respond to Sandra Gomez new...

Luis Rodriguez

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet.

Alex Garbanzo

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint.

George Bonilla

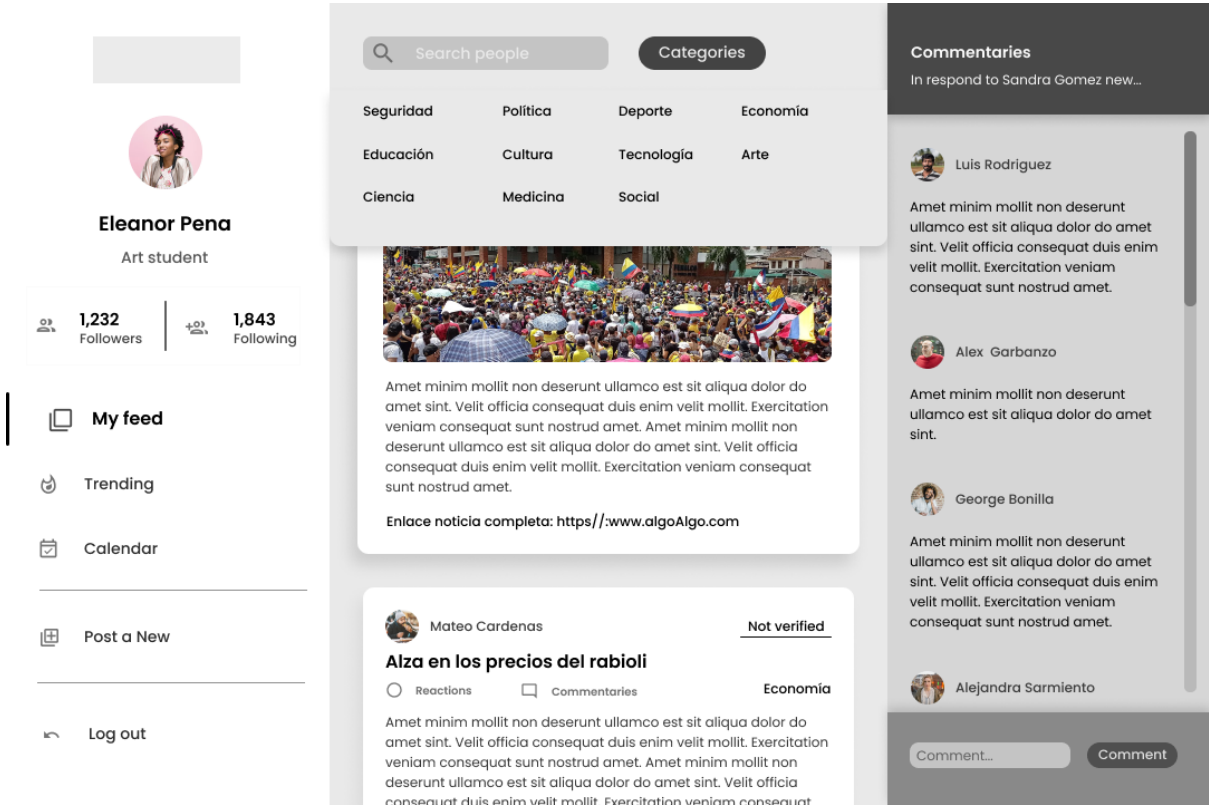
Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet.

Alejandra Sarmiento

Comment...

Comment

Menú inicial, filtro categorías




Trending




Calendario


Perfil principal




Eleanor Pena
Art student





1,232
Followers





1,843
Following

 My feed


 Trending

 Calendar


 Post a New

 Log out


Edición de perfil




Eleanor Pena
Art student





1,232
Followers





1,843
Following

 My feed


 Trending

 Calendar

 Post a New

 Log out

Creación de post



Eleanor Pena
187 posts 1,232 followers 1,843 following 140 verified posts
Art student

Edit profile


Posts

Statistics

Reacted posts


Título noticia

Not verified



☐ Reactions


☐ Commentaries



Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet.

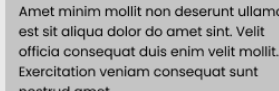
Título noticia

Verified



☐ Reactions


☐ Commentaries



Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet.


Título noticia

Verified




☐ Reactions

☐ Commentaries




Título noticia


Verified



☐ Reactions

☐ Commentaries





Change profile picture

Name

Eleanor Pena


Email

EleanorPena@gmail.com


Bio


Art student


Confirm changes





Eleanor Pena
Art student



1,232
Followers



1,843
Following

 My feed

 Trending

 Calendar

 Post a New

 Log out

Topic

Seguridad

Política

Deporte

Economía

Educación

Cultura

Tecnología

Arte

Ciencia

Medicina

Social

Tittle

A text with less than 70 characters

Example...

Details

A text with less than 500 characters

Example...

Image

Select an image from your files to upload

Optional *


Add file

Path


.../access.log-2019-07-19.google_only.png


Upload post


Estadísticas perfil





Eleanor Pena
Art student



1,232
Followers



1,843
Following


 My feed

 Trending

 Calendar

 Post a New

 Log out



Eleanor Pena
Art student

187 posts

1,232 followers

1,843 following

140 verified posts

Edit profile

Posts

Statistics

Reacted posts

Average reactions per post is the 65% of people that follows you


Average reactions - 820

Average comments - 65

Your reactions: 764

Your comments: 214

Export statistics



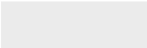
Fake posts - 0%


Verified posts - 52%

Unverified posts - 48%


Graph with monthly reports


Estadísticas post








Eleanor Pena
 Art student



1,232
 Followers



1,843
 Following


 My feed


 Trending


 Calendar


 Post a New


 Log out

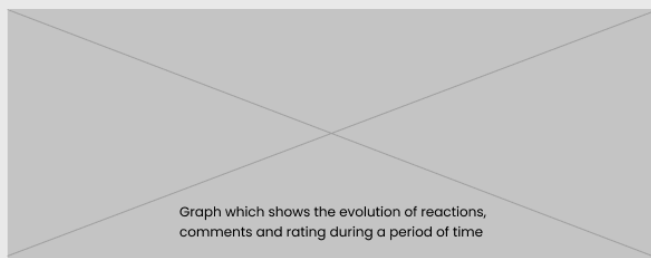
Statistics of the post: News title

Título noticia
Verified

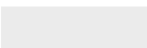
☐ Reactions
 ☐ Commentaries




Amet minim mollit non deserunt ullamco
 est sit aliqua dolor do amet sint. Velit
 officia consequat dui enim velit mollit.
 Exercitation veniam consequat sunt
 nostrud amet.





Post reaccionados








Eleanor Pena
 Art student



1,232
 Followers



1,843
 Following



 My feed


 Trending


 Calendar


 Post a New


 Log out



Eleanor Pena
Edit profile

187 posts
 1,232 followers
 1,843 following
 140 verified posts

Art student


Posts

Statistics

Reacted posts

Título noticia
verified

☐ Reactions
 ☐ Commentaries



Amet minim mollit non deserunt ullamco
 est sit aliqua dolor do amet sint. Velit
 officia consequat dui enim velit mollit.
 Exercitation veniam consequat sunt
 nostrud amet.


Título noticia
verified

☐ Reactions
 ☐ Commentaries

Amet minim mollit non deserunt ullamco
 est sit aliqua dolor do amet sint. Velit
 officia consequat dui enim velit mollit.
 Exercitation veniam consequat sunt
 nostrud amet.


Título noticia
Not verified

☐ Reactions
 ☐ Commentaries




Título noticia
verified

☐ Reactions
 ☐ Commentaries





Lista personas

perfil ajeno





Eleanor Pena
Art student


 **1,232**
Followers





1,843
Following


 My feed

 Trending

 Calendar

 Post a New


 Log out





Username
187 posts 1,232 followers 1,843 following w verified posts
User's description

Posts

Título noticia Not verified


 Reactions


 Commentaries



Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet.


Título noticia Verified


 Reactions


 Commentaries

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet.


Título noticia Verified


 Reactions


 Commentaries



Título noticia Verified

 Reactions

 Commentaries



Requerimientos no funcionales

A continuación se listan los requerimientos no funcionales de Vistara:

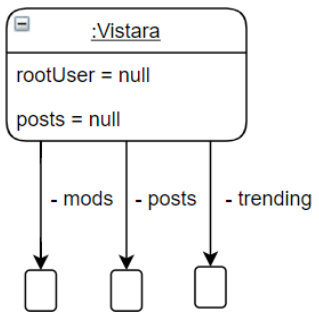
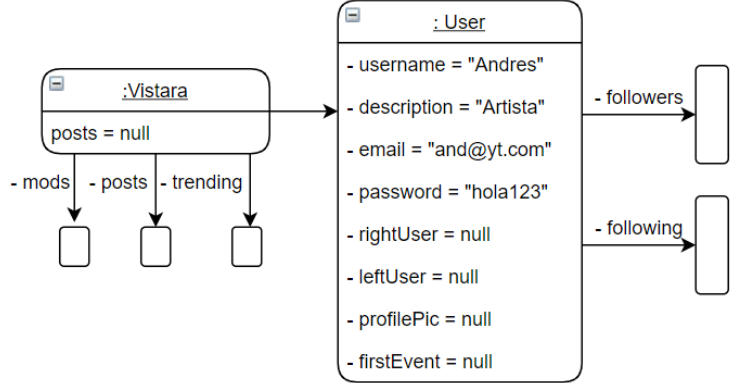
RNF 1: Recargar cada 5 minutos el apartado de *trending* que los usuarios pueden visualizar, calculando cada 5 minutos el rating de los posts de la aplicación.

RNF 2: Presentar una interfaz amigable con el usuario en la cual este se puede guiar adecuadamente.

RNF 3: Hacer uso de JfoeniX y FontAwesomeFX para hacer una interfaz que asemeja patrones de material design.

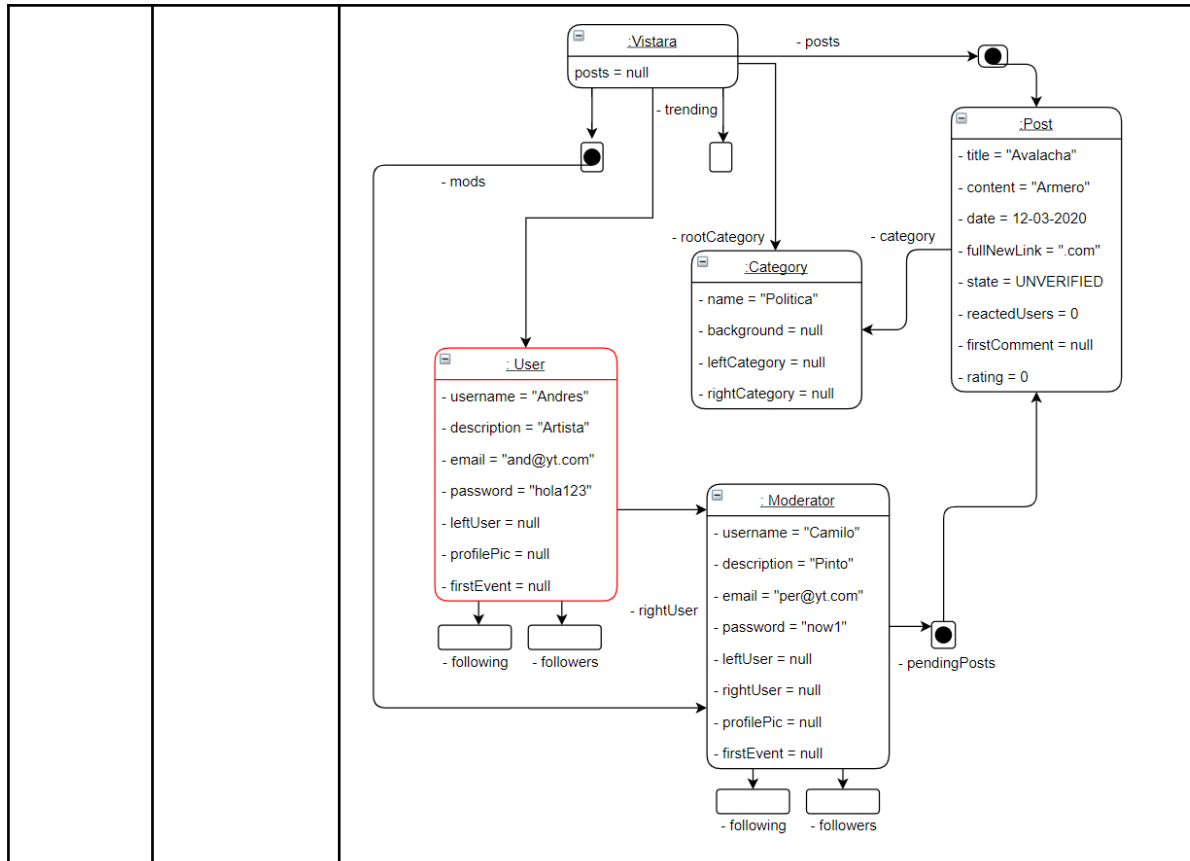
Diseño de Casos de Prueba

Configuración de los Escenarios

Nombre	Clase	Escenario
	Vistara	<p>No se tiene ninguna publicación (post), usuarios, eventos, comentarios o categorías..</p> 
setupScenario2	Vistara	<p>No se tiene ninguna publicación (post), eventos o comentarios. Se cuenta con un único usuario en la raíz del árbol binario.</p> 

setupScenary3	Vistara	<p>Se tienen 2 usuarios en el árbol binario, sin ningún contenido adicional. Las acciones se realizarán sobre el usuario en rojo.</p> <pre> graph TD Vistara[Vistara] --> mods[] Vistara --> posts[] Vistara --> trending[] posts --> User1[User: Andres] User1 --> following1[following] User1 --> followers1[followers] User1 -- rightUser --> User2[User: Camilo] User2 --> following2[following] User2 --> followers2[followers] </pre>
setupScenary4	Vistara	<p>Se tiene una única categoría en la raíz del árbol de categorías</p> <pre> graph TD Vistara[Vistara] --> mods[] Vistara --> posts[] Vistara --> trending[] Vistara -- rootCategory --> Category[Category: Politica] </pre>
setupScenary5	Vistara	<p>Existe un usuario y un post en la aplicación.</p> <pre> graph TD Vistara[Vistara] --> mods[] Vistara --> posts[] Vistara --> trending[] Vistara -- rootUser --> User[User: Camilo] posts --> Post[Post: Avalacha] </pre>
setupScenary6	Vistara	<p>Existe una única categoría y un único usuario en la aplicación.</p>

setupScenary7	Vistara	<p>Existen dos usuarios, uno de ellos es moderador y no existen categorías.</p>
Vistara	setupScenar y8	<p>Existen dos usuarios, uno de ellos es moderador, adicionalmente existe una categoría y un post pendiente por verificar.</p>



Diseño de Casos de Prueba

Objetivo de la Prueba: Validar que se creen y añadan correctamente los usuarios al árbol binario de usuarios de la aplicación

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	addUser	setupScenariy1	username = "Andres" email = "and@yt.com" password = "Hola123"	true El usuario es creado correctamente y es añadido en la raíz del árbol.
Vistara	addUser	setupScenariy2	username = "Cami" email = "cam@yt.com" password = "cam456"	true El usuario es creado correctamente y es añadido en el costado derecho del árbol
Vistara	addUser	setupScenariy1	username = "" email = "" password = "Hola123"	EmptyFieldsException No se crea el usuario y tampoco se añade este al árbol binario de búsqueda. La raíz del árbol sigue siendo null.
Vistara	addUser	setupScenariy2	username = "Andres" email = "mx@yt.com" password = "cam456"	RepeatedUsernameException No se crea el usuario y tampoco se añade este al árbol binario de

				búsqueda debido a que ya existe un usuario con dicho username.
--	--	--	--	--

Objetivo de la Prueba: Validar que un usuario puede iniciar sesión adecuadamente o se lanza la excepción correspondiente

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	login	setupScen ary1	username = "Andres" password = "Hola123"	InvalidUserException Se lanza la excepción de que el usuario no coincide con ninguno de los usuarios en la aplicación. En esta caso debido a que no hay usuarios en la aplicación
Vistara	login	setupScen ary2	username = "Andres" password = "mal"	InvalidUserException Se lanza la excepción debido a que la contraseña es incorrecta y no coincide con la contraseña del usuario.
Vistara	login	setupScen ary2	username = "Andres" password = "Hola123"	Retorna el usuario correspondiente a la clave y nombre indicados. Se permite el ingreso a la aplicación.

Objetivo de la Prueba: Validar que se puede modificar adecuadamente los datos de un usuario

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	editUser	setupScen ary2	newUsername = "Camilo" newPassword = "123" newEmail = "me@po.co" newDesc = "Artist" newPic = null	true Los valores del usuario cambiaron por los nuevos parámetros indicados, verificando que no exista ningún otro usuario con el nuevo username.
Vistara	editUser	setupScen ary3	newUsername = "Camilo" newPassword = "123" newEmail = "me@po.co" newDesc = "Artist" newPic = null	RepeatedUsernameException No se cambian los valores debido a que el nuevo username coincide con el de un usuario que ya existe.
Vistara	editUser	setupScen ary2	newUsername = "" newPassword = "123" newEmail = "me@po.co" newDesc = "Artist" newPic = null	EmptyFieldsException No se actualiza la información debido a que existe uno de los parámetros son una cadena vacía.

Objetivo de la Prueba: Validar que se esté haciendo la búsqueda de un usuario adecuadamente en el árbol binario y que se está retornando en caso de estar o se arroja un error en caso contrario

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	searchUser	setupScen ary2	username = "Andres"	Retorna el usuario que tiene como username "Andres"
Vistara	searchUser	setupScen ary3	username = "Camilo"	Retorna el usuario que tiene como username "Camilo"
Vistara	searchUser	setupScen ary2	username = "Camilo"	NullPointerException Indicando que no se encontró ningún usuario con los datos (username) indicados.

Objetivo de la Prueba: Validar que se está buscando adecuadamente una categoría según su nombre.

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	searchCategory	setupScena ry1	category = "Política"	NullPointerException Se lanza una excepción indicando que la categoría no existe
Vistara	searchCategory	setupScena ry4	category = "Política"	Retorna la categoría correspondiente con el nombre ingresado por parámetro: "Política"

Objetivo de la Prueba: Comprobar que un usuario esté en capacidad de seguir a otro usuario en la aplicación

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	followUser	setupScen ary3	followed = "Andres" follower = "Camilo"	Incrementa en una unidad el número de seguidores de "Andres", a su vez añadiendo a "Camilo" a su lista de followers. Igualmente, se incrementa en una unidad el número de usuarios que el follower "Camilo" sigue, también se añade a "Andres" al listado de usuarios que sigue "Camilo"
Vistara	followUser	setupScen ary2	followed = "Andres" follower = "Camilo"	InvalidUserException No se modifica el número de followers de "Andres" debido a que no existe ningún usuario llamado "Camilo".

Objetivo de la Prueba: Comprobar que un usuario puede reaccionar adecuadamente a una publicación con un “me interesa”

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	reactPost	setupScen ary5	usuario con username = “Camilo” post con título = “Avalancha”	Se incrementa en una unidad el número de reacciones del post y se añade el post al listado de reactedPosts del usuario, incrementando el tamaño de esta lista en una unidad.

Objetivo de la Prueba: Validar que un usuario puede crear una publicación adecuadamente.

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	createP ost	setupScen ary6	currentDate = 20-03-2020 title = “Title” content = “content” category = “Politica” fullNewLink = “go.com”	Se creará el post, añadiendo así el nuevo post al listado de publicaciones de la aplicación y del usuario. Con esto, se incrementará en una unidad el tamaño del listado. Este post será enviado a revisión de un moderador inmediatamente.
Vistara	createP ost	setupScen ary6	currentDate = 20-03-2020 title = “” content = “content” category = “Politica” fullNewLink = “go.com”	EmptyFieldsException Debido a que alguno de los campos está vacío, el post no será creado y se arrojará la excepción anterior. El listado de posts de la aplicación no se modificará ni el listado de posts hechos por el usuario.

Objetivo de la Prueba: Validar que un moderador puede convertir a un usuario normal en moderador si cumple con las condiciones dadas

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	upgrade User	setupScen ary2	verifiedPosts = 0 currentDate = LocalDateTime.now() creationDate = 5/04/2021	No se crea el moderador. El usuario permanece igual y no se modifica el árbol binario de usuarios
Vistara	upgrade User	setupScen ary2	verifiedPosts = 20 currentDate = LocalDateTime.now() creationDate = LocalDateTime.now()	No se crea el moderador. El usuario permanece igual y no se modifica el árbol binario de usuarios

Vistara	upgradeUser	setupScenary2	verifiedPosts = 20 currentDate = LocalDateTime.now() creationDate = 5/04/2021	El usuario original es removido del árbol binario, añadiendo su reemplazo como moderador en el árbol binario de usuarios.
---------	-------------	---------------	--	---

Objetivo de la Prueba: Validar que un moderador puede crear una nueva categoría de manera apropiada

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	AddCategory	setupScenary1	name = "Religion"	Se crea una nueva categoría y el número de elementos en el árbol de categorías incrementa a 1.
Vistara	AddCategory	setupScenary4	name = "Religion"	Se crea una nueva categoría y el número de elementos en el árbol de categorías incrementa a 2.

Objetivo de la Prueba: Validar que una publicación es eliminada del programa

Clase	Método	Escenario	Valores de Entrada	Resultado
Vistara	deletePost	setupScenary8	La publicación con título = "Avalancha"	El post es removido del listado de publicaciones del usuario que lo creó, del listado de posts de la aplicación y del listado de posts a verificar de un moderador. Todos los anteriores listados disminuyen en un unidad. El total de posts en la aplicación es 0
Vistara	deletePost	setupScenary8	La publicación con título = ""	EmptyFieldsException No se elimina ningún post de la aplicación y todos los listados se mantienen igual con el mismo tamaño.
Vistara	deletePost	setupScenary7	La publicación con título = "Avalancha"	NullPointerException Se lanza una excepción indicando que no existe ningún post en la aplicación. El tamaño del listado de posts en la aplicación se mantiene como 0.

Objetivo de la Prueba: Validar que una publicación fue verificada exitosamente

Clase	Método	Escenario	Valores de Entrada	Resultado
-------	--------	-----------	--------------------	-----------

Vistara	verifyPost	setUpScenario8	VerificationStatus = VERIFIED	El estado de verificación de la publicación cambiará a VERIFIED.
Vistara	verifyPost	setUpScenario8	verificationStatus = FAKE	El estado de verificación de la publicación cambiará a FAKE_NEW.

A continuación se presenta el diagrama de clases del programa. En caso de que la resolución no sea visible, en el proyecto encontrará una imagen de este y un formato pdf individual.

