



**UNIVERSIDAD DE SANTIAGO DE CHILE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA INFORMÁTICA**

Paradigmas de Programación

Laboratorio N°4

Alumno: Cristhofer Parada Salazar

Sección: B-2.

Profesor: Víctor Flores Sánchez.

Santiago – Chile
1-2021

TABLA DE CONTENIDOS

Contents

Alumno: Cristhofer Parada SalazarSección: B-2	2
1.1.1 Descripción del problema.....	4
1.1.2 Descripción del paradigma utilizado	4
2.1.1 Análisis del problema respecto a sus requisitos específicos.....	5
2.1.2 Diseño de la solución.	6
2.1.3 Aspectos de implementación.....	7
2.1.4 Instrucciones de uso.	8
2.2 Resultados y autoevaluación.	9
Requerimientos Funcionales:	10
2.3 Conclusión.	11
CAPÍTULO 3. ANEXO.....	12

CAPÍTULO 1. INTRODUCCIÓN

El siguiente informe cuenta con el objetivo de mostrar al lector el desarrollo de la solución del cuarto laboratorio de la asignatura Paradigmas de Programación. Se mostrará una breve descripción del problema a resolver, el paradigma utilizado para implementar la solución y las herramientas que se utilizaron. Además, se comparará la implementación con este paradigma respecto a los anteriores (funcional, lógico y orientado a objetos) que se utilizaron en entregas pasadas. También se mostrarán pequeñas maneras de uso de la solución.

1.1.1 Descripción del problema

En el semestre 1/2021, para la entrega N°4 del laboratorio de la asignatura Paradigmas de Programación, a los estudiantes se les solicitó la implementación de una Red Social, la cual tenía como funciones principales acciones como registrarse en la red social, crear publicaciones, comentar publicaciones, seguir a otros usuarios, etc. Para ello, se les solicitó a los alumnos que el programa debiese estar implementado usando el paradigma de orientado a objetos y, además, el paradigma dirigido por eventos. Para ello se dio la opción a los estudiantes de usar el lenguaje de programación Java o C# y con ello dicha solución en esta ocasión debe contar con interfaz gráfica. Para la implementación de esta solución se decidió por llevarse a cabo utilizando C# con WPF (Windows Presentation Foundation) como la biblioteca de entorno gráfico.

1.1.2 Descripción del paradigma utilizado

En esta ocasión, se solicitó que se utilice el paradigma POO (programación Orientada a Objetos) en conjunto con el paradigma dirigido por eventos. La característica principal del paradigma orientado a objetos es llevar una idea o estructura de la vida real, asignarle atributos y capacitarlo con métodos o acciones propias, buscando crear como el nombre del paradigma lo menciona, un objeto.

Por otro lado, el paradigma dirigido por eventos tiene como característica principal eventos que dirigen la estructura y flujo del programa, es decir, se le da la libertad al usuario de controlar el flujo del programa a través de los eventos que el provoca, en comparación con la programación imperativa que es el programador que mantiene un flujo constante del programa el cual el usuario no tiene forma de manipular.

CAPÍTULO 2. SOLUCIÓN DEL PROBLEMA

2.1.1 Análisis del problema respecto a sus requisitos específicos.

Para el desarrollo de la solución se solicitan los siguientes requisitos funcionales específicos:

- Implementar abstracciones apropiadas para el problema mediante el uso de clases y estructuras que modelen de forma correcta el problema entregado.
- Implementación de una interfaz gráfica que permita al usuario interactuar con las distintas funcionalidades del programa
- Función authentication: Función que permite que registrar usuarios en una red social, iniciar sesión en una red social o desconectarse de una red social.
- Función post: Permite a un usuario con sesión iniciada realizar una nueva publicación, la cual puede ser propia o con usuarios etiquetados
- Función follow: Permite al usuario con sesión iniciada poder seguir a otro usuario.
- Función share: Funcionalidad que permite al usuario con sesión iniciada compartir el contenido de una publicación.

Además de las funciones antes mencionadas, se a la libertad de escoger otras funciones que se pueden implementar en el programa, específicamente en este caso, se implementaron las funciones de comment y like:

- Función comment: permite a un usuario comentar una publicación siempre y cuando esta última exista.
- Función react: permite a un usuario con sesión iniciada agregar una reacción a una publicación, siempre y cuando esta última exista.

2.1.2 Diseño de la solución.

La solución fue llevada a cabo utilizando como IDE la aplicación Visual Studio Community en el lenguaje C#, utilizando como biblioteca gráfica la llamada WPF, la cual utiliza el lenguaje declarativo XAML y la lógica de la vista está dada por C#. Gracias a la IDE utilizada, no fue necesario enfocarse en aprender XAML dado que la IDE entregaba herramientas que facilitaban su uso, como la creación de botones, cambios de background, tamaño de la ventana, etc.

Para el diseño de la solución, fue tomado como base el análisis y diagrama del laboratorio anterior, el diseño actual para esta entrega puede ser encontrado en la sección de Anexo. El programa sigue el patrón de arquitectura MVC (model, view, controller) que permite mantener un orden y dividir el programa en:

- Modelo: encargado de definir todas las clases u objetos del proyecto.
- Controlador: quien utiliza los elementos del modelo para generar acciones entre ellos.
- Vista: cuenta con la visualización grafica visible al usuario y se encuentra comunicada con el controlador.

Dada la arquitectura utilizada para la implementación de la solución, el siguiente paso fue crear los distintos modelos correspondientes a una red social, que en este caso corresponden a las clases de Usuario, Publicacion, React y la Socialnetwork como tal. Luego se continuó con la clase del controlador, la cual cuenta con todos los requerimientos funcionales descritos en el punto 2.1.1, esto se llevó a cabo sin el uso de una vista puesto que al ya estar bien implementado en el lenguaje Java de la entrega pasada, había una alta certeza que los requerimientos estaban correctos. Finalmente, gran parte del trabajo se concentró en crear sección de vistas que cuenta con una gran cantidad de ventanas que utilizan los métodos del controlador. La combinación de estas acciones da como resultado la aplicación como tal con una interfaz grafica en donde el usuario controla los eventos que puede realizar la solución.

2.1.3 Aspectos de implementación.

Este programa se realizó en el lenguaje C#, utilizando como IDE el programa Visual Studio Community 2019 en su versión 16.11.1, utilizando como biblioteca gráfica la biblioteca WPF y el .NET Framework objetivo fue la versión 4.7.2. Se decidió tomar este lenguaje dado al interés de aprender otro lenguaje adicional a Java y la facilidad que esto llevaba dado que hay bastante similitudes entre ambos. Esta solución solo puede ser ejecutada en ordenadores que cuenten con Windows como su sistema operativo.

2.1.4 Instrucciones de uso.

En comparación con las entregas anteriores, su uso es bastante intuitivo dado que toda la información es dada al usuario de una manera comprensible visualmente, asemejándolo a programas de uso cotidiano.

Para ejecutar el programa es necesario abrir el ejecutable Lab4.exe, el cual abrirá la ventana inicial de la aplicación que mostrará la opción de poder registrarse dentro de la red social y la opción de iniciar sesión, se recomienda que el usuario se cree una cuenta y luego ingrese con la cuenta recién creada, o también ingresar con el usuario “Harunomi” con contraseña “password1” el cual cuenta ya con algunas publicaciones y follows previamente cargados dentro de la red social.

2.2 Resultados y autoevaluación.

En esta sección se hará una revisión de los resultados, para ello se verificarán los requisitos no funcionales y funcionales solicitados

Requerimientos no funcionales:

Lenguaje y Herramientas	Logrado: el lenguaje usado es C# en su versión 8.0 o superior y el uso de WPF como biblioteca de entorno gráfico.
Interacciones del programa	Logrado: todas las interacciones del programa se realizan mediante el uso de una interfaz gráfica.
Interacciones con el programa: Todas las interacciones con el programa deben ser mediante consola/terminal.	Logrado: El usuario solamente puede interactuar por la red social mediante el uso del menú por consola.
Uso del paradigma	Logrado: Se implemento de manera correcta el paradigma orientado a objetos y el paradigma dirigido por eventos.
Organización del código	Logrado: El programa se encuentra bien organizado siguiendo el patrón MVC.
Diagrama de análisis	Logrado: Se crea un diagrama de análisis previo a la implementación de la solución.
Diagrama de diseño	Logrado: Se presenta un diagrama final con la implementación después de creada la implementación de la solución.
Historial: al menos 10 commits en un mínimo de 2 semanas, con un mínimo de 10 commits.	No logrado: Si bien es cierto se logró el mínimo de commits, a los cuales en realidad se deben restar 5 por eliminar carpetas subidas innecesariamente, estos no logran el rango de ser en una semana. Primer commit: 21/08/2021. ultimo commit: 23/08/2021
Prerrequisitos de funciones:	Logrado, cada función tiene su prerrequisito implementado como se solicitó.

Requerimientos Funcionales:

Todas las funciones cumplen con los requerimientos funcionales exigidos por la coordinación, por lo que este punto también fue respetado en su gran totalidad.

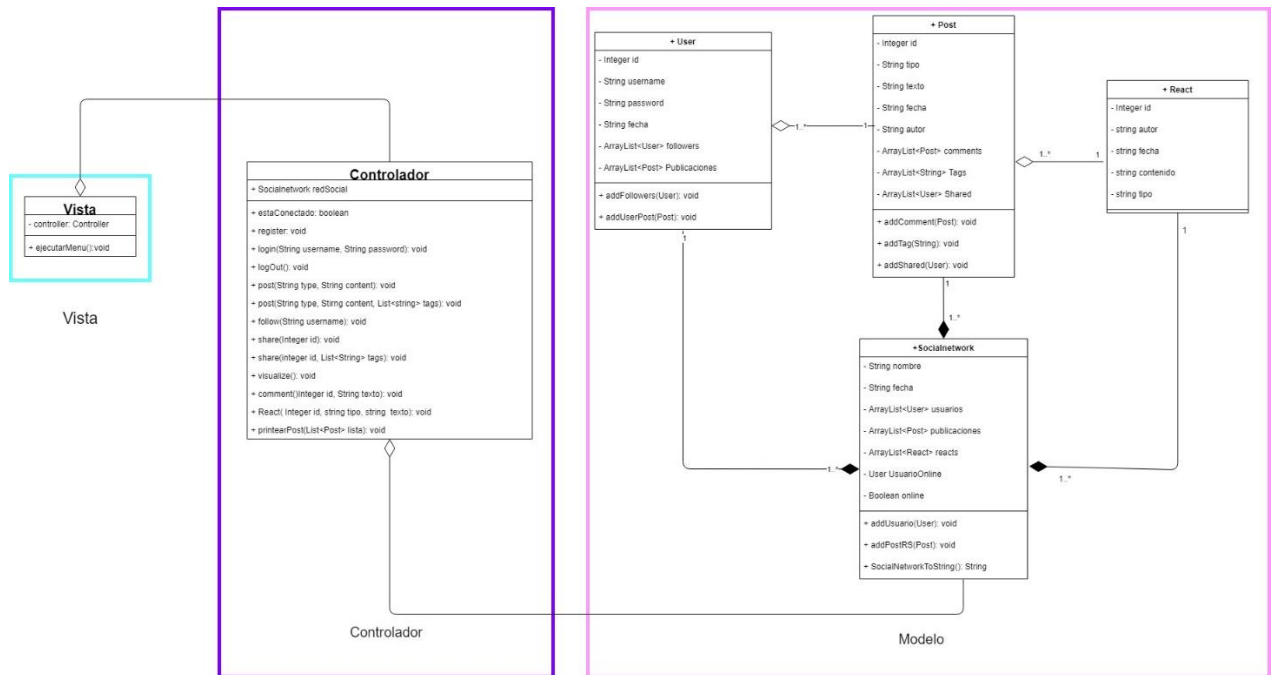
2.3 Conclusión.

Para finalizar, se puede concluir a partir de los resultados anteriores vistos en la sección 2.2 que se pudo realizar una correcta implementación de la solución al problema utilizando un nuevo lenguaje y creando finalmente, al menos para el desarrollador, su primera interfaz gráfica para un programa.

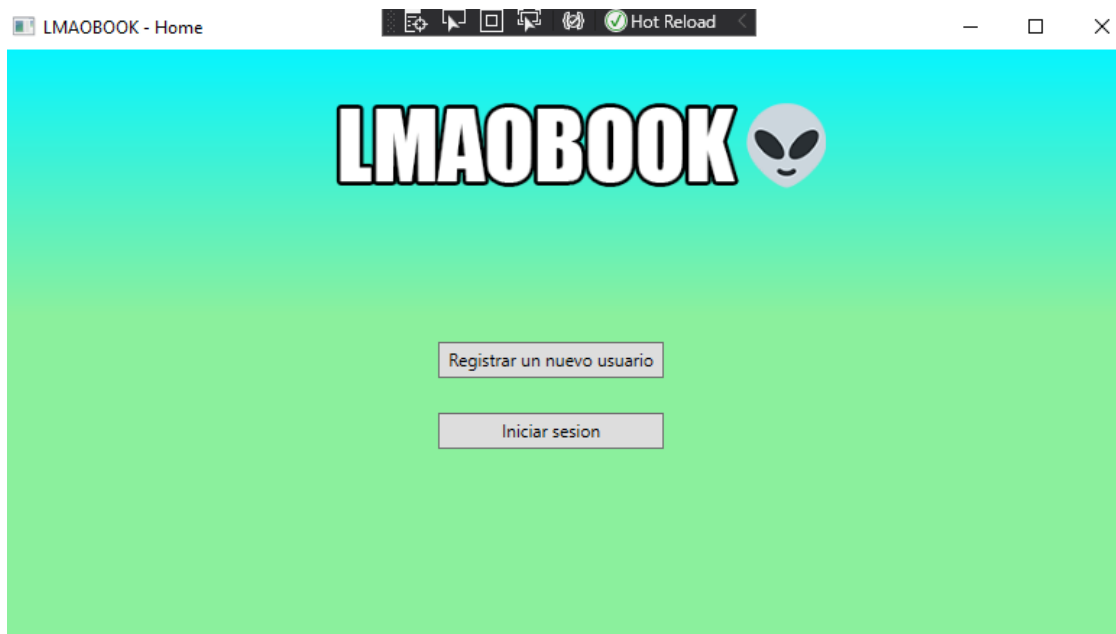
En este caso, no se logró cumplir con el rango de commits, por temas de estar agobiado mentalmente, pero se está bastante satisfecho con el trabajo logrado, hubo un buen aprendizaje del paradigma y se pudo crear una implementación al problema acorde a lo que se requería. Aspectos para mejorar, sigue siendo intentar mantener una buena organización personal, para poder mantener la misma cantidad de commits, pero en el rango que se pide de 2 semanas.

Personalmente me hubiese gustado hacer una interfaz gráfica bastante más bonita e ir jugando con las distintas herramientas que entrega el IDE y WPF como tal, las cuales son bastantes y muy entretenidas. Aún así, para el tiempo empleado en realizar la solución, estoy bastante satisfecho.

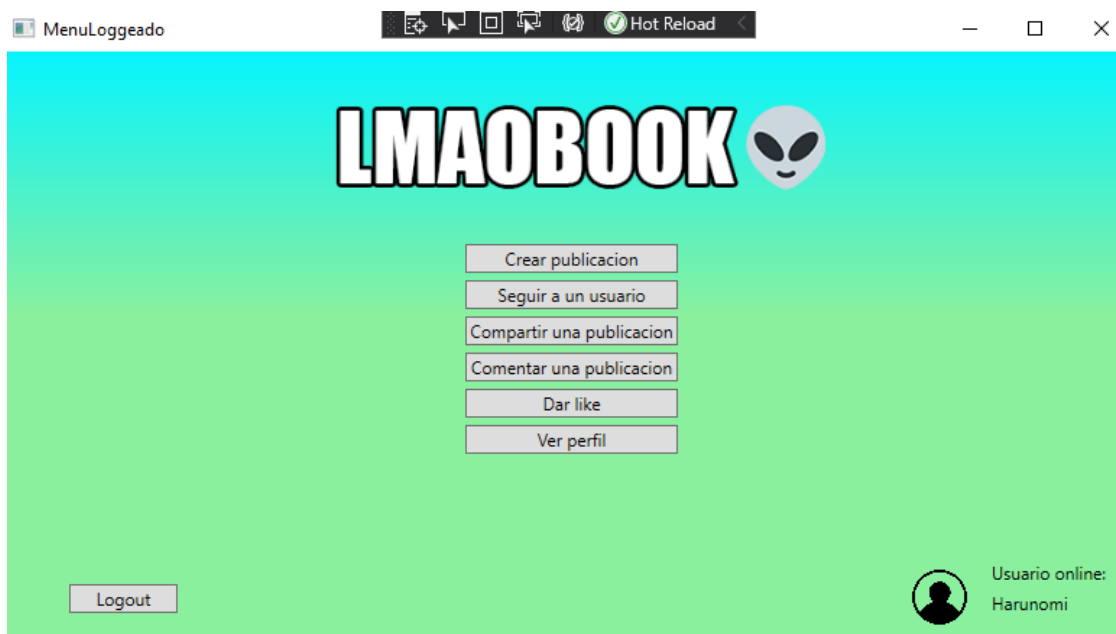
CAPÍTULO 3. ANEXO



Anexo 1: Diagrama de análisis



Anexo 3: Ventana al abrir el programa



Anexo 4: Menú de un usuario con session iniciada