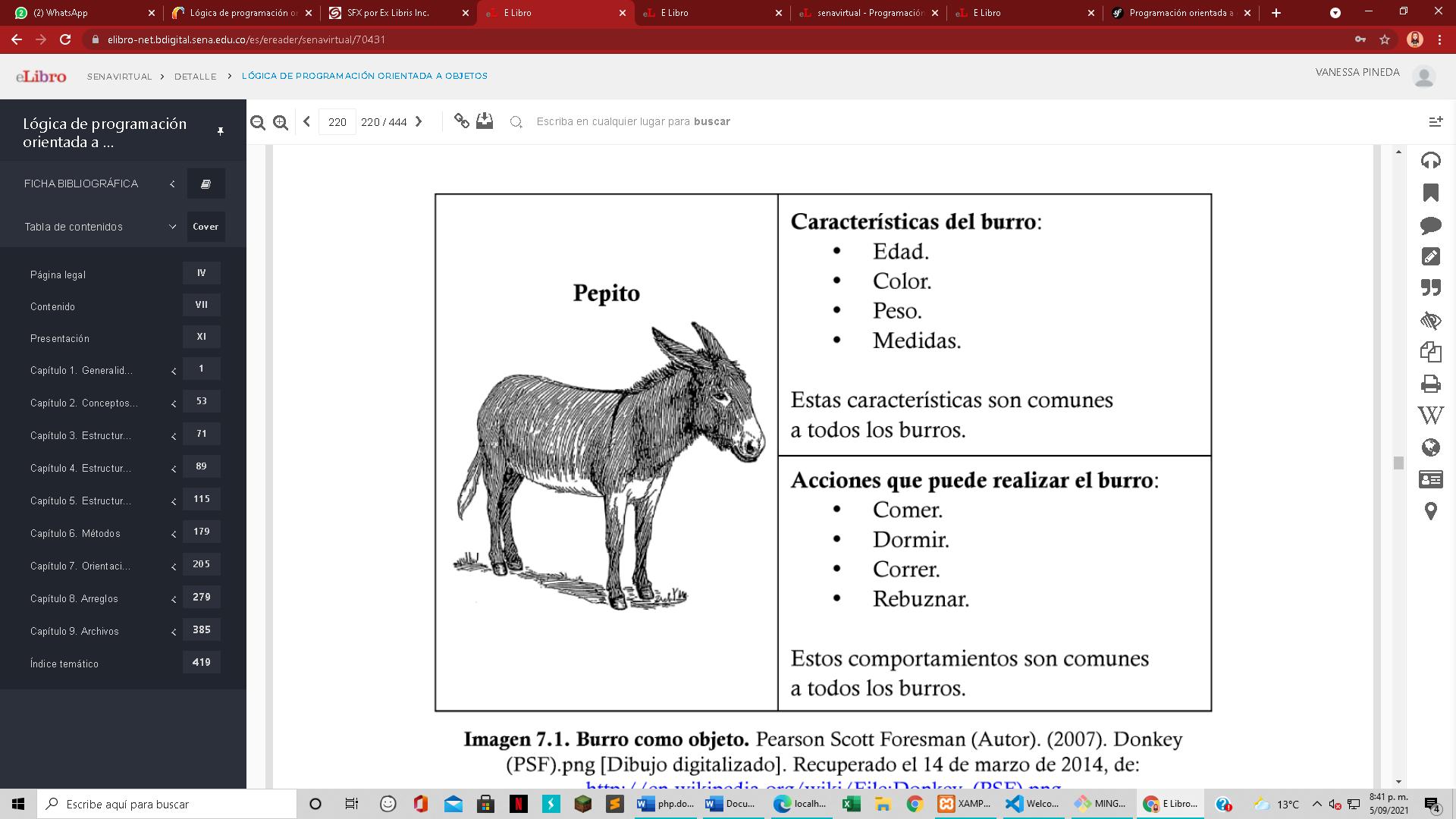
**PROGRAMACION ORIENTADA A OBJETOS**

Con la programación orientada a objetos (Poo) se aumenta el modularidad de los programas y la reutilización de los mismos. además, la Poo se diferencia de la programación clásica porque utiliza técnicas nuevas como el polimorfismo, el encapsulamiento, la herencia, etc.

La programación orientada a objetos rompe definitivamente con el paradigma de la programación estructurada, surge el concepto de “objeto”. todos los programas se reducen a objetos que tienen una serie de atributos, y, asociados, un comportamiento o procedimientos llamados “métodos”. estos objetos, que son instancias de clases, interaccionarán unos con otros y de esa manera se diseñarán aplicaciones y programas.

Programar bajo el paradigma de programación orientada a objetos (POO) consiste en simular o modelar los objetos del mundo real. En un establo, por ejemplo, se pueden identificar diferentes animales como entes con características y acciones propias. Como ejemplo introductorio se puede tomar un burro.

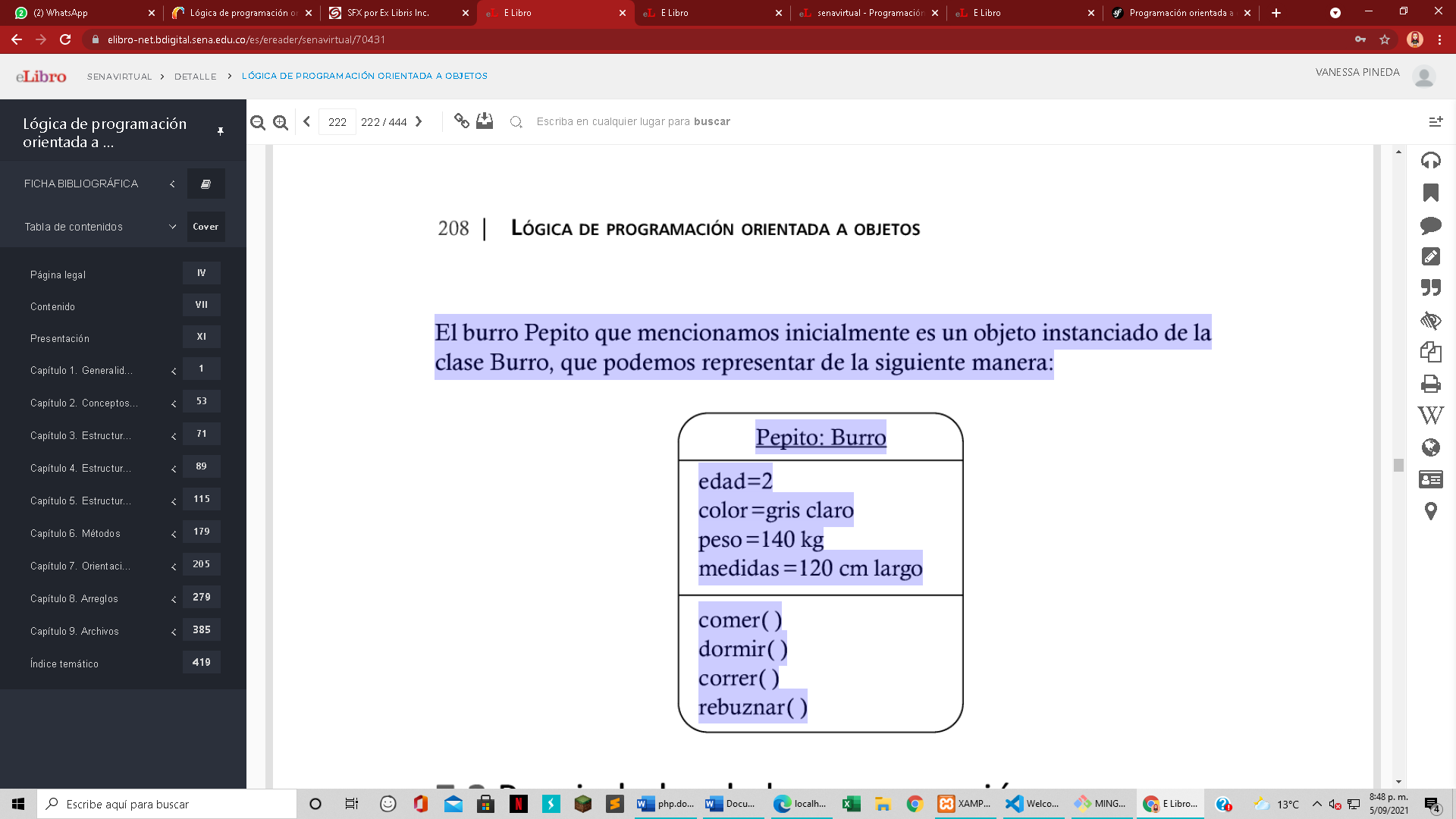
Imagen 7.1. Burro como objeto. Pearson Scott Foresman (Autor). (2007). Donkey (PSF).png [Dibujo digitalizado]. Recuperado el 14 de marzo de 2014, de: http://en.wikipedia.org/wiki/File:Donkey\_(PSF).png



podemos deducir la definición de clase y objeto. Una clase es una plantilla para crear objetos que constituye una abstracción del mundo real, como la clase Burro, la clase persona, la clase cuenta, etcétera. Las clases se nombran como sustantivos en singular y poseen variables que definen la información que se desea almacenar y métodos que definen las acciones que se desean realizar. Las variables se nombran como sustantivos y los métodos como verbos en infinitivo.

un objeto es una instanciación de una clase, es decir, la materialización de la clase. Cuando se instancia un objeto se asignan datos a las variables de la clase y se pueden ejecutar los métodos.

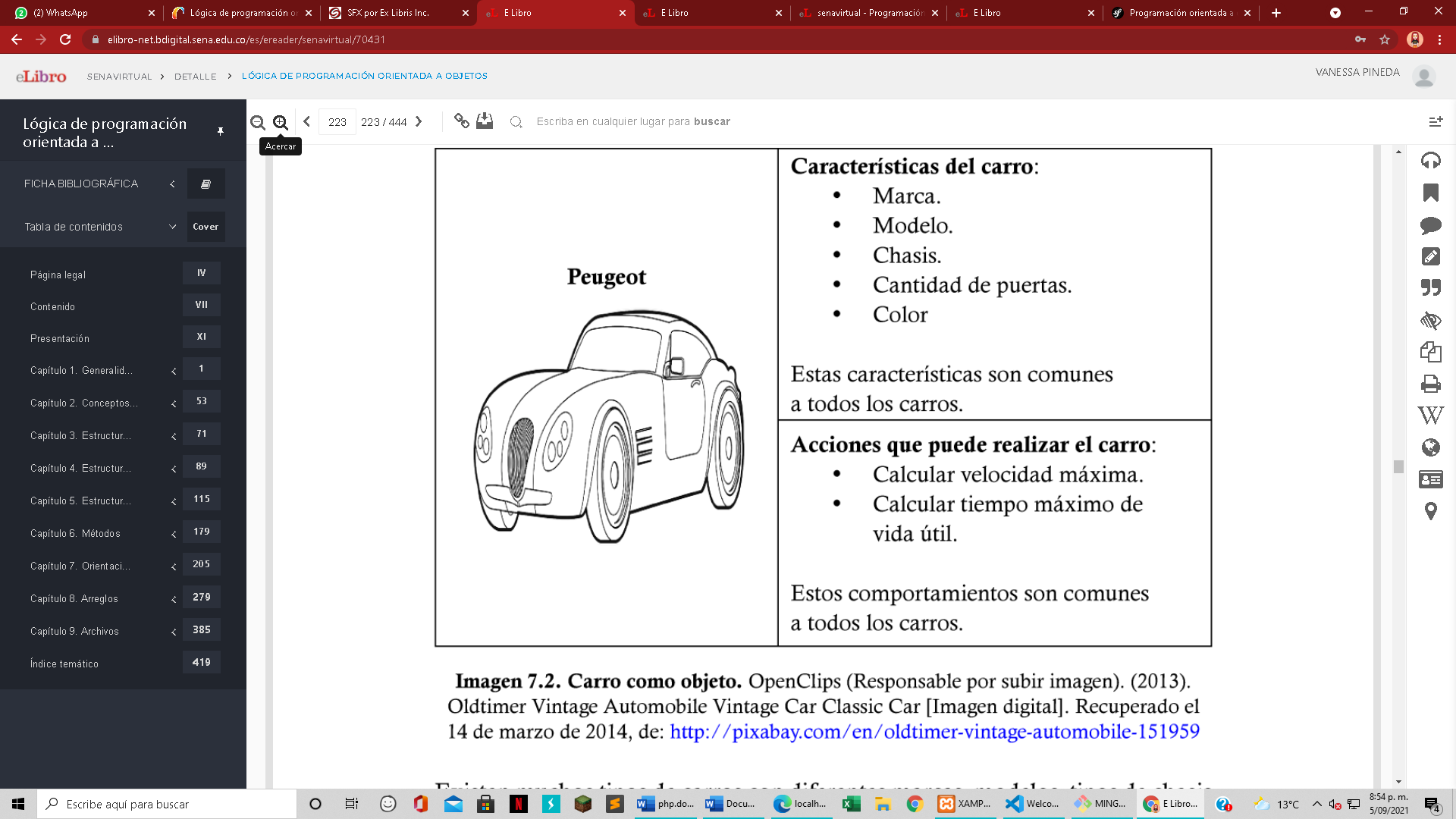
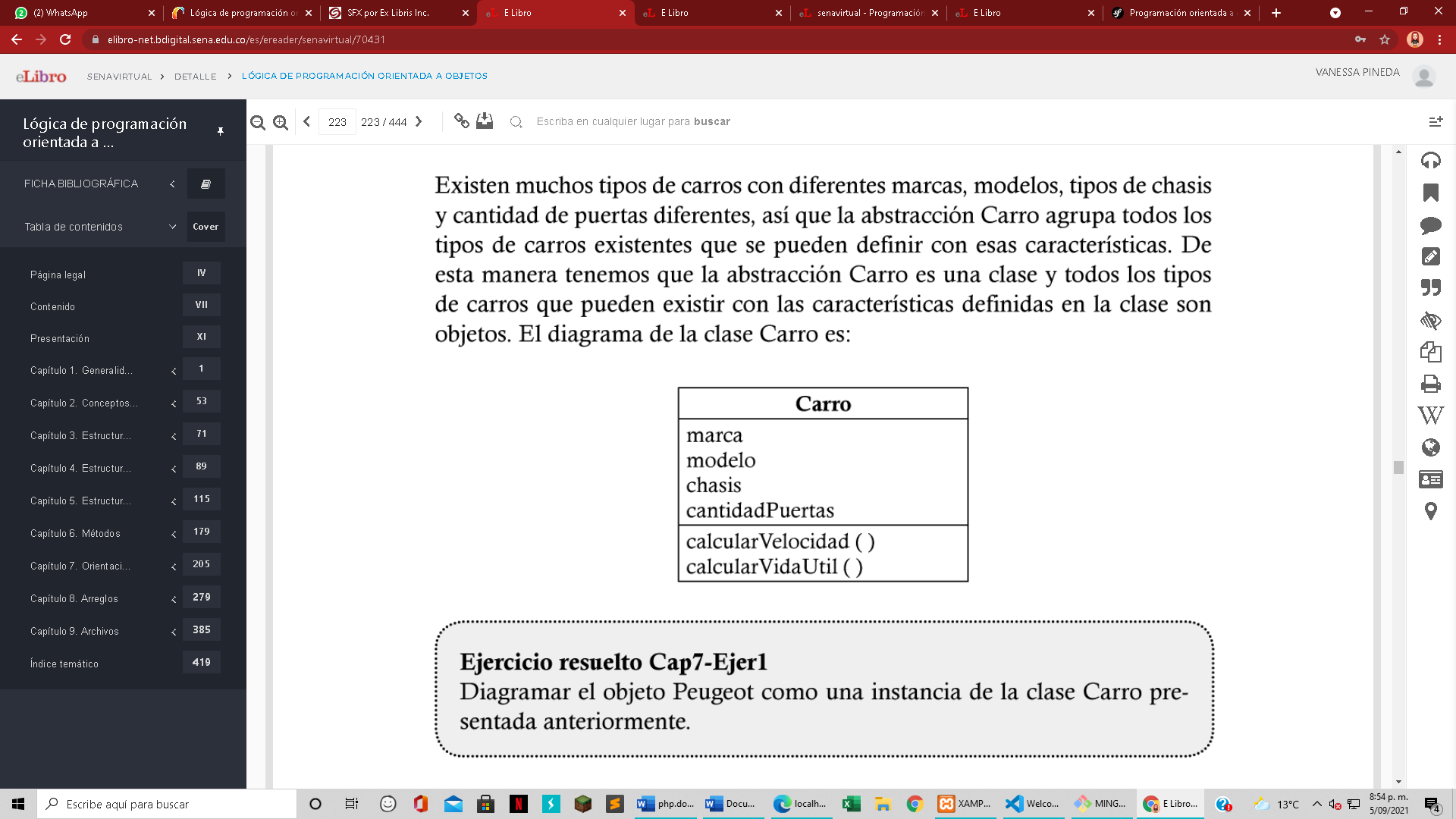
El burro Pepito que mencionamos inicialmente es un objeto instanciado de la clase Burro, que podemos representar de la siguiente manera:



Propiedades de la programación orientada a objetos

La programación objetual define un conjunto de propiedades básicas que los lenguajes de programación orientados a objetos deben cumplir:

• **Abstracción:** Es la acción de identificar las cualidades y acciones que un objeto puede realizar



• **Encapsulamiento:** Es la propiedad que tienen las clases de agrupar las características y las acciones relacionadas con una abstracción bajo una misma unidad de programación.

• **Ocultamiento de información:** Protege los objetos restringiendo y controlando el acceso en la clase que los define.

**Especificadores o modificadores de acceso**

condicionan el acceso de las variables, de los métodos de una clase, definiendo su nivel de ocultamiento.

(+) Público: El elemento puede ser accedido desde cualquier clase. Si es un dato miembro, cualquier clase puede acceder al elemento. Si es un método, cualquier clase puede invocarlo.

(-) Privado: Solo se puede acceder al elemento desde métodos miembros de la clase donde se encuentra definido, o solo puede invocarse desde otro método de la misma clase.

(#) Protegido: Proporciona acceso público para las clases derivadas (véase sección de herencia) y acceso privado para el resto de clases.

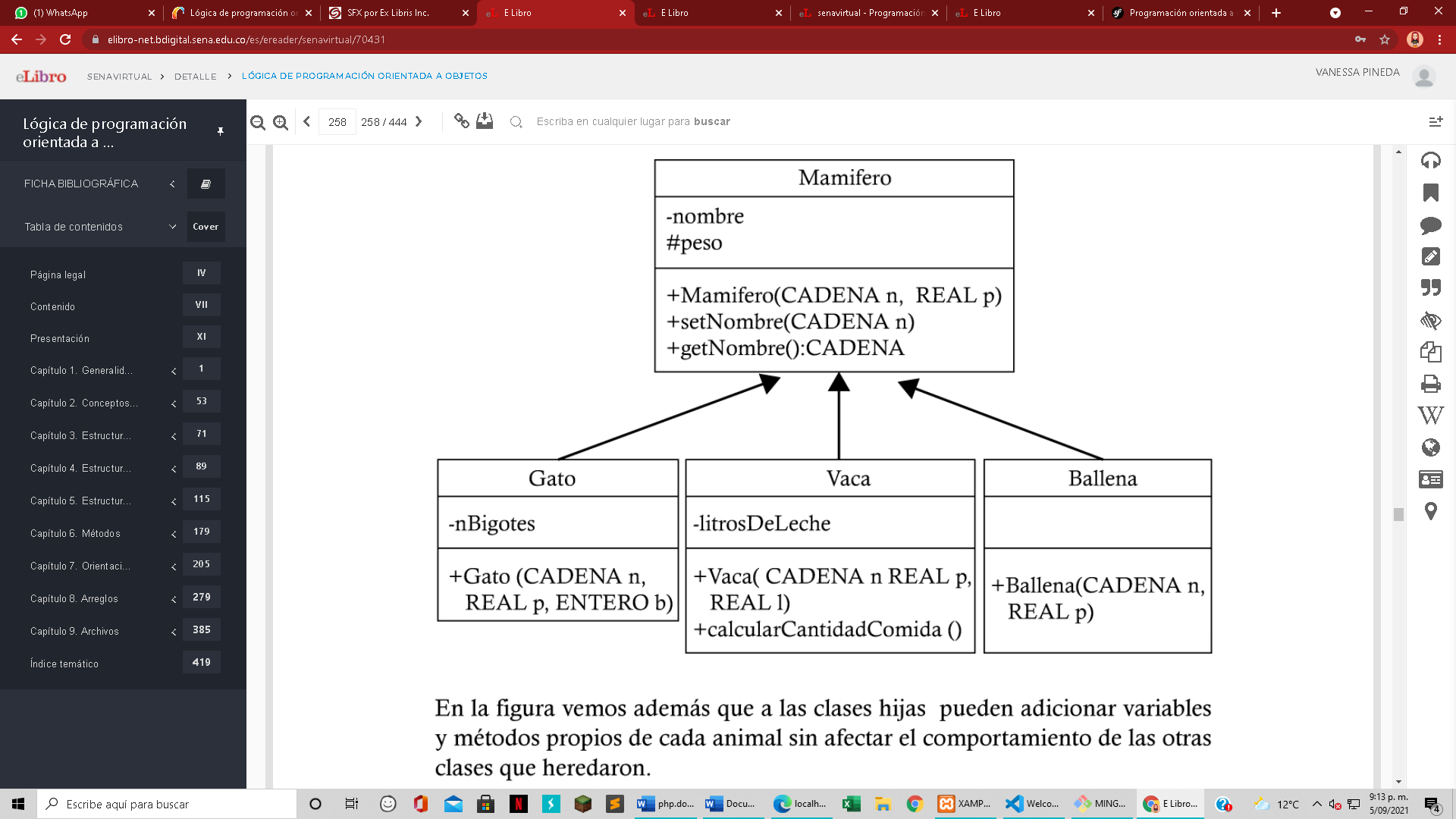
( ) Sin modificador: Se puede acceder al elemento desde cualquier clase que esté en la misma ubicación (carpeta) donde se define la clase.

• **Sobrecarga.** Permite a una misma clase tener varios métodos con el mismo nombre, inclusive con el mismo valor de retorno, pero con diferentes pará- metros.

• **Polimorfismo.** Permite a una misma clase tener varios métodos con el mismo nombre, inclusive con el mismo valor de retorno, pero con diferentes pará- metros.

• **Herencia y Reutilización.**

Al igual que los hijos heredamos características de nuestros padres, en la POO podemos obtener clases hijas con el mismo contenido de su clase padre. De esta manera se obtiene una reutilización de las clases, la cual es una de las propiedades más deseadas en la programación ya que con lleva un ahorro de tiempo y de líneas de código. La reutilización ha sido abordada en la programación estructural por medio de funciones.



**Referencia bibliografica**

Oviedo Regino, E. (2015). Lógica de programación orientada a objetos. Ecoe Ediciones. https://elibro-net.bdigital.sena.edu.co/es/ereader/senavirtual/70431?page=6