



Testing y calidad de software

Documentación Taller 1

Universidad Andrés Bello

Ingeniería en Computación e Informática

6 de noviembre de 2021

Profesora:

Sarita Gonzalez Catalán

Alumnos:

Alonso Ruiz

Nicolás Muñoz

Claudio Pérez

Tabla de contenidos

Análisis del problema	3
Grupos de datos	3
Datos de login	3
Datos personales	3
Datos de IMC	4
Historias de usuario	5
Historia de usuario 1 (login)	5
Historia	5
Escenarios	5
Historia de usuario 2 (registro)	6
Historia	6
Escenarios	6
Historia de usuario 3	7
Historia	7
Escenarios	7
Diseño	8
Dominio	8
Navegación	9
Diseño de las pantallas	10
Requerimientos e instalación	11

Análisis del problema

Grupos de datos

A partir de los requerimientos del enunciado se identificaron los siguientes grupos de datos:

Datos de login

Correo	<ul style="list-style-type: none">• Obligatorio• Cadena.• No repetido.• Formato válido (<usuario>@<dominio>)
Contraseña	<ul style="list-style-type: none">• Obligatorio• Cadena.• Entre 8 y 20 caracteres• Debe ser hasheada

Datos personales

Nombres	<ul style="list-style-type: none">• Obligatorio• Cadena.• Máximo 100 caracteres
Apellidos	<ul style="list-style-type: none">• Obligatorio• Cadena• Máximo 100 caracteres
Edad	<ul style="list-style-type: none">• Obligatorio.• Número entero.• Entre 15 y 70 (inclusives)
Sexo	<ul style="list-style-type: none">• Obligatorio• Cadena.• Solo "F" para femenino y "M" para masculino (insensible a mayúsculas y minúsculas)

Datos de IMC

Peso	<ul style="list-style-type: none">• Obligatorio• Número decimal.• Mayor a 0
Altura	<ul style="list-style-type: none">• Obligatorio• Número decimal.• Mayor a 0
Fecha en que se peso	<ul style="list-style-type: none">• Obligatorio• Formato dd/mm/yyyy (días y meses como número)
Hora en que se peso	<ul style="list-style-type: none">• Obligatorio• Formato hh:mm (hora en formato de 24 horas)

Historias de usuario

Historia de usuario 1 (login)

Historia

Como	Usuario
Quiero	tener una cuenta privada
Para	tener un ingreso seguro a mi información en la aplicación

Escenarios

Escenario 1	Usuario quiere ingresar a una cuenta con datos correctos de log in.
Dado que	se encuentra en la pantalla de ingreso (log in)
Cuando	el usuario ingresa un correo registrado
Y	la contraseña asociada a ese correo es correcta
Entonces	el sistema lo redirige a la pantalla del menú logueado

Escenario 2	Usuario quiere ingresar a una cuenta existente con contraseña errónea.
Dado que	Se encuentra en la pantalla de ingreso (log in)
Cuando	Se ingresa un correo registrado
Pero	se ingresa una contraseña que no corresponde al correo indicado
Entonces	El sistema muestra un mensaje indicando que los datos de ingreso son inválidos.
Y	Luego redirige al menú deslogueado

Escenario 3	Usuario quiere ingresar con una cuenta que no existe
Dado que	Se encuentra en la pantalla de ingreso (log in)
Cuando	NO existe un correo registrado
Y	sin importar la contraseña que se ingrese
Entonces	El sistema muestra un mensaje indicando que los datos de ingreso son inválidos.
Y	Luego redirige al menú deslogueado

Historia de usuario 2 (registro)

Historia

Como	Usuario
Quiero	poder registrar mis datos personales
Para	que el sistema me de una atención personalizada

Escenarios

Escenario 1	Usuario quiere crear una cuenta
Dado que	El usuario ingresó a la pantalla de registro
Cuando	Ingresa sus datos correctamente
Entonces	se crea una cuenta de usuario
Y	me muestra un mensaje de que la cuenta fue creada exitosamente.
Y	me redirige a la pantalla del menú deslogueado.

Escenario 2	Usuario quiere crear una cuenta con un campo inválido
Dado que	El usuario ingresó a la pantalla de registro
Cuando	Ingresa uno de los campos de login o registro (ver grupos de datos)
Entonces	se muestra un mensaje indicando que el campo es invalido (ver las reglas para los campos de login y registro en los grupos de datos)
Y	se pide que ingrese ese campo nuevamente hasta que lo ingresa correctamente.

Historia de usuario 3

Historia

Como	Usuario
Quiero	saber mi indice de masa corporal
Para	saber mi estado de salud.

Escenarios

Escenario 1	Usuario quiere calcular el IMC correctamente
Dado	Ya ingresó a su cuenta
Y	El usuario ingresó a la pantalla cálculo IMC
Cuando	Ingresa los campos requeridos correctamente
Entonces	se muestra un mensaje indicando los datos personales del usuario
Y	además se muestran los resultados del IMC (peso, altura, valor del IMC y su correspondiente evaluación)

Escenario 2	Usuario ingresa de manera incorrecta algún dato de IMC.
Dado	Ya ingresó a su cuenta
Y	El usuario ingresó a la pantalla cálculo IMC
Cuando	Ingresa uno de los campos incorrectamente
Entonces	se muestra un mensaje indicando que el campo es invalido (ver las reglas para los campos de IMC en los grupos de datos)
Y	se pide que ingrese ese campo nuevamente hasta que lo ingresa correctamente.

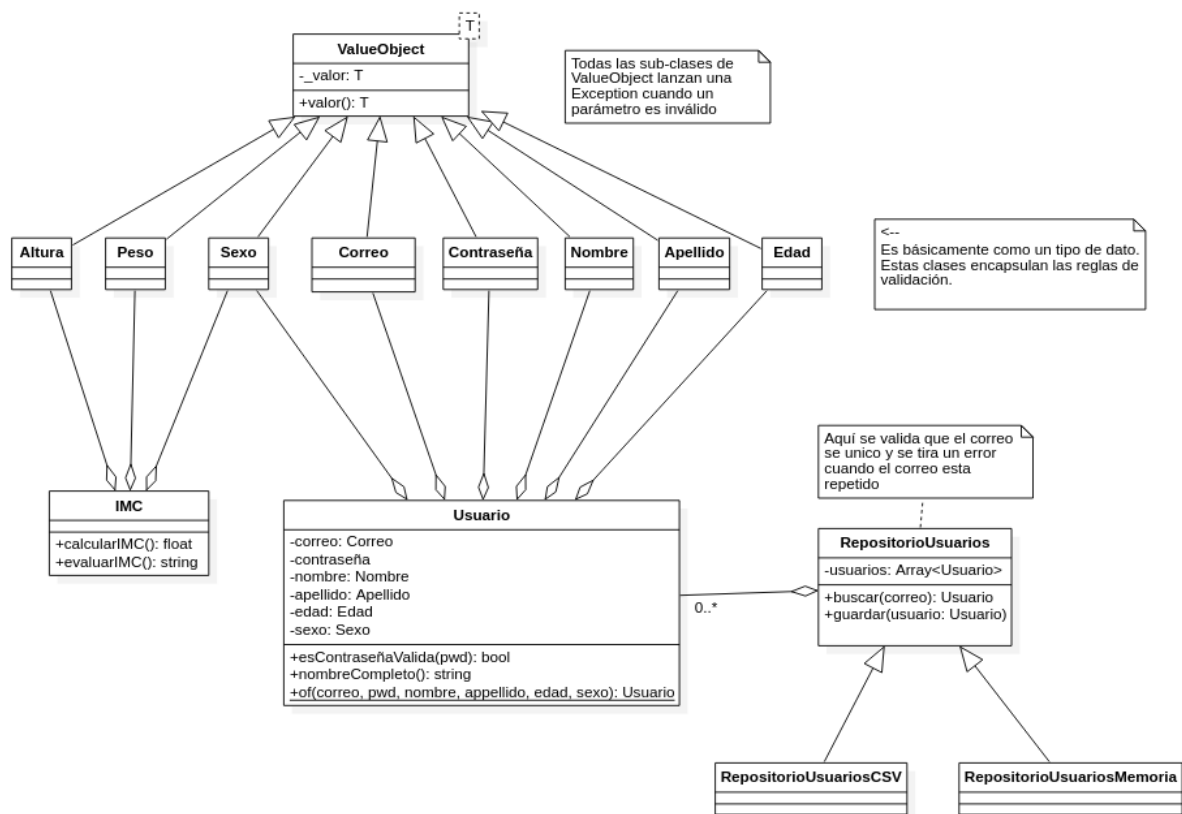
Diseño

Para el desarrollo del software se decidió dividir el problema en 2 partes:

- **Dominio:** contiene toda la lógica con respecto a los usuarios, sus campos y el cálculo del IMC.
- **Pantallas:** contiene toda la lógica para obtener la información del usuario y el despliegue correcto de los resultados.

Dominio

El dominio del problema se puede ver en el siguiente diagrama de clases:



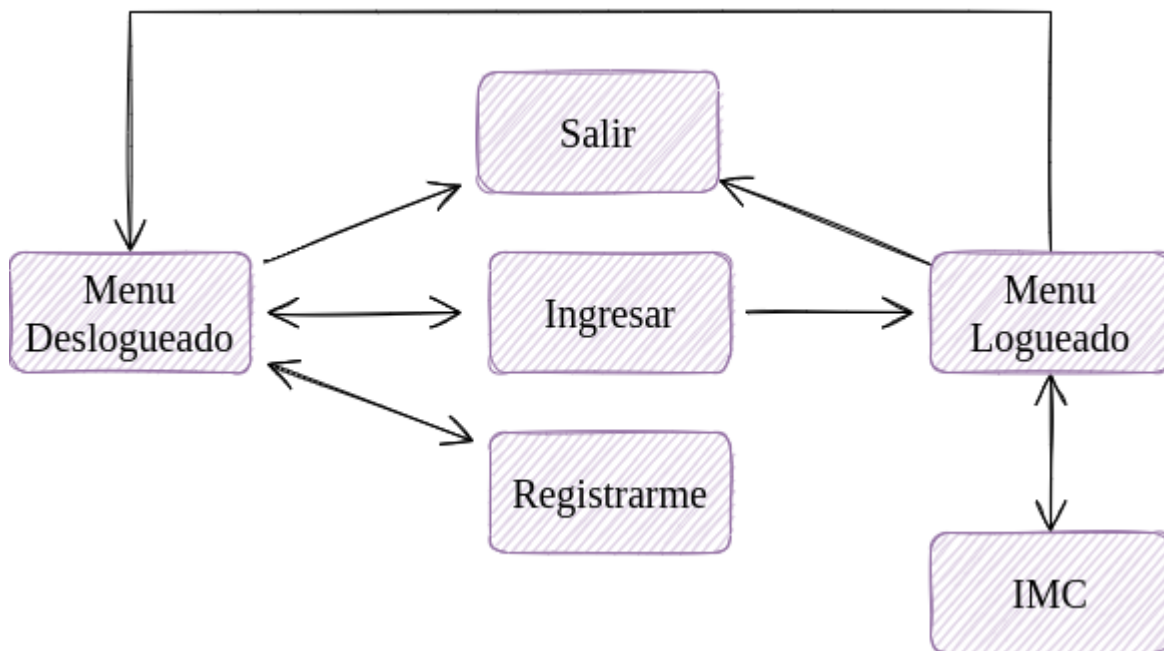
En este diagrama vemos que todas las que heredan de **ValueObject** representan un campo de información importante para el problema y cuyo objetivo principal es encapsular las reglas de validación de dicho campo.

Notar que la validación que guarda relación con que el correo del usuario sea único se lleva a cabo en el **RepositorioUsuarios** y no en la clase **Correo**. Esto se debe a que es esta clase la que guarda un conjunto de usuarios y es la que puede determinar si un correo está repetido o no.

Navegación

Para recopilar la información del usuario se modeló la interfaz del programa como una serie de pantallas entre las que se puede navegar entre sí y donde cada una se encarga de hacer una sola cosa en específico.

La siguiente figura muestra las pantallas presentes en el programa y cuales son las pantallas a las que uno se puede mover.



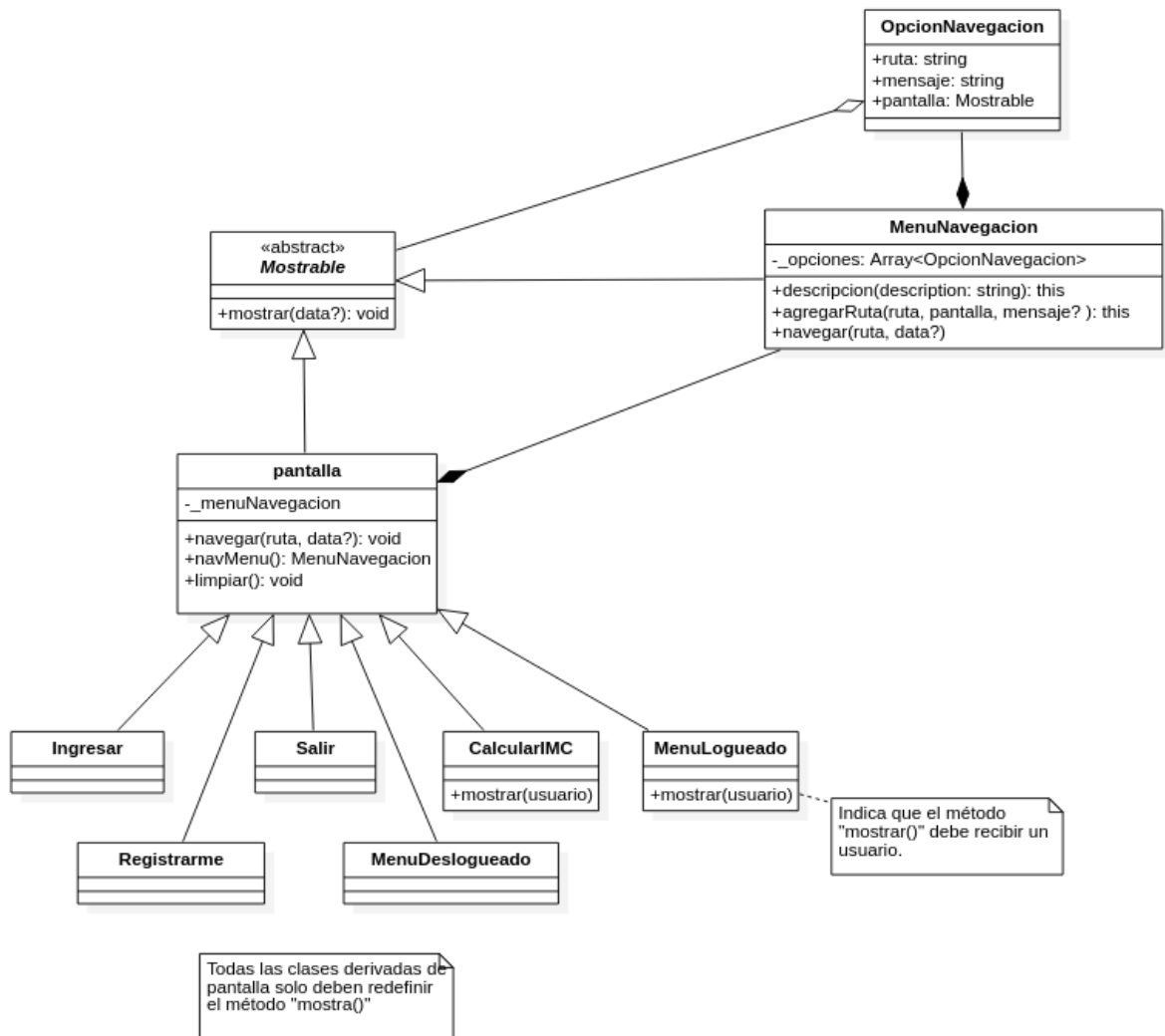
Una flecha simple indica que me puedo mover desde la pantalla A a la pantalla B, pero no al revés. Por ejemplo, desde la pantalla **Ingresar** podemos navegar a la pantalla **Menú Logueado**, pero no podremos navegar *desde* **Menú Logueado** a la pantalla **Ingresar**.

De la misma forma una flecha doble indica que la navegación es bidireccional, es decir, se puede navegar desde la pantalla A a la B y viceversa. En el diagrama podemos ver que es posible navegar desde la pantalla **Menú Deslogueado** a la pantalla **Registrarme** y de la misma forma podremos navegar desde **Registrarme** a la pantalla **Menu Logueado**.

La forma exacta de cómo y cuándo se produce dicha navegación queda a criterio de la implementación de esa pantalla.

Diseño de las pantallas

La siguiente figura muestra el diseño para poder realizar la navegación entre pantallas indicadas en la sección anterior.



En la figura vemos que cada pantalla hereda desde una clase base **Pantalla** que tiene ya implementado el cómo navegar de una pantalla a otra y solo necesita sobrescribir el método `mostrar()`, que es el que incluye toda la lógica de una pantalla particular.

Requerimientos e instalación

Tanto los requerimientos del programa como sus instrucciones de instalación se encuentran en el README del repositorio, el cual se puede acceder desde el siguiente link:

<https://github.com/alonsox/Testing-Taller1>

Si bien junto a este documento se adjunta el código fuente del programa se recomienda instalar siguiendo las instrucciones oficiales del repositorio.