**PROYECTO CENTRAL**

**ENTREGA 2 DISEÑA**

**Trabajo escrito presentado a la profesora**

**ANGELA MARÍA VALDERRAMA**

**en el curso Lógica y Representación I**

**Integrantes:**

CAMILO ANDRÉS PÉREZ PADILLA

ANDRÉS FELIPE SALAZAR RAMOS

****

**UNIVERSIDAD DE ANTIOQUIA**

**PROGRAMA DE INGENIERÍA DE SISTEMAS VIRTUAL**

**SEMESTRE FEBRERO-JUNIO 2023**

Tabla de contenido

[1. Diagrama de clases 3](#_Toc135231571)

[1.1 Clase Persona 3](#_Toc135231572)

[*1.1.1* Atributos de la clase *Persona* 3](#_Toc135231573)

[1.1.2 Métodos de la clase *Persona* 4](#_Toc135231574)

[*1.2* Clase *Recordatorio* 4](#_Toc135231575)

[1.2.1 Atributos de la clase Recordatorio 4](#_Toc135231576)

[1.2.2 Métodos de la clase Recordatorio 5](#_Toc135231577)

[1.3 Clase Registro 5](#_Toc135231578)

[1.3.1 Atributos de la clase Registro 5](#_Toc135231579)

[1.3.2 Métodos de la clase Registro 5](#_Toc135231580)

[2 Interfaces de la aplicación 6](#_Toc135231581)

[2.1 Home 6](#_Toc135231582)

[2.2 Registro y caracterización inicial 7](#_Toc135231583)

[2.3 Vista Recordatorios y validación de consumo de agua 7](#_Toc135231584)

[2.4 Vista informe 8](#_Toc135231585)

[3 Código fuente del Proyecto 8](#_Toc135231586)

[3.1 Clase Persona 8](#_Toc135231587)

[3.2 Clase Recordatorio 8](#_Toc135231588)

[3.3 Clase Registro 9](#_Toc135231589)

[4 Instancias de las clases creadas 9](#_Toc135231590)

[Bibliografía 10](#_Toc135231591)

Tabla de Ilustraciones

[Figura 1. Diagrama de clases de la aplicación 3](#_Toc135231592)

[Figura 2 Clase Persona 3](#_Toc135231593)

[Figura 3 Clase Recordatorio 4](#_Toc135231594)

[Figura 4 Clase Registro 5](#_Toc135231595)

[Figura 5 Conexión del flujo entre las vistas de la aplicación 6](#_Toc135231596)

[Figura 6 Vista de home de la aplicación 7](#_Toc135231597)

[Figura 7 Vista de notificación y validación del consumo de agua 8](#_Toc135231598)

[Figura 8 Vista informe de consumo de agua 8](#_Toc135231599)

[Figura 9 Clase Persona muestra del código fuente 9](#_Toc135231600)

[Figura 10 Clase Recordatorio muestra del código fuente 9](#_Toc135231601)

[Figura 11 Clase Registro muestra del código fuente 9](#_Toc135231602)

[Figura 12 Instancias de las clases diseñadas y programadas 10](#_Toc135231603)

# Diagrama de clases

En la siguiente figura se presenta el diagrama de clases de la aplicación. Debido a que la aplicación permitirá el registro de distintos usuarios se diseña una clase ***Persona*** que permitirá crear instancias para guardar como propiedades los siguientes elementos:

Diagrama

Descripción generada automáticamente

Figura 1. Diagrama de clases de la aplicación

## Clase Persona

La siguiente figura establece los atributos y métodos definidos para la clase Persona.

Diagrama

Descripción generada automáticamente

Figura Clase Persona

### Atributos de la clase *Persona*

* Id: Este id permitirá identificar el usuario que ingresa a la aplicación y conservará un tipo de dato entero
* Nombre: El nombre se utilizará como un label para comunicar la información al usuario correspondiente en distintos puntos de la aplicación. Este campo se define como un string.
* Edad: Una de las características primordiales es su edad. La aplicación está diseñada de tal manera que la sugerencia de ingesta de agua se hará para personas mayores de edad. Este atributo de la clase persona se tomará como un entero.
* Estatura: La estatura es uno de los parámetros que determinarán la cantidad necesaria de ingesta de agua. Se guardará este dato por cada instancia de la aplicación considerando un número float.
* Peso: El peso se considerará también como un atributo de tipo float para las instancias de la clase ***Persona***. En este caso es además uno de los factores importantes a la hora de calcular la cantidad de agua recomendada para que una persona ingiera.

### Métodos de la clase *Persona*

En la figura 1 se presenta la clase Persona con distintos métodos que permitirán conocer y calcular distinta información sobre el usuario que ingresa a la aplicación:

* calcularCantidadAgua() : Este método define la cantidad recomendada para el usuario según sus parámetros definidos en la vista de caracterización inicial
* consultarHistoria(): Este método permite al usuario consultar la historia de consumos y cumplimiento de notificaciones.
* actualizarHistoria(): Este método busca actualizar la historia de un usuario cuando se confirma una cantidad consumida.
* horaDespierto(): Cada usuario podrá definir la hora en que despierta, de esta manera esté método guarda el valor para tenerlo en cuenta sobre las notificaciones y cuando se pueden realizar
* horaDormir(): Cada usuario podrá definir en la aplicación la hora en la que se va a dormir. De este manera la aplicación impedirá nuevas notificaciones en horarios no permitidos por el usuario.

## Clase *Recordatorio*

Diagrama

Descripción generada automáticamente

Figura Clase Recordatorio

### Atributos de la clase Recordatorio

* Id: este parámetro se guarda con un valor entero para detallar los recordatorios y poder acceder a futuro a ellos.
* Intervalo: El intervalo del recordatorio es un parámetro que se define según se instancia un nuevo recordatorio. Este intervalo es sugerido para el usuario según la caracterización inicial.
* ultimoRecordatorio: Podemos conocer dinámicamente cuando se estableció el último recordatorio para el usuario. De esta manera se puede rastrear en cada paso el cumplimiento o no del anterior para poder notificar sobre la constancia y cumplimiento seguido de la cantidad de agua que se toma.

### Métodos de la clase Recordatorio

* recordar(): Esta clase por ahora se implementa un método para recordar al usuario y lanzar una notificación a los usuarios.
* calcularHoras(): Debido a que se conocerá del usuario la hora de despertar y la hora de dormir podemos establecer con este método la cantidad de minutos que hay entre esos momento de tal manera que se puedan establecer los recordatorios en ciertos espacios de tiempo.

## Clase Registro

Diagrama

Descripción generada automáticamente

Figura Clase Registro

### Atributos de la clase Registro

* Id: El registro o complimiento de una ingesta de agua se crea con un identificador de tipo entero.
* Cantidad: Una vez se confirma por el usuario que realizó un consumo de agua también genera un detalle de la cantidad de agua bebida.
* Fecha: Cada uno de los nuevos consumos es rastreado con una fecha para posteriormente consultar el consumo en distintas unidades de tiempo (semanas, días, meses).

### Métodos de la clase Registro

* Exportar(): El usuario podrá realizar una exportación de los registros o consumos realizados en distintas unidades de tiempo.
* Importar(): En etapa de desarrollo se necesitará un método para importar algunos registros base que permitan genera un dashboard de muestra para el usuario.

# Interfaces de la aplicación

Las interfaces de la aplicación comprenden 4 etapas principales. La primera de ellas considera que debe existir un usuario válidado, se debe ingresar con un usuario y contraseña a la aplicación. La segunda vista establece que el usuario puede ingresar o modificar información de su caracterización, esto es importante pues según los datos ingresados la aplicación internamente hará el calculo de la cantidad de agua que debe ingerir en el día. La tercera etapa se refleja en una vista donde se le notificará mediante mensaje y según un recordatorio establecido internamente que debe realizar consumo de agua. Desde esta misma vista el usuario podrá realizar la validación de la cantidad consumida y la aplicación guardará un nuevo registro con fecha hora y cantidad consumida. Con base en los datos recopilados durante cierto tiempo en la vista anterior, la aplicación podrá realizar un tablero de seguimiento del consumo con distintos datos guardados.

Figura Conexión del flujo entre las vistas de la aplicación

La figura anterior determina el flujo que seguiría un usuario para que funcione adecuadamente. Desde el registro su caracterización, recordatorio y validación de consumo hasta finalmente obtener los reportes históricos.

## Home

|  |  |
| --- | --- |
| Idealmente la vista de home permitirá al usuario realizar el login con un usuario y contraseña. | Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto  Descripción generada automáticamente  Figura Vista de home de la aplicación |

## Registro y caracterización inicial

|  |  |
| --- | --- |
| Esta vista tomará datos relevantes de cada usuario para poder sugerirle un consumo de agua al día.  Con base además de las horas disponibles en el día se establecerán los intervalos de tiempo en que se debe notificar al usuario sobre el consumo de agua.  Esto se reflejará en la vista de “recordatorios y validación” | Interfaz de usuario gráfica, Aplicación  Descripción generada automáticamente |

## Vista Recordatorios y validación de consumo de agua

|  |  |
| --- | --- |
| En esta vista el usuario podrá recibir las notificaciones y cantidades sugeridas por la aplicación.  Por otro lado, desde esta misma vista el usuario podrá confirmar y validar la cantidad de agua que ha consumido en el momento. | Interfaz de usuario gráfica, Aplicación  Descripción generada automáticamente  Figura Vista de notificación y validación del consumo de agua |

## Vista informe

|  |  |
| --- | --- |
| Finalmente, desde la vista informe el usuario podrá hacer consultas sobre el histórico de las validaciones que se han realizado en un periodo de tiempo.  El usuario podrá consultar en distintas maneras su información y además podrá realizar una exportación de la información a manera de tabla en Excel. | Gráfico, Gráfico de líneas  Descripción generada automáticamente  Figura Vista informe de consumo de agua |

# Código fuente del Proyecto

Para este proyecto se crea la clase principal **main**. En las siguientes imágenes se muestran algunas capturas de pantalla de la aplicación en su etapa de desarrollo.

## Clase Persona

Esta clase implementa los atributos y métodos descritos en el diagrama de clases de la aplicación.

Texto

Descripción generada automáticamente

Figura Clase Persona muestra del código fuente

## Clase Recordatorio

Esta clase implementa los atributos y métodos descritos en el diagrama de clases de la aplicación.

Texto

Descripción generada automáticamente

Figura Clase Recordatorio muestra del código fuente

## Clase Registro

Esta clase implementa los atributos y métodos descritos en el diagrama de clases de la aplicación.

Texto

Descripción generada automáticamente

Figura Clase Registro muestra del código fuente

# Instancias de las clases creadas

A continuación, se relaciona una imagen donde en la clase principal del proyecto se considera las instancias de las clases definidas

Texto

Descripción generada automáticamente

Figura Instancias de las clases diseñadas y programadas

# Bibliografía

* <https://www.bbc.com/mundo/noticias/2016/04/160401_salud_alimentacion_deshidratacion_efectos_no_beber_agua_lb>
* <https://www.who.int/news-room/fact-sheets/detail/healthy-diet>
* <https://www.who.int/news-room/fact-sheets/detail/drinking-water>
* <https://www.accioncontraelhambre.org/es/te-contamos/blog-testimonios/agua-el-primer-alimento#:~:text=El%20agua%2C%20que%20conforma%20el,a%C3%BAn%20mayor%20en%20los%20ni%C3%B1os>.