



Proyecto Final de Programación

Gestor de Proyectos

Alumno: Juanma Vizcaíno Barquero

ÍNDICE

ÍNDICE	2
Breve Descripción:	2
Duración:	2
Enlaces:	3
ANÁLISIS (F1)	3
OBJETIVOS de la aplicación	3
DISEÑO (F2)	4
ESTRUCTURAS DE DATOS	6
2 Colecciones usadas:	6
CODIFICACIÓN (F3)	7
Historial de versiones:	7
PRUEBAS (F4)	8
FICHERO INICIAL:	9
FICHERO tras la sesión actual *.ser:	10
AUTOEVALUACIÓN:	11

Breve Descripción:

Este es un proyecto propuesto por la profesora de programación del ciclo de Desarrollo de Aplicaciones Web del IES ÁGORA, como proyecto final de la asignatura. Desarrollado en Java, realiza una gestión de proyectos de software, incluyendo altas, bajas, filtros por campos...

En este proyecto se pone a prueba las capacidades aprendidas en el curso, programación modular, uso de colecciones, tratamiento de excepciones, testing, documentación...

Duración:

Fecha Inicio: 15 de Mayo 2024

Fecha Fin: 7 de Junio de 2024

Enlaces:

Link al repositorio (Drive): <https://drive.google.com/>

Descarga el repositorio del proyecto desde la plataforma de GitHub:

https://github.com/JuanmaDAW/Gestor_Proyectos_Juanma_Vizcaino.git

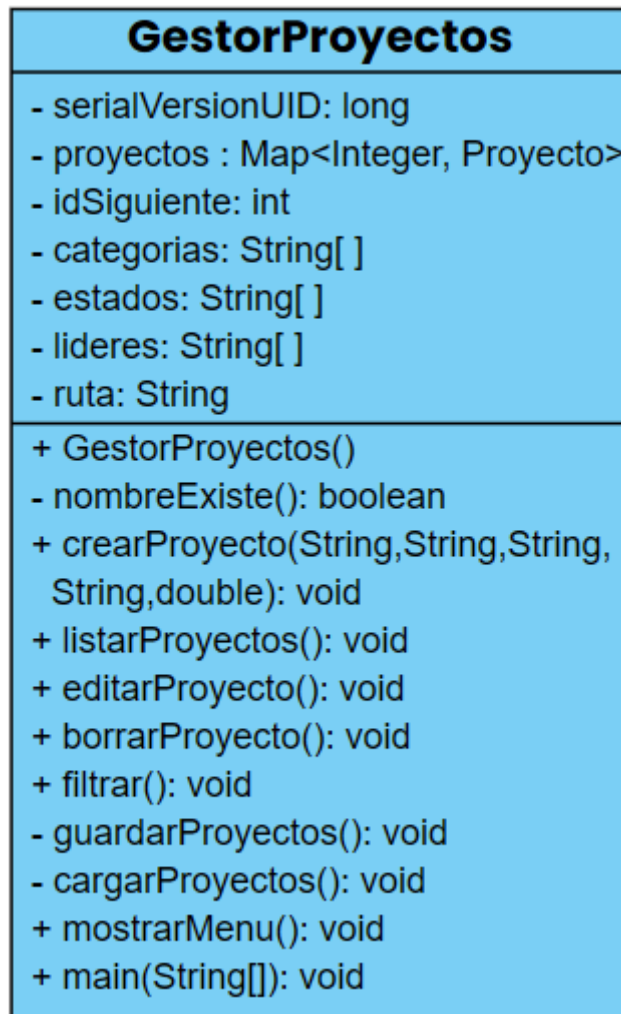
ANÁLISIS (F1)

OBJETIVOS de la aplicación

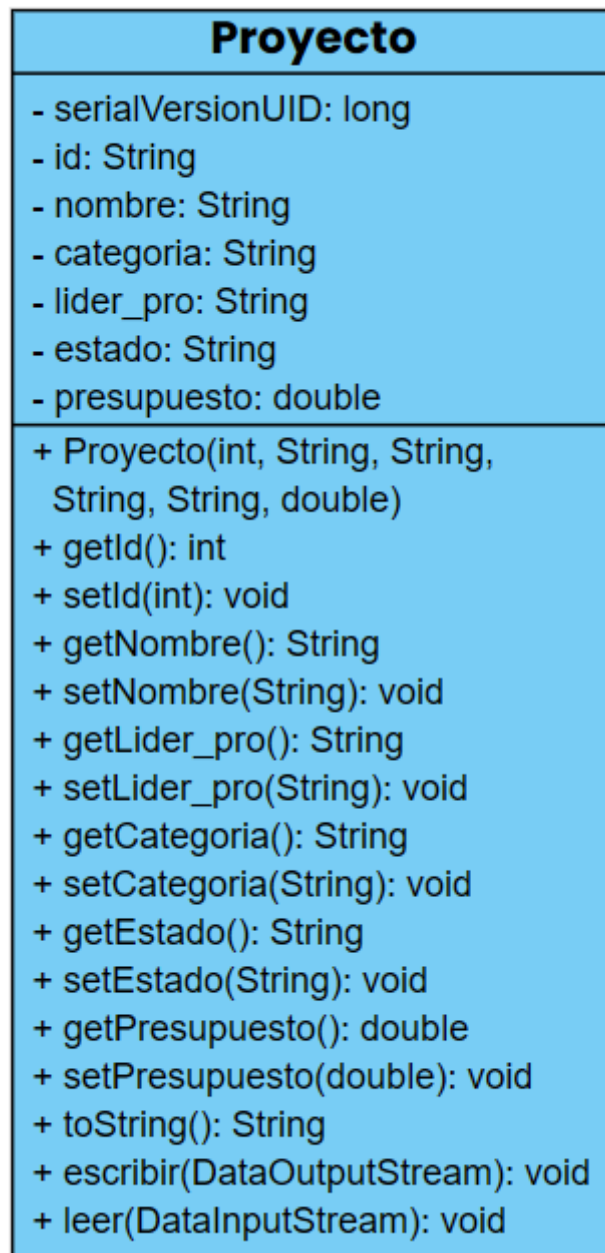
- Crear proyectos
- Listar proyectos existentes
- Modificar proyectos
- Borrar proyectos
- Volcar los datos a un fichero al cerrar el programa y recuperarlos al abrirlo
- Filtrar por el campo que se quiera, nombre del proyecto, lider, categoría, presupuesto etc
- Control de excepciones

DISEÑO (F2)

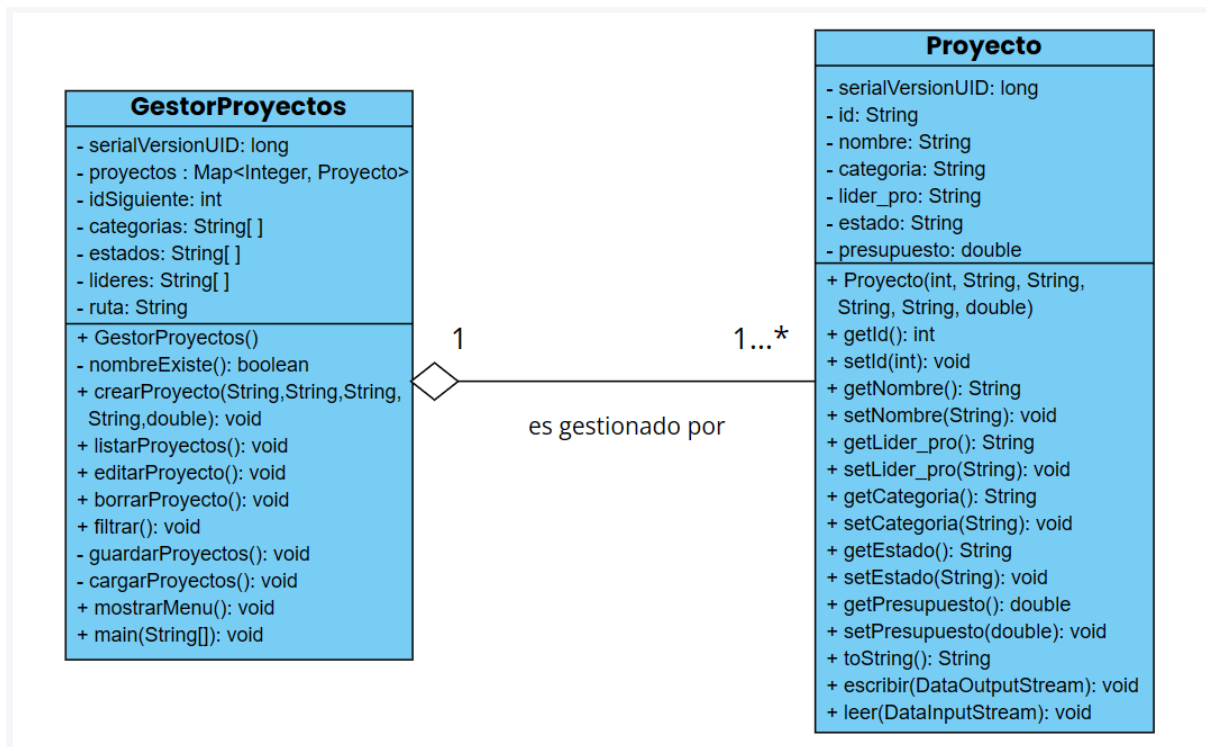
UML- clase principal GestorProyectos



UML- clase secundaria Proyecto



UML- Relaciones de clases



ESTRUCTURAS DE DATOS

1.- Arrays usados:

Se han utilizado tres arrays de tipo String para guardar las Categorías, los Estados y los Líderes del proyecto.

```
private static String categorias[] = { "Big Data", "I.Artificial", "Ciberseguridad", "Desarrollo Web" };
private static String estados[] = { "Pendiente", "En proceso", "Finalizado" };
private static String lideres[] = { "Marco", "Lidia", "Juanma", "Dani" };
```

2 Colecciones usadas:

La colección elegida es Hashmap

```
private static Map<Integer, Proyecto> proyectos = new HashMap<>();
```

➤ **Uso de la colección:** Se utiliza para guardar los objetos de tipo Proyecto, además de asociar un ID único a cada uno de ellos.

➤ **Ventajas de usar HashMap:** Ofrece acceso rápido y eficiente a los datos para búsqueda, inserción y eliminación de datos. Además, garantiza la unicidad de las claves, evitando duplicados, y proporciona flexibilidad para almacenar y gestionar cualquier tipo de objeto.

CODIFICACIÓN (F3)

Historial de versiones:

📁 Versión 1.0

Descripción:

Primera versión, estática, usando una colección de ArrayList y mostrando un menú para crear proyectos.

📁 Versión 2.0

Descripción:

Segunda versión, igual que la anterior pero adaptando las clases con programación modular.

☐ Versión 3.0

Descripción:

Versión 3, versión con un menú básico, usando un fichero en txt, pero los datos no son persistentes todavía, al cerrar el programa los datos se pierden, y por último he añadido a esta nueva versión un control de excepciones.

☐ Versión 4.0 (Final)

Descripción:

Versión 4, incluye un menú más avanzado, con más opciones, incluida la opción de filtrar por campos, además los datos que se guardan son serializables y se almacenan de manera permanente en el fichero, en este caso proyectos.ser.

PRUEBAS (F4)

(Tester's)

Nombre Tester	Sugerencias	Estado
Abraham	Cambiar opciones del menú de posición, validar los campos vacíos	Realizada ▾
David	Cambiar el formato de la salida por consola	Realizada ▾
Raúl	Mostrar campos donde haya opciones para elegir	Realizada ▾
Jesús	Crear excepción cuando no haya proyectos en el fichero y mostrarla	Realizada ▾
Maruan	Siempre que se termina de elegir una opción volver a repetir el menú, hasta que se elija la opción de salir	Realizada ▾

Carlos	No permitir texto en una entrada de tipo numérica	Realizada ▾
--------	---------------------------------------------------	-------------

FICHERO INICIAL:

```

proyectos.ser: Bloc de notas
Archivo Edición Formato Ver Ayuda
|  |  | Conecta | Big Data | Lidia
En proceso@G~
BrainStorm ^I.Artificial | Marco
FinalizadoA\A      | | Escudo | Ciberseguridad | Dani
En proceso@ej

```

Salida por consola:

```

Menú:
1. Crear Nuevo Proyecto
2. Listar Proyecto
3. Modificar Proyecto
4. Borrar Proyecto
5. Filtrar por campo
6. Salir
Seleccione una opción: 2

```

ID	Nombre	Categoria	Lider	Estado	Presupuesto
1	Conecta	Big Data	Lidia	En proceso	20000.0
2	BrainStorm	I.Artificial	Marco	Finalizado	350000.0
3	Escudo	Ciberseguridad	Dani	En proceso	50000.0

FICHERO tras la sesión actual *.ser:

```
proyectos.ser: Bloc de notas
Archivo Edición Formato Ver Ayuda
|  |  |Conecta |Big Data |Lidia
En proceso@Ó~
BrainStorm |I.Artificial |Marco
FinalizadoA|\A      | |Escudo |Ciberseguridad |Dani
En proceso@èj      |
Web Ágora |Desarrollo Web |Juanma
Finalizado@ã~      | |Kali |Ciberseguridad |Dani
En proceso@Øj      | |Apache |Desarrollo Web |Juanma   Pendiente@R@      | |Babel |I.Artificial |Lidia
En procesoA.,,€    | |Estadista |Big Data |Marco
Finalizado@ÈX
```

Salida por consola:

```
Menú:
1. Crear Nuevo Proyecto
2. Listar Proyecto
3. Modificar Proyecto
4. Borrar Proyecto
5. Filtrar por campo
6. Salir
Seleccione una opción: 2
|
ID      Nombre      Categoria      Lider      Estado      Presupuesto
-----
1       Conecta      Big Data      Lidia      En proceso   20000.0
2       BrainStorm    I.Artificial  Marco      Finalizado   350000.0
3       Escudo          Ciberseguridad  Dani      En proceso   50000.0
4       Web Ágora       Desarrollo Web  Juanma     Finalizado   40000.0
5       Kali           Ciberseguridad  Dani      En proceso   25000.0
6       Apache         Desarrollo Web  Juanma     Pendiente    32000.0
7       Babel          I.Artificial    Lidia      En proceso   1000000.0
8       Estadista      Big Data       Marco      Finalizado   14000.0
```

AUTOEVALUACIÓN:

- **Debilidades**

Al principio el tema de los ficheros, con todos los tipos que hay para leer y escribir, los tipos de ficheros como .dat, .ser, o .obj no me dejaban claro cuál usar y cómo usarlos.

Además considero una debilidad el hecho de no haber pedido más testeos a mis compañeros para mejorar más la aplicación.

- **Fortalezas**

Resolver los problemas que me han surgido de manera eficiente, tanto investigando en internet como buscando ayuda por parte de mis compañeros.

Además creo que tengo facilidad a la hora de organizar la documentación del proyecto.

- **Si tuviera más tiempo**

En ese caso dedicaría el tiempo a añadir más funcionalidades, arreglar los bug que pueda tener, y añadir más validaciones para perfeccionar la aplicación.

- **Conclusiones personales**

He disfrutado mucho trabajando en el proyecto, he aprendido sobre excepciones, entrada y salida de datos en un fichero, testing, documentación...

Además de aprender a implementar todos los conocimientos aprendidos en el curso, a mejorar gracias a los tester, a ser crítico cuando tu eres el tester, a trabajar con un proyecto lo más parecido a una aplicación que usa una base de datos pero en nuestro caso usando ficheros y almacenando allí la información persistente.

En definitiva, creo que los proyectos son una de las mejores maneras de aprender, ya que te obligas a implementar todos los conocimientos que sabes para poder completar los objetivos propuestos.