

# El Compresor

Proyecto PROP Cuatrimestre Otoño 2019/20

## Miembros del grupo

- Boné Ribó, Aleix aleix.bone@est.fib.upc.edu Gonzàlez Godoy, Alex alex.gonzalez.godoy@est.fib.upc.edu Herrero Pons, Alex alex.herrero.pons@est.fib.upc.edu Mercadé Plasencia, Albert albert.mercade.plasencia@est.fib.upc.edu

## Lista de clases y documentos implementados por cada miembro del grupo:

- Clases:
  - **JPEG**: Aleix Boné
  - **LZ78**: Alex González
  - **LZW**: Alex Herrero
  - **LZSS**: Albert Mercadé
  - **Folder**: Aleix Boné
  - **JPEGBlock**: Aleix Boné
  - **Huffman**: Aleix Boné
  - **PpmImage**: Aleix Boné
  - **BitSetL**: Aleix Boné
  - **IO**: Alex González
  - **Statistics**: Alex Herrero
  - **CtrlDomini**: Albert Mercadé
  - **CtrlPresentacio**: Albert Mercadé
  - **Main**: Alex Herrero
- **Interfaz gráfica de usuario**: toda hecha por Albert Mercadé
- **Juegos de prueba**: todos hechos por Alex González
- **Drivers**: Alex Herrero
- **Documentación**:
  - **Diagramas de clase/Casos de uso**: Albert Mercadé
  - **Especificación detallada**: Alex González
  - **Explicación juegos de prueba**: Alex González
  - **Doxygen**: Alex Herrero
  - **Explicación ED & Algoritmos**: Alex Herrero

## Lista librerías externas

- **Forms\_rt**: Hemos implementado la interfaz gráfica de usuario usando el *designer* que nos proporcionaba el propio *IntelliJ IDEA* y necesitamos la librería **Forms\_rt**, de la cual usamos la versión 7.0.3, para generar todas las inicializaciones de los componentes de gráficos que usamos (JLabel, JSpinner, JButton, etc) acorde a lo especificado en el *designer*.

- **API Guardian/Hamcrest Core/JUnit Jupiter/Junit Jupiter Api/JUnit Platform Console Standalone:** Todas estas librerías las usamos para hacer los test con JUnit, muy necesarios para poder probar todos los componentes de nuestro código por separado mientras los desarrollábamos o mejorábamos.

## Lista de funcionalidades implementadas:

### 1a Entrega:

- Los algoritmos LZs solo comprimían y descomprimían archivos de tipo `.txt`.
- El algoritmo JPEG comprimía archivos de tipo `.ppm`.
- Se generaban estadísticas de compresión y descompresión.

### 2a Entrega:

Además de las funcionalidades implementadas en la 1a entrega. - **Obligatorio:** - Se permite comprimir y descomprimir carpetas (con subcarpetas y/o todo tipo de archivos). - En esta segunda entrega hemos implementado una interfaz gráfica con la que interactúa el usuario, en lugar de con la consola. - Con esta interfaz gráfica se permite la visualización de un fichero de entrada y su correspondiente descompresión después de aplicar el proceso de compresión/descompresión con un cierto algoritmo. - **Opcional:** - Hemos mejorado los algoritmos LZs de forma que ahora comprimen y descomprimen todo tipo de archivos. - Además el algoritmo LZ78 ahora implementa la compresión con **Tree**.

## Diagramas

### Diagrama Casos de Uso

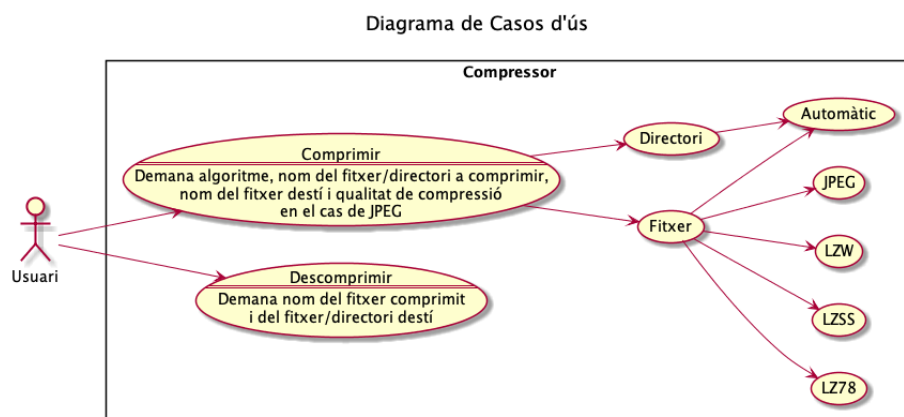


Figure 1: Diagrama Casos d'ús

## Diagramas de Clases

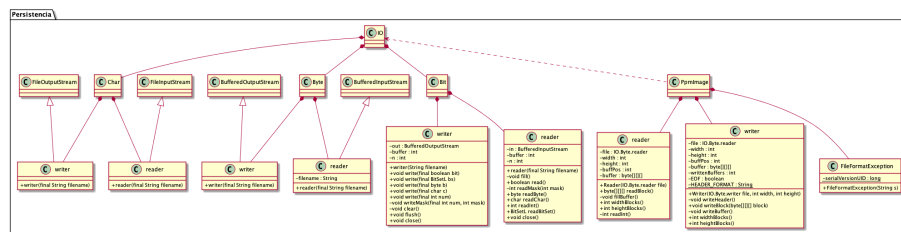
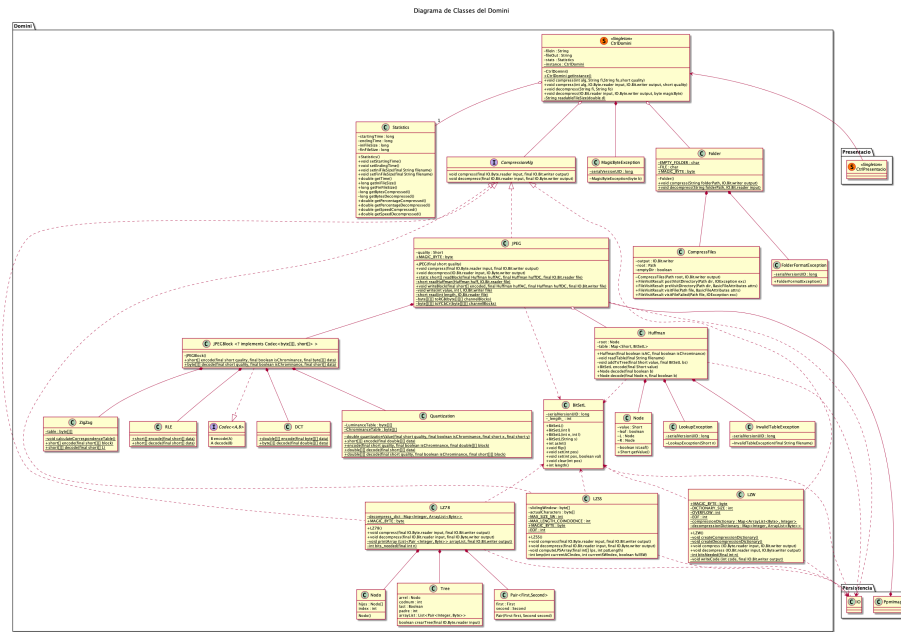


Diagrama de classes de Presentacio

