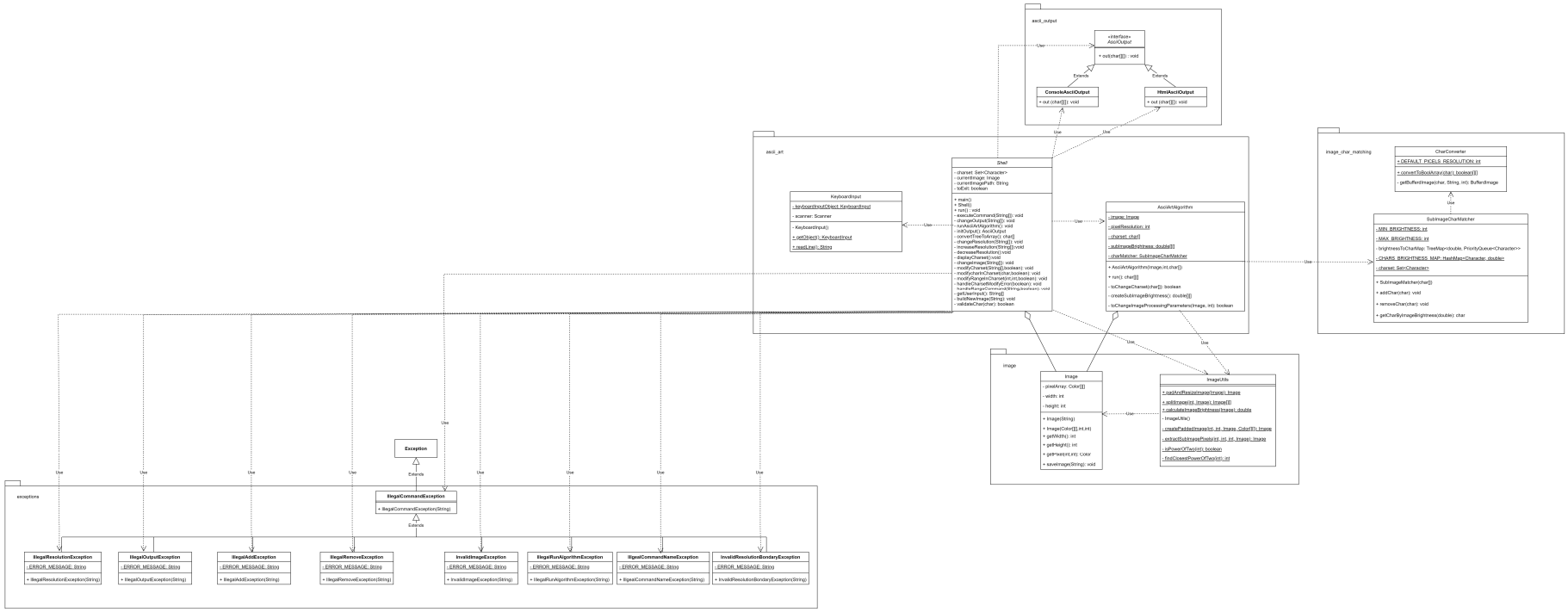
ascii_output
«interface»
AsciiOutput
+ out(char[][]) : void
Extends
Extends
ConsoleAsciiOutput
+ out (char[][]): void
HtmlAsciiOutput
+ out (char[][]): void
Use
Use
Use
ascii_art
Shell
- charset: Set<Character>
- currentImage: Image
- currentImagePath: String
- toExit: boolean
+ main()
+ Shell()
+ run() : void
- executeCommand(String[]): void
- changeOutput(String[]): void
- runAsciiArtAlgorithm(): void
- initOutput(): AsciiOutput
- changeConsole(String[]): void
- changeResolution(String[]): void
- increaseResolution(String[]):void
- decreaseResolution():void
- displayCharset():void
- changeImage(String[]): void
- modifyCharsetString([]boolean): void
- modifyCharsetChar(char,boolean): void
- modifyRangeCharset(int,int,boolean): void
- handleCharsetModifyError(boolean): void
- handleRangeCommand(String,boolean): void
- getUserInputs(): String[]
- buildNewImage(String): void
- validateChar(char): boolean
Use
Use
Keyboardinput
- keyboardInputObject: KeyboardInput
- scanner: Scanner
- KeyboardInput()
+ getObject(): KeyboardInput
+ readLine(): String
Use
AsciiArtAlgorithm
- image: Image
- pixelResolution: int
- charset: char[]
- subImageBrightness: double[][]
- charMatcher: SubImageCharMatcher
+ AsciiArtAlgorithm(Image,int,char[])
+ run(): char[][]
- toChangeCharset(char[]): boolean
- createSubImageBrightness(): double[][]
- toChangeImageProcessingParameters(Image, int): boolean
Use
Use
Use
image_char_matching
CharConverter
+ DEFAULT_PICELS_RESOLUTION: int
+ convertToBoolArray(char): boolean[][]
- getBuffredImage(char, String, int): BufferdImage
Use
SubImageCharMatcher
- MIN_BRIGHTNESS: int
- MAX_BRIGHTNESS: int
- brightnessToCharMap: TreeMap<double, PriorityQueue<Character>>
- CHARS_BRIGHTNESS_MAP: HashMap<Character, double>
- charset: Set<Character>
+ SubImageMatcher(char[])
+ addChar(char): void
+ removeChar(char): void
+ getCharByImageBrightness(double): char
image
Image
- pixelArray: Color[][]
- width: int
- height: int
+ Image(String)
+ Image(Color[][], int,int)
+ getWidth(): int
+ getHeight(): int
+ getPixel(int,int): Color
+ saveImage(String): void
Use
ImageUtils
+ readAndResize(String,Image): Image
+ addBlank(int, Image): Image[][]
+ calculateImageBrightness(Image): double
- ImageUtils()
- createPaddedImage(int, int, Image, Color[][]): Image
- extractSubImagePixels(int, int, int, Image): Image
- isPowerOfTwo(int): boolean
- findClosestPowerOfTwo(int): int
Use
Use
Use
Use
Use
Use
Use
Use
Use
Use
Exception
Extends
IllegalCommandException
+ IllegalCommandException(String)
Extends
exceptions
IllegalResolutionException
- ERROR_MESSAGE: String
+ IllegalResolutionException(String)
IllegalOutputException
- ERROR_MESSAGE: String
+ IllegalOutputException(String)
IllegalAddException
- ERROR_MESSAGE: String
+ IllegalAddException(String)
IllegalRemoveException
- ERROR_MESSAGE: String
+ IllegalRemoveException(String)
InvalidImageException
- ERROR_MESSAGE: String
+ InvalidImageException(String)
IllegalRunAlgorithmException
- ERROR_MESSAGE: String
+ IllegalRunAlgorithmException(String)
IllegalCommandNameException
- ERROR_MESSAGE: String
+ IllegalCommandNameException(String)
InvalidResolutionBoundaryException
- ERROR_MESSAGE: String
+ InvalidResolutionBoundaryException(String)