

1 - Descrição das classes

Classes `Label`, `Message` and `Prompt` (from the `swing` package) have been omitted since they represent logs of global functions or widgets and are never instantiated as normal concept classes.

Constructors with no arguments, abstract methods or interface methods implementations, and other possibly inherited methods have been omitted.

pt. tecnico . utilif

Display

- `mi`: `Dialog`
- `title`: `String` = ""
- `text`: `String` Builder

- + `Display(title: String)`
- + `Display(mi: Dialog, title: String)`
- + `add(toAdd: Object): Display`
- + `addAll(items: Collection<?>): Display`
- + `addhere(toAdd: Object): Display`
- + `addNewLine(toAdd: Object, force: boolean): Display`
- + `<<find>> display(): void`
- + `<<find>> displayText(): void`
- + `popup(toPop: Object): void`
- + `popup(toPop: Collection<?>): void`
- + `clear(): void`

Dialog

- `<<find>> ACTION_CHANNEL: String` = "mi"
- `<<find>> CHANNEL_SWING: String` = "swing"
- `<<find>> CHANNEL_NEW_SWING: String` = "newswing"
- `<<find>> CHANNEL_TEXT: String` = "test"

- + `UI: Dialog`

- `backend: InteractionDriver`

- + `Dialog(backend: InteractionDriver)`
- + `open(menu: Menu): void`
- + `fill(form: Form): void`
- + `render(title: String, text: String): void`
- + `close(): void`

<<Interface>>

InteractionDriver

- + `open(menu: Menu): void`
- + `fill(form: Form): void`
- + `render(title: String, text: String): void`
- + `close(): void`

Menu

- _mi : Dialog
- .title : String
- .commands : Command<?>[]

```
+ Menu(mi : Dialog, title : String,
      commands : Command<?>...)
+ Menu(title : String, commands : Command<?>...)
+ title() : String
+ size() : int
+ entry(n : int) : Command<?>
+ entries() : Command<?>[]
+ open() : void
```

Dialog Menu

```
# <<find>> execute() : void
```

```
Receiver: class <<abstract>>
Command
```

- .lost : boolean
- .title : String
- # <<find>> - receiver : Receiver
- # - valid : Predicate<Receiver>
- <<find>> - form : Form
- # <<find>> - display : Display

```
+ Command(lost : boolean, title : String)
+ Command(lost : boolean, title : String,
          receiver : Receiver)
+ Command(lost : boolean, title : String,
          receiver : Receiver, valid : Predicate<Receiver>)
+ Command(title : String, receiver : Receiver)
+ Command(title : String, receiver : Receiver,
          valid : Predicate<Receiver>)
+ title() : String
+ isLost() : boolean
+ isValid() : boolean
+ addBooleanField(key : String, prompt : String) : void
+ addRealField(key : String, prompt : String) : void
+ addIntegerField(key : String, prompt : String) : void
+ addStringField(key : String, prompt : String) : void
+ addOptionField(key : String, prompt : String,
                options : String...) : void
+ booleanField(key : String) : Boolean
+ realField(key : String) : Double
+ integerField(key : String) : Integer
+ stringField(key : String) : String
+ optionField(key : String) : String
+ <<find>> performCommand() : void
# <<abstract>> execute() : void
```

Exceptions:

CommandException

pt. mechanics. milit. forms

Form

- _ui: Dialog
- _title: String
- _fields: Map<String, Field<?>>

- + Form(title: String)
- + Form(ui: Dialog, title: String)
- + title(): String
- + entries(): Collection<Field<?>>
- add(key: String, in: Field<?>): void
- + addBooleanField(key: String, label: String): void
- + addRealField(key: String, label: String): void
- + addIntegerField(key: String, label: String): void
- + addOptionField(key: String, label: String, options: String...): void
- get(key: String, type: String): Object
- + booleanField(key: String): Boolean
- + stringField(key: String): String
- + optionField(key: String): String
- + realField(key: String): Double
- + integerField(key: String): Integer
- + parse(): Form
- + parse(clear: boolean): Form
- + clear(): void
- + confirm(prompt: String): Boolean
- + requestInteger(prompt: String): Integer
- + requestReal(prompt: String): Double
- + requestString(prompt: String): String
- + requestOption(prompt: String, options: String...): String

Exceptions:

FormException

FieldNone

FieldNone(prompt: String)
+ isReadOnly(): boolean

FieldString

FieldString(prompt: String)

<< abstract >>

Field

- prompt: String
- _clear: boolean
- # _value: Type
- # Field(prompt: String)
- + prompt(): String
- + set(value: Type): void
- + value(): void
- + clear(): void
- # dirty(): void
- + cleared(): boolean
- + isReadOnly(): boolean
- + << abstract >> parse(in: String): boolean

FieldBoolean

- << final >> BOOLEAN_WORD_YES: String = "yes"
- << final >> BOOLEAN_WORD_NO: String = "no"
- << final >> BOOLEAN_CHAR_YES: char = 'y'
- << final >> BOOLEAN_CHAR_NO: char = 'n'

FieldBoolean(prompt: String)

FieldOption

- _options: String[]
- FieldOption(prompt: String, options: String[])

FieldReal

FieldReal(prompt: String)

FieldInteger

FieldInteger(prompt: String)

pt. tecnico. milib. text

Composite PrintStream

- streams: Collection<PrintStream>
- error: boolean = false

CompositePrintStream(PrintStreams: PrintStream...)

<<final>> add(ps: PrintStream): void

+ checkError(): boolean

+ close(): void

+ flush(): void

+ print(b: boolean): void

+ print(c: char): void

+ print(c: char[]): void

+ print(d: double): void

+ print(f: float): void

+ print(i: int): void

+ print(l: long): void

+ print(obj: Object): void

+ print(str: String): void

+ println(b: boolean): void

+ println(c: char): void

+ println(c: char[]): void

+ println(d: double): void

+ println(f: float): void

+ println(i: int): void

+ println(l: long): void

+ println(obj: Object): void

+ println(str: String): void

+ setError(): void

+ write(buf: byte[], off: int, len: int): void

+ write(b: int): void

+ write(b: byte[]): void

<<interface>>

Property

WRITE_INPUT: String = "writeInput"

BOTH_CHANNELS: String = "both"

INPUT_CHANNEL: String = "in"

OUTPUT_CHANNEL: String = "out"

LOG_CHANNEL: String = "log"

Text Interaction

- in: BufferedReader

- out: PrintStream

- log: PrintStream

- writeInput: boolean

- readString(prompt: String): String

- readInteger(prompt: String): int

Exceptions:

IOException

Only Java classes that have been
extended or implemented are explicitly
represented

