



Serial numbers for serialization have been omitted

All of the methods and attributes were defined separately in order to better understand the connections established between classes and packages.

## Dialog

```
-<<final>> ACTION_CHANNEL: String = "u1"
-<<final>> CHANNEL_SWING: String = "swing"
-<<final>> CHANNEL_NEW_SWING: String = "newswing"
-<<final>> CHANNEL_TEXT: String = "text"
+ U1: Dialog = new Dialog()
- backend: InteractionDriver

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

## Text Interaction

```
-in: BufferedReader = new BufferedReader (new
InputStreamReader (System.in))
-out: PrintStream = System.out
-log: PrintStream = null
-writeInput: boolean
-readString (prompt: String): String
-readInteger (prompt: String): int
```

## Display

```
-ui: Dialog
-title: String = ""
-text: StringBuilder = new StringBuilder ()

+ add (toAdd: Object): Display
+ addAll (items: Collection<?>): Display
+ addLine (toAdd: Object): Display
+ addNewLine (toAdd: Object, force: boolean):
Display
+ <<final>> display(): void
+ <<final>> displayText: void
+ popop (toPop: Object): void
+ popop (toPop: Collection<?>): void
+ clear(): void
```

## Composite PrintStream

```
-stream: Collection<PrintStream> = new ArrayList<>()
-error: boolean = false
+ <<final>> add (ps: PrintStream): void
```

Message, Prompt, Property and InteractionDriver have been omitted as they represent bags of global functions on variables and are never instantiated as normal concept classes.

DoOpen Menu

```
# <<final>> execute(); void
```

CommandException

Runtime EOFException

Menu

```
-ui: Dialog  
-title: String  
-commands: Command<?>[]  
  
+size(): int  
+entry(n, int): Command<?>  
+open(): void
```

Message has been omitted as it represents bags of global functions on variables and is never instantiated as a normal concept class.

Receiver  
Receiver: Class  
Command

```
-last: boolean  
-title: String  
# <<final>> -receiver: Receiver  
# -valid: Predicate<Receiver> = receiver → true  
- <<final>> -form: Form  
# <<final>> -display: Display
```

```
+isValid(): boolean  
+addBooleanField(key: String, prompt: String, bid)  
+addRealField(key: String, prompt: String, bid)  
+addIntegerField(key: String, prompt: String, bid)  
+addStringField(key: String, prompt: String, bid)  
+addOptionField(key: String, prompt: String, options: String, bid): void  
+booleanField(key: String): Boolean  
+realField(key: String): Double  
+integerField(key: String): Integer  
+stringField(key: String): String  
+optionField(key: String): String  
+ <<final>> performCommand(): void  
# <<abstract>> execute(): void
```

Declaro por  
minha honra  
que este  
diagrama foi  
realizado apenas  
pelos elementos  
que constituem  
o grupo de  
projeto.

Wander



## Form

- ui: Dialog
- title: String
- fields: Map<String, Field<?>> = new LinkedHashMap<>()
- +entries(): Collection<Field<?>>
- +field(key: String): Field<?>
- add(key: String, in: Field<?>): void
- +addBooleanField(key: String, label: String): void
- +addStringField(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
- +can firm (prompt: String): boolean
- +requestInteger(prompt: String): Integer
- +requestReal(prompt: String): Double
- +requestString(prompt: String): String
- +requestOption(prompt: String, options: String...): String

!Type, Class!  
«abstract»  
Field

- prompt: String
- clear: boolean
- #\_value: Type
- +set(value: Type): void
- +clear(): void
- #dirty(): void
- +isReadOnly(): boolean
- +«abstract» parse(in: String): boolean

### Field Integer

+parse(in: String): boolean

### Form Exception

### Field Boolean

- «final»>> BOOLEAN-WORD-YES: String = "sim"
- «final»>> BOOLEAN-WORD-NO: String = "nao"
- «final»>> BOOLEAN-CHAR-YES: char = 's'
- «final»>> BOOLEAN-CHAR-NO: char = 'n'
- +parse(in: String): boolean

### Field None

+parse(in: String): boolean

+isReadOnly(): boolean

### Field Option

- options: String[]
- +parse(in: String): boolean

### Field Real

+parse(in: String): boolean

### Field String

+parse(in: String): boolean

Message has been omitted as it represents bags of global functions on variables and is never instantiated as a normal concept class.