



Interaction Driver

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

Display

- ui : Dialog
- title: String
- text: StringBuilder
- + add (to Add: Object) : Display
- + addAll (to item: Collection) : Display
- + addLine (to Add: Object) : Display
- + addNewline (to Add: Object, force: Boolean) : Display
- + display () : void
- + display (to Put: Object) : void
- + display (to Put: Collection) : void
- + clear () : void

Dialog

- /

+ Dialog UI: Dialog

- interaction: Interaction Driver

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

Menu

- id: String
- title: String
- commands: Commands
- + title() : String
- + id() : int
- + sub (m: id) : Command
- + activate () : Command
- + clear () : void

Do Open Menu

~ execute () : void

Command

- last : boolean
- title: String
- receiver: Receiver
- workflow : Predictive
- form: Form
- display: Display

+

- + title() : String
- + isLast() : boolean
- + isValid () : boolean
- + addBooleanField (key: String, value: String) : void
- + addRealField (key: String, value: String) : void
- + addIntegerField (key: String, value: String) : void
- + addStringField (key: String, value: String) : void
- + booleanField (key: String) : Boolean
- + realField (key: String) : Double
- + integerField (key: String) : Integer
- + stringField (key: String) : String
- + runCommand () : void

#

Command Exception

- menuItemID : long

Form	Field	Test Interaction
<ul style="list-style-type: none"> - <code>ini</code>: Dictionary - <code>title</code>: String - <code>fields</code>: Map<String, Field>> 	<ul style="list-style-type: none"> - <code>isInit</code> > String - <code>class</code>: boolean - <code>value</code>: Type 	<ul style="list-style-type: none"> - <code>ini</code>: BufferedReader - <code>out</code>: PrintWriter - <code>log</code>: PrintStream - <code>uninitializedValue</code>
<ul style="list-style-type: none"> + <code>title()</code>: String + <code>enhance</code>: Collection<Field>> + <code>field (key: String) : Field</code>>> + <code>add (key: String, ini: Field<?, ?>): void</code> + <code>add BooleanField (key: String, value: String): void</code> + <code>add StringField(key: String, value: String): void</code> + <code>add RealField(key: String, value: String): void</code> + <code>add IntegerField(key: String, value: String): void</code> - <code>get (key: String, type: String): Object</code> + <code>BooleanField (key: String): Boolean</code> + <code>StringField (key: String): String</code> + <code>realField (key: String): Double</code> + <code>IntegerField (key: String): Integer</code> + <code>func(): Form</code> + <code>func(class: Boolean): Form</code> + <code>clear(): void</code> + <code>copyFrom (what: String): Boolean</code> + <code>registerInteger(key: String): Integer</code> + <code>registerReal(key: String): Double</code> + <code>registerString(key: String): String</code> 	<ul style="list-style-type: none"> + <code>func(): String</code> + <code>set (name: Type): void</code> + <code>value(): Type</code> + <code>class(): void</code> - <code>dirty(): void</code> + <code>cleanUp(): boolean</code> + <code>is ReadOnly(): boolean</code> + <code>func (in: String): boolean</code> 	<ul style="list-style-type: none"> + <code>clear(): void</code> + <code>open(name: Name): void</code> + <code>fill (fun: Function): void</code> + <code>render (value: String, out: String): void</code> - <code>readString (func: String): String</code> - <code>readString (func: String): int</code>
	<p>Field Boolean</p> <ul style="list-style-type: none"> - <code>BooleanField(key: String)</code> - <code>BooleanField(name: String)</code> - <code>BooleanField(value: String)</code> + <code>func (in: String): boolean</code> 	<p>ComparablePrintStream</p> <ul style="list-style-type: none"> - <code>streams</code>: Collection<PrintStream> - <code>error</code>: PrintStream
	<p>Field Integer</p> <ul style="list-style-type: none"> + <code>func (in: String): boolean</code> 	<ul style="list-style-type: none"> + <code>add (fun: PrintStream): void</code> + <code>checkEven(): Boolean</code> + <code>clear(): void</code> + <code>flush(): void</code> + <code>func (b: Boolean): void</code> + <code>func (b: boolean): void</code> + <code>func (c: char): void</code> + <code>func (d: double): void</code> + <code>func (f: float): void</code> + <code>func (i: int): void</code> + <code>func (l: long): void</code> + <code>func (u: (void)): void</code> + <code>func (v: String): void</code>
	<p>Field None</p> <ul style="list-style-type: none"> + <code>func (in: String): boolean</code> + <code>is ReadOnly(): boolean</code> 	
	<p>Field String</p> <ul style="list-style-type: none"> + <code>func (in: String): boolean</code> 	
	<p>Field Real</p> <ul style="list-style-type: none"> + <code>func (in: String): boolean</code> 	<ul style="list-style-type: none"> + <code>return(): void</code> + <code>unshift (buf: Byte[]), off: int, len: int): void</code> + <code>wrote (b: byte[], len): void</code> + <code>writeln (b: byte[]): void</code>

Declaro por minha honra que este diagrama foi
realizado apenas pelos ~~meus~~ elementos que compõem o
Gráfico do Projeto.

Yann E.