

pt. tecnico . utilbox . forms

<< abstract >>

Field

- prompt : String
- clear : boolean
- # - value : Type

+ prompt() : String
+ set(value:Type) : void
+ clear() : void
dirty() : void
+ cleared() : boolean
+ isReadOnly() : boolean
+ parse(in: String) : boolean {abstract}

Field Boolean

- BOOLEAN_WORD_YES : String {final}
- BOOLEAN_WORD_NO : String {final}
- BOOLEAN_CHAR_YES : char {final}
- BOOLEAN_CHAR_NO : char {final}

+ parse(in: String) : boolean

Field Integer

+ 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

Form Exception

- serialVersionUID : long {final}

Field

- ui : Dialog
- title : String
- fields : Map<String, Field<?>>

+ title() : String
+ 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
+ addRealField(Key: String, label: String) : void
+ addIntegerField(Key: String, label: String) : void
+ addOptionField(Key: String, label: String, option: 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(clean: boolean) : Form
+ clean() : void
+ confirm(prompt: String) : Boolean
+ requestInteger(prompt: String) : Integer
+ requestReal(prompt: String) : Double
+ requestString(prompt: String) : String
+ requestOption(prompt: String, options: String...) : String

<<interface>>

Message

+ keyAlreadyExists (form: String, key: String)

+ formNotFilled (form: String): String

+ keyNotFound (form: String, key: String)

+ typeMismatch (form: String, key: String, actualType: String, requestedType: String): String

pt.tecnico.util.messages

<<abstract>>

Command

- - last: Boolean

- - title: String

- receiver: Receiver { final }

- valid: Predicate

- - form: Form { final }

- display: Display { final }

+ title(): String { final }

+ isLast(): 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

+ performCommand(): void { final }

execute(): void { abstract }

<< abstract >>

CommandException

- serialVersionUID: long {final}

DoOpenMenu

- serialVersionUID: long {final}

Menu

- -ui: Dialog

- -title: String

- -commands: Command<?>[]

+ title(): String

+ size(): int

+ atny(n: int): Command<?>

+ entries(): Command<?>[]

+ open(): void

<< interface >>

Message

+ operationFailed(error: String): String

pt. tecnica . util. text

Composite Print Stream

```
- streams : Collection  
- error : boolean  
  
~ add( ps: PrintStream ): void { final }  
+ checkError(): boolean  
+ close(): void  
+ flush(): void  
+ print( b: boolean ): void  
+ print( c: char ): void  
+ print( e: 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( e: 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 >>

Message

```
+ invalidOperation( error: String ): String  
+ invalidOption(): String  
+ errorEOF( e: EOFException ): String  
+ errorIO( ioe: IOException ): String  
+ errorInvalidNumber( e: NumberFormatException ): String  
+ errorREOF( e: RuntimeEOFException ): String  
+ inputError( e: FileNotFoundException ): String  
+ errorClosingInput( e: IOException ): String  
+ endOfInput(): String  
+ outputError( e: FileNotFoundException ): String  
+ logError( e: FileNotFoundException ): String
```

<< interface >>

Prompt

```
+ option(): String  
+ exit(): String
```

<< interface >>

Property

```
+ WRITE_INPUT: String  
+ BOTH_CHANNELS: String  
+ INPUT_CHANNEL: String  
+ OUTPUT_CHANNEL: String  
+ LOG_CHANNEL: String
```

pt. tecnico. util. text

Runtime EOF Exception

~ serialVersion UID : long { final }

Text Interaction

- -in : BufferedReader
- -out : PrintStream
- -log : PrintStream
- -whiteInput : boolean

+ close(): void

+ open (menu: Menu): void

+ fill (form: Form): void

+ render (title: String, text: String): void

- readString (prompt: String): String

- readInteger (prompt: String): int

<< interface >>

Interaction Driver

~ open (menu: Menu): void

~ fill (form Form): void

~ render (title: String, text: String): void

~ close(): void

pt. tecnico. util

Dialog

- ACTION_CHANNEL : String { final }

- CHANNEL_SWING : String { final }

- CHANNEL_NEW_SWING : String { final }

- CHANNEL_TEXT : String { final }

+ UI : Dialog

- -backend : InteractionDriver

+ 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 (items: Collection<?>): Display

+ addLine (to Add: Object): Display

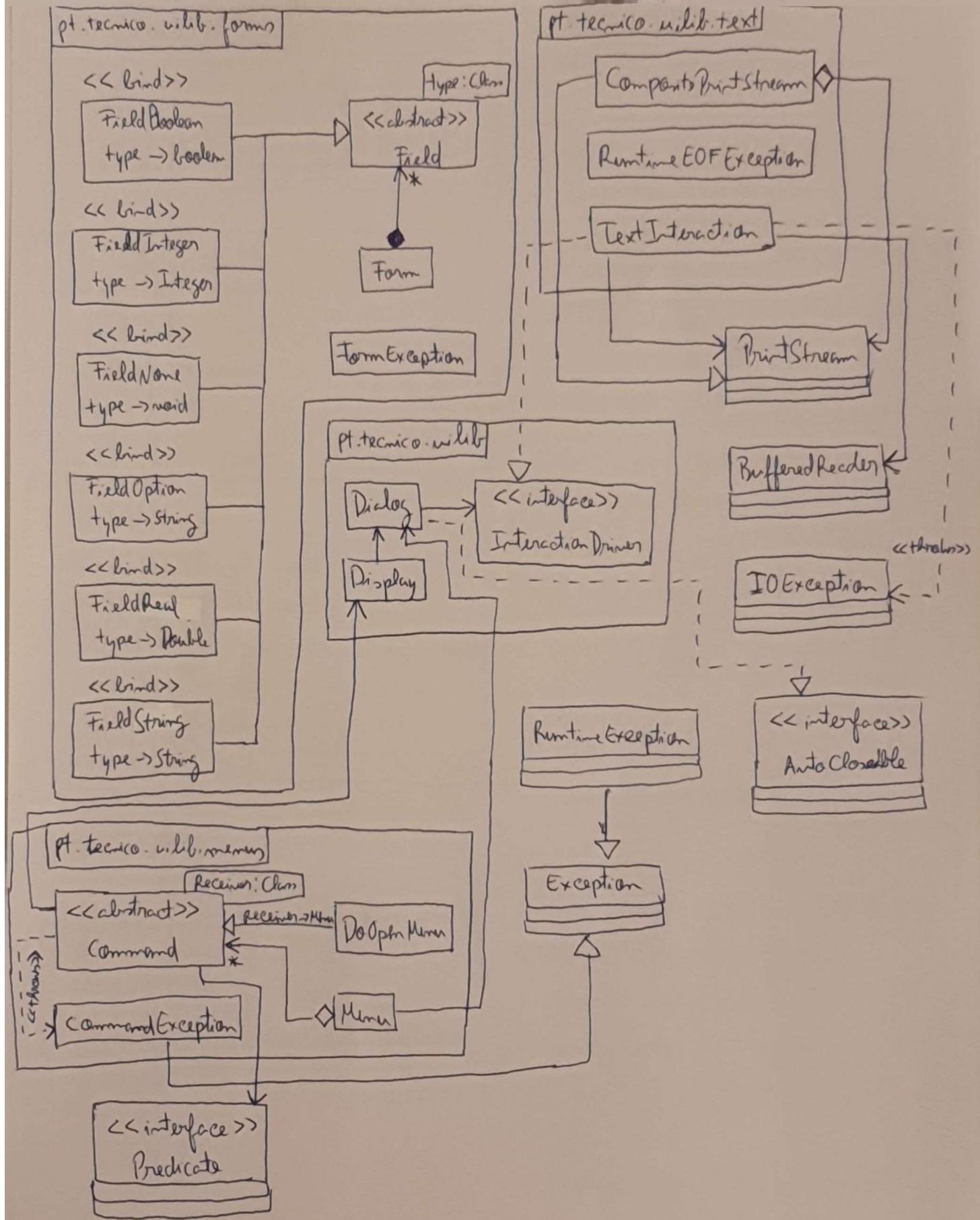
+ display (): void { final }

+ displayText (): void { final }

+ popup (toPop: Object): void

+ clear (): void

+ popup (toPop: Collection<?>): void



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

Declaro por minha honra que este diagrama foi realizado apenas pelos elementos do grupo das classes chicken.