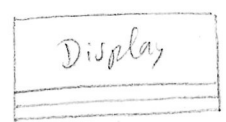
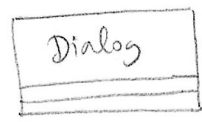
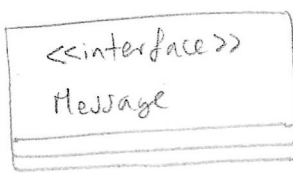
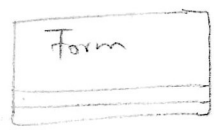
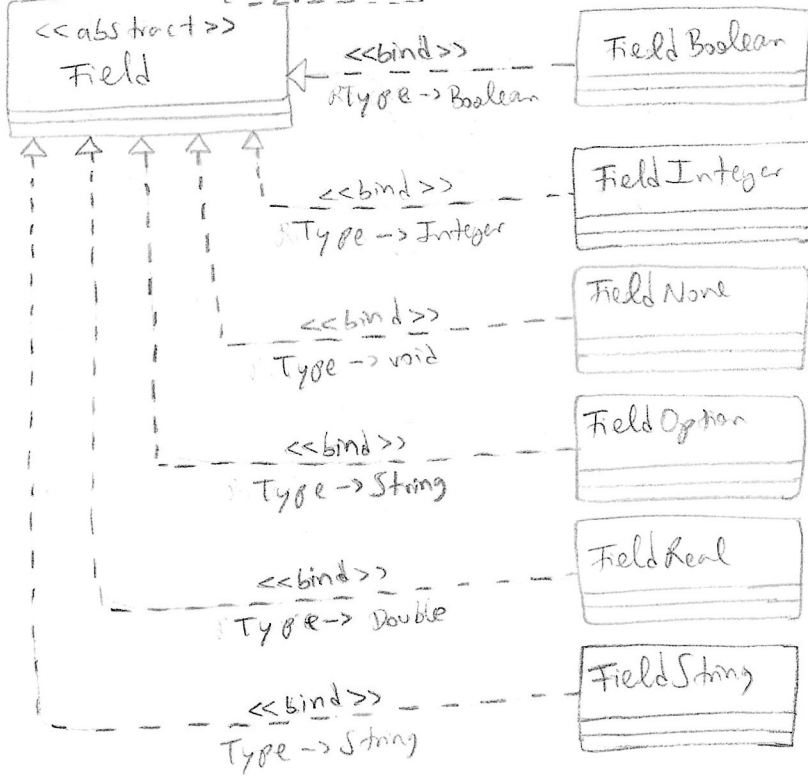
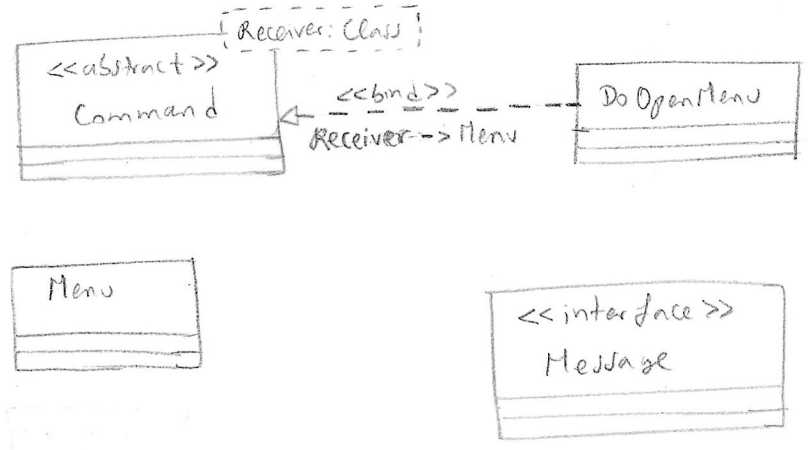


pt. tecnico. utilib. forms

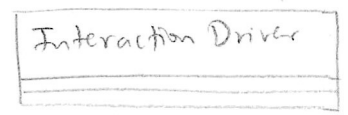
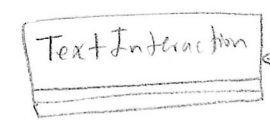
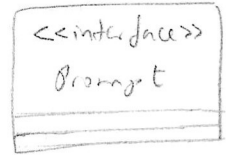
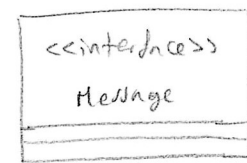
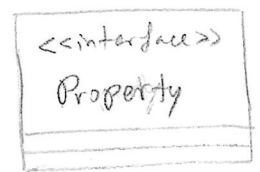
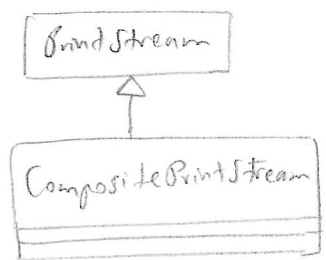
Type: Class



pt. tecnico. utilib. menus



pt. tecnico. utilib. text



opt. tecnico. visib. forms:

<<abstract>>

Field

- \_prompt : String  
- \_clear : boolean  
# \_value : Type

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

Field Boolean

- <<final>> BOOLEAN\_WORD\_YES : String  
- <<final>> BOOLEAN\_WORD\_NO : String  
- <<final>> BOOLEAN\_CHAR\_YES : String  
- <<final>> BOOLEAN\_CHAR\_NO : String

+ parse (in : String) : boolean

Field Integer

+ parse (in : String) : boolean

Field None

+ parse (in : String) : boolean  
+ isReadOnly() : boolean

Field Option

+ parse (in : String) : boolean

Field Real

+ parse (in : String) : boolean

Field String

+ parse (in : String) : boolean

Form

- \_ui : Dialog  
- \_title : String  
- fields : Map <String, Field <Type>>

+ get (Key : String, type : String) : Object  
+ title() : String  
+ entries() : Collection <Field <Type>>  
+ field (Key : String) : Field <Type>  
+ add (Key : String, in : Field <Type>) : void  
+ add BooleanField (Key : String, label : String) : void  
+ add StringField (Key : String, label : String) : void  
+ add IntegerField (Key : String, label : String) : void  
+ add OptionField (Key : String, label : String, options : String...) : void  
+ 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  
+ canForm (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) : String  
+ formNotFilled (form : String) : String  
+ KeyNotFound (form : String, Key : String) : String  
+ typeMismatch (form : String, Key : String, actualType : String, requestedType : String) : String

• pt.tecnico.vilib.menus:

<<abstract>>

Command

- \_last: boolean

- \_title: String

# <<final>> \_receiver: Receiver

# \_valid: Predicate <Receiver>

- <<final>> \_form = Form

# <<final>> \_display = Display

+ <<final>> title(): String

+ 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

+ <<final>> performCommand(): void

# <<abstract>> execute(): void

DoOpen Menu

+ <<final>> execute(): void

Menu

- \_ui: Dialog

- \_title: String

- \_commands: Command <?> [ ]

+ title(): String

+ size(): int

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

+ entries(): Command <?> [ ]

+ open(): void

<<interface>>

Message

+ operationFailed(error: String): String

opt. tecnico, vlib. text:

### ComparativePrintStream

- \_streams : Collection <PrintStream>

- \_error : boolean

~ << 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: float): 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

+ error EOF(e: EOFException): String

+ error IO(e: IOException): String

+ error InvalidNumber(e: NumberFormatException): String

+ error REOF(e: Runtime EOFException): String

+ inputError(e: FileNotFoundException): String

+ error ClosingInput(e: IOException): String

+ endOfInput(): String

+ outputError(e: FileNotFoundException): String

+ logError(e: FileNotFoundException): String

### TextInteraction

- \_in: BufferedReader

- \_out: PrintStream

- \_log: PrintStream

- \_writeInput: 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>> Prompt

+ option(): String

+ exit(): String

### <<interface>> Property

~ WRITE\_INPUT: String

~ BOTH\_CHANNELS: String

~ INPUT\_CHANNEL: String

~ OUTPUT\_CHANNEL: String

~ LOG\_CHANNEL: String

## Dialog

- <<final>> ACTION\_CHANNEL: String  
 - <<final>> CHANNEL\_SWING: String  
 - <<final>> CHANNEL\_NEW\_SWING: String  
 - <<final>> CHANNEL\_TEXT: String  
 - \_backend: InteractionDriver

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

## Display

- \_ui: Dialog  
 - title: String  
 - text: String Builder

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

## <<interface>>

### InteractionDriver

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

Declaro por minha honra que este diagrama  
 elementos que constituem o grupo de projeto.

Declaro por minha honra que este diagrama  
 elementos que constituem o grupo de projeto.

foi realizado apenas pelos

Francisco Alôido

foi realizado apenas pelos

Tiago Santos