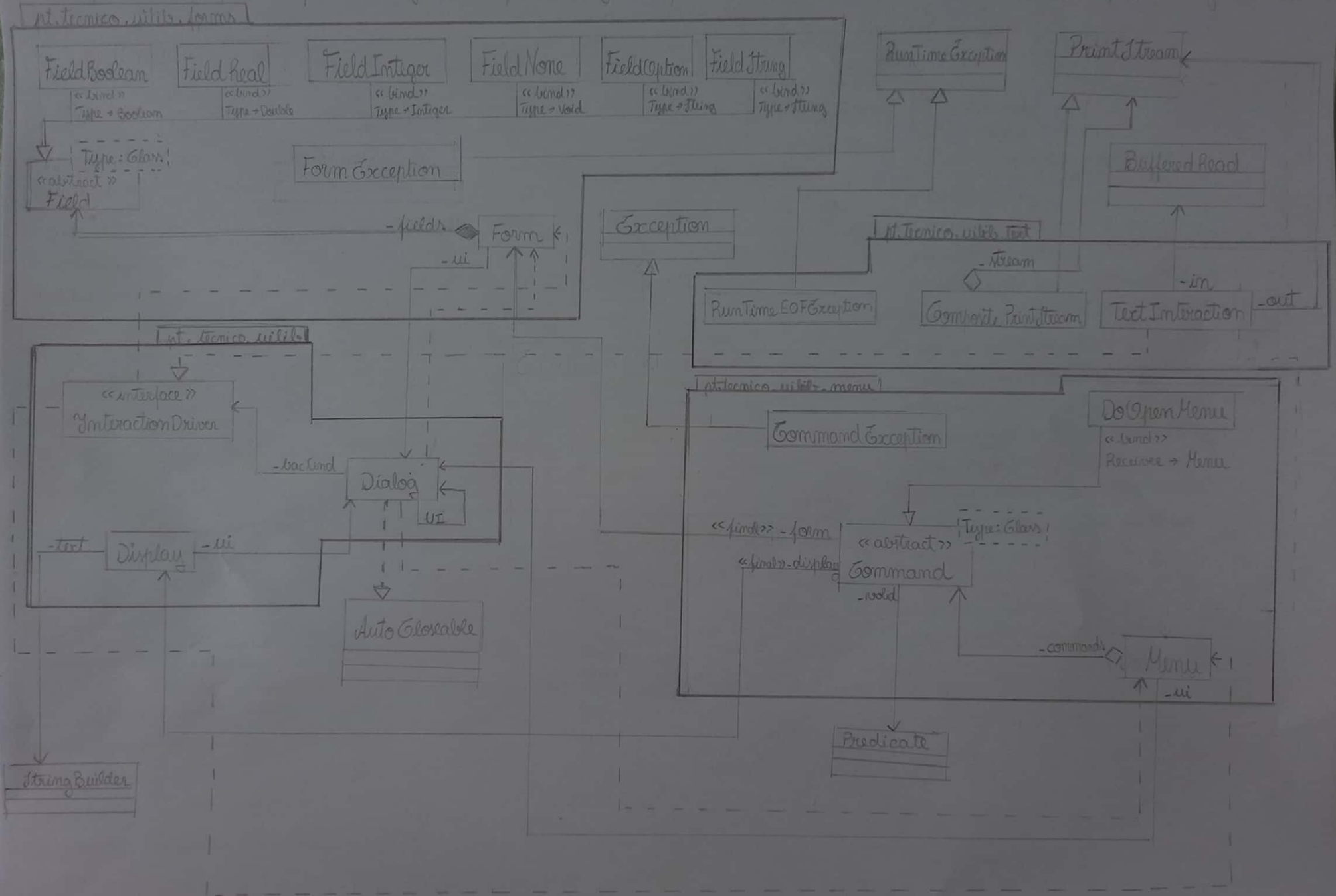


Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo do projeto. Raquel Braunschweig  
 Declaro por minha honra que este diagrama foi realizado apenas pelos elementos que constituem o grupo do projeto. Tiago Ferreira



«Abstract» Field	Type: Class
- _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 + toString(): String	

FieldString
+ parse(in: String): boolean
FieldReal
+ parse(in: String): boolean
FieldOption
- _options: String[] + parse(in: String): boolean
FieldNone
+ parse(in: String): boolean + isReadOnly(): boolean

FieldBoolean
- <<final>> BOOLEAN.WORD_YES: String - <<final>> BOOLEAN.WORD_NO: String - <<final>> BOOLEAN.CHAR_YES: String - <<final>> BOOLEAN.CHAR_NO: String + parse(in: String): boolean + toString(): String
FieldInteger
+ parse(in: String): String
DoOpenMenu
# <<final>> execute(): void

Form
- _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, 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(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

Display
- _ui: Dialog - _title: String - _text: StringBuilder
+ add(toAdd: Object): Display + addAll(items: Collection<?>): Display + addSize(toAdd: Object): Display + addNewSize(toAdd: Object, force: boolean): Display + display(): void + popup(toPop: Object): void + popup(toPop: Collection<?>): void + clear(): void

Classes 'Message', 'Prompt' and 'Properties' have been omitted since they are specific to the implementation and not relevant for the overall structure of the project.

FieldExecution
+ parse(in: String): String
FieldForm
+ parse(in: String): Form
FieldFormList
+ parse(in: String): List<Form>
FieldFormList
+ parse(in: String): List<Form>
FieldFormList
+ parse(in: String): List<Form>

Declaring a class in Java. The class is a blueprint for creating objects. It defines the attributes and methods of the objects. The class is declared using the keyword 'class' followed by the class name and a curly brace. The class can have attributes (variables) and methods (functions). The class is instantiated using the 'new' keyword followed by the class name and parentheses. The class is used to create objects of the class. The class is a blueprint for creating objects. It defines the attributes and methods of the objects. The class is declared using the keyword 'class' followed by the class name and a curly brace. The class can have attributes (variables) and methods (functions). The class is instantiated using the 'new' keyword followed by the class name and parentheses. The class is used to create objects of the class.

**Receiver Class**

**Command**

- last: boolean
- title: String
- # << final >> receiver: Receiver
- # - valid: Predicate <Receiver>
- << final >> form: Form
- # << final >> display: Display
- + << final >> title(): String
- + 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

**Dialog**

- << final >> ACTIONCHANNEL: String
- << final >> CHANNELSWING: String
- << final >> CHANNEL\_NEW\_SWING: String
- << final >> CHANNEL\_TEXT: String
- + UI: Dialog
- backend: InteractionDriver
- + open(menu: Menu): void
- + fill(form: Form): void
- + render(title: String, text: String): void
- + close: void

**CommandException**

- << final >> serialVersionUID: long
- + << final >> toString(): String

**Menu**

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

**Runtime EOF Exception**

- << final >> serialVersionUID: long

**FormException**

- << final >> serialVersionUID: long

**SwingMessage**

- << final >> serialVersionUID: long

**SwingMenu**

- << final >> serialVersionUID: long
- option: int
- + actionPerformed(event: ActionEvent): void

**SwingForm**

- << final >> serialVersionUID: long
- inputs: Map<Field<?>, JTextField>
- + have(): boolean

**SwingDialogMenu**

- << final >> serialVersionUID: long
- option: int
- menu: Menu
- out: JTextArea
- outputTitleLabel: JLabel
- buttonMenu: JButton[]
- selectedButton: JButton
- createOutputPanel(): JPanel
- + updateOutput(title: String, output: String): void
- createMenuPanel(): JPanel
- + actionPerformed(event: ActionEvent): void
- updateValidButtons(): void
- executeOption(option: int): void
- optionId: int
- clearAndHide(): void
- showErrorMessage(title: String, msg: String): void

**SwingDialogForm**

- << final >> RED\_BORDER: Border
- << final >> NO\_BORDER: Border
- << final >> serialVersionUID: long
- inputs: Map<Field<?>, JTextField>
- + createFormPanel(form: Form): JPanel
- + actionPerformed(event: ActionEvent): void
- have(): boolean

**InteractionDriver**

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

**Composite PrintStream**

- stream: 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(c: char): void
- + println(b: boolean): 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

~~JwingPanel~~

- <<final>> serialVersionUID : long
- # - end : boolean
- # <<final>> - lock : Object
- + actionPerformed (event : ActionEvent) : void
- sleep (mills : int) : void
- await () : void

~~JwingInteraction~~

- <<final>> serialVersionUID : long
- <<final>> MAIN\_TITLE : String
- + open (menu : Menu) : void
- + fill (form : Form) : void
- + render (title : String, text : String) : void
- + close () : void
- message (title : String, msg : String) : void

New JwingInteraction

- <<final>> serialVersionUID : long
- <<final>> MAIN\_TITLE : String
- currentDialogMenu : JwingDialogMenu
- + open (menu : Menu) : void
- + fill (form : Form) : void
- + render (title : String, text : String) : void
- + close () : void
- message (title : String, msg : String) : void

Test Interaction

- in : Buffered Reader
- out : Print Stream
- write Input : boolean
- + close () : void
- + open (menu : Menu) : void
- + fill (form : Form) : void
- + render (title : String, text : String) : void
- <<final>> readString (prompt : String) : String
- <<final>> readInteger (prompt : String) : int