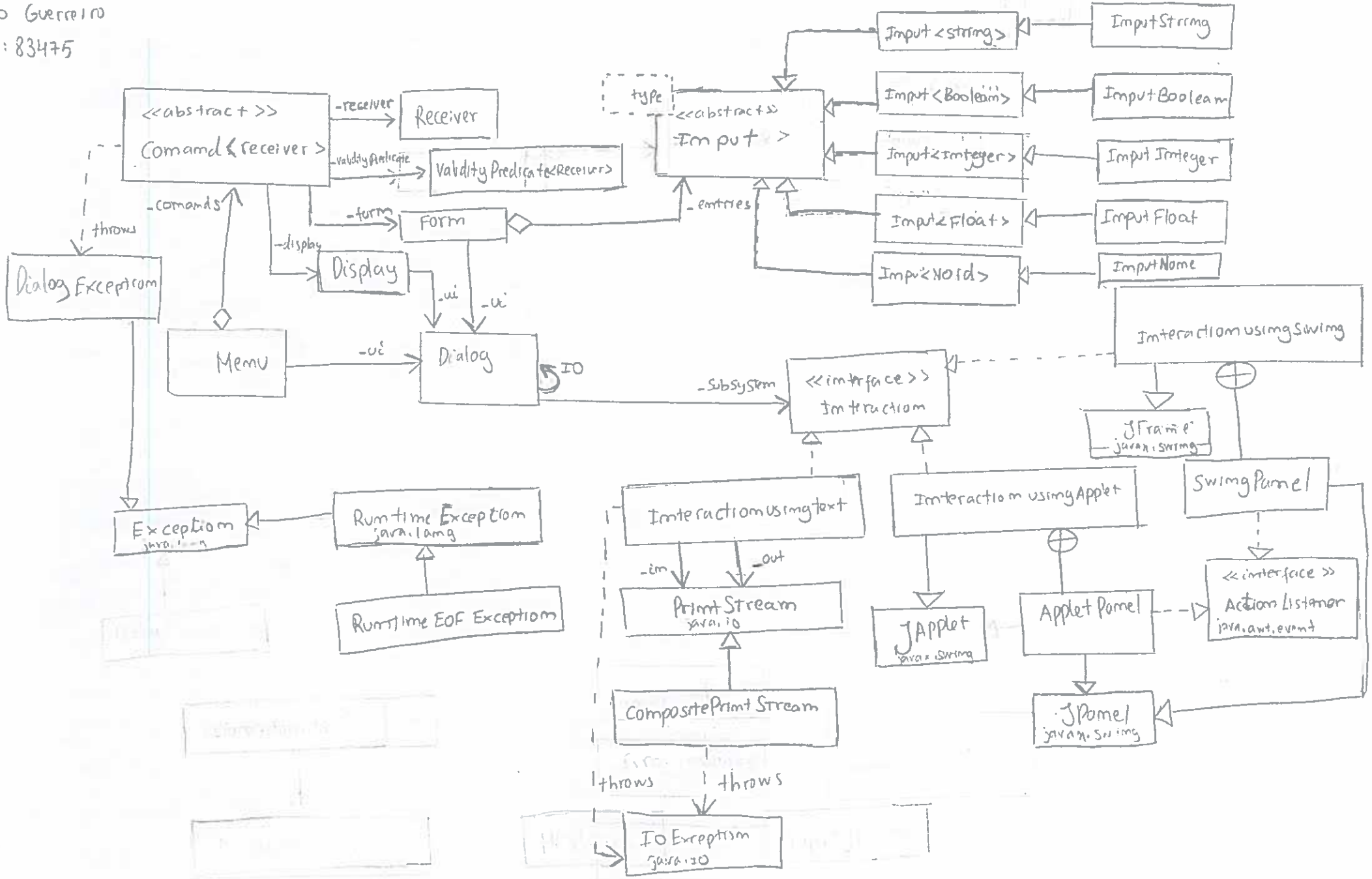
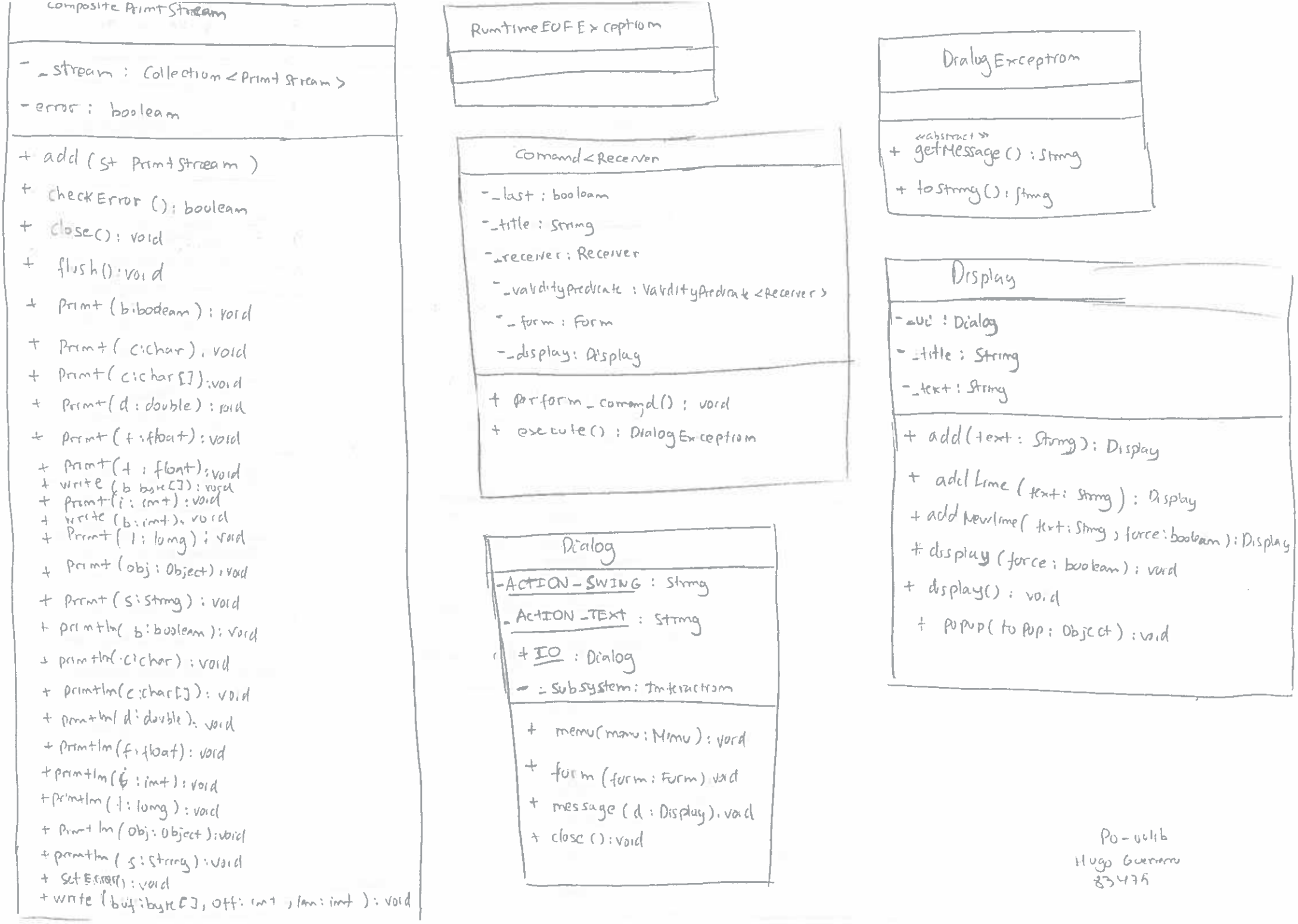
PO- UUIcb

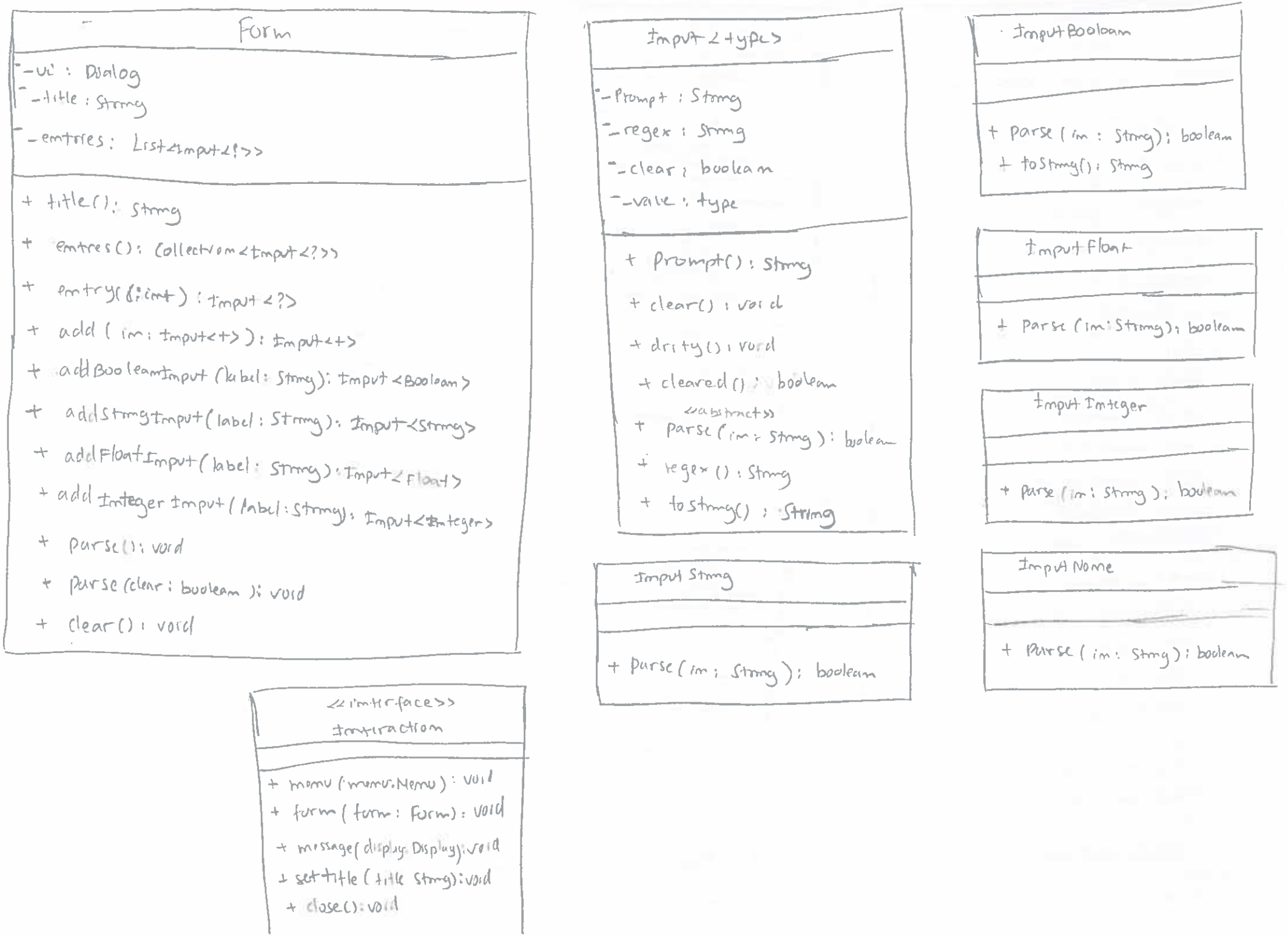
Hugo Guerreiro

nº : 83475

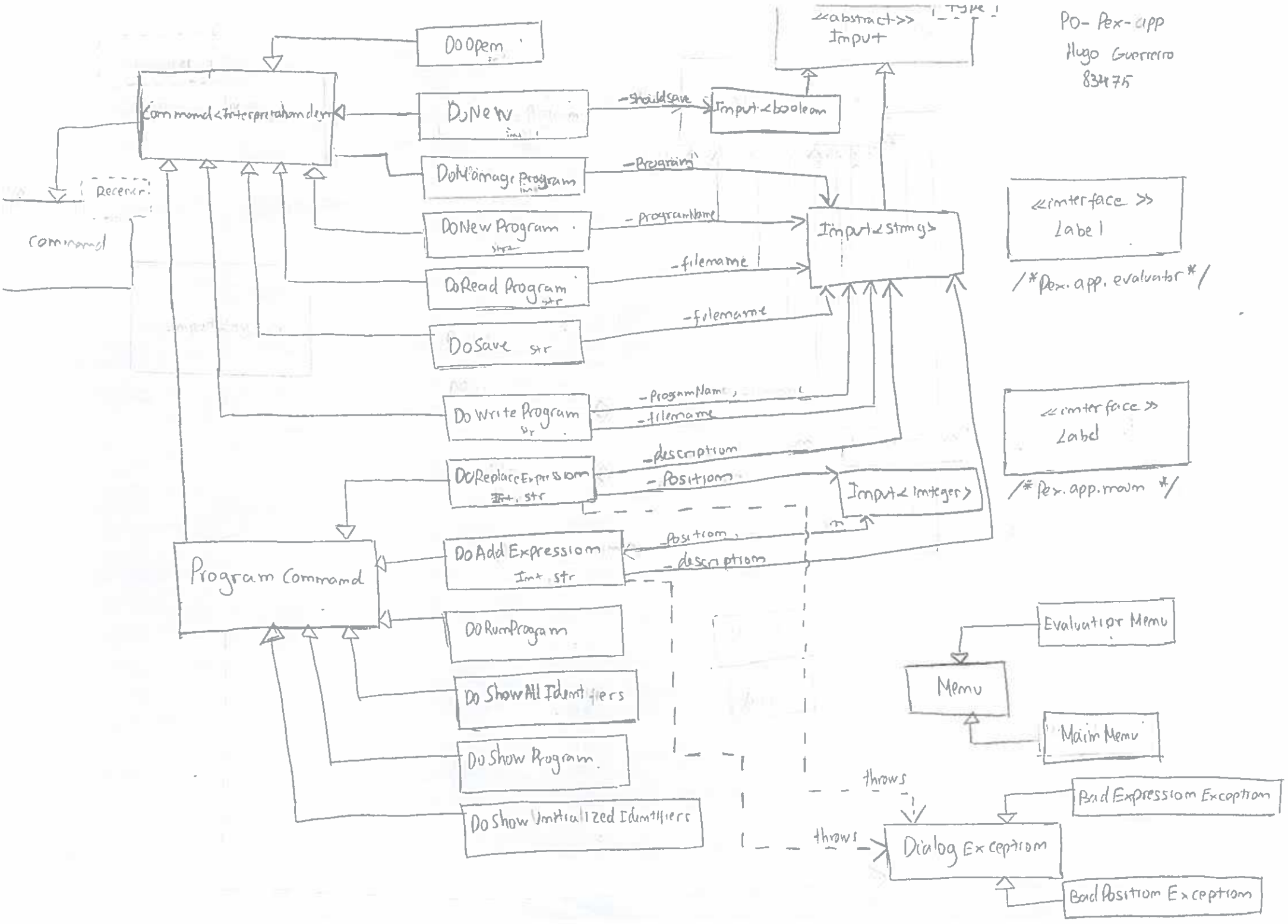


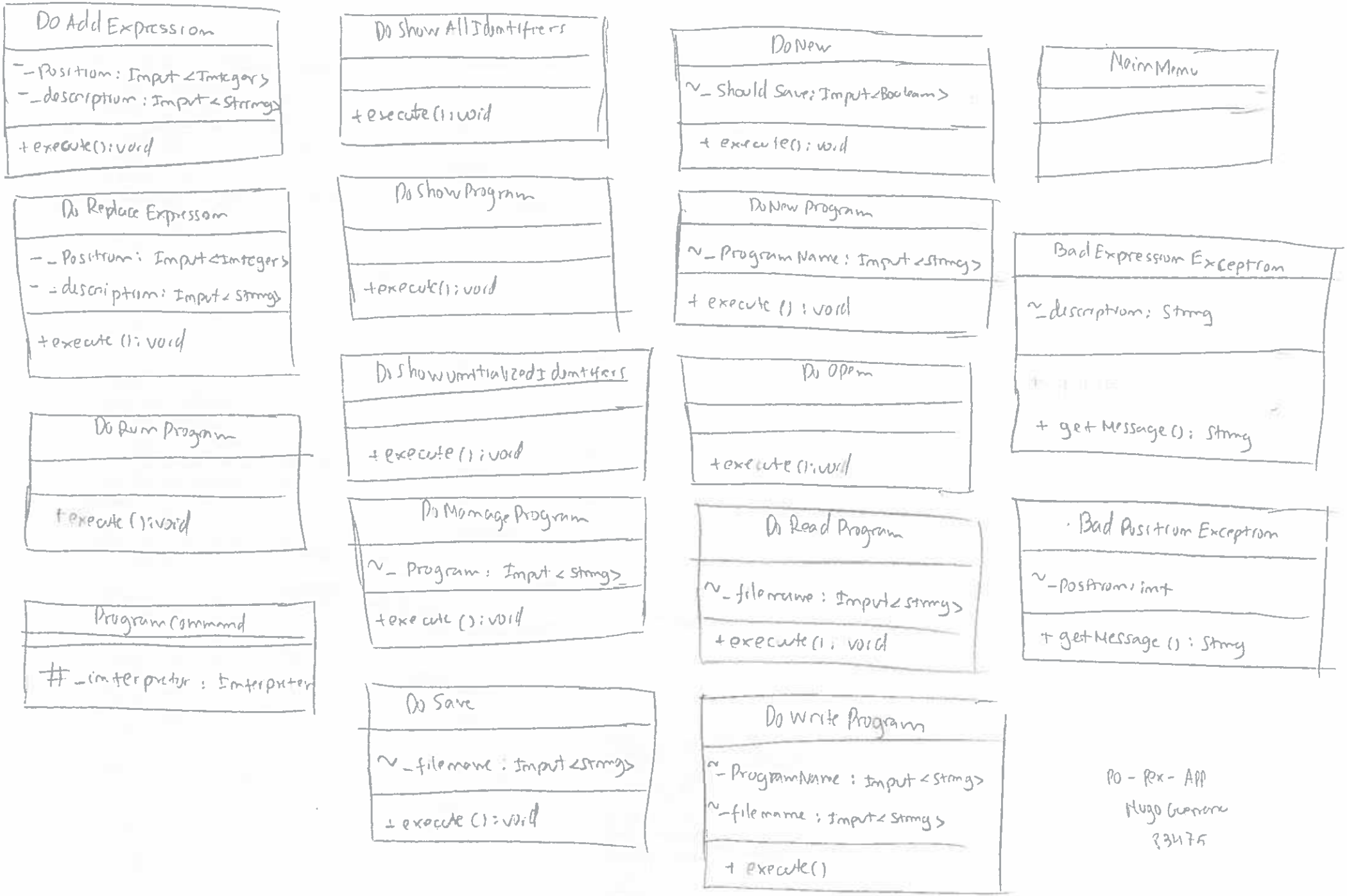


PO-uml
Hugo Guerrero
83475


Form
-ui : Dialog
-title : String
-entries : List<Input<?>>
+ title() : String
+ entries() : Collection<Input<?>>
+ entry(i:int) : Input<?>
+ add(in : Input<t>) : Input<t>
+ addBooleanInput(label : String) : Input<Boolean>
+ addStringInput(label : String) : Input<String>
+ addFloatInput(label : String) : Input<Float>
+ addIntegerInput(label : String) : Input<Integer>
+ parse() : void
+ parse(clear : boolean) : void
+ clear() : void
Input<type>
-Prompt : String
-regex : String
-clear : boolean
-value : type
+ prompt() : String
+ clear() : void
+ dirty() : void
+ cleared() : boolean
+ «abstract» parse(in : String) : boolean
+ regex() : String
+ toString() : String
Input Boolean
+ parse(in : String) : boolean
+ toString() : String
Input Float
+ parse(in : String) : boolean
Input Integer
+ parse(in : String) : boolean
Input String
+ parse(in : String) : boolean
Input Name
+ parse(in : String) : boolean
«interface»
Interaction
+ menu(menu : Menu) : void
+ form(form : Form) : void
+ message(display : Display) : void
+ setTitle(title : String) : void
+ close() : void


Interaction Using Applet
- menu(): void
+ init(): void
+ close(): void
+ setTitle(title: String): void
+ menu(m: Menu): void
+ form(f: Form): void
+ message(d: Display): void
+ message(s: String, title: String): void
AppletPanel
- _opt: int
- _end: boolean
- _inputs: Map<Input<?>, JTextField>
+ actionPerformed(evt: ActionEvent): void
- sleep(m: int): void
- await(): void
- optNum(): int
- parse(): boolean
Interaction Using Swing
+ close(): void
+ menu(m: Menu): void
+ form(f: Form): void
+ message(d: Display): void
+ message(s: String, title: String)
SwingPanel
- _opt: int
- _end: boolean
- _ins: Map<Input<?>, JTextField>
- _lock: Object
+ actionPerformed(evt: ActionEvent)
- sleep(m: int): void
- await(): void
- optNum(): int
- parse(): boolean
InteractionUsingText
- _in: BufferedReader
- _out: PrintStream
- _log: PrintStream
- _writeInput: boolean
+ closeDown(): void
+ menu(m: Menu): void
+ form(form: Form): void
+ message(display: Display)
+ setTitle(title: String): void
+ close(): void
- readInteger(prompt: String): int
- readString(prompt: String): String
- println(text: String): void
- print(text: String): void
Menu
- _title: String
- _commands: Comand<?>[]
- _ui: Dialog
+ title(): String
+ size(): int
+ entry(n: int): Command<?>
+ entries(): Command<?>[]
+ open(): void
ValidityPredicate<Receiver>
~ _receiver: Receiver
+ isValid(): boolean
B-valib
Hugo Guerra
83475

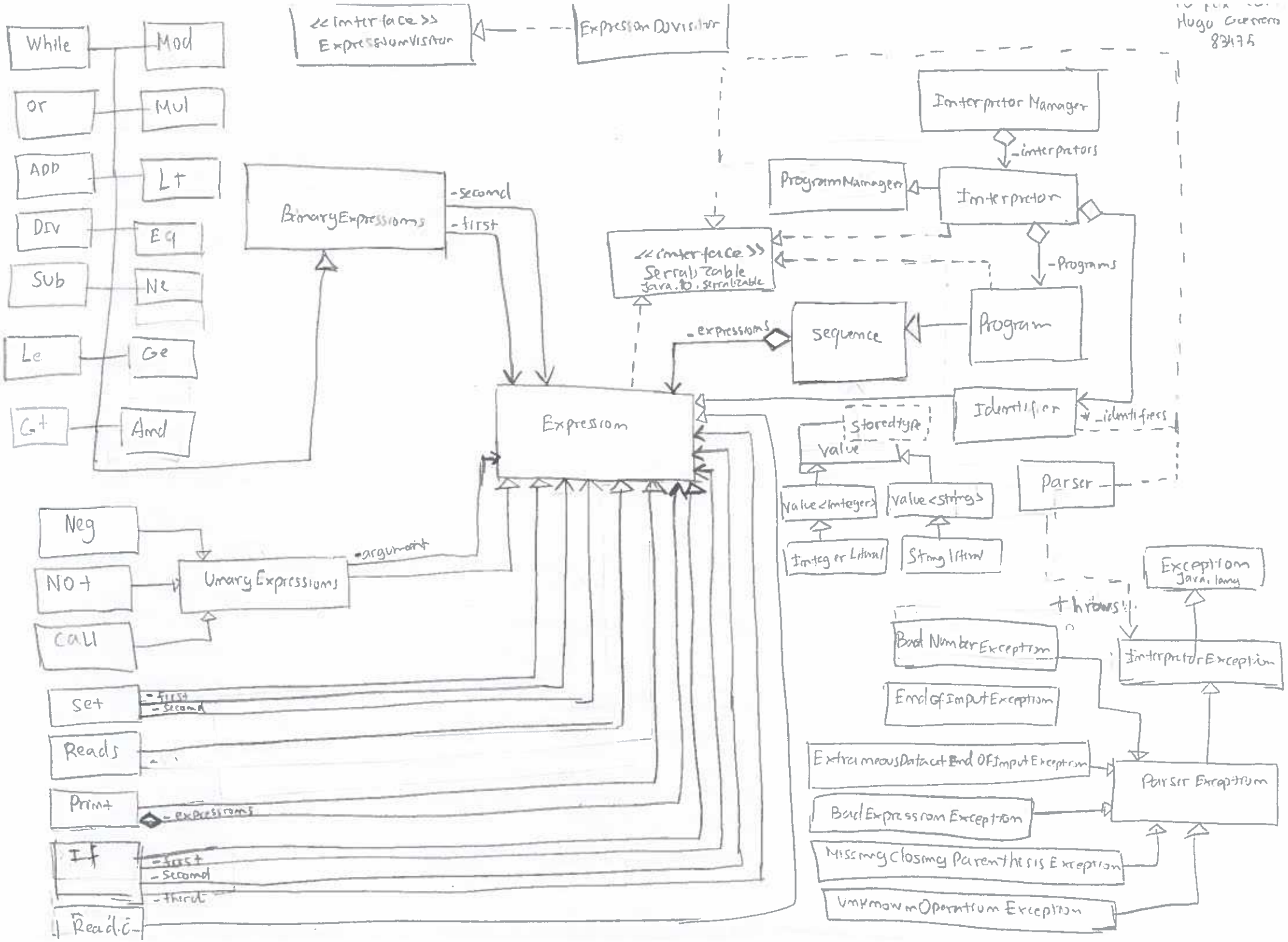

PO- Pex-app
Hugo Guerrero
83475
<<abstract>>
Input
Type
DoOpen
Command<Interpretahandler>
DoNew
-shouldSave
Input<boolean
Receiver
command
DoManage Program
-Programs
DoNew Program
-programName
Input<String>
<<interface>>
Label
/*Pex.app.evaluator*/
DoRead Program
-filename
DoSave str
-filename
DoWrite Program
-programName
-filename
<<interface>>
Label
/*Pex.app.main*/
DoReplaceExpression
-description
-position
Input<Integer>
Program Command
DoAddExpression
-position
-description
DoRunProgram
DoShowAllIdentifiers
DoShowProgram
DoShowUnitialized Identifiers
Evaluator Menu
Menu
Main Menu
throws
BadExpressionException
throws
DialogException
BadPositionException

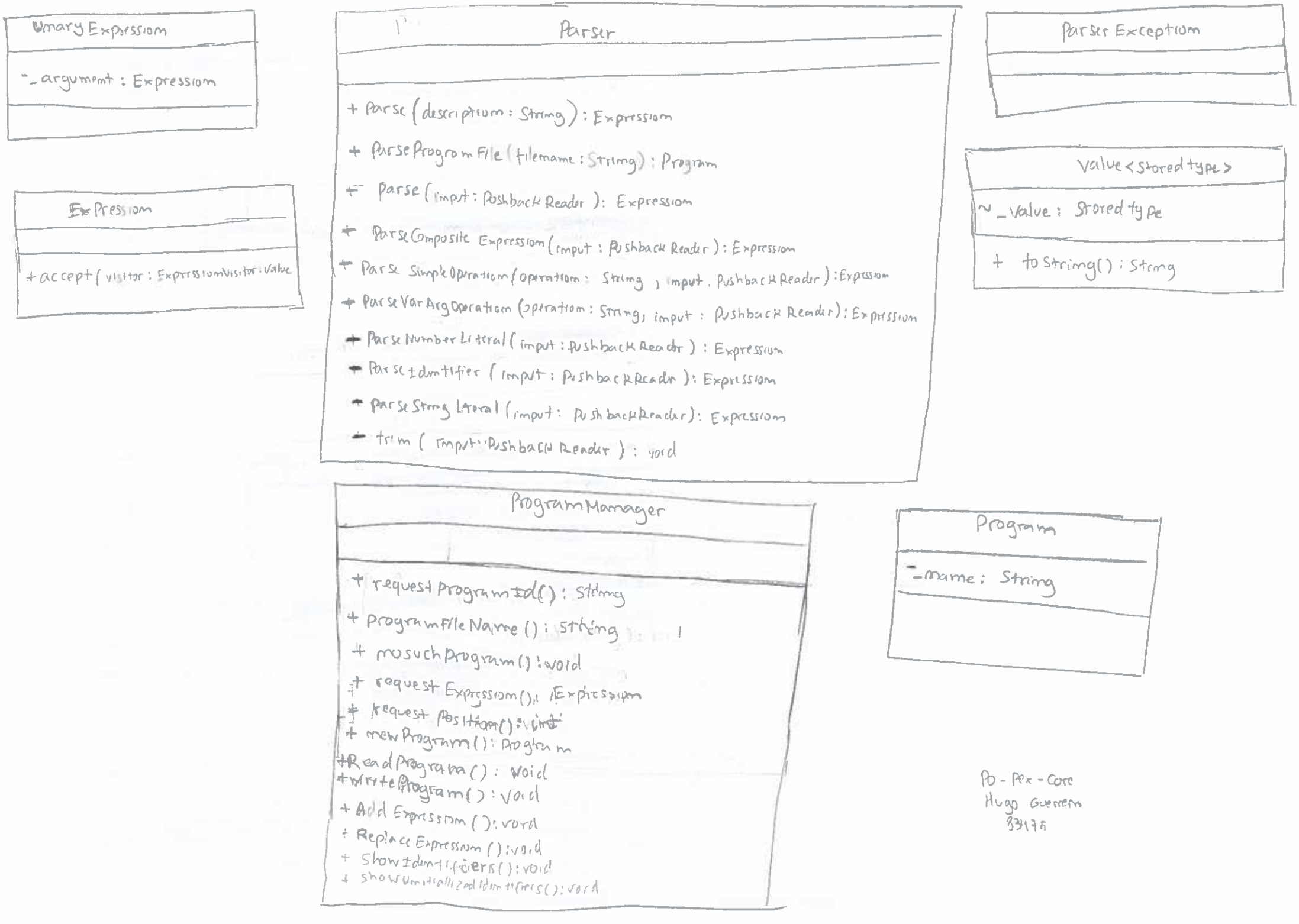


PO - Rex - APP

Hugo Guerrero

73475


Hugo Guerrero
83175
While
Mod
or
Mul
ADD
Lt
Div
Eq
Sub
Ne
Le
Ge
Gt
And
<<Interface>>
ExpressionVisitor
ExpressionDVisitor
BinaryExpressions
-second
-first
Neg
NOt
CALL
UnaryExpressions
-argument
Set
Reads
Print
-expressions
If
-first
-second
-third
Read.c
Interpretor Manager
-interpretors
ProgramManager
Interpretor
-Programs
<<Interface>>
Serializable
java.io.Serializable
Sequence
Program
-expressions
Expression
Identifier
-identifiers
StoredType
Value
Value<Integer>
Value<String>
IntegerLitral
StringLitral
Parser
throws
Exception
java.lang
BadNumberException
EndOfInputException
InterpretorException
ExtraneousDataAtEndOfInputException
BadExpressionException
MissingClosingParenthesisException
UnknownOperationException
ParserException


UnaryExpression
-_argument : Expression
Parser
+ Parse(description : String) : Expression
+ ParseProgramFile(filename : String) : Program
+ parse(input : PushbackReader) : Expression
+ ParseCompositeExpression(input : PushbackReader) : Expression
+ ParseSimpleOperation(operation : String, input, PushbackReader) : Expression
+ ParseVarArgOperation(operation : String, input : PushbackReader) : Expression
+ ParseNumberLiteral(input : PushbackReader) : Expression
+ ParseIdentifier(input : PushbackReader) : Expression
+ ParseStringLiteral(input : PushbackReader) : Expression
+ trim(input : PushbackReader) : void
ParserException
Value<storedtype>
~_value : storedtype
+ toString() : String
Expression
+ accept(visitor : ExpressionVisitor.Value
ProgramManager
+ requestProgramId() : String
+ programFileName() : String
+ noSuchProgram() : void
+ requestExpression() : IExpression
+ requestPosition() : int
+ newProgram() : Program
+ ReadPrograma() : void
+ WriteProgram() : void
+ Add Expression() : void
+ Replace Expression() : void
+ ShowIdentifiers() : void
+ showUninitializedIdentifiers() : void
Program
-_name : String
PO - Pex - Core
Hugo Guerrero
33175

<<interface>>
ExpressionVisitor

+ visit(eq: Eq): Value
+ visit(ne: Ne): Value
+ visit(le: Le): Value
+ visit(ge: Ge): Value
+ visit(gt: Gt): Value
+ visit(and: And): Value
+ visit(seq: Sequence): Value
+ visit(print: Print): Value
+ visit(neg: Neg): Value
+ visit(not: Not): Value
+ visit(call: Call): Value
+ visit(if: If): Value
+ visit(read: Read): Value
+ visit(read: Reads): Value
+ visit(while: While): Value
+ visit(set: Set): Value
+ visit(or: Or): Value
+ visit(add: Add): Value
+ visit(div: Div): Value
+ visit(sub: Sub): Value
+ visit(mod: Mod): Value
+ visit(mul: Mul): Value
+ visit(lt: Lt): Value
+ visit(exp: Expression): Value

ExpressionDoVisitor

+ visit(eq: Eq): value
+ visit(ne: Ne): value
+ visit(le: Le): value
+ visit(ge: Ge): value
+ visit(gt: Gt): value
+ visit(and: And): value
+ visit(seq: Sequence): value
+ visit(print: Print): value
+ visit(neg: Neg): value
+ visit(not: Not): value
+ visit(call: Call): value
+ visit(if: If): value
+ visit(read: Read): value
+ visit(read: Reads): value
+ visit(while: While): value
+ visit(set: Set): value
+ visit(or: Or): value
+ visit(add: Add): value
+ visit(div: Div): value
+ visit(sub: Sub): value
+ visit(mod: Mod): value
+ visit(mul: Mul): value
+ visit(lt: Lt): value
+ visit(exp: Expression): value

Interpretor

- _Programs: Hashmap<String, Program>
- _ProgramFile: String
- _identifiers: Hashmap<String, values>

+ evaluateExpression(expression: Expression): Value
+ interpreteExpressions(expressions: Expression): value
+ SaveProgram(filename: String): void
+ SaveProgram(): void
+ interpreteExpressions(filename: String): value
+ evaluateExpressions(filename: String): value

InterpretorManager

- _Interpretors: Hashmap<String, Interpretor>

+ OpenInterpretor(): void
+ SaveInterpretorState(): void
+ NewInterpretor(): void
+ exit(): void
+ RestartInterpretor(filename: String): void
+ RemoveInterpretor(filename: String): void

Po Pex-ros
Hugo Guerrero
83475


Identifier
-name: String
+accept(visitor: ExpressionVisitor): V
IntegerLitera)
+accept (visitor : ExpressionVisitor):
StringLitera)
+accept(visitor: ExpressionVisitor):
BadExpressionException
-description: String
BadNumberException
-description: String
EndOfInputException
ExtraneousDataAtEndOfInputException
-c: int
InterpreterException
MissingCloseingParanthesisExeption
UnKnownOperationExeption
-discription: String
Or
+accept (visitor : ExpressionVisitor): Value
Add
+accept(visitor: ExpressionVisitor): Value
Div
+accept(visitor: ExpressionVisitor): Value
Sub
+accept(visitor: ExpressionVisitor): Value
Mod
+accept(visitor: ExpressionVisitor): Value
Mul
+accept(visitor: ExpressionVisitor): Value
Lt
+accept(visitor: ExpressionVisitor): Value