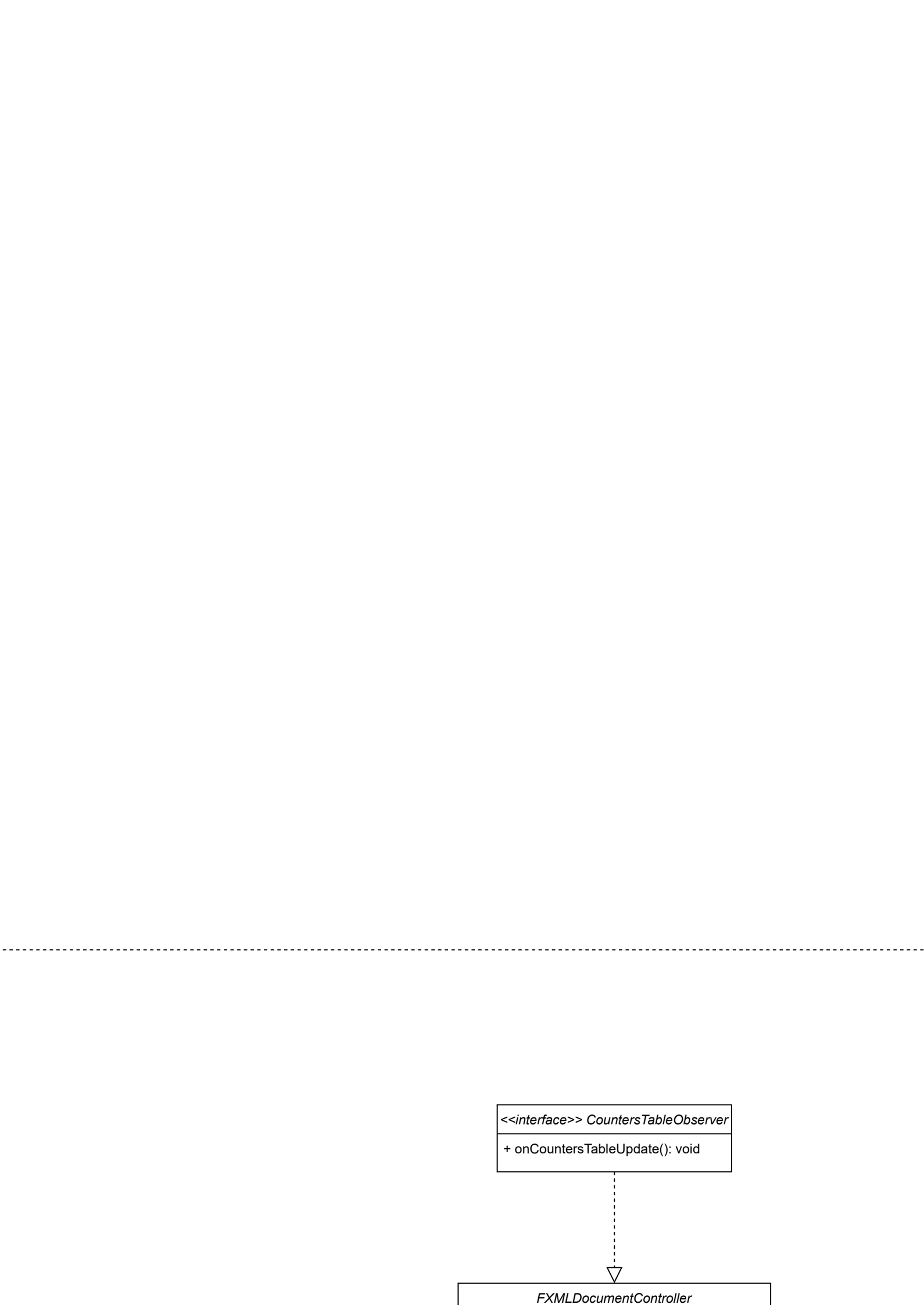
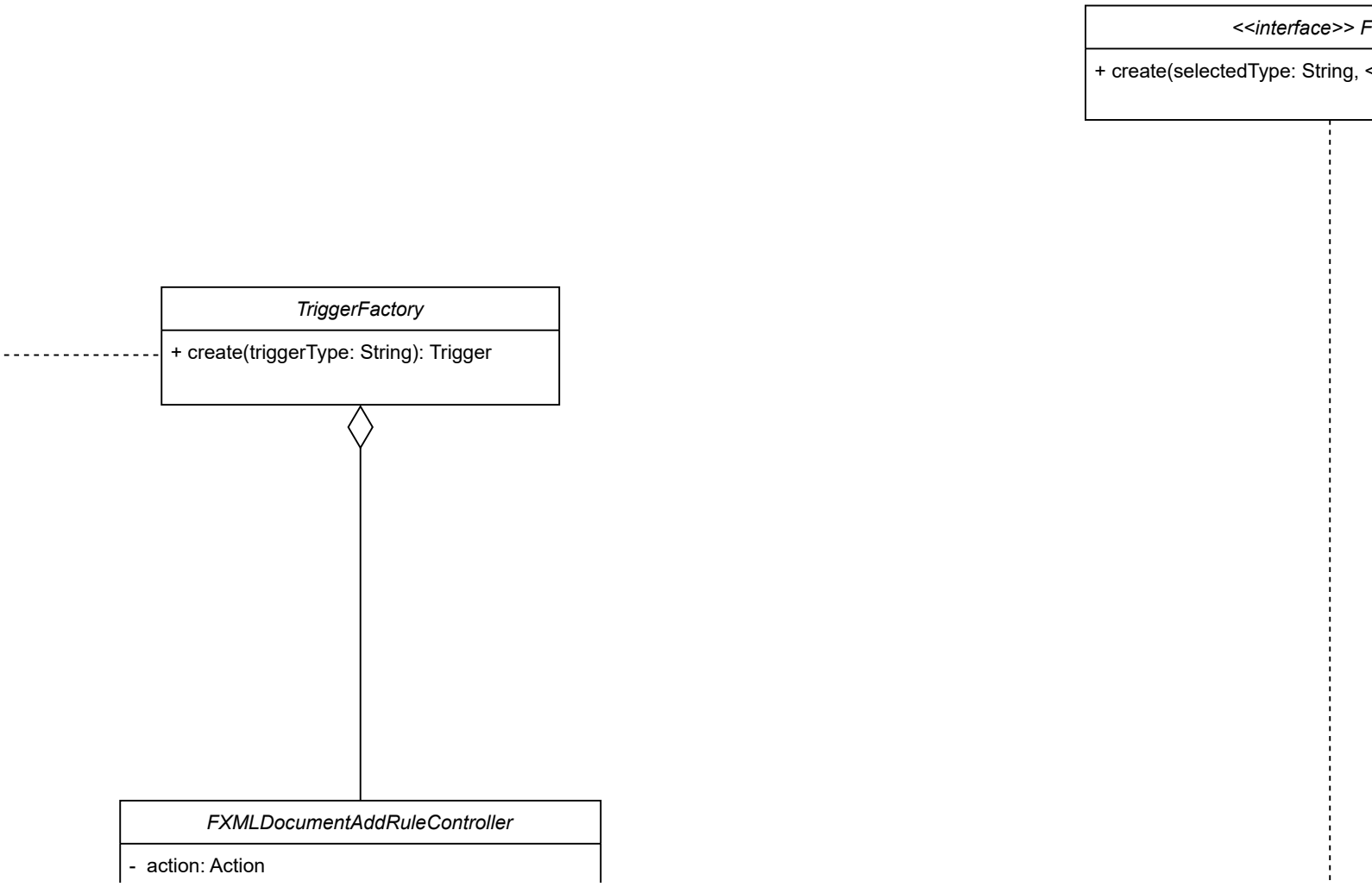
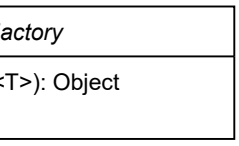
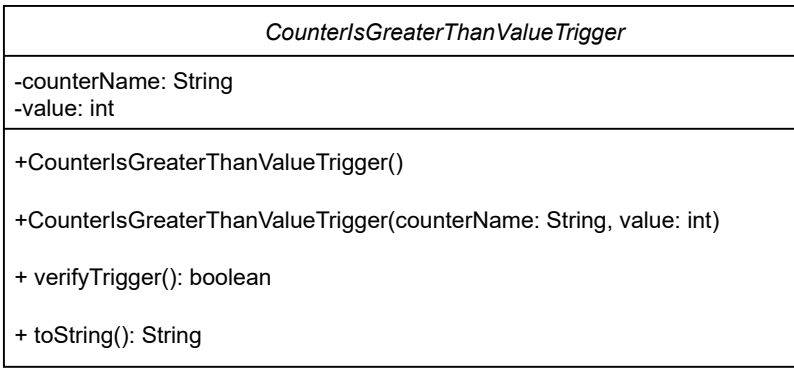
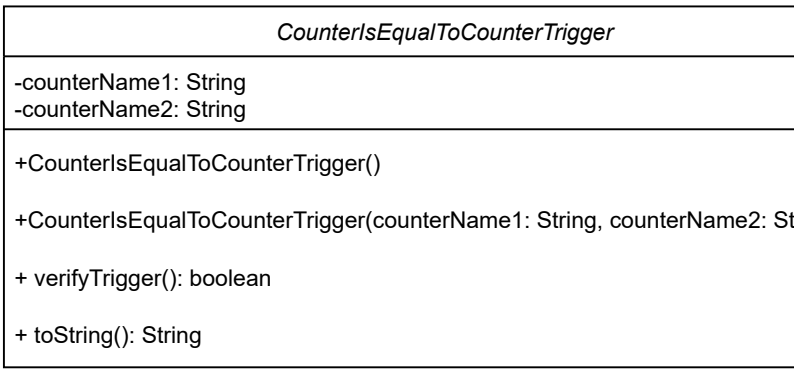
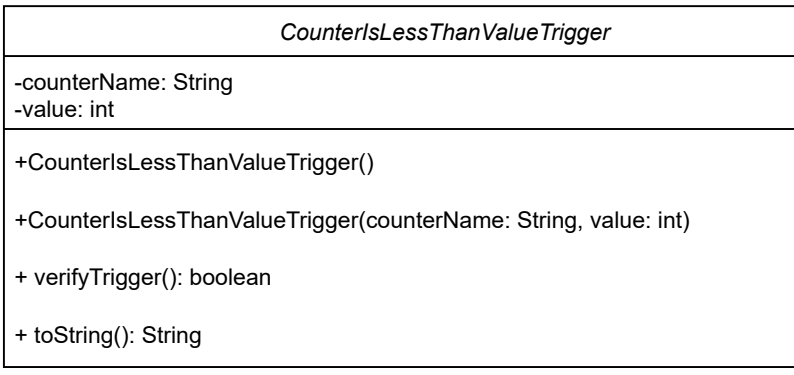

<<interface>> CountersTableObserver
+ onCountersTableUpdate(): void
FXMLDocumentController

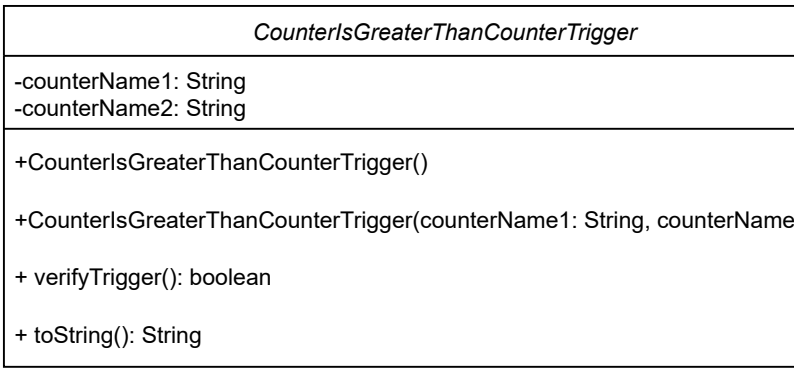

TriggerFactory
+ create(triggerType: String): Trigger
FXMLDocumentAddRuleController
- action: Action

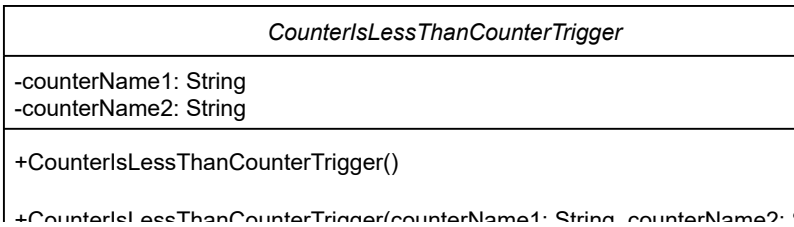

actory
<T>): Object

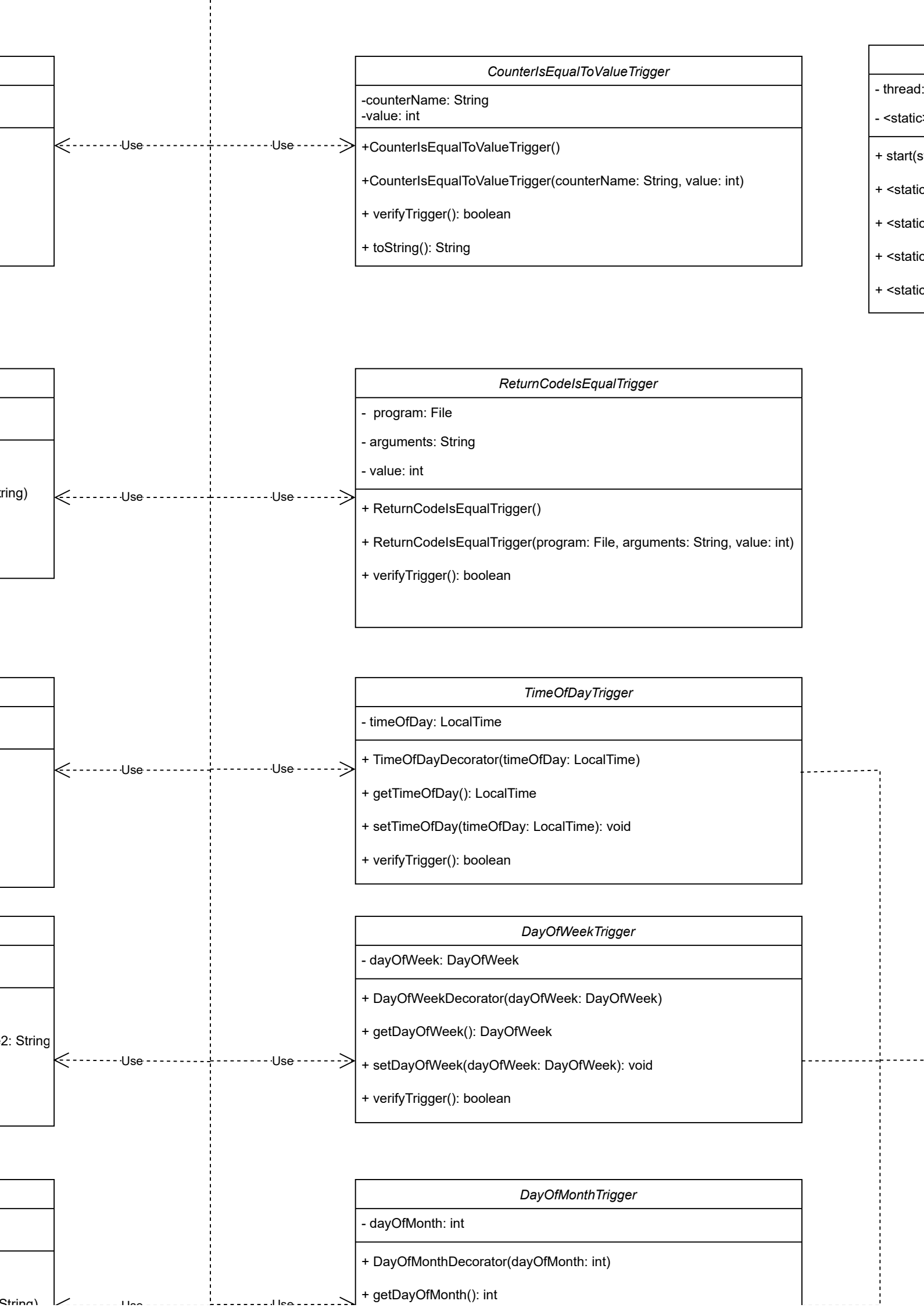

CounterIsGreaterThanValueTrigger
-counterName: String
-value: int
+CounterIsGreaterThanValueTrigger()
+CounterIsGreaterThanValueTrigger(counterName: String, value: int)
+ verifyTrigger(): boolean
+ toString(): String

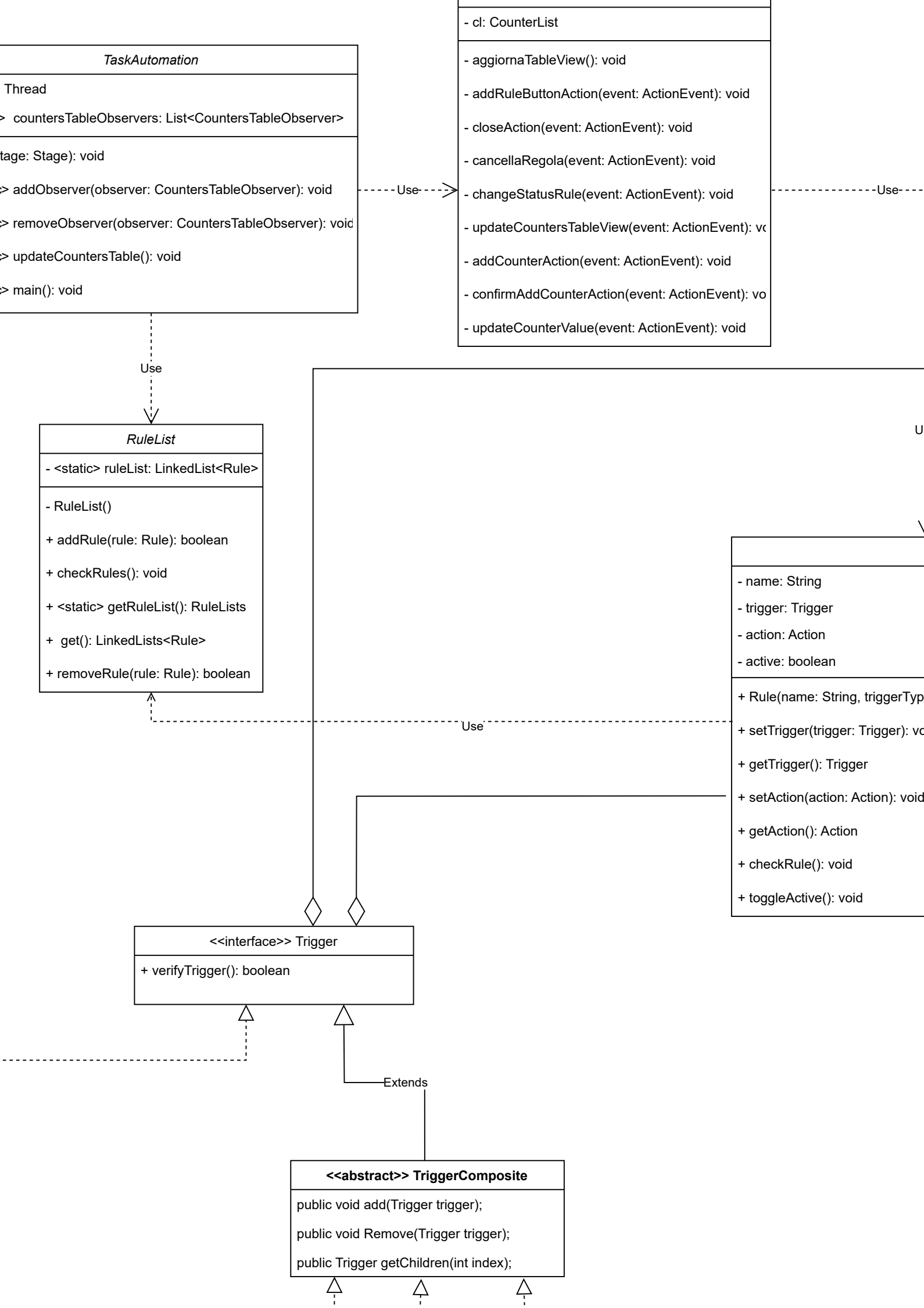

CounterIsEqualToCounterTrigger
-counterName1: String
-counterName2: String
+CounterIsEqualToCounterTrigger()
+CounterIsEqualToCounterTrigger(counterName1: String, counterName2: St
+ verifyTrigger(): boolean
+ toString(): String

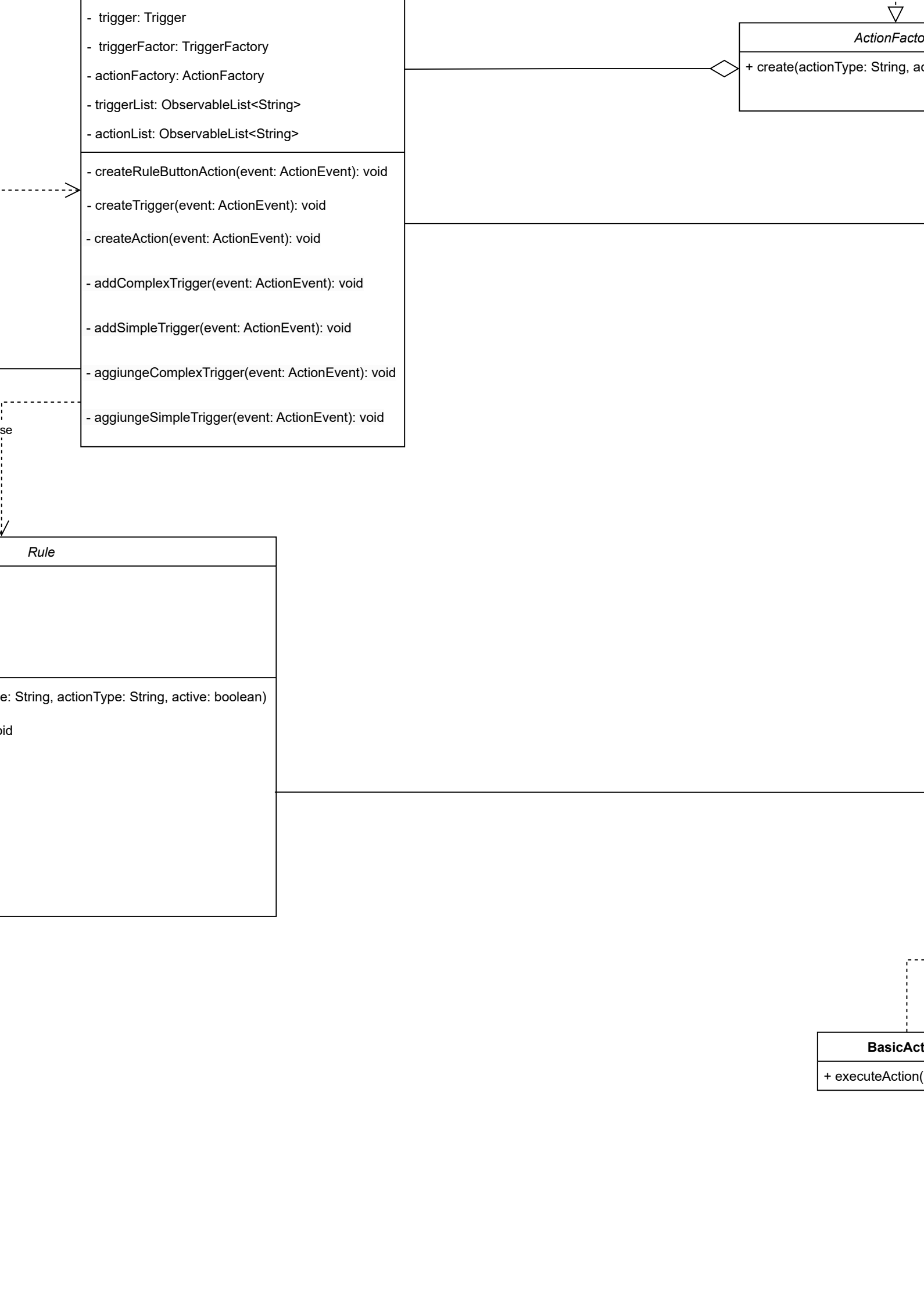

CounterIsLessThanValueTrigger
-counterName: String
-value: int
+CounterIsLessThanValueTrigger()
+CounterIsLessThanValueTrigger(counterName: String, value: int)
+ verifyTrigger(): boolean
+ toString(): String

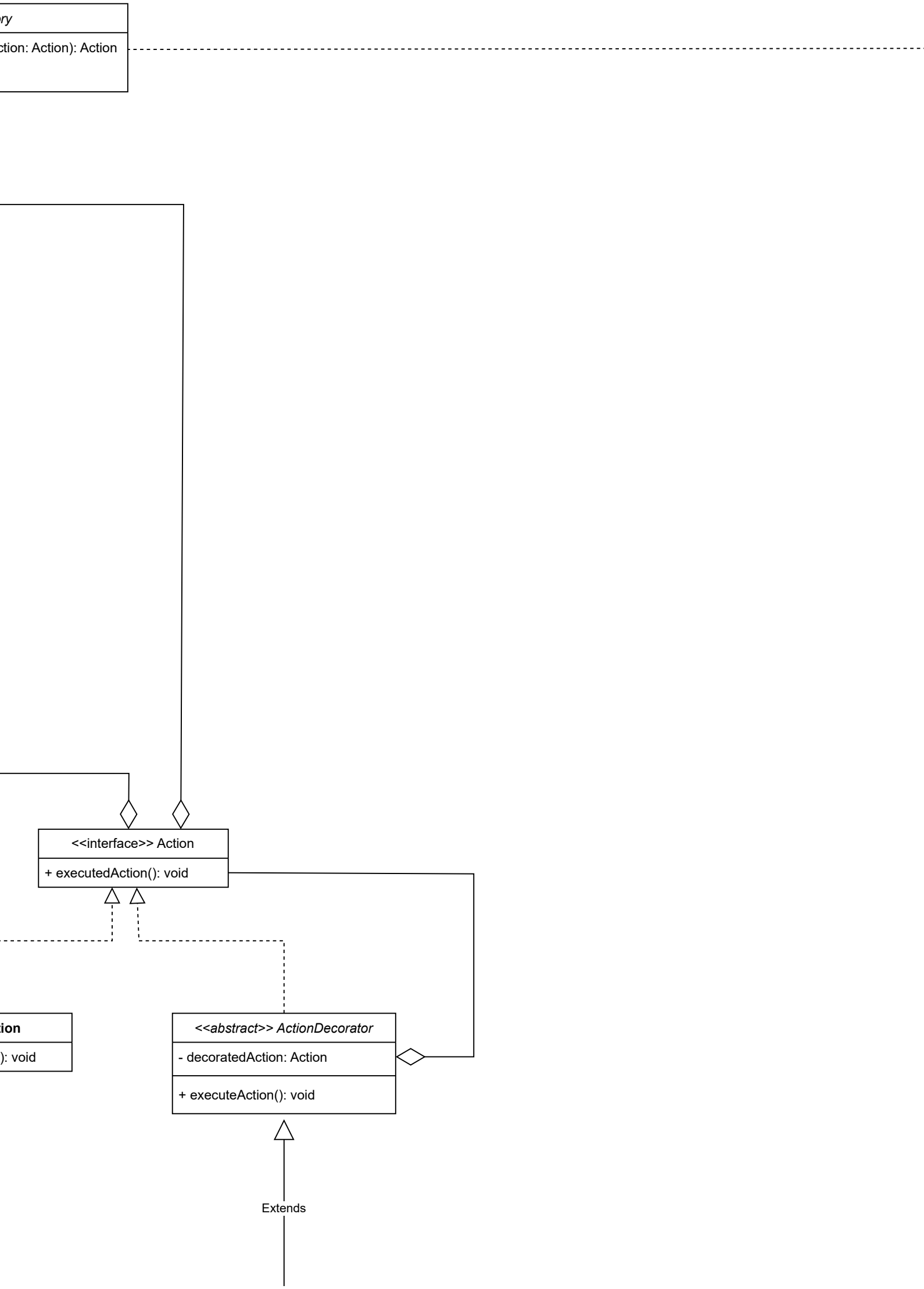

CounterIsGreaterThanCounterTrigger
-counterName1: String
-counterName2: String
+CounterIsGreaterThanCounterTrigger()
+CounterIsGreaterThanCounterTrigger(counterName1: String, counterName
+ verifyTrigger(): boolean
+ toString(): String


CounterIsLessThanCounterTrigger
-counterName1: String
-counterName2: String
+CounterIsLessThanCounterTrigger()


CounterIsEqualToValueTrigger
-counterName: String
-value: int
+CounterIsEqualToValueTrigger()
+CounterIsEqualToValueTrigger(counterName: String, value: int)
+ verifyTrigger(): boolean
+ toString(): String
Use
Use
ReturnCodeIsEqualTrigger
- program: File
- arguments: String
- value: int
+ ReturnCodeIsEqualTrigger()
+ ReturnCodeIsEqualTrigger(program: File, arguments: String, value: int)
+ verifyTrigger(): boolean
Use
Use
TimeOfDayTrigger
- timeOfDay: LocalTime
+ TimeOfDayDecorator(timeOfDay: LocalTime)
+ getTimeOfDay(): LocalTime
+ setTimeOfDay(timeOfDay: LocalTime): void
+ verifyTrigger(): boolean
Use
Use
DayOfWeekTrigger
- dayOfWeek: DayOfWeek
+ DayOfWeekDecorator(dayOfWeek: DayOfWeek)
+ getDayOfWeek(): DayOfWeek
+ setDayOfWeek(dayOfWeek: DayOfWeek): void
+ verifyTrigger(): boolean
Use
Use
DayOfMonthTrigger
- dayOfMonth: int
+ DayOfMonthDecorator(dayOfMonth: int)
+ getDayOfMonth(): int
Use
Use


- cl: CounterList
- aggiornaTableView(): void
- addRuleButtonAction(event: ActionEvent): void
- closeAction(event: ActionEvent): void
- cancellaRegola(event: ActionEvent): void
- changeStatusRule(event: ActionEvent): void
- updateCountersTableView(event: ActionEvent): vo
- addCounterAction(event: ActionEvent): void
- confirmAddCounterAction(event: ActionEvent): vo
- updateCounterValue(event: ActionEvent): void
Use
Use
TaskAutomation
Thread
countersTableObservers: List<CountersTableObserver>
tage: Stage): void
addObserver(observer: CountersTableObserver): void
removeObserver(observer: CountersTableObserver): void
updateCountersTable(): void
main(): void
Use
RuleList
- <static> ruleList: LinkedList<Rule>
- RuleList()
+ addRule(rule: Rule): boolean
+ checkRules(): void
+ <static> getRuleList(): RuleLists
+ get(): LinkedLists<Rule>
+ removeRule(rule: Rule): boolean
Use
- name: String
- trigger: Trigger
- action: Action
- active: boolean
+ Rule(name: String, triggerTyp
+ setTrigger(trigger: Trigger): vo
+ getTrigger(): Trigger
+ setAction(action: Action): void
+ getAction(): Action
+ checkRule(): void
+ toggleActive(): void
<<interface>> Trigger
+ verifyTrigger(): boolean
Extends
<<abstract>> TriggerComposite
public void add(Trigger trigger);
public void Remove(Trigger trigger);
public Trigger getChildren(int index);


- trigger: Trigger
- triggerFactor: TriggerFactory
- actionFactory: ActionFactory
- triggerList: ObservableList<String>
- actionList: ObservableList<String>
- createRuleButtonAction(event: ActionEvent): void
- createTrigger(event: ActionEvent): void
- createAction(event: ActionEvent): void
- addComplexTrigger(event: ActionEvent): void
- addSimpleTrigger(event: ActionEvent): void
- aggiungeComplexTrigger(event: ActionEvent): void
- aggiungeSimpleTrigger(event: ActionEvent): void
ActionFacto
+ create(actionType: String, ac
se
Rule
e: String, actionType: String, active: boolean)
oid
BasicAct
+ executeAction(


ry
ction: Action): Action
<<interface>> Action
+ executedAction(): void
tion
): void
<<abstract>> ActionDecorator
- decoratedAction: Action
+ executeAction(): void
Extends

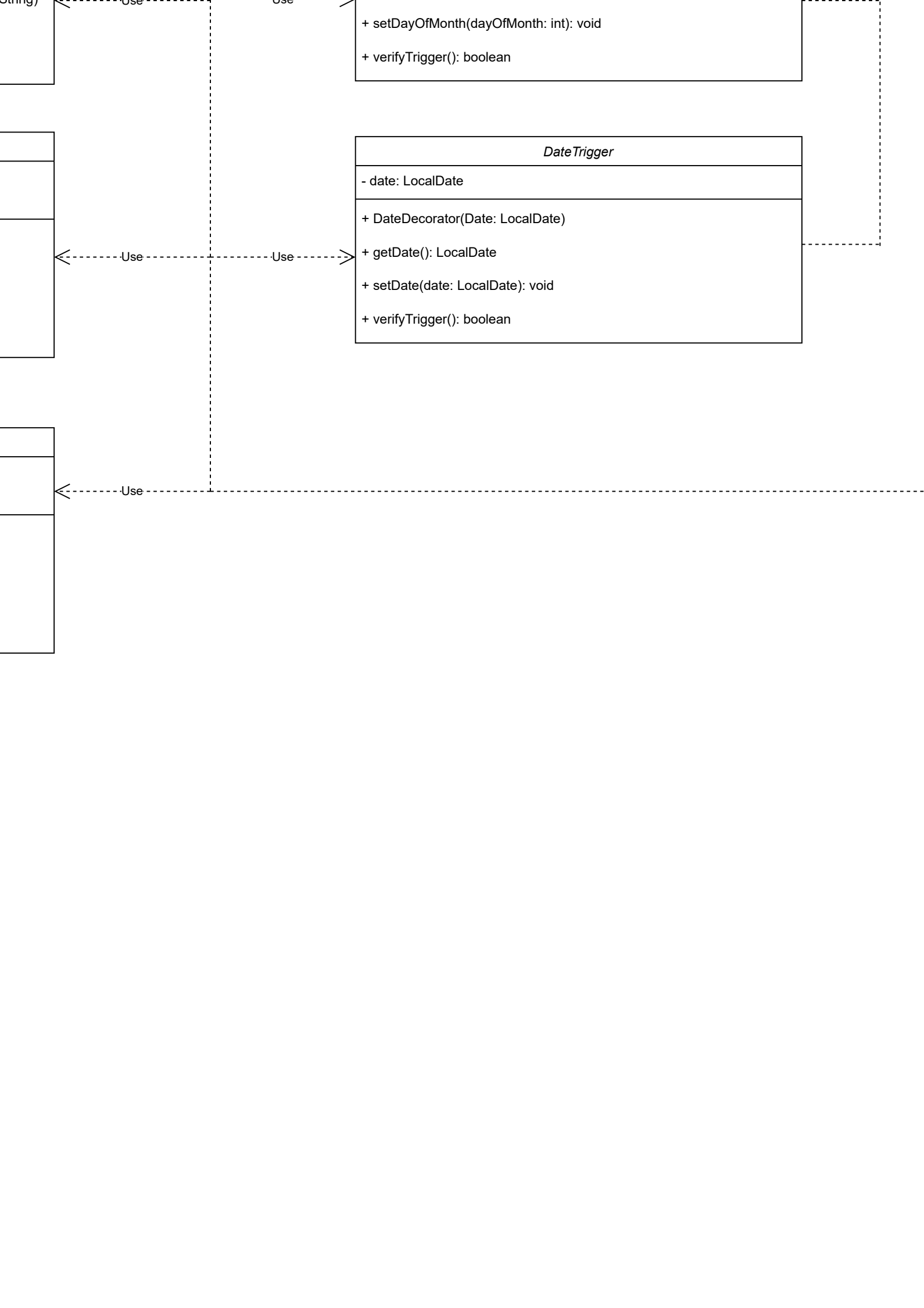
+CounterIsLessThanCounterTrigger(counterName1: String, counterName2:

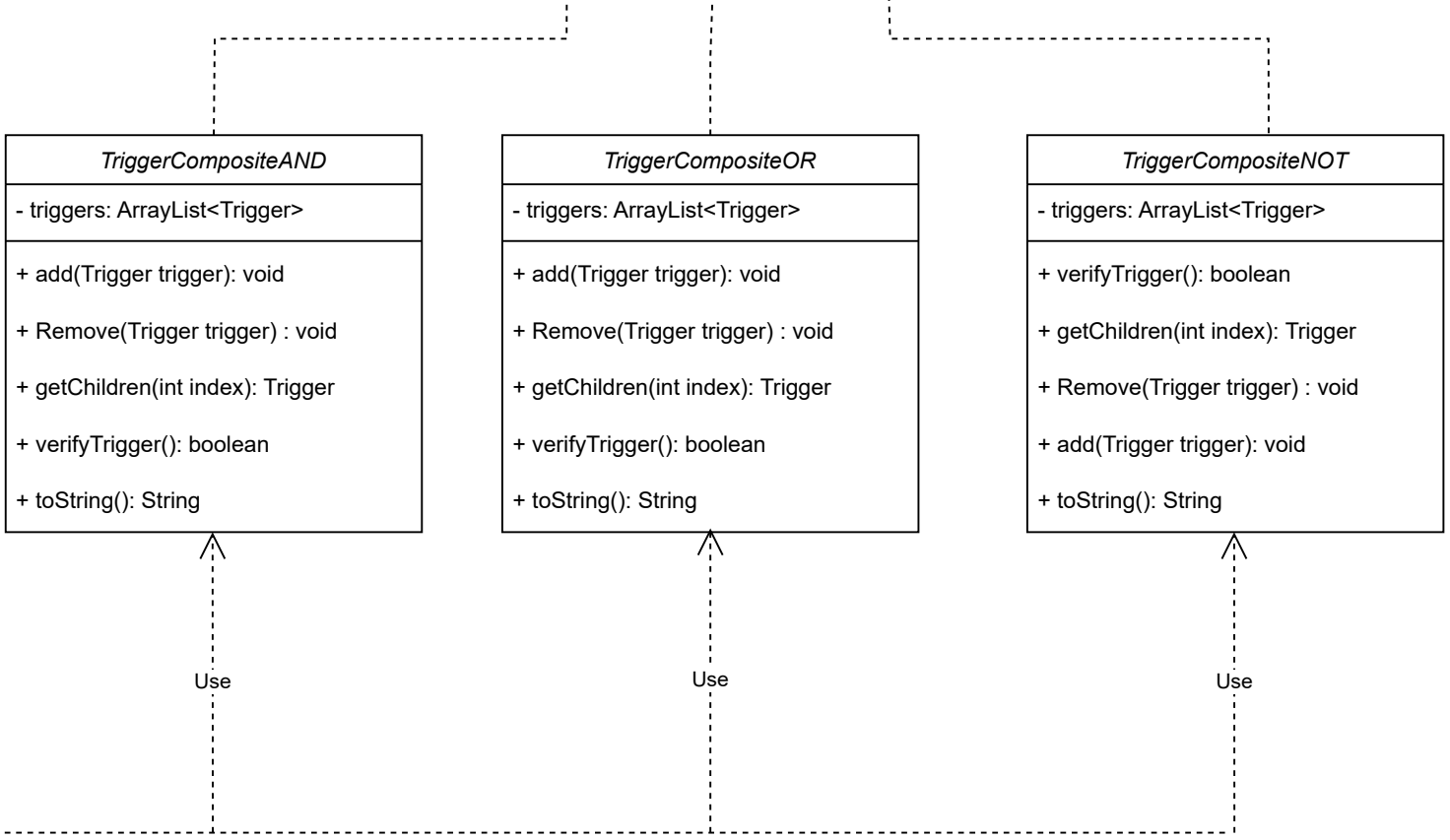
+ verifyTrigger(): boolean

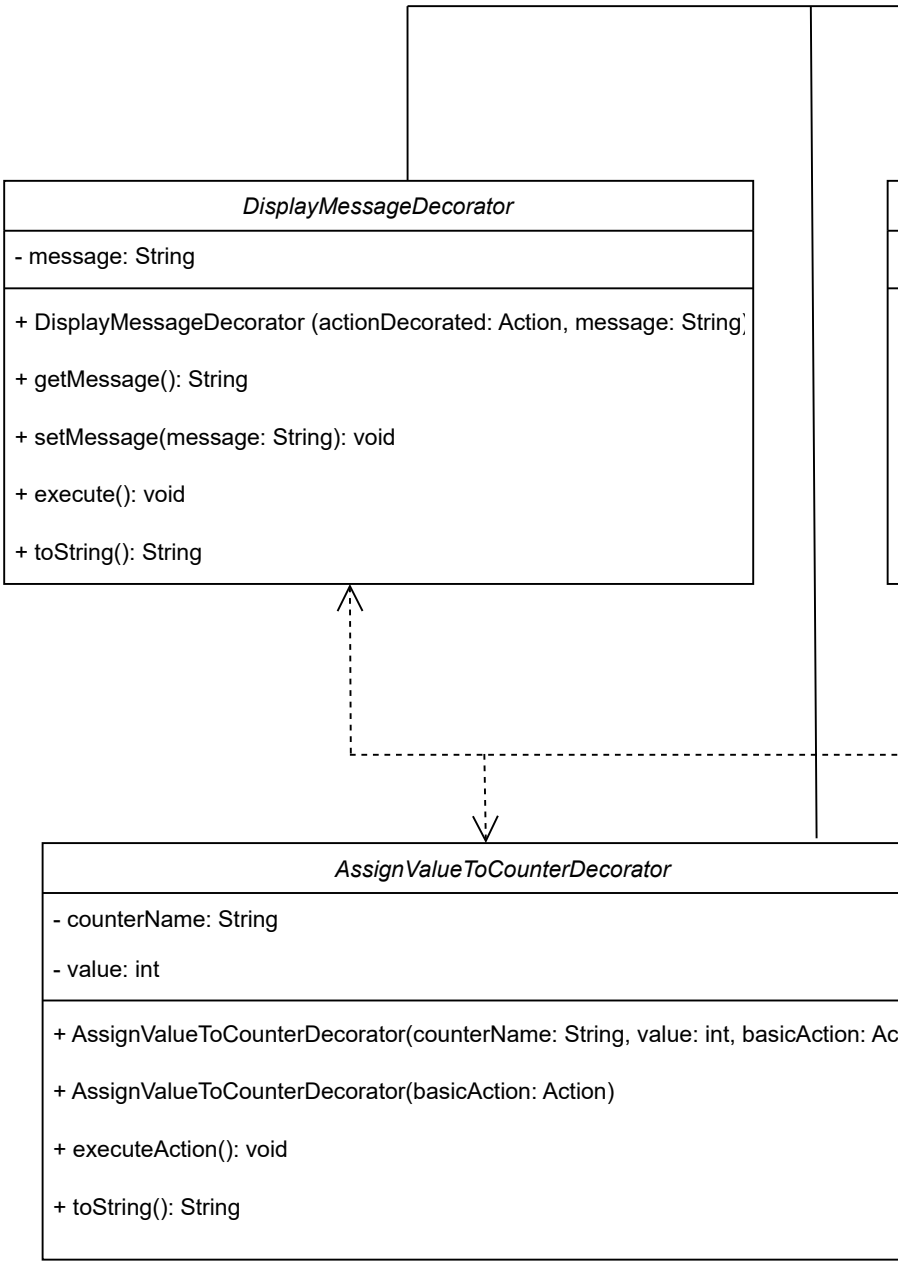
+ toString(): String

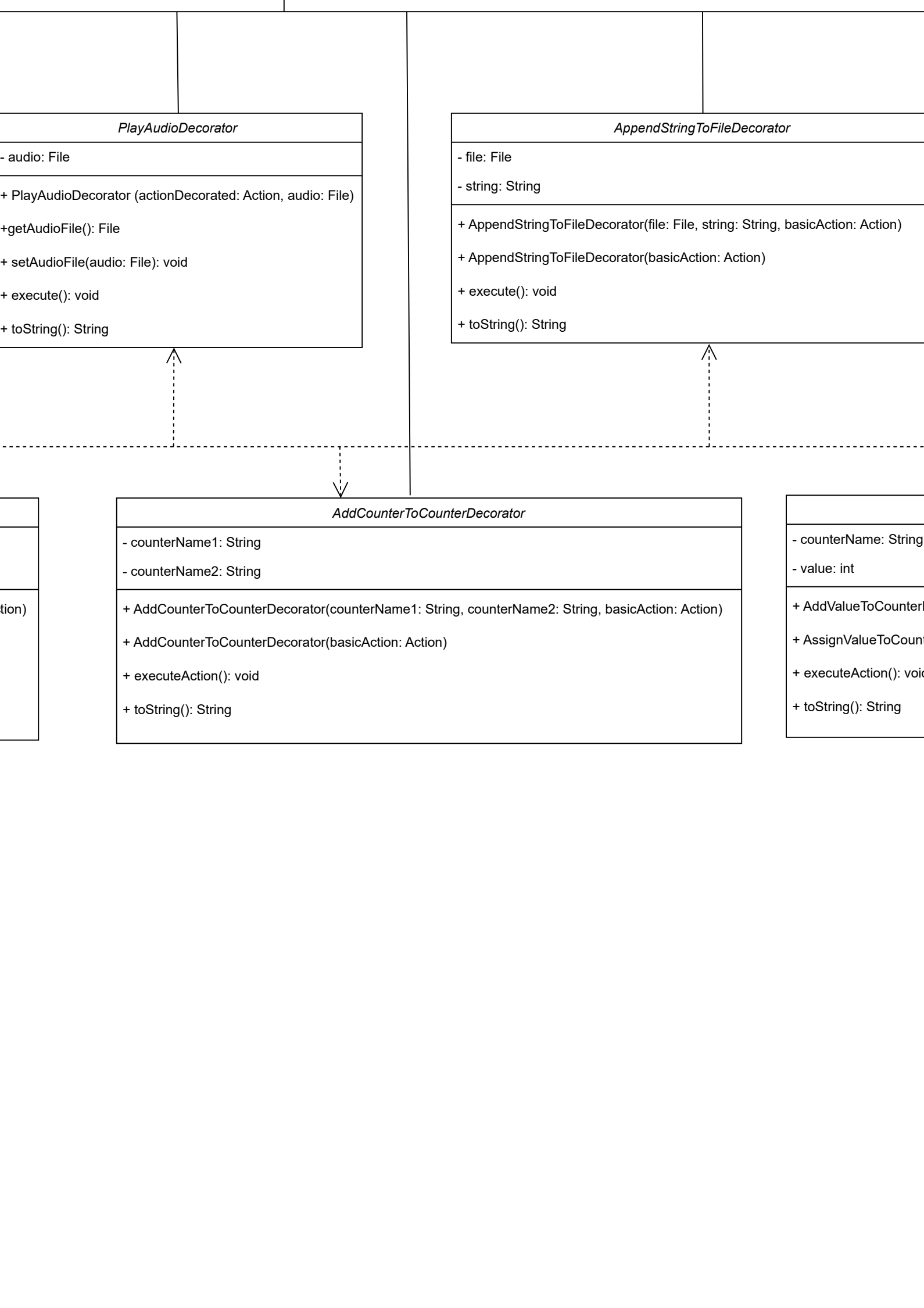
FileExistsTrigger
-fileName: String -dir: File
+FileExistsTrigger() +FileExistsTrigger(fileName: String, dir: File) + verifyTrigger(): boolean + toString(): String

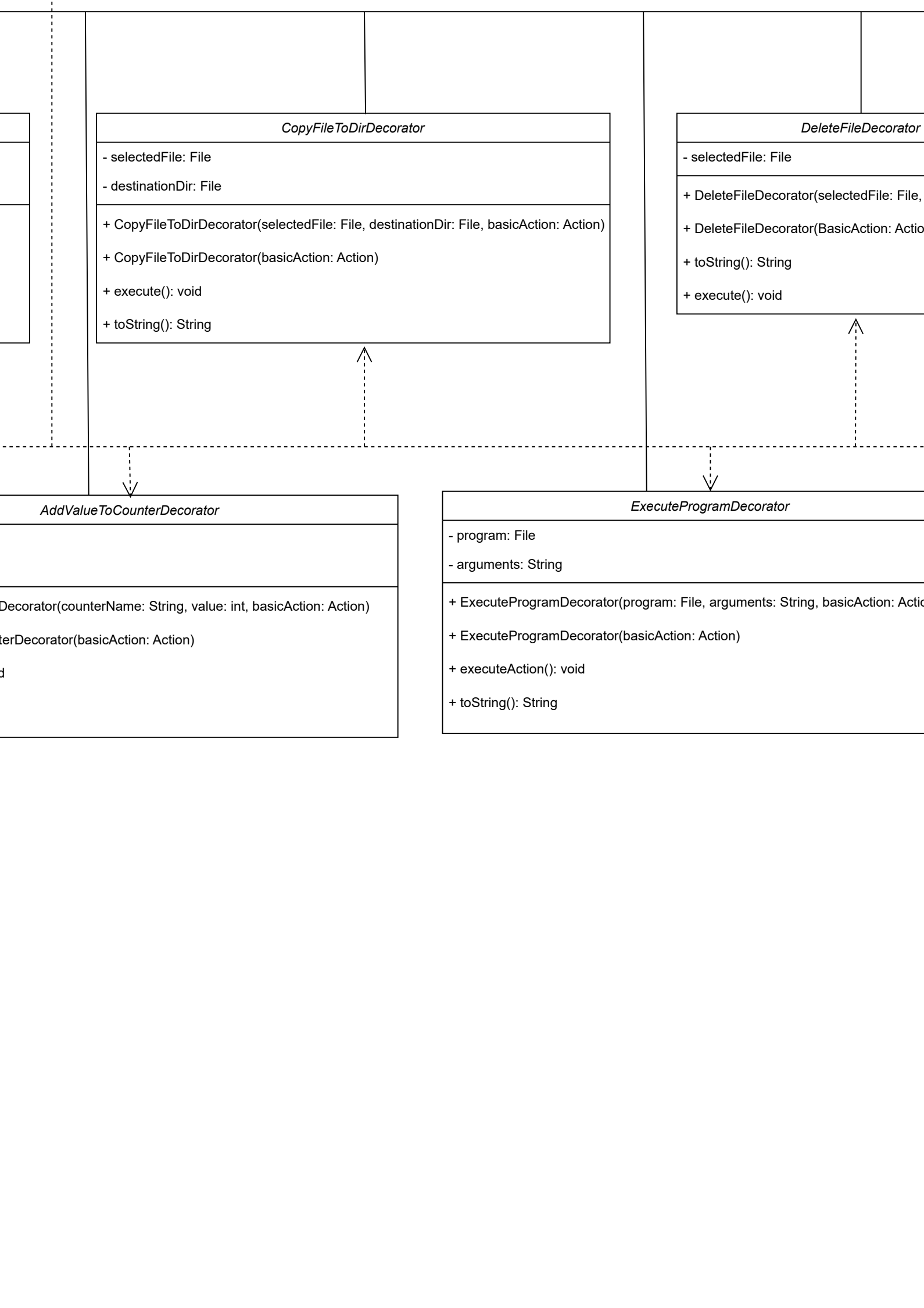
FileSizeIsGreaterTrigger
-fileName: String -size: long
+FileSizeIsGreaterTrigger +FileSizeIsGreaterTrigger(fileName: String, size: long) + verifyTrigger(): boolean + toString(): String

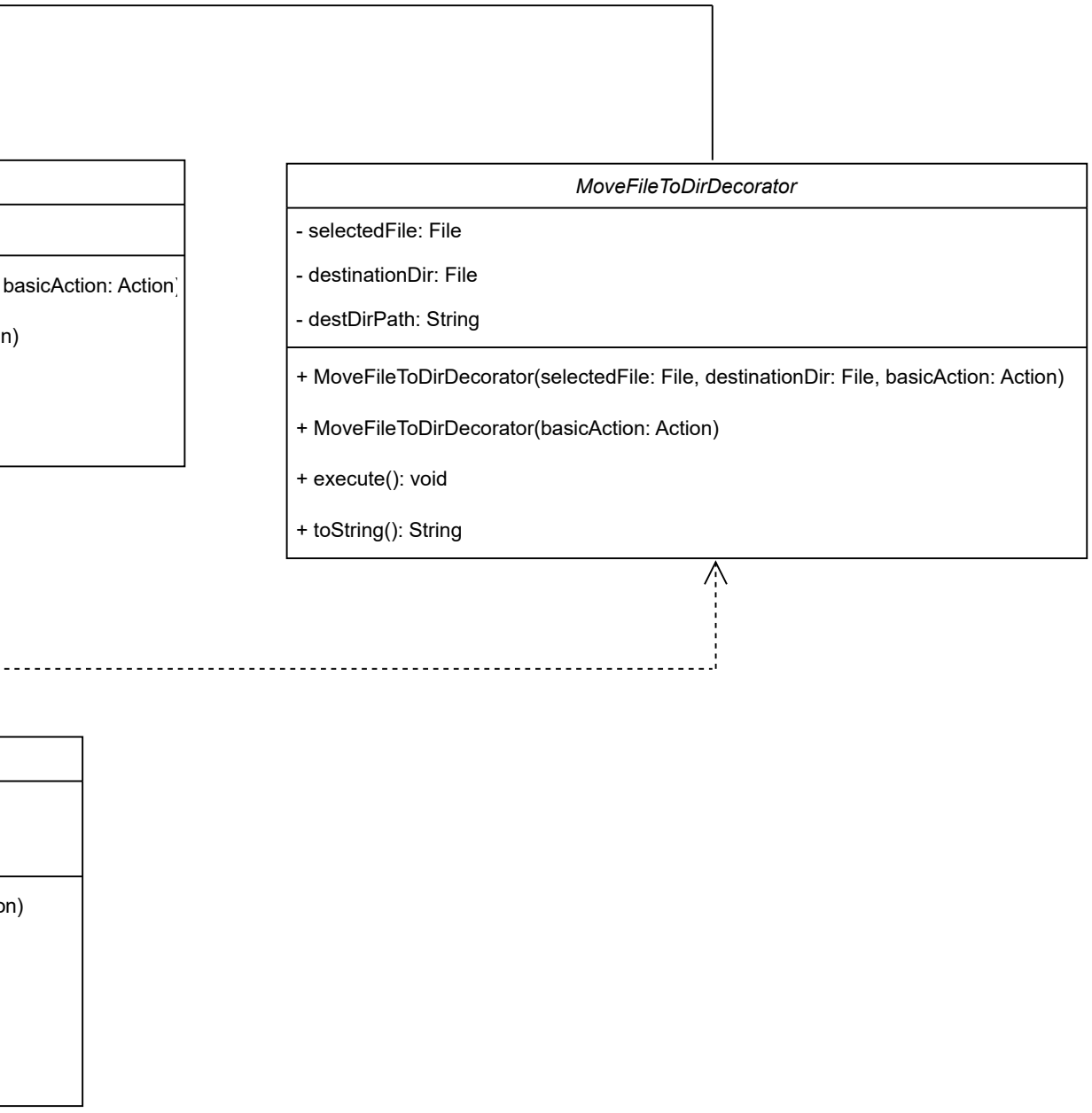

+ setDayOfMonth(dayOfMonth: int): void
+ verifyTrigger(): boolean
DateTrigger
- date: LocalDate
+ DateDecorator(Date: LocalDate)
+ getDate(): LocalDate
+ setDate(date: LocalDate): void
+ verifyTrigger(): boolean
Use
Use
Use
Use


TriggerCompositeAND
- triggers: ArrayList<Trigger>
+ add(Trigger trigger): void
+ Remove(Trigger trigger) : void
+ getChildren(int index): Trigger
+ verifyTrigger(): boolean
+ toString(): String
TriggerCompositeOR
- triggers: ArrayList<Trigger>
+ add(Trigger trigger): void
+ Remove(Trigger trigger) : void
+ getChildren(int index): Trigger
+ verifyTrigger(): boolean
+ toString(): String
TriggerCompositeNOT
- triggers: ArrayList<Trigger>
+ verifyTrigger(): boolean
+ getChildren(int index): Trigger
+ Remove(Trigger trigger) : void
+ add(Trigger trigger): void
+ toString(): String
Use
Use
Use


DisplayMessageDecorator
- message: String
+ DisplayMessageDecorator (actionDecorated: Action, message: String)
+ getMessage(): String
+ setMessage(message: String): void
+ execute(): void
+ toString(): String
AssignValueToCounterDecorator
- counterName: String
- value: int
+ AssignValueToCounterDecorator(counterName: String, value: int, basicAction: Ac
+ AssignValueToCounterDecorator(basicAction: Action)
+ executeAction(): void
+ toString(): String


PlayAudioDecorator
- audio: File
+ PlayAudioDecorator (actionDecorated: Action, audio: File)
+getAudioFile(): File
+ setAudioFile(audio: File): void
+ execute(): void
+ toString(): String
AppendStringToFileDecorator
- file: File
- string: String
+ AppendStringToFileDecorator(file: File, string: String, basicAction: Action)
+ AppendStringToFileDecorator(basicAction: Action)
+ execute(): void
+ toString(): String
AddCounterToCounterDecorator
- counterName1: String
- counterName2: String
+ AddCounterToCounterDecorator(counterName1: String, counterName2: String, basicAction: Action)
+ AddCounterToCounterDecorator(basicAction: Action)
+ executeAction(): void
+ toString(): String
- counterName: String
- value: int
+ executeAction(): voi
+ toString(): String


CopyFileToDirDecorator
- selectedFile: File
- destinationDir: File
+ CopyFileToDirDecorator(selectedFile: File, destinationDir: File, basicAction: Action)
+ CopyFileToDirDecorator(basicAction: Action)
+ execute(): void
+ toString(): String
DeleteFileDecorator
- selectedFile: File
+ DeleteFileDecorator(selectedFile: File,
+ DeleteFileDecorator(BasicAction: Actio
+ toString(): String
+ execute(): void
AddValueToCounterDecorator
Decorator(counterName: String, value: int, basicAction: Action)
terDecorator(basicAction: Action)
d
ExecuteProgramDecorator
- program: File
- arguments: String
+ ExecuteProgramDecorator(program: File, arguments: String, basicAction: Actio
+ ExecuteProgramDecorator(basicAction: Action)
+ executeAction(): void
+ toString(): String


basicAction: Action
n)
MoveFileToDirDecorator
- selectedFile: File
- destinationDir: File
- destDirPath: String
+ MoveFileToDirDecorator(selectedFile: File, destinationDir: File, basicAction: Action)
+ MoveFileToDirDecorator(basicAction: Action)
+ execute(): void
+ toString(): String
on)