REVISION HISTORY

Date	Version	Description	Author
30.11.2021	1.1	File filled.	Başak BALCI
15.12.2021	2.1	Corrections made.	Başak BALCI

TABLE OF CONTENTS

Revision History **1**

1 Introduction **3**

1.1 References ***3***

1.1.1 Project References 3

2 Software Architecture overview **3**

3 Software design description **4**

3.1 Component 1 ***4***

3.1.1 Component interfaces 4

3.1.2 Component design description 4

3.1.3 Workflows and algorithms 6

3.1.4 Software requirements mapping 6

4 COTS Identification **6**

1 Introduction

This document describes the design of the Pushdown Automata software system.

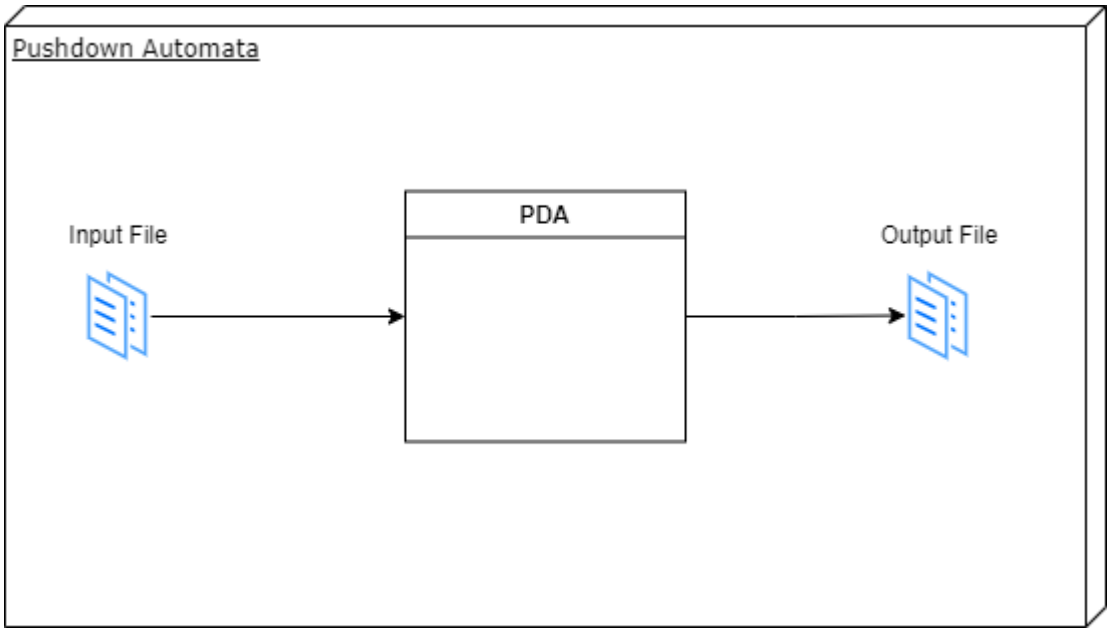
1.1 References

1.1.1 Project References

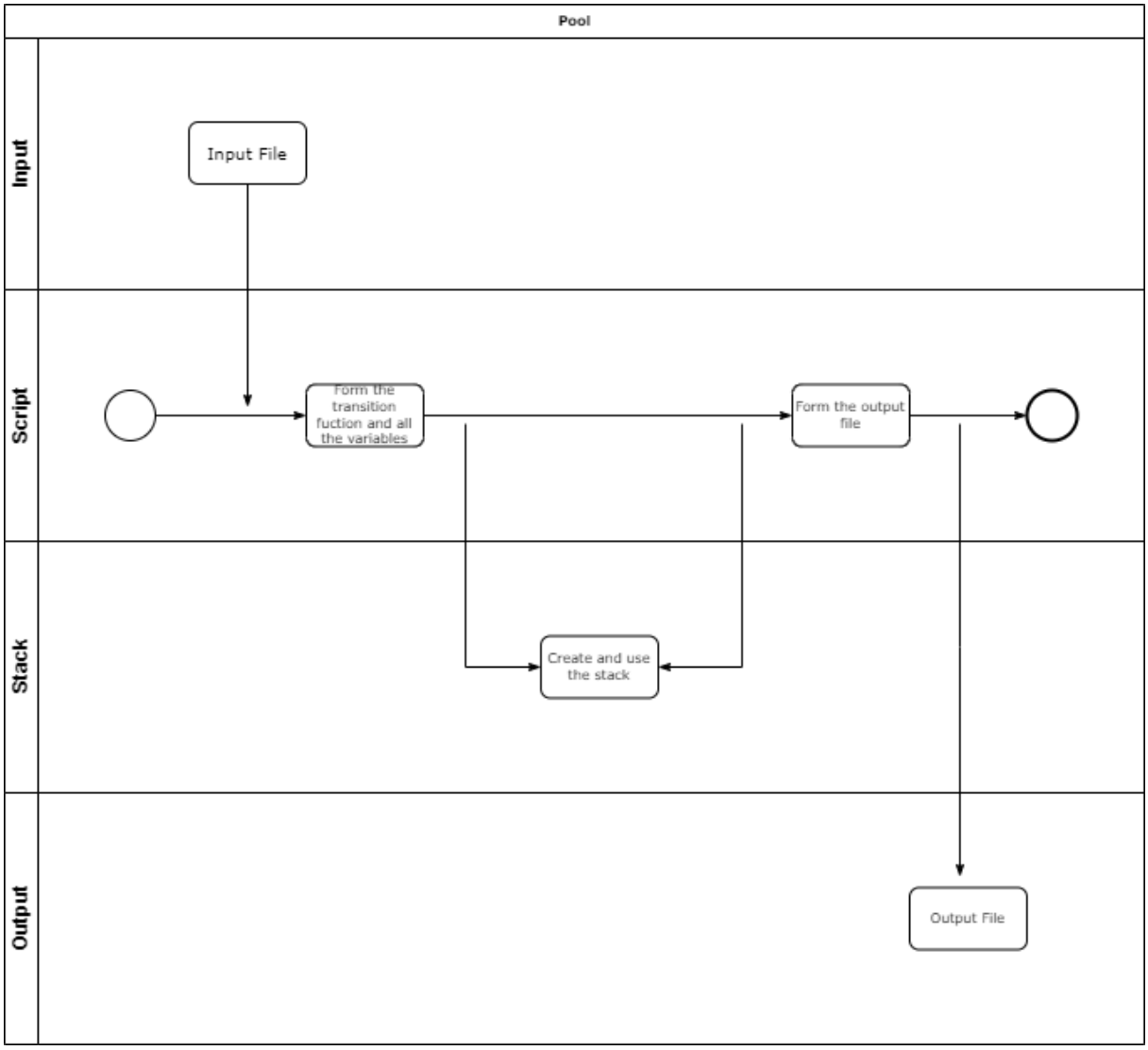
#	Document Identifier	Document Title

2 Software Architecture overview

The package diagram.



The flowchart diagram.



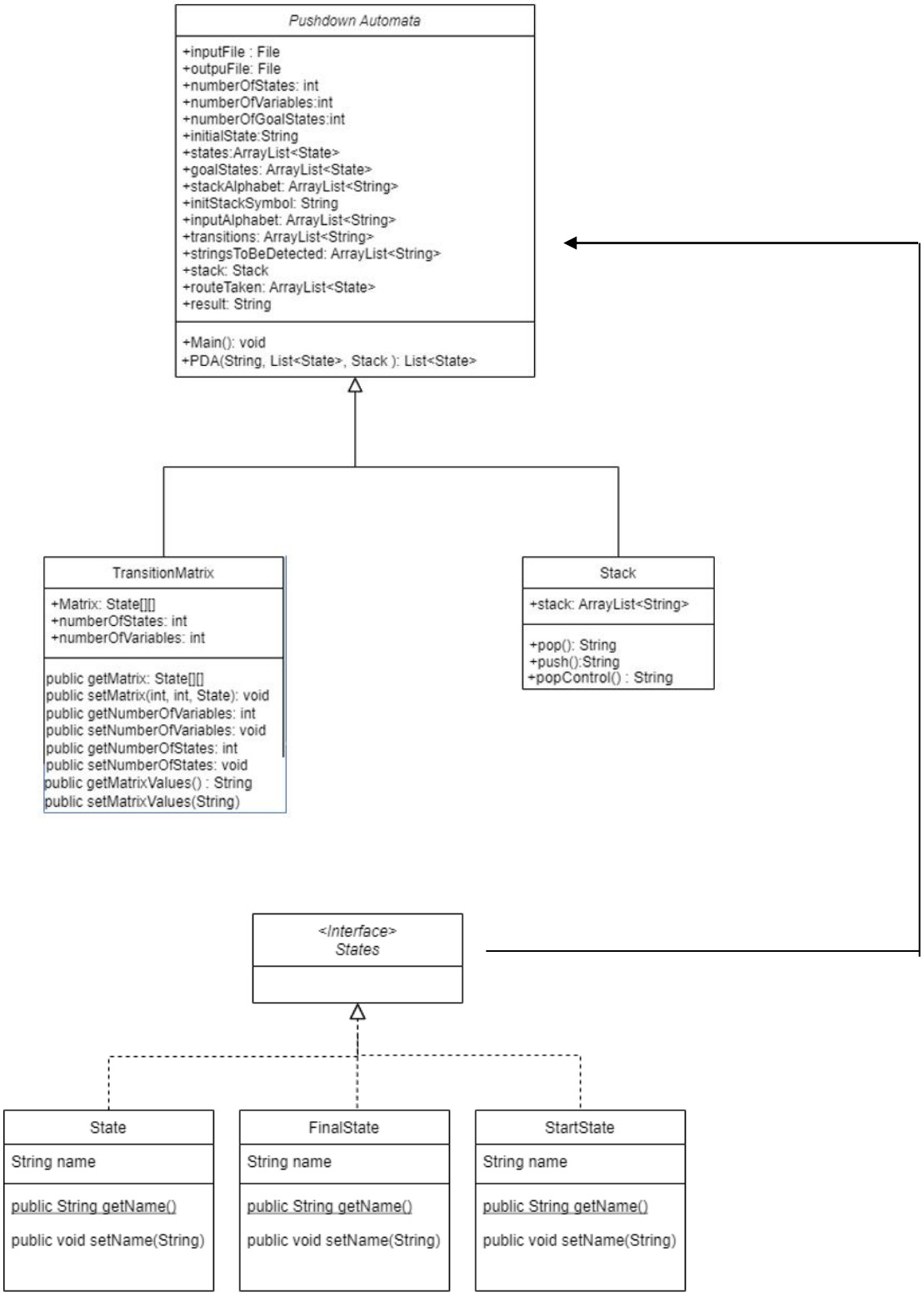
3 Software design description

3.1 Component 1

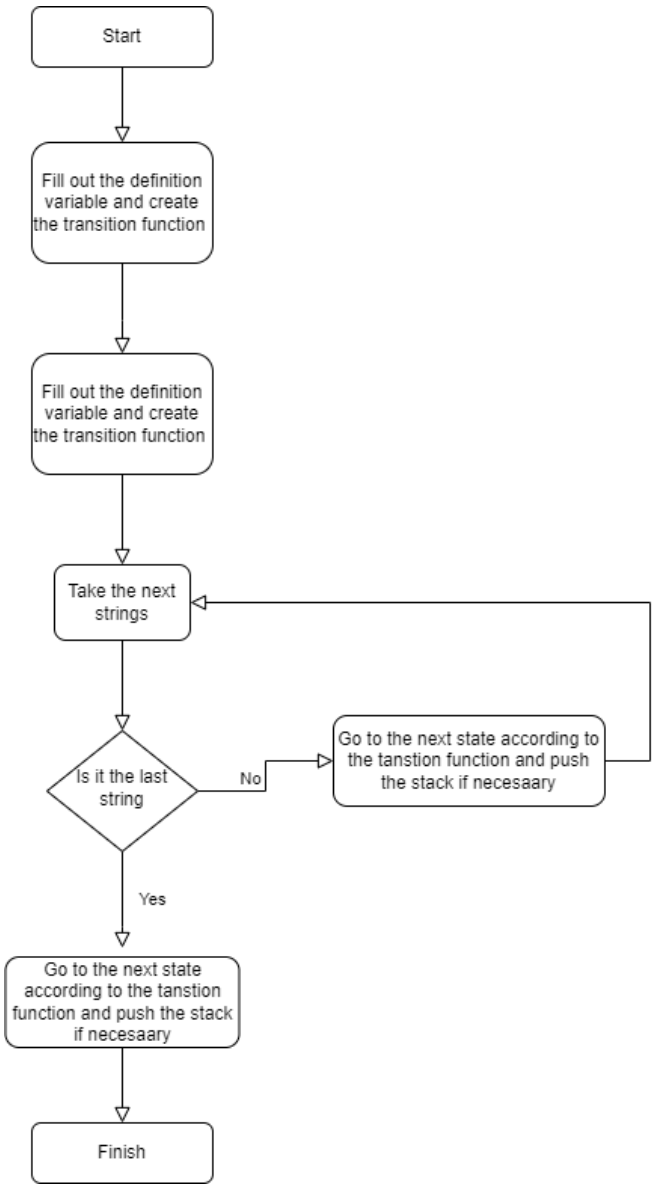
3.1.1 Component interfaces

PDA: Simulates the PDA, takes the input file and creates the output file.
TransitionFunction: This class represents the transition function matrix.
State: This class represents the states. Stack: This class represents the stack.

3.1.2 Component design description


Pushdown Automata
+inputFile : File
+outpuFile: File
+numberOfStates: int
+numberOfVariables:int
+numberOfGoalStates:int
+initialState:String
+states:ArrayList<State>
+goalStates: ArrayList<State>
+stackAlphabet: ArrayList<String>
+initStackSymbol: String
+inputAlphabet: ArrayList<String>
+transitions: ArrayList<String>
+stringsToBeDetected: ArrayList<String>
+stack: Stack
+routeTaken: ArrayList<State>
+result: String
+Main(): void
+PDA(String, List<State>, Stack): List<State>
TransitionMatrix
+Matrix: State[][]
+numberOfStates: int
+numberOfVariables: int
public getMatrix: State[][]
public setMatrix(int, int, State): void
public getNumberOfVariables: int
public setNumberOfVariables: void
public getNumberOfStates: int
public setNumberOfStates: void
public getMatrixValues() : String
public setMatrixValues(String)
Stack
+stack: ArrayList<String>
+pop(): String
+push():String
+popControl() : String
<Interface>
States
State
String name
public String getName()
public void setName(String)
FinalState
String name
public String getName()
public void setName(String)
StartState
String name
public String getName()
public void setName(String)

3.1.3 Workflows and algorithms



3.1.4 Software requirements mapping

Not Applicable.

4 COTS Identification

Not Applicable.