

CS410 Project-1 DFA		
Doc # DFA-SDD	Version: 1.0	Page 1 / 5

REVISION HISTORY

Date	Version	Description	Author
18/10/2021	0.5	Filled Sections	Emin Sadikhov
19/10/2021	0.7	Added required diagrams	Emin Sadikhov
5/11/2021	1.0	Revised and finalized	Emin Sadikhov

TABLE OF CONTENTS

Table of Contents

REVISION HISTORY1

1 *Introduction3*

2 *Software Architecture overview3*

3 *Software design description5*

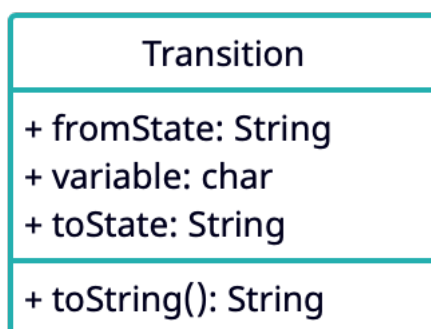
CS410 Project-1 DFA		
Doc # DFA-SDD	Version: 1.0	Page 3 / 5

1 Introduction

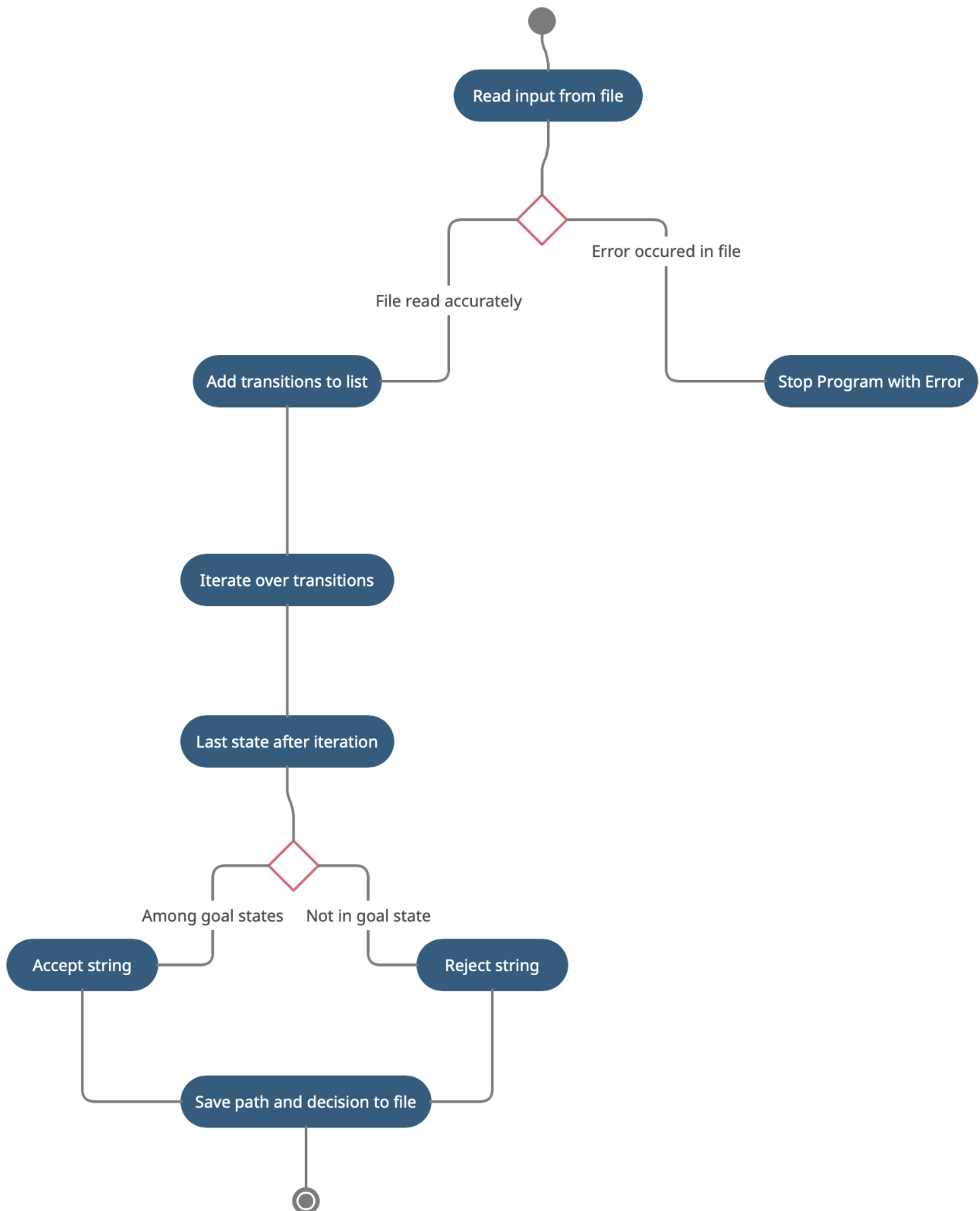
This document describes the design of the Deterministic Finite Automate software system. Once a string comes to a program, DFA should either accept or reject it. Program should read values from an input file that includes all necessary information regarding states, variables, transitions, and strings to test. Output file should include the path taken by the program and either accept or reject for a particular string.

2 Software Architecture overview

Class diagram:



Activity Diagram:



CS410 Project-1		
Doc # DFA-SDD	Version: 1.0	Page 5 / 5

3 Software design description

This program is implemented with Java programming language and is indented to run on console. No GUI is available for the application. Initially, I started with reading all the necessary variable details from the input file. Later, I created a Transition class that stores two states, from which and to which, and a variable that makes that transition. All the transitions are stores in the List. Next, we will be processing the input strings. While considering each transition, I will also store the path taken so far. Once all the characters are processed in the string, I will check the state I am at, and will control if it is among the goal states. If it is, then the string is accepted for that DFA, and if not, it is rejected. Finally, the path taken along with the result will be stored on the output.txt file.

Below you can find sample input and output for the program. The information given inside parenthesis is the description of the input and output. They are given to better understand each line of input for human. In the real input for the program, they will be removed as it is not necessary for the computer.

Input:

2 (number of states)
2 (number of variables)
1 (number of goal states)
q1 q2 (states)
q2 (goal state(s))
a b (variables)
q1 a q1 (from q1 with a to q1)
q1 b q2 (from q1 with b to q2)
q2 a q2 (from q2 with a to q2)
q2 b q1 (from q2 with b to q1)
aba (string to be detected)
ababababa (string to be detected)

Output:

q1 q2 q2 (route taken)
Accepted
q1 q2 q2 q1 q1 q2 q2 q1 q1 (route taken)
Rejected

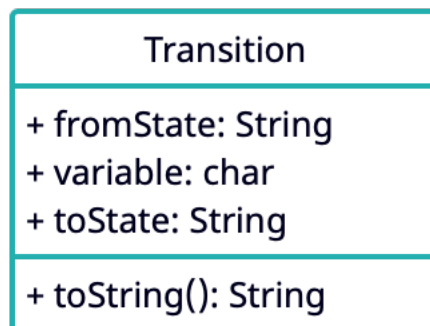
CS410 Project-1		
Doc # DFA-SDD	Version: 1.0	Page 6 / 5

3.1 Transition

3.1.1 Component Interface

Transition class consists of three instance variables. FromState, toState which are Strings, and one variable which is of type character. Main purpose of this class is to store all the transitions that happens in the DFA.

3.1.2 Component Design Description



4 COTS Identification

COTS (Commercial off the shell) libraries used:

Not Applicable