CTL Model Checker

Project Report

Submitted by:

Akhilesh Padmanabhuni Net_ID: bdu3

Harish Pagidipala Net_ID: h_p161

Pallavi Krishna Reddy Net_ID: osg12

Harshitha Srinivas Net_ID: jrm512

Navya Sahithi Velugubantla Net_ID: hnr48

Under the guidelines of Professor:

Dr. Rodion Podorozhny

Dept. Computer Science,

Texas State University

San Marcos, Texas -78666

Aim:

To develop a CTL model checking and CTL temporal logic verification tool as a Java Standalone application.

Description:

- The UI enables the user to upload the file that contains the definition of the kripke structure against which the property and a CTL formula is verified.
- After uploading the file, the CTL formula would be entered in the text box specified as CTL formula
- User can select the state from the drop-down box.
- Upon clicking the check button, the application would perform syntax verification and if there's any problem with the Kripke structure an error message would be displayed.
- The result whether the kripke structure for specified state holds for the formula or not would be displayed in the Result text box only if the kripke structure is successfully verified against the formula.
- The application throws an error message if kripke structure is not parsed completely.

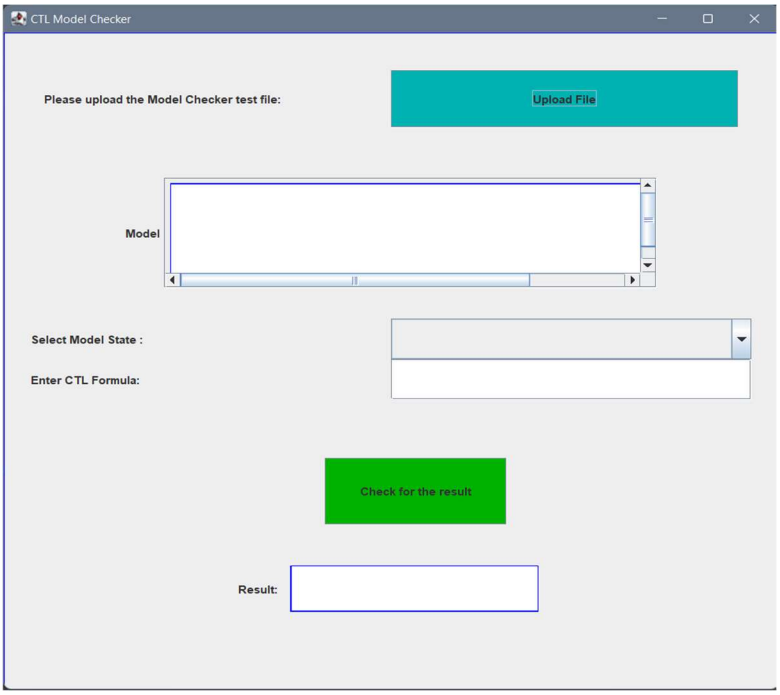


Figure 1: GUI of the Model System

Figure 2: Loading a Kripke Structure Test File

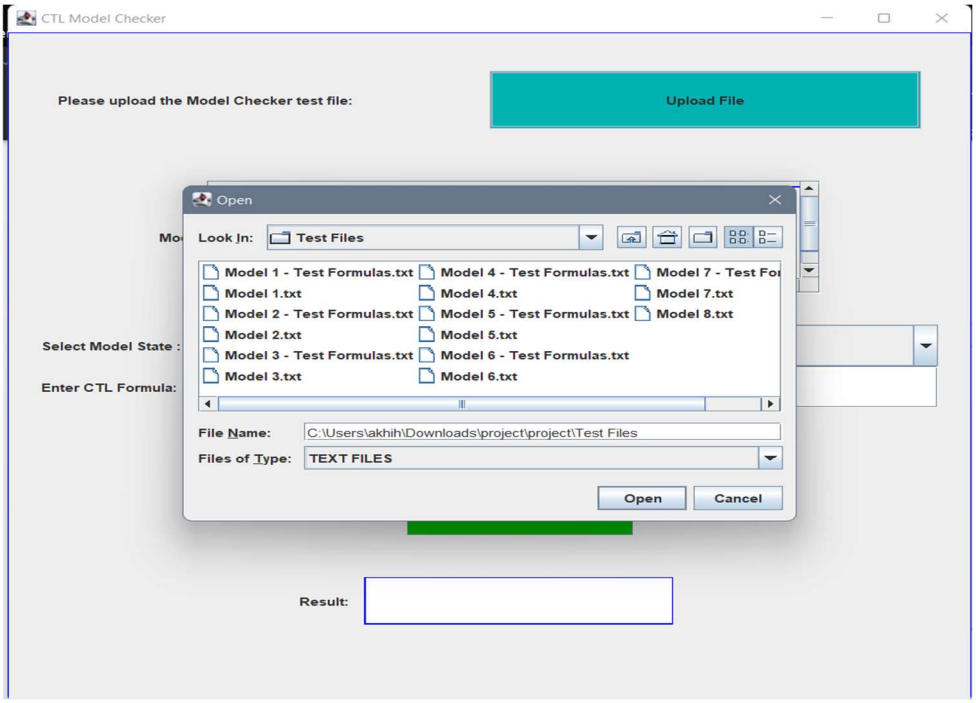
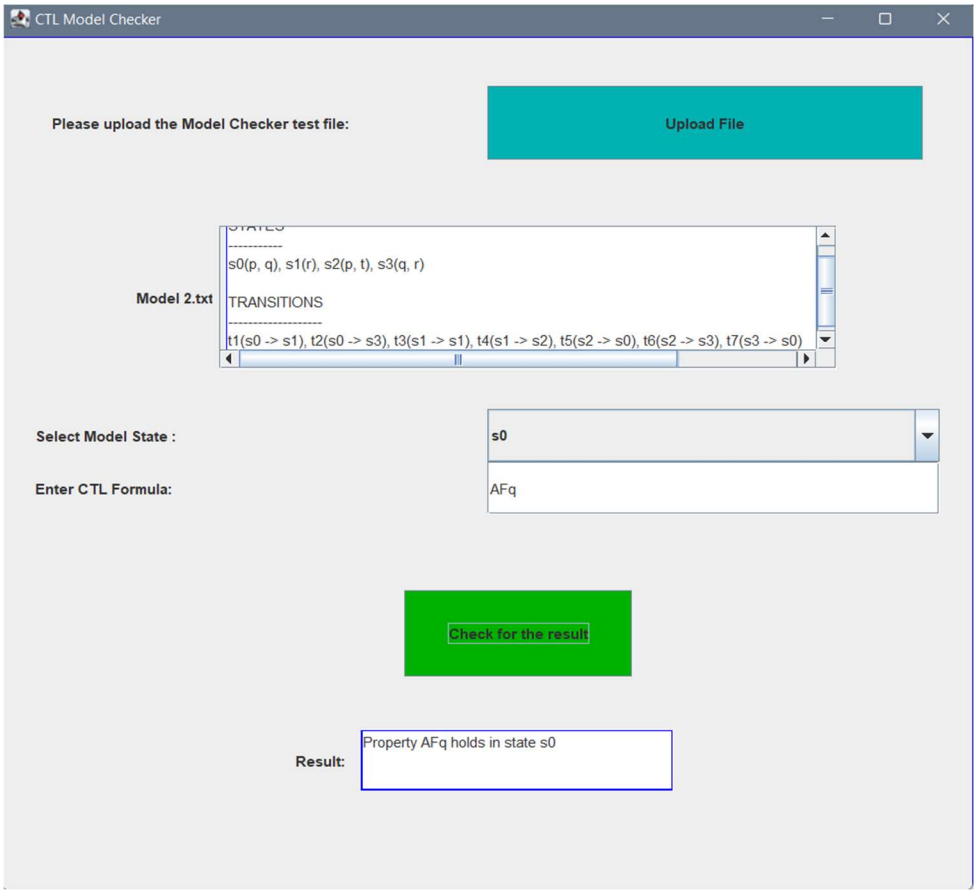


Figure 3: The overview of the application



Acceptance testcases

Sample Testcases are specified for model 1, model 3, model 6 and model 8 for the CTL Formulas

Model 1

i) CTL formula: EG(r->t)

Starting state: s1

Property EG(r->t) holds in state s1

ii) CTL formula: AXq and A(pUq)

Starting state: s2

Property AXq and A(pUq) does not hold in state s2

Model 3

i) CTL formula: AGq

Starting state: s5

Property AGq does not hold in state s5

ii) CTL formula: EGq

Starting state: s7

Property EGq does not hold in state s7

Model 6

i) CTL formula: EFp

Starting state: s1

Property EFp holds in state s1

ii) CTL formula: AFp

Starting state: s4

Property EGq holds in state s4

Model 8

i) CTL formula: E(start U err)

Starting state: s2

Property E(start U err) holds in state s2

ii) CTL formula: EGheat

Starting state: s7

Property EGheat holds in state s7

Illustration of acceptance testcases with screenshots of all the UI elements.

Model 1

i) CTL formula: EG(r->t)

Starting state: s1

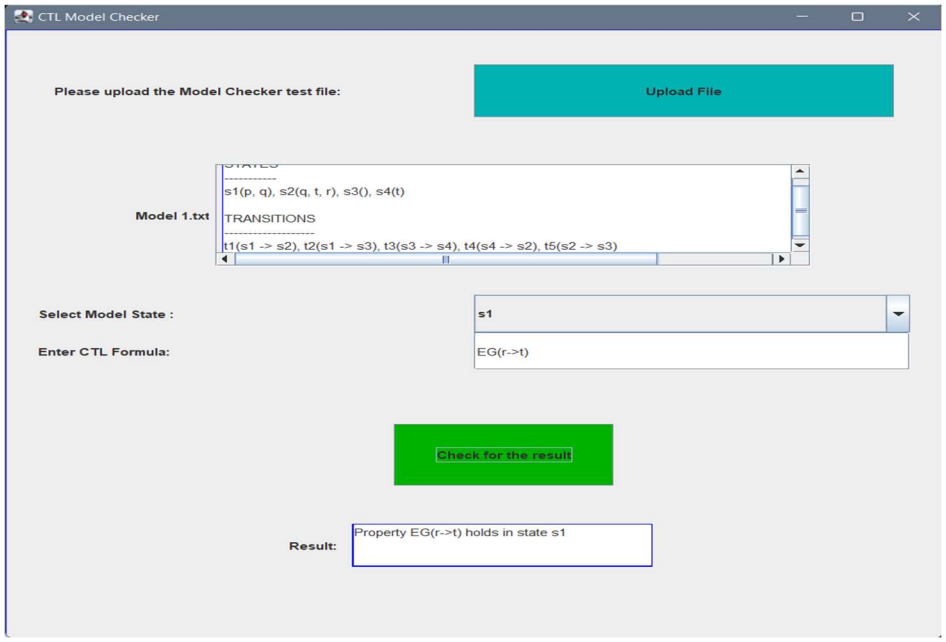
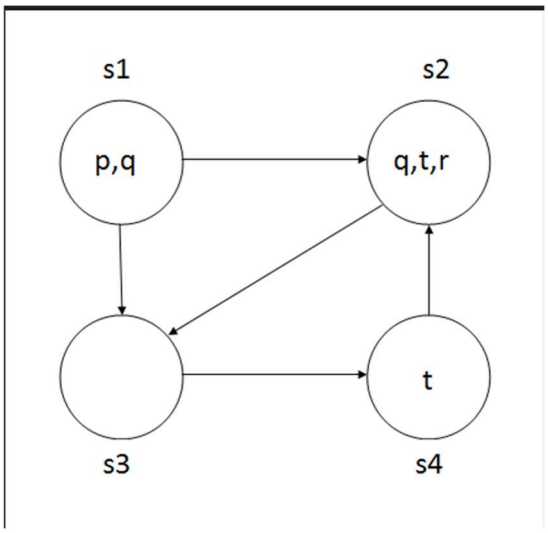


Figure 4: Property EG(r->t) holds in state s1



Kripke Structure for model 3

i) CTL formula: AGq

Starting state: s5

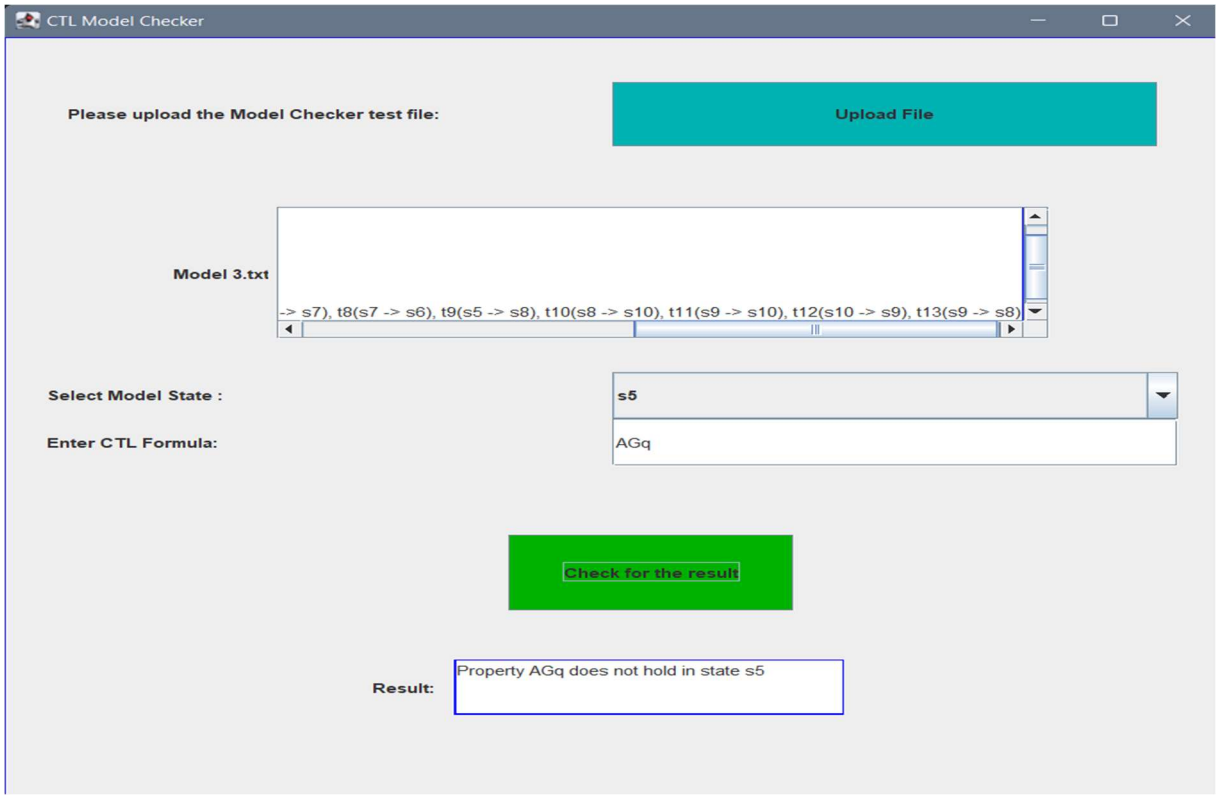
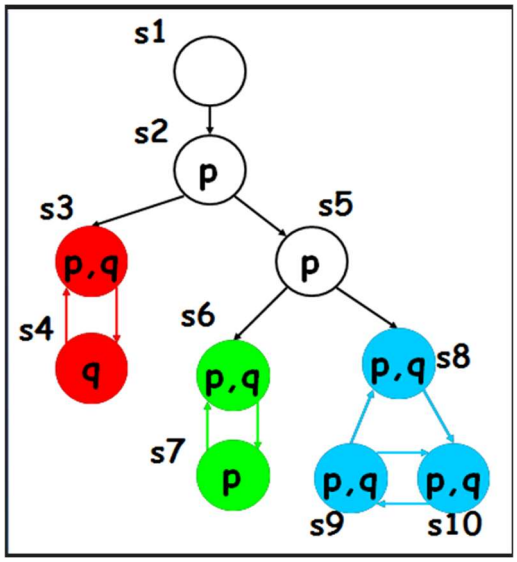


Figure 5: Property AGq does not hold in state s5



Kripke Structure for model 6

i)CTL formula: EFp

Starting state: s1

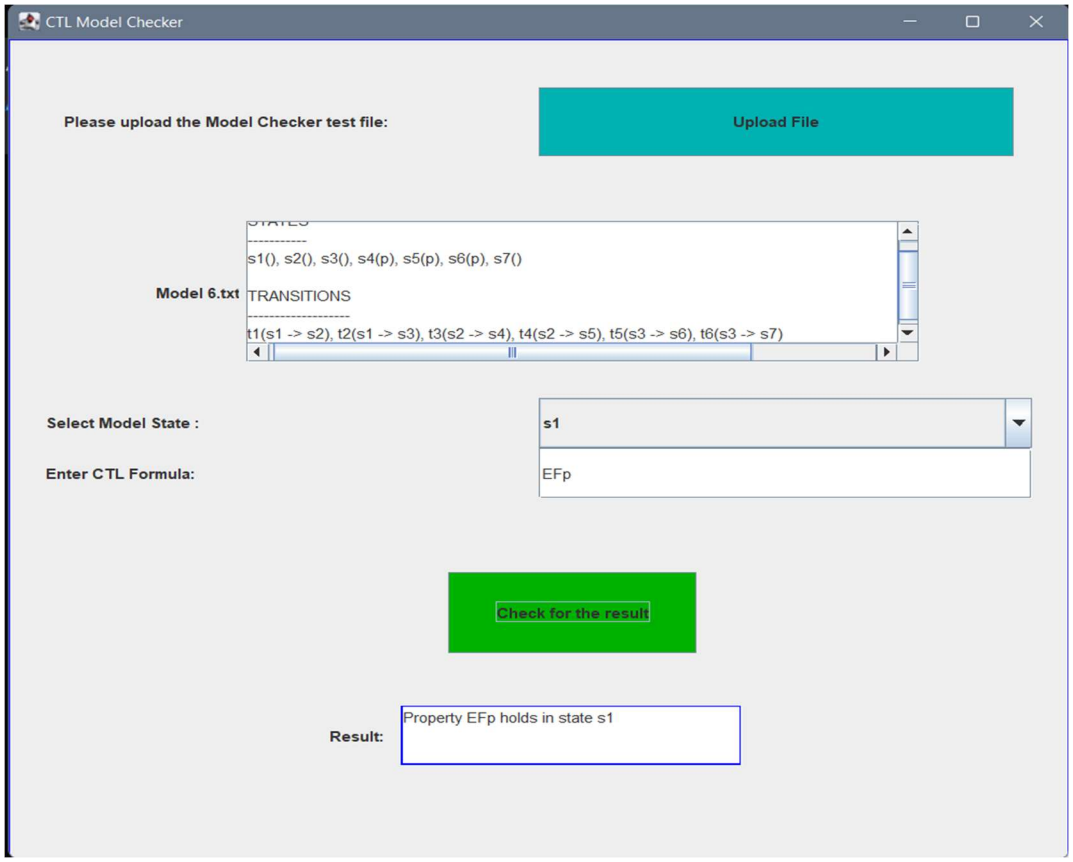
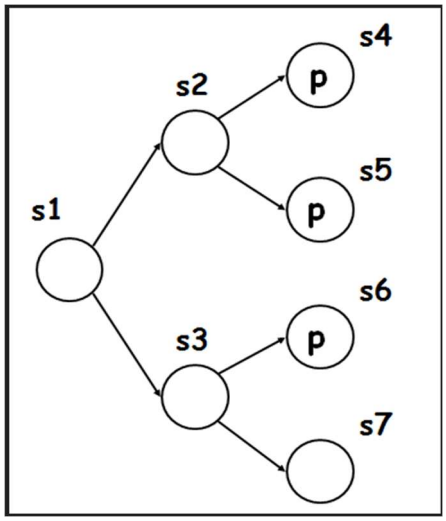


Figure 6: Property EFp holds in state s1



Kripke Structure for model 8

i)CTL formula: E(start U err)

Starting state: s2

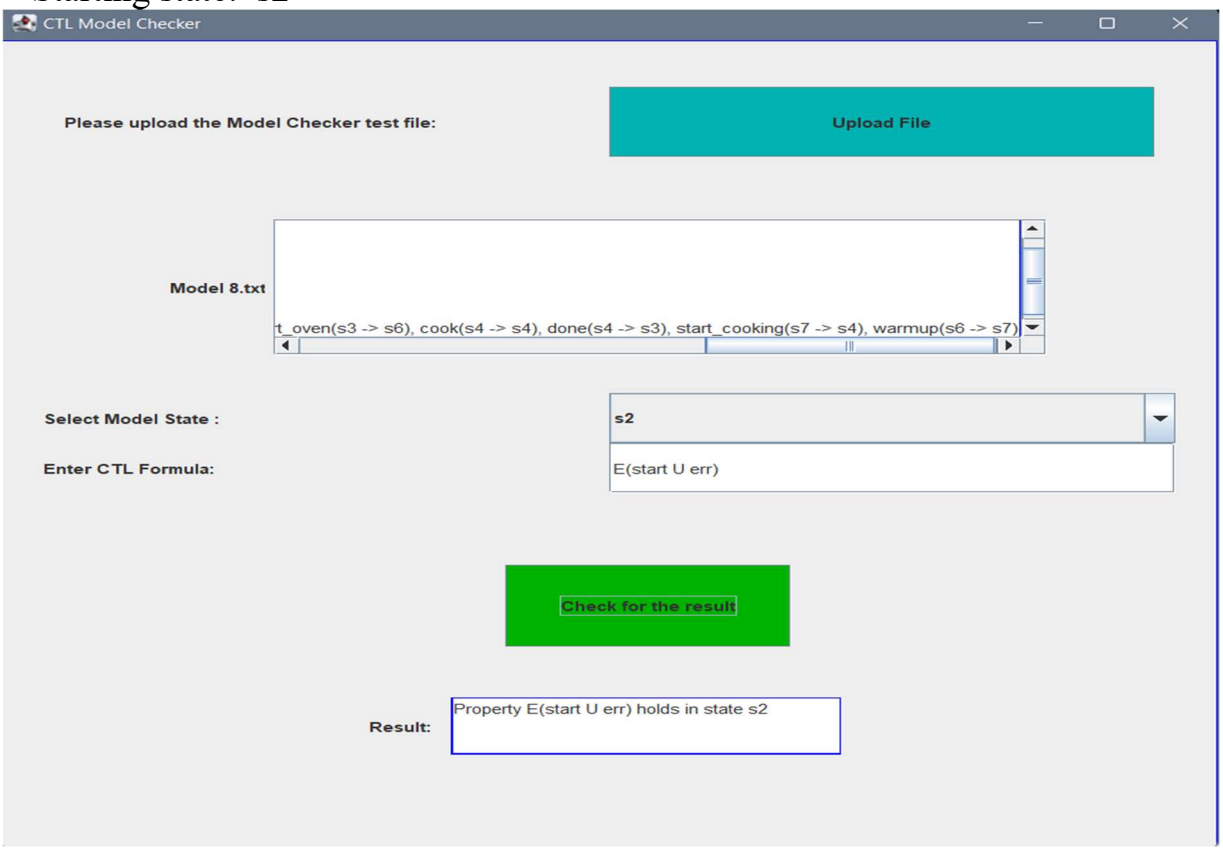
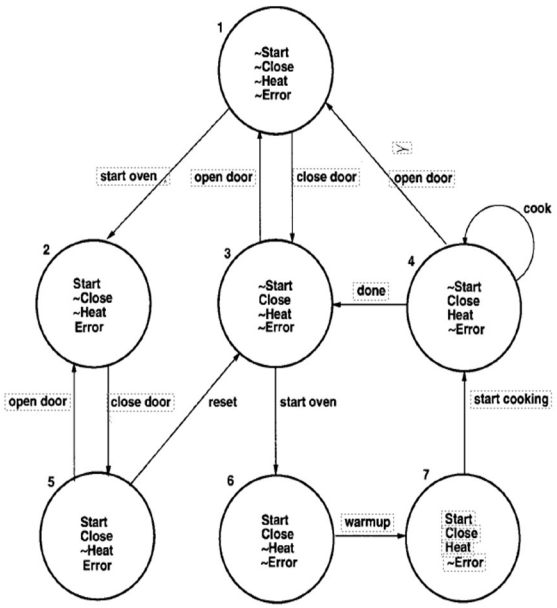
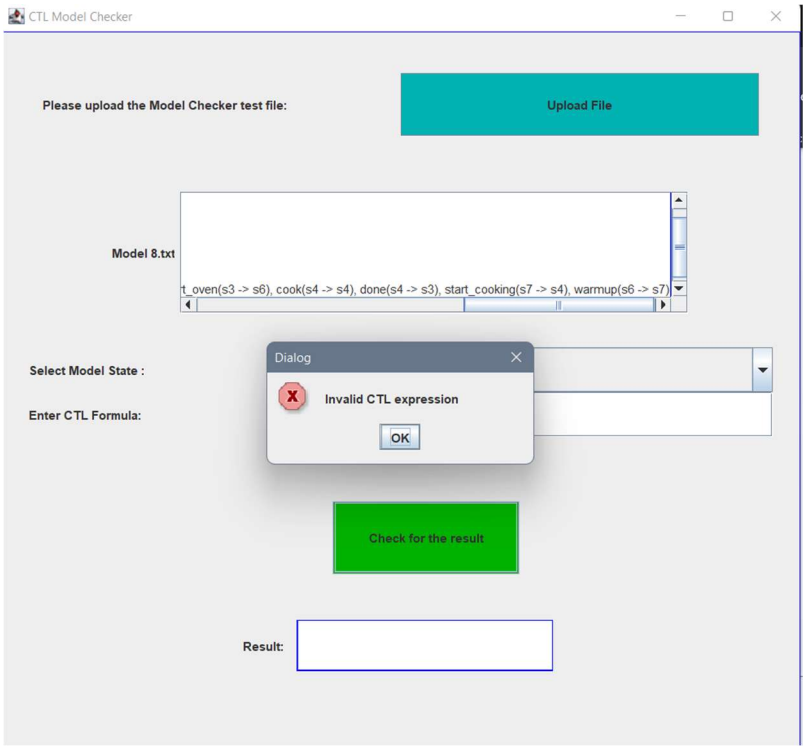


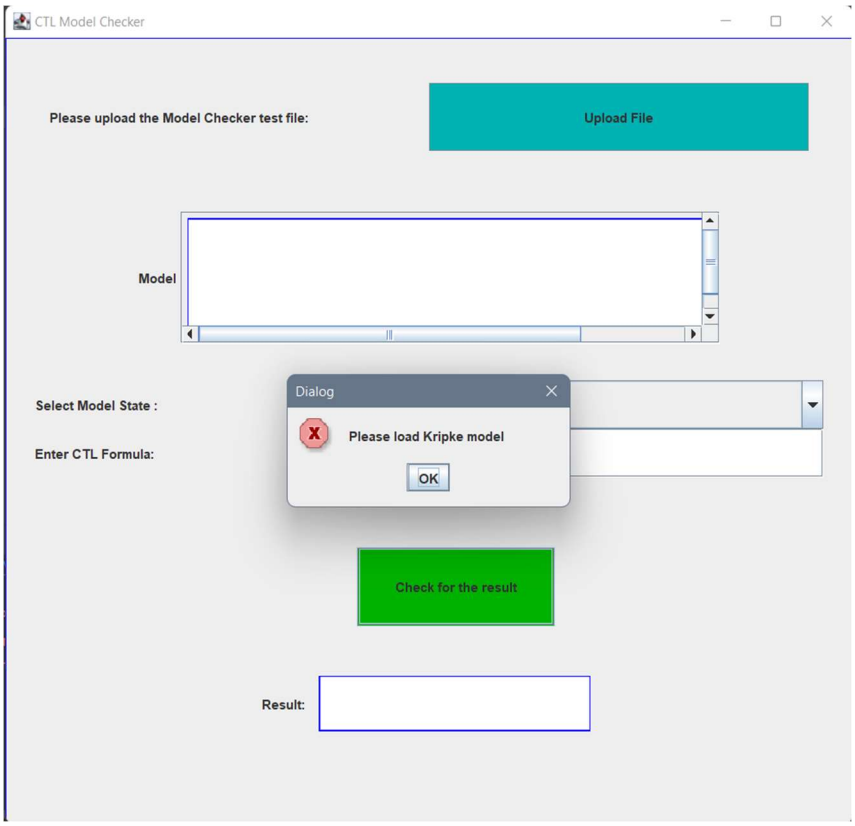
Figure 7: Property E(start U err) holds in state s2



Error messages if CTL formula is invalid:



Error message if kripke structure is not parsed:



UML class diagram for the software system

UML Class diagram present in the modelCheckCTL directory named UML_Diag.png

Source code (archive of directory structure starting from modelCheckCTL dir)

Source code is present in the “src” folder.

Tools used Eclipse IDE, IntelliJ IDE, JDk 14.0.1