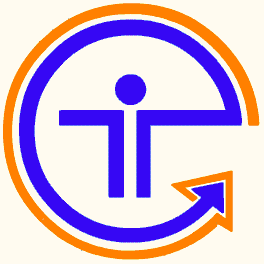
CENTRO DE ENSEÑANZA TECNICA INDUSTRIAL



By:

Daniel Arturo Fernández Raygoza #13310091

Edgar Jesús Robles Najar #12110275

Date: June 13 of 2016

Final Project:

“Expert system for personal computer hardware fail diagnostic”

**Expert system for personal computer hardware fail diagnostic**

**Justification:**

The user will introduce some symptoms related to the fail and the expert system will suggest the probable hardware parts that are causing the failure. After that the user can change that hardware part or repair option to try to solve the problem. This is going to help inexpert users to troubleshoot problems with their computer. It can be used as an educative tool for noob technicians.

Also it reduces the waste of time searching in internet the symptoms of a certain fail, is easy to use because it makes simple questions to the user that can be answered by yes or no.

In case of multiple failures the user can solve one by one and consult again the expert system for more help until the computer works again.

We are going to use CLIPS to create the expert system algorithm. Also we used GitHub to work better and faster in team, and allow other people to contribute adding more capabilities because the code is open to anyone and are free to improve it.

**Code:**

;;;======================================================

;;; Hardware troubleshoot Expert System

;;;

;;; This expert system diagnoses some desk and mobile

;;; PC hardware issues.

;;;

;;;

;;; To execute merely, load, reset and run.

;;;======================================================

;;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;;\* DEFFUNCTIONS \*

;;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(deffunction ask-question (?question $?allowed-values)

(printout t ?question)

(bind ?answer (read))

(if (lexemep ?answer)

then (bind ?answer (lowcase ?answer)))

(while (not (member ?answer ?allowed-values)) do

(printout t ?question)

(bind ?answer (read))

(if (lexemep ?answer)

then (bind ?answer (lowcase ?answer))))

?answer)

(deffunction yes-or-no-p (?question)

(bind ?response (ask-question ?question yes no y n si no ))

(if (or (eq ?response yes) (eq ?response y))

then TRUE

else FALSE))

;;;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;;;\* PC STATE RULES \*

;;;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(defrule power-on

(declare (salience 100))

(initial-fact)

=>

(if (yes-or-no-p "can your computer turn on(yes/no)?")

then

(assert (computer-turn-on yes))

else

(assert (computer-turn-on no))

)

)

(defrule Blue-issue

(declare (salience 80))

(computer-turn-on yes)

=>

(if (yes-or-no-p "Have you seen a blue screen(yes/no)?")

then

(assert (computer-blue-screen yes))

(assert (repair "Execute a hard drive failures test."))

else

(assert (computer-blue-screen no))

)

)

(defrule Batterie-RTC-issue

(declare (salience 80))

(computer-turn-on yes)

(computer-blue-screen no)

=>

(if (yes-or-no-p "Have you seen on launch a warning at beginning and include a work like (batterie or RTC)(yes/no)?")

then

(assert (repair "Change the RTC battery that is in the motherboard."))

(assert (computer-BIOS-warning yes))

else

(assert (computer-BIOS-warning no))

)

)

(defrule Nostart-OS-issue

(declare (salience 80))

(computer-turn-on yes)

(computer-blue-screen no)

(computer-BIOS-warning no)

=>

(if (yes-or-no-p "Have you seen on initial launch a warning like this 'No format to boot' (yes/no)?")

then

(assert (computer-BIOS-NOBOOT yes))

(assert (repair "Recover the MBR.(Master Boot Register)"))

else

(assert (computer-BIOS-NOBOOT no))

)

)

(defrule NoTurn-on-PC

(declare (salience 80))

(computer-turn-on no)

=>

(assert (repair "Please check if power cable have a good connection or power supply has been properly powered."))

)

(defrule NoTurn-on-PC2

(declare (salience 70))

(computer-turn-on no)

=>

(if (yes-or-no-p "Power cable is good and power supply has been powered(yes/no)?")

then

(assert (repair "Motherboard or internal circuitry may have a several problem. Consult your PC technician."))

else

(assert (repair "Buy a new power cord or replace power supply unit according to your PC if the cables and power are good."))

)

)

(defrule Low-resolution-issue-yes

(declare (salience 80))

(computer-turn-on yes)

(computer-blue-screen no)

(computer-half-screen yes)

=>

(if (yes-or-no-p "There is not video or in low resolution(yes/no)?")

then

(assert (computer-half-screen yes))

(assert (repair "Please check if video card driver are correct installed or the computer has last drivers updates."))

else

(assert (computer-half-screen no))

)

)

(defrule HDD-issue-appear

(declare (salience 70))

(computer-turn-on yes)

(computer-blue-screen yes)

=>

(if (yes-or-no-p "HDD test found any error in the PC(yes/no)?")

then

(assert (computer-HDD-damage yes))

(assert (repair "Hard drive is damage,replace or format it."))

else

(assert (computer-HDD-damage no))

)

)

(defrule half-screen-issue

(declare (salience 80))

(computer-turn-on yes)

(computer-BIOS-NOBOOT no)

(computer-blue-screen no)

(computer-BIOS-warning no)

=>

(if (yes-or-no-p "Have you seen a half screen(yes/no)?")

then

(assert (computer-half-screen yes))

else

(assert (computer-half-screen no))

)

)

(defrule RAM-issue

(declare (salience 80))

(computer-turn-on yes)

(computer-blue-screen yes)

(computer-HDD-damage no)

=>

(if (yes-or-no-p "Are RAM memories correct placed(yes/no)?")

then

(assert (RAM-memories-damage yes))

(assert (repair "Remplace RAM memories."))

else

(assert (RAM-memories-damage no))

(assert (repair "Make sure RAM memories are correct placed on mother board slots."))

)

)

(defrule No-problem-or-no-data

(declare (salience 50))

(computer-turn-on yes)

(computer-blue-screen no)

(computer-BIOS-warning no)

(computer-BIOS-NOBOOT no)

(computer-half-screen no)

=>

(printout t crlf "There is no problem with your computer or there is not in the expert system database."crlf )

(printout t crlf "Press any key to exit"crlf)

(bind ?anything (readline))

(exit)

)

;;;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;;;\* STARTUP AND REPAIR RULES \*

;;;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

(defrule system-banner ""

(declare (salience 110))

=>

(system "reset")

(printout t crlf crlf)

(printout t "Hardware troubleshoot Expert System for PC")

(printout t crlf crlf))

(defrule print-repair ""

(declare (salience 90))

(repair ?item)

=>

(printout t crlf crlf)

(printout t "Suggest :")

(printout t crlf crlf)

(assert (ask-done yes))

(format t " %s%n%n%n" ?item))

(defrule ask-if-end ""

(declare (salience 0))

(ask-done yes)

=>

(if (yes-or-no-p "This solved the problem(yes/no)?")

then

(printout t "Goodbye..." crlf crlf)

(exit)

else

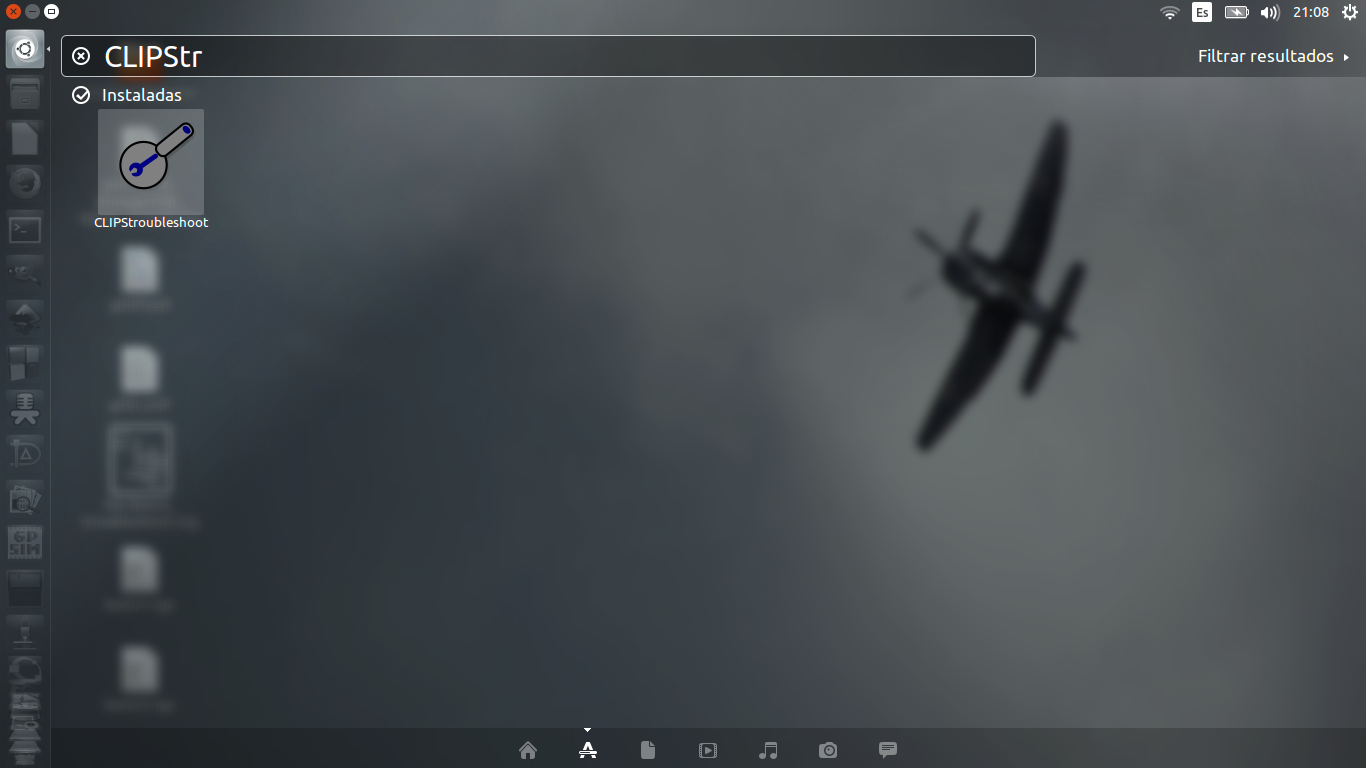
(reset)

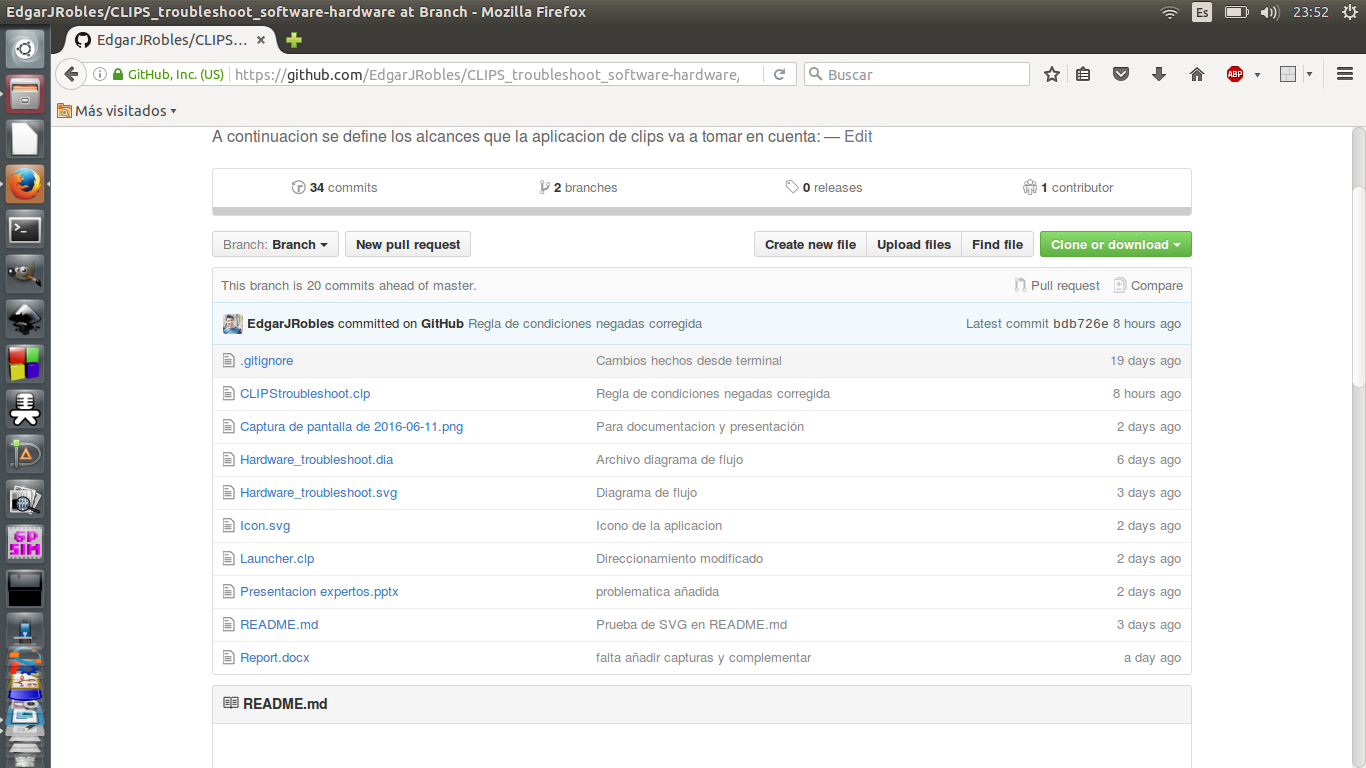
(run)

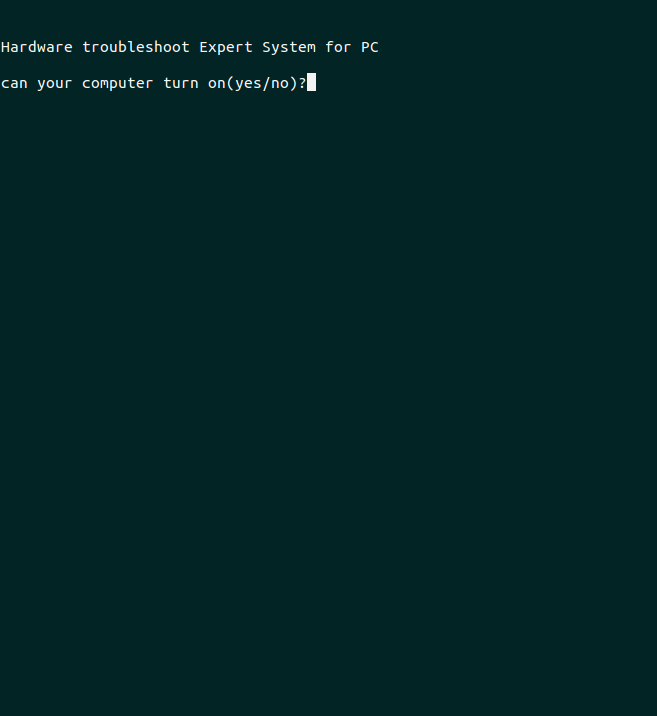
)

)

**Screen shots:**

****

****

****

**Conclusion:**

This expert system is capable of help a noob user to repair hardware issues with his computer. At the same time the user gets familiar with the symptoms of a certain fail. Is easy to use and simple, so it could be a great educative resource for training technicians and anyone who wants to learn how to fix computers. Is true that it needs a much larger database in order to help with more complex fails, but it can be done with ease and anyone can contribute because is open source, you can be found in: <https://github.com/EdgarJRobles/CLIPS_troubleshoot_software-hardware>

Also we learn how to implement an expert system for a real life solution that can be helpful for the society. It don’t needs to be very complex, so by these way much more people can help in the development, not only advanced users that would not be interested in helping. This is why we decided to keep this easy and understandable.