extends Node

#Handle Post Discussion Page

var url = "discussion"

func \_ready():

root.set\_screen\_orientation(0)

$PostDiscussion\_Btn.connect("pressed",self,"post\_discussion\_pressed")

$Post\_Discussion\_Success\_Popup.hide()

$RichTextLabel3.hide()

pass

func post\_discussion\_pressed():

var discussion\_title = $TextEdit.get\_text()

var discussion\_details = $TextEdit2.get\_text()

if (discussion\_title == "" && discussion\_details == ""):

$RichTextLabel3.clear()

$RichTextLabel3.add\_text("Please input discussion title and details")

$RichTextLabel3.show()

elif (discussion\_title == ""):

$RichTextLabel3.clear()

$RichTextLabel3.add\_text("Please input discussion title")

$RichTextLabel3.show()

elif (discussion\_details == ""):

$RichTextLabel3.clear()

$RichTextLabel3.add\_text("Please input discussion details")

$RichTextLabel3.show()

else:

$loading.popup()

$RichTextLabel3.clear()

var user\_id = userModel.getUserId()

var datetime\_dictionary = OS.get\_datetime()

var second = datetime\_dictionary["second"]

if (second <= 9):

second = '0' + str(datetime\_dictionary["second"])

var time\_created = str(datetime\_dictionary["year"]) + '-' + str(datetime\_dictionary["month"]) + '-' + str(datetime\_dictionary["day"]) + ' ' + str(datetime\_dictionary["hour"]) + ':' + str(datetime\_dictionary["minute"]) + ':' + str(second)

var apiUrl = url + '/' + user\_id

var data = {"title": discussion\_title, "details": discussion\_details, "time": time\_created}

apiController.apiCallPost(data,apiUrl)

apiController.apiCallPost(data,apiUrl)

yield(apiController, "request\_completed")

var result = apiController.getResult()

var responseCode = apiController.getResponseCode()

$loading.hide()

if (str(responseCode) == "200"):

$Post\_Discussion\_Success\_Popup.show()

else:

$RichTextLabel3.add\_text("Error posting discussion. Please try again!")

$RichTextLabel3.show()

func \_notification(what):

if (what == MainLoop.NOTIFICATION\_WM\_GO\_BACK\_REQUEST):

root.return\_to\_last()

func \_input(delta):

if Input.is\_action\_pressed('ui\_cancel'):

if(userModel.getUserRole() == 'Student'):

root.return\_to\_last()