Rina API

# Page elements

* Avatar
* Talkbox
* Choicebox
* Alertbox
* Appbox

# Rina App API

manifest.json

* name
* version
* description

main.py

* void emote(Emotions emotion)
  + Sets the avatar emotion to the given emotion
* void say(String str)
  + Makes the talkbox display the passed string
* String ask(String str, [List<String> choices])
  + Makes talkbox display str as question and returns user input
  + If choices are given, displays choices as button choices
* void alert(String str)
  + Displays an alert dialogue box with the str which can be closed by clicking OK.
* bool confirm(String str)
  + Displays an confirm dialogue box with the str which can be closed with either OK (returns true) or Cancel (returns false)
* void openAppBox(React.Component comp)
  + Opens up the Appbox and generates the React Component comp
* void closeAppBox()
  + Closes the Appbox
* void end()
  + Ends the app
* bool varExists(String varname)
  + Checks if a variable with varname has been created
* void createVar(String varname, [Anytype value])
  + Creates a variable with the name varname with a null value or an optional given value
* void createVarIfAbsent(String varname, [Anytype value])
  + Creates a variable with the name varname with a null value or an optional given value IF it doesn’t already exist
* void setVar(String varname, Anytype value)
  + Sets the value of a variable named varname to the given value
* Anytype getVar(String varname)
  + Returns the value of the variable named varname
* void destroyVar(String varname)
  + Destroys the variable with the name varname

# Skin API

manifest.json

* name
* version
* description

# Other

* enum Emotions
  + HAPPY
  + SAD
  + ANGRY
  + BLUSHING
  + NEUTRAL
  + SERIOUS
  + SMUG
  + WINKING

When app begins:

* Create a new thread with the RinaApp instance
* Thread sends back first command and waits for next call

Upon next call:

* Make thread move on to next command and send it back
* Then thread waits for next call

Note:

* Thread is destroyed when app closes
* App closes upon session end
* Thread is linked to both session and request