

Virtual Pet

Interface Design - Components



Properties

TableArrangement1: Columns - 4

updateClock: 1000

All buttons: Width - 75 pixels, Height - 75 pixels

moodClock: 10000

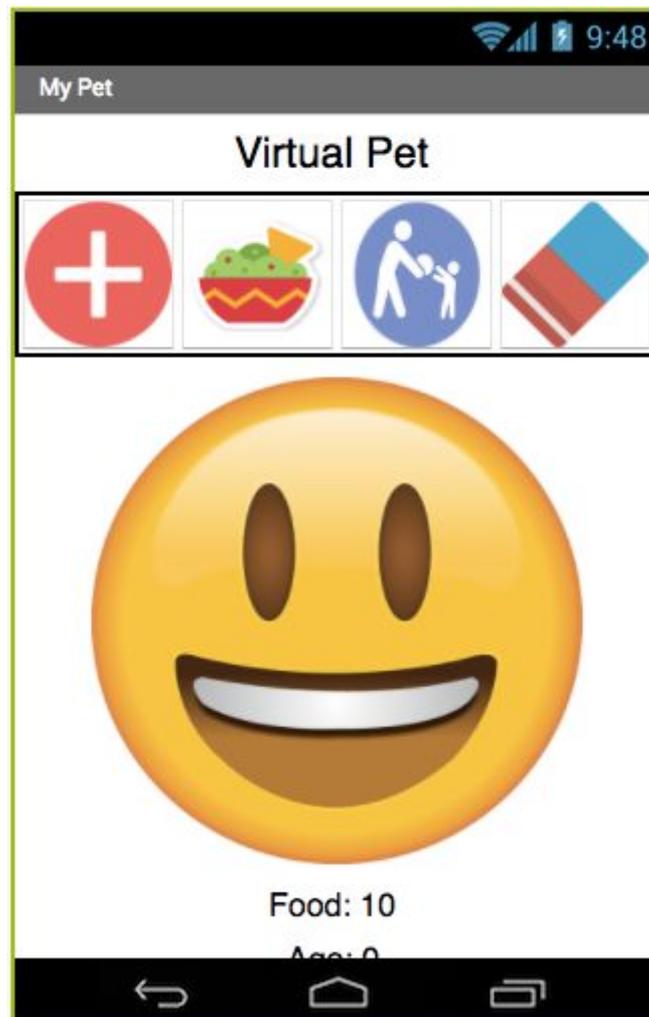
All labels: Font size - 18 pixels

lifeClock: TimeInterval - 30000

foodClock: TimeInterval - 20000

NOTE: You can download your own Emoji images from here: [Emoji](#)

How it looks



Blocks

The following blocks of codes setup variables that will store all the data.



When the user clicks on the 'New' button the following code will be executed. It resets the variables to their initial state and alerts the user that their pet is alive!

```

when new_pet_btn .Click
do
  call Notifier1 .ShowAlert
  notice "Your pet is alive "
  set global age to 0
  set global waste to 0
  set global mood to 0
  set global food to 10

```

When your pet is complaining of being hungry you must click the 'Feed' button. This will add one to the food counter. Although, you cannot overfeed the pet!

```

when feed_pet_btn .Click
do
  if get global food ≠ 10
  then
    set global food to get global food + 1
    set food_lbl .Text to join "Food: "
    get global food

```

We need to constantly check if your emoji/pet is hungry and we have achieved this by using a clock component that fires every second. The code below checks if your emoji/pet's food level is below a certain threshold and if so it will display a 'sad' emoji icon.

NOTE: The emoji images are uploaded in the 'Designer' view via the media section.

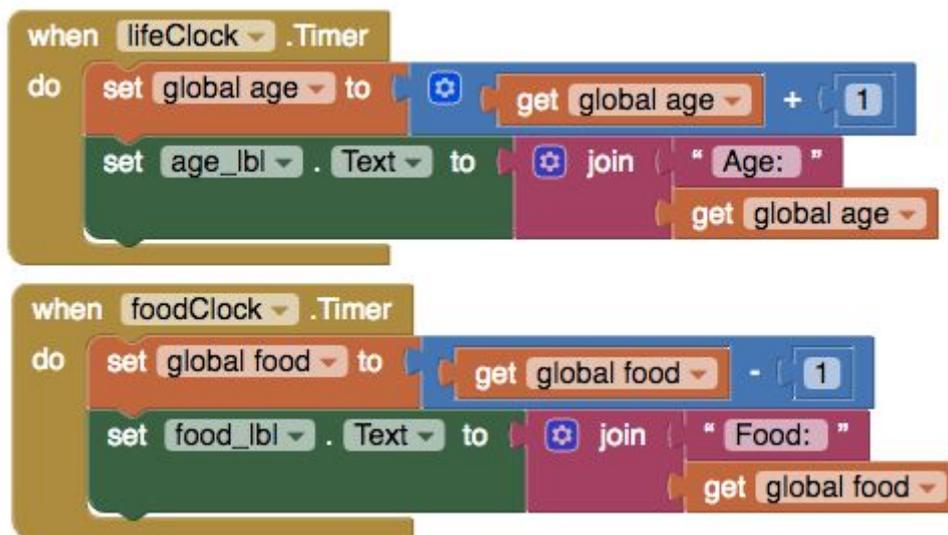
```

when updateClock .Timer
do
  if get global food < 7
  then
    set Emoji_img .Picture to "Sad_Face_Emoji.png "
  if get global food ≥ 7
  then
    set Emoji_img .Picture to "Smiling_Emoji_with_Eyes_Opened.png "

```

The code below is executed after a certain amount of time and will increase you emoji/pets age by one and decrease their food by one. The 'Clock' component is very important in App

Inventor as it allows us to execute code after a certain period of time rather than waiting for user interaction.



Challenges

- Implement the 'Clear' button (looks like an eraser). When this is clicked, all the attributes are reset to their initial state.
- Your pet cannot be happy all the time so add a feature that displays different moods! You will have to change the code in the 'update clock' event.
- Add a new button that allows the user to interact with the emoji/pet in a different way.