# Projets

Lua

Dropbox

# Processing

Codea

# Github

FaceTime
Camera
Contacts
Notes
Reminders
Safari
1

Main | Code | WriteText | FormatText | Scrollable | ThemeDefault

```
    return y - self.dr.height*.3 < self.limitDown and self.limitDown - self.dr.height*.3 or self.limitUp - self.dr.height*.3
end

function Scrollable:back(name, time, easing, callback) -- launch a tween for back to normal state
    if not self:isOut() then return end
    time = time or .2
    easing = easing or tween.easing.quintOut
    self[name] = tween(time, self.tr, {
        y = self:getLimit()
    }, easing, callback or function()
        self[name] = nil
    end)
end

function Scrollable:stop(...) -- stop a tween if not nil
    for _,tw in pairs{arg} do
        if self[tw] then
            tween.stop(self[tw])
        end
    end
end

function Scrollable:draw(customDrawBefore)
    pushStyle()
    spriteMode(CENTER)
    pushMatrix()
    translate(self.tr.x, self.tr.y)
    self.dr:draw(customDrawBefore)
    popMatrix()
    if self.debug then self:debugDraw() end
    popStyle()
end

function Scrollable:debugDraw()
    -- draw limits
    strokeWidth(3)
    stroke(255, 0, 0)
    line(0, self.limitUp, WIDTH, self.limitUp)
    stroke(0, 0, 255)
    line(0, self.limitDown, WIDTH, self.limitDown)

    alinea(WIDTH-5, self.tr.y + self.dr.height*.5, .75)
```

Made with Codea

# Codeaception

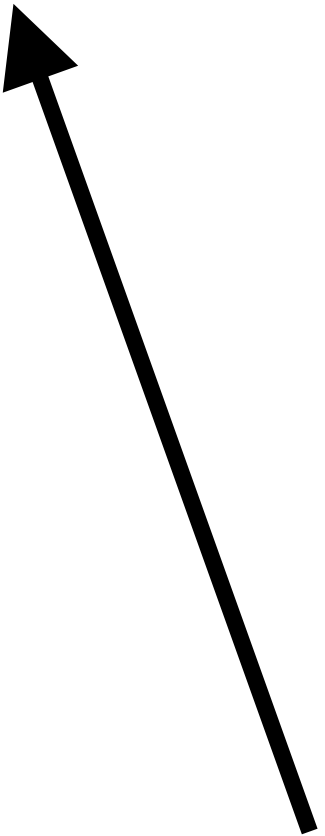
# APIs

# **ExpressLua** (WebServer)

like express for Node.js

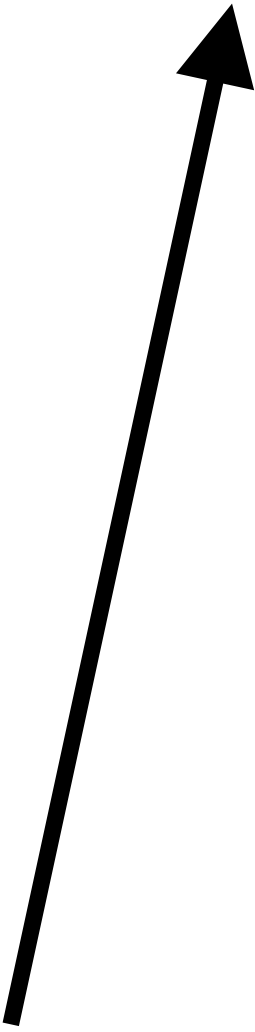
# AirCodea

edit code in web browser with hot-reload



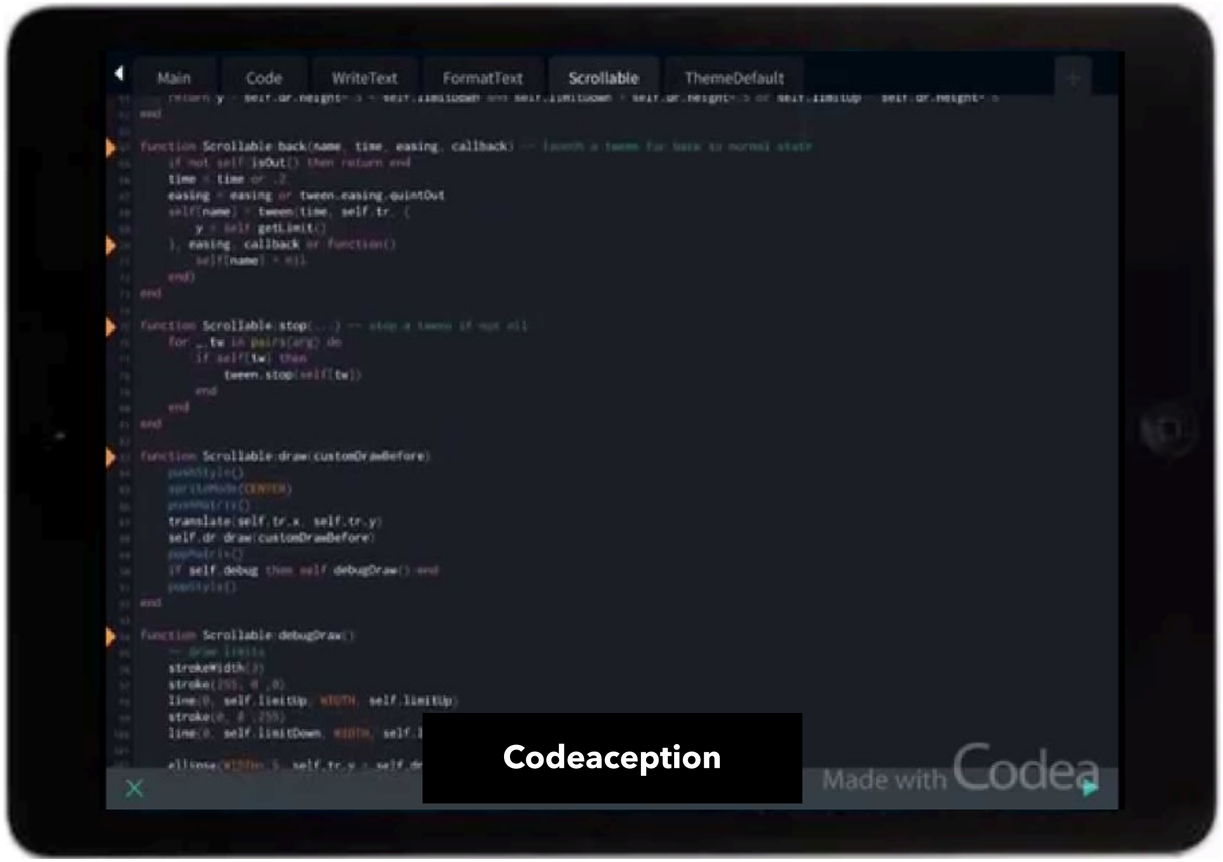
# Required

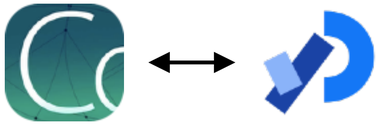
package manager + version control



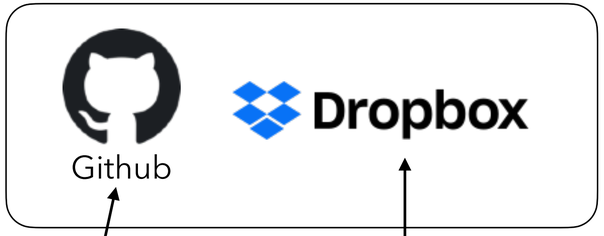
# SyncCodea

# Projets






**APIs**




**SyncCodea**

**Required**
package manager + version control

**ExpressLua** (WebServer)
like express for Node.js

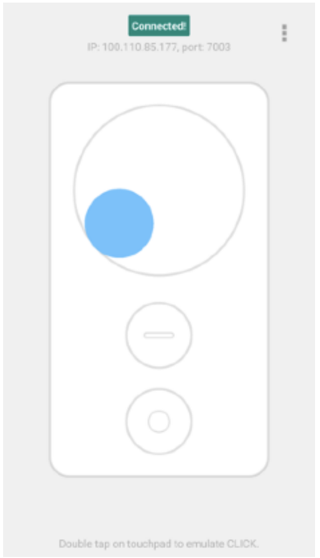
**AirCodea**
edit code in web browser with hot-reload

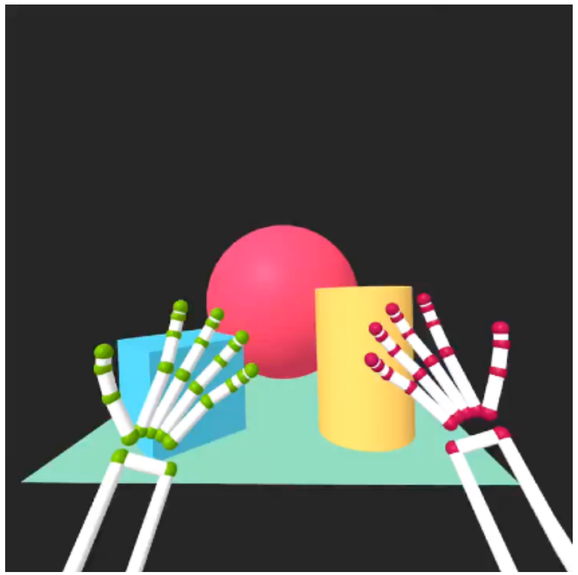
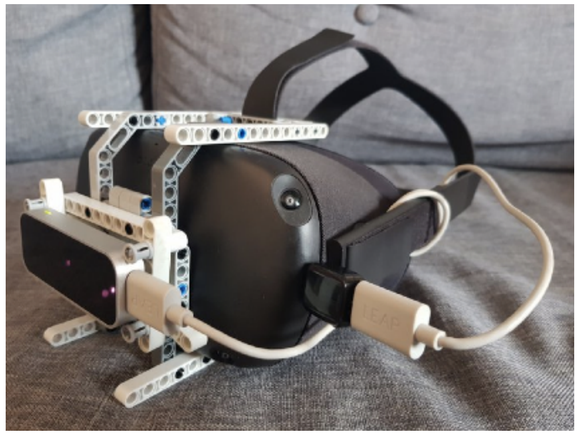
# Projets

VIRTUALHERE

+ Unity
- Network
- Remote

Controller Emulator for iOs





hubs by moz://a