MARYTINI

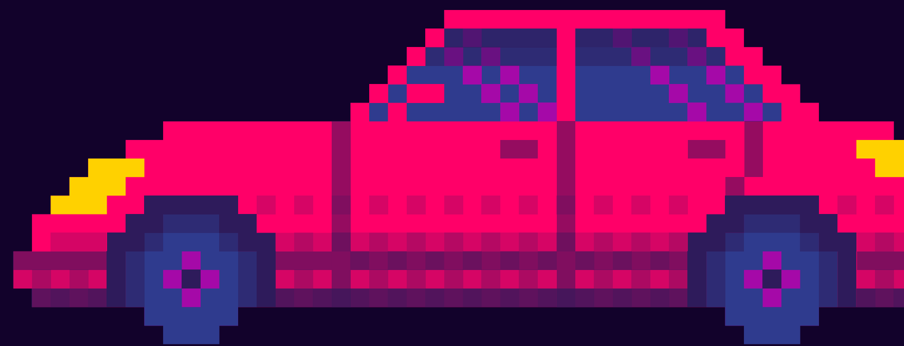
DEVELOPMENT BY:
SARANSH SONDHI
ARNAV AGRAWAL

INTRODUCTION

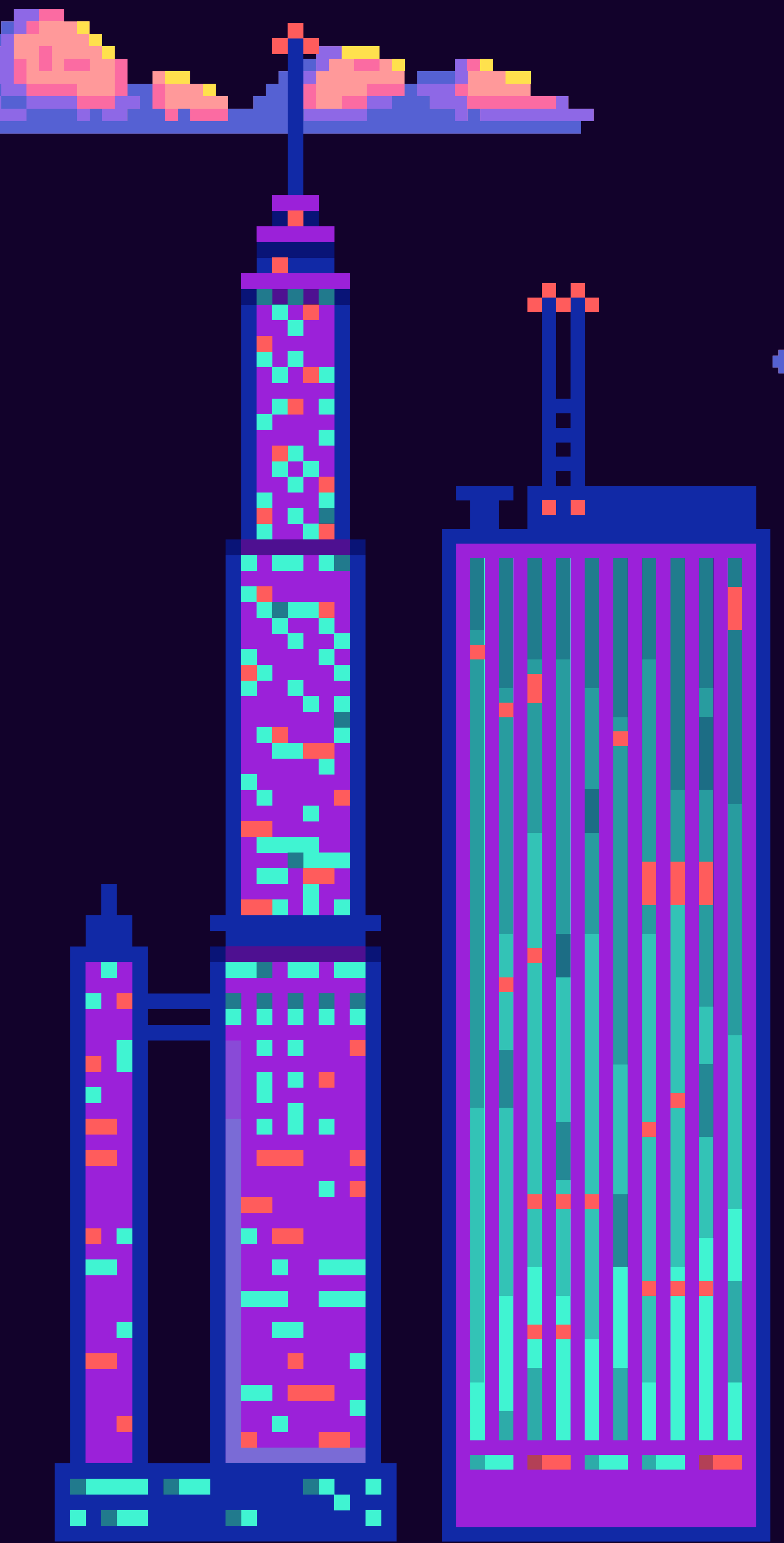
A ROGUELIKE CARD GAME, MADE AS COMPUTER WEB APPLICATION

PLAYERS TAKE ON THE ROLE OF A BARTENDER,
PLAYERS MUST PIECE TOGETHER DETAILS ABOUT THEIR CUSTOMERS' TARGET DRINKS.

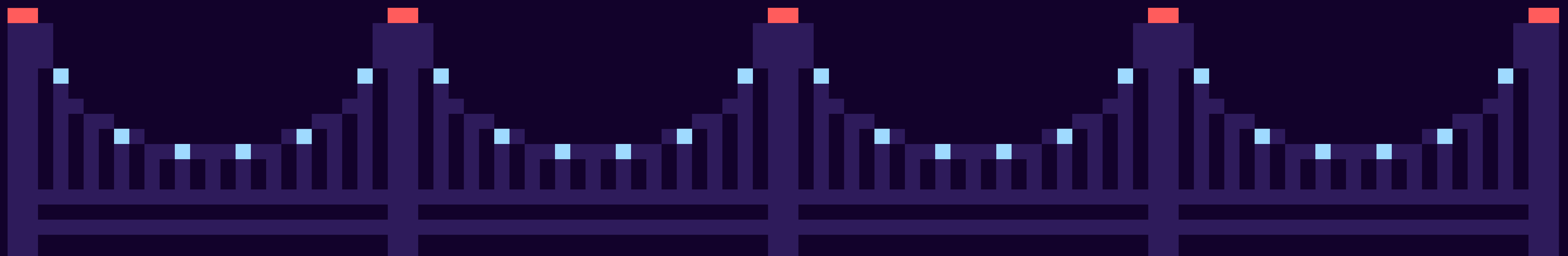
EARN POINTS BY STRATEGICALLY MIXING INGREDIENTS IN AN ALCHEMICAL FASHION

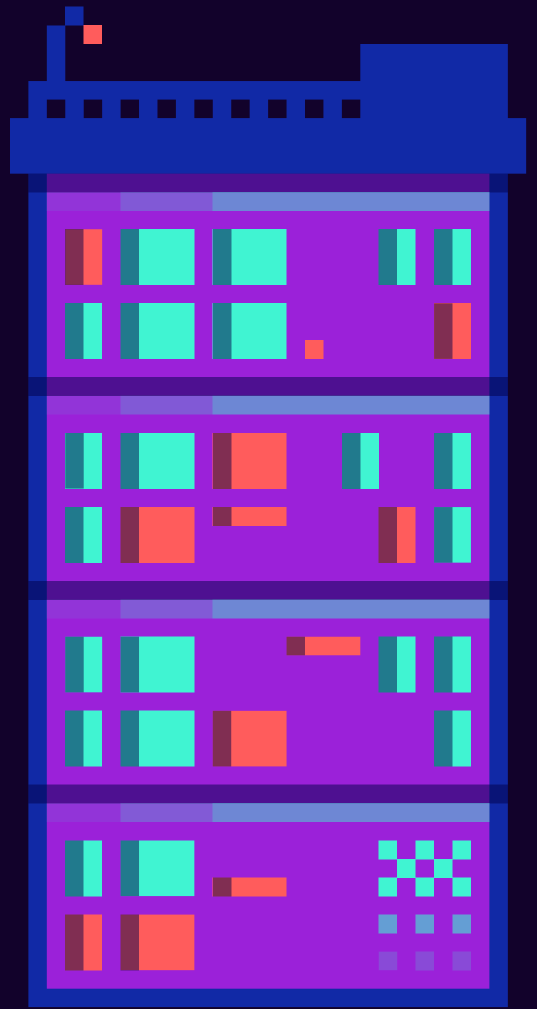


INSPIRATION AND CONCEPT



DRAWING HEAVY INSPIRATION FROM BALATRO, WE AIMED TO BUILD A DECKBUILDING ROGUELIKE WITH THE THEME OF BARTENDING TO MAKE DRINKS. A DRINK DESCRIPTION IS GIVEN AND THE PLAYER MUST DRAW CARDS AND WHIP UP A DRINK TO SATISFACTION AND HENCE EARN PROFITS, THE GOAL WAS SIMPLY TO AIM FOR THE HIGHEST SCORE. ADDITIONALLY WE THOUGHT OF SECONDARY ELEMENTS WHERE THE PLAYER IS UNDERCOVER AS AN INFORMATION BROKER.



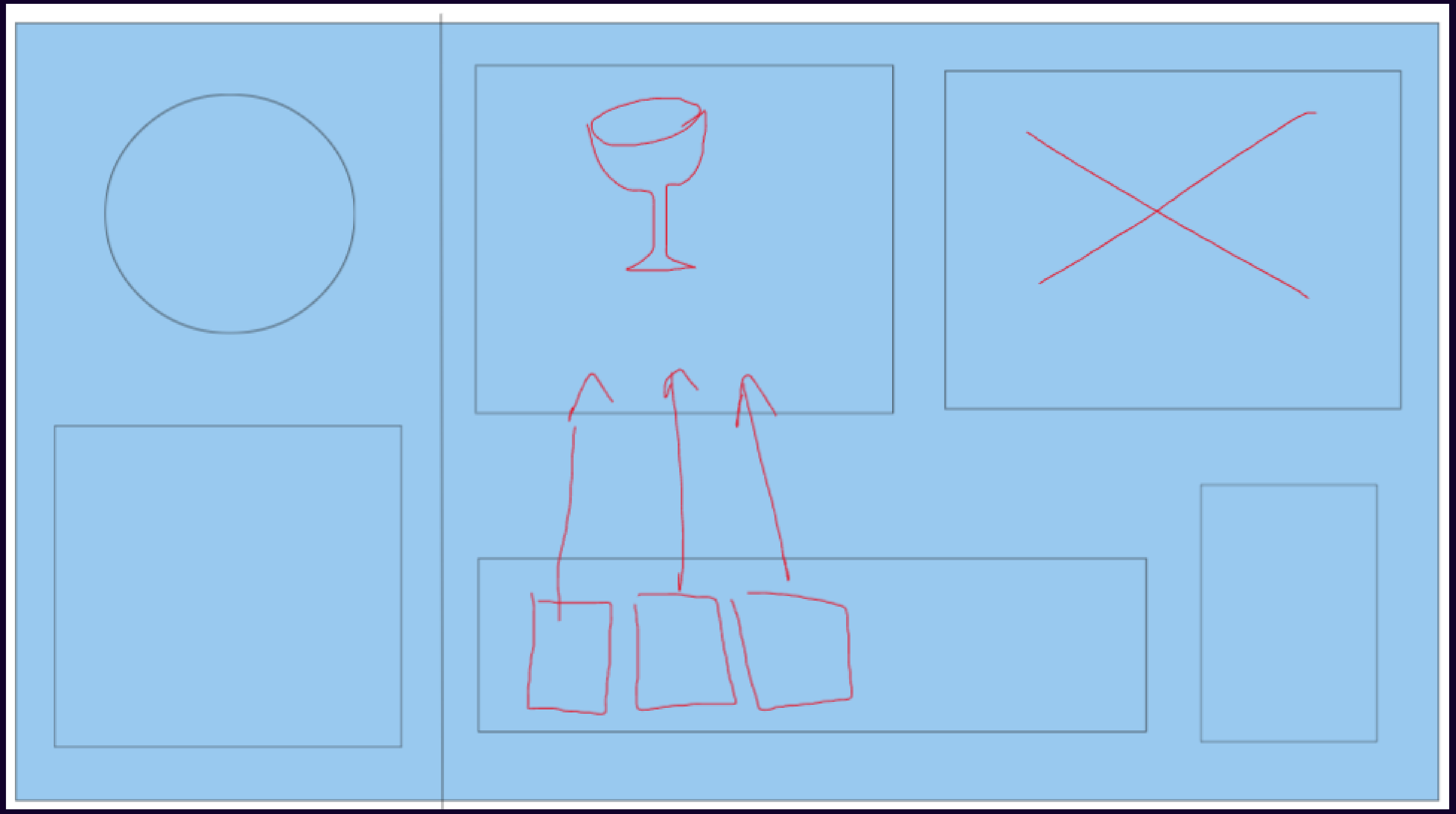


INITIAL IDEAS AND EVOLUTION

01 INGREDIENTS ARE AVAILABLE TO PLAYERS ON A DECK, PULLING A CARD COSTS SOME MONEY

02 2 TIER MIXING WHERE YOU TOOK TWO INGREDIENTS TO MAKE ANOTHER SPECIAL ONE

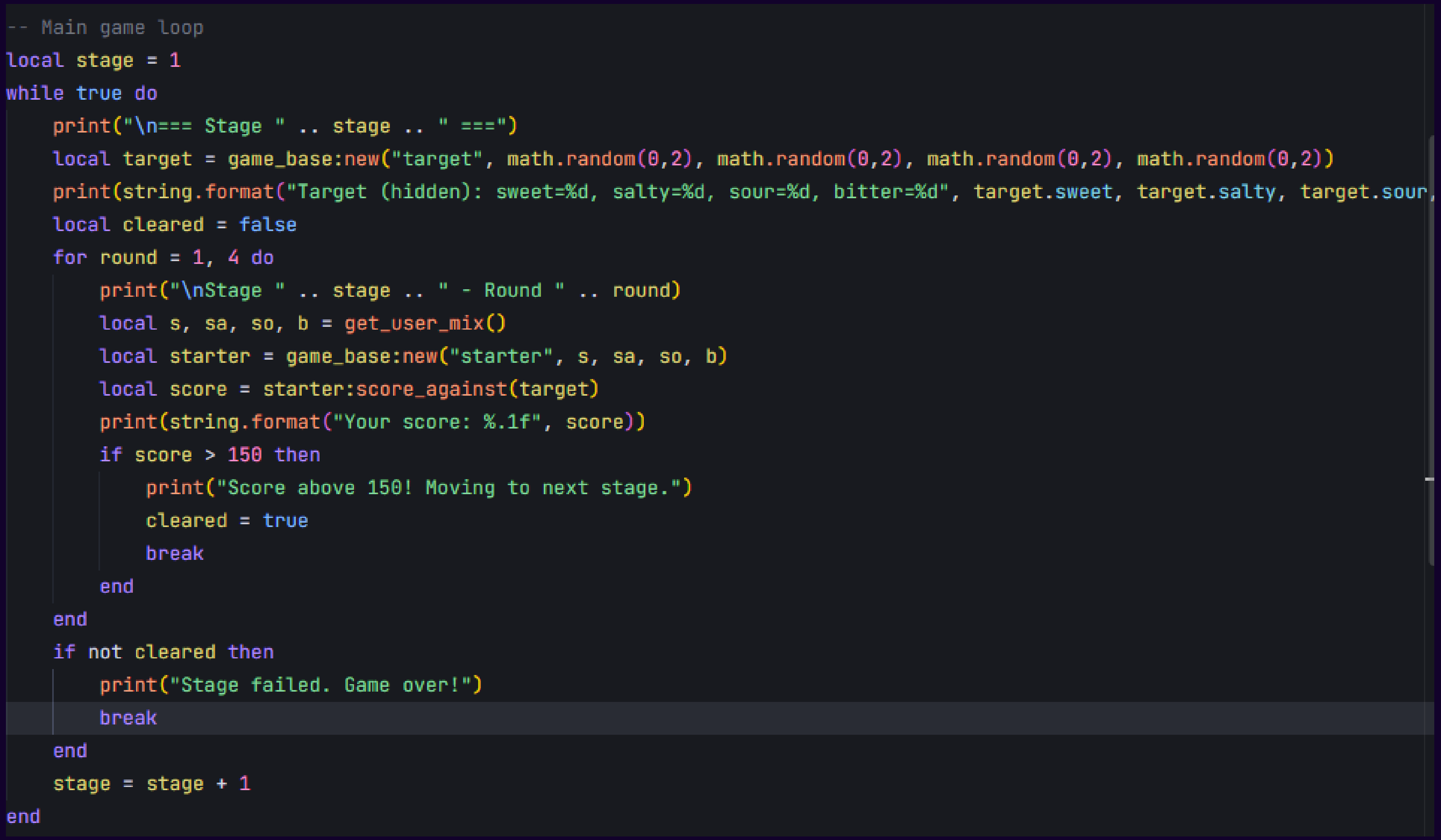
03 CREATING SCORE CALCULATION METRICS FROM MULTIPLE PARAMETERS



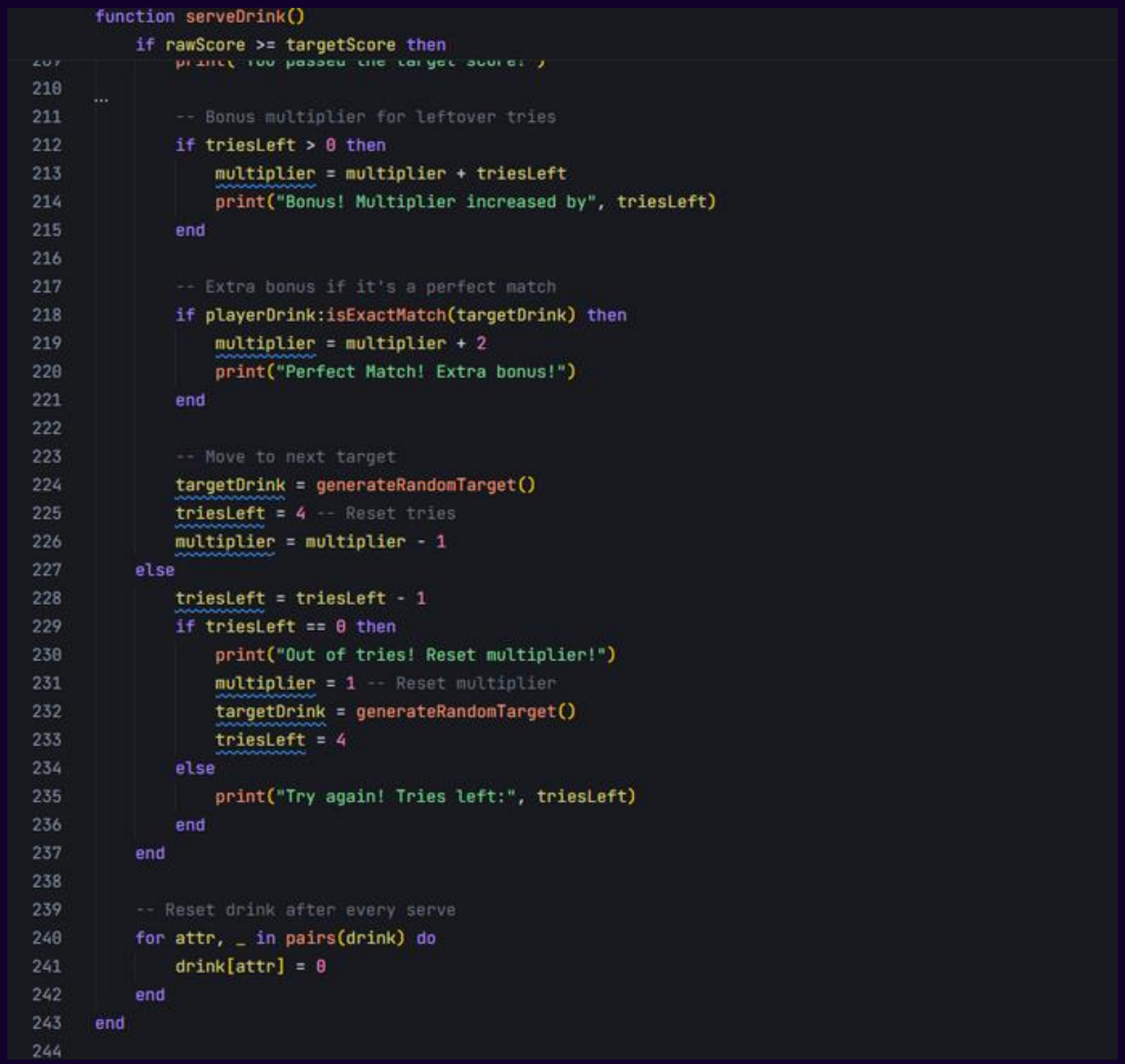
LEARNING AND DEVELOPMENT

01 LEARNING BASICS OF LUA SCRIPTING

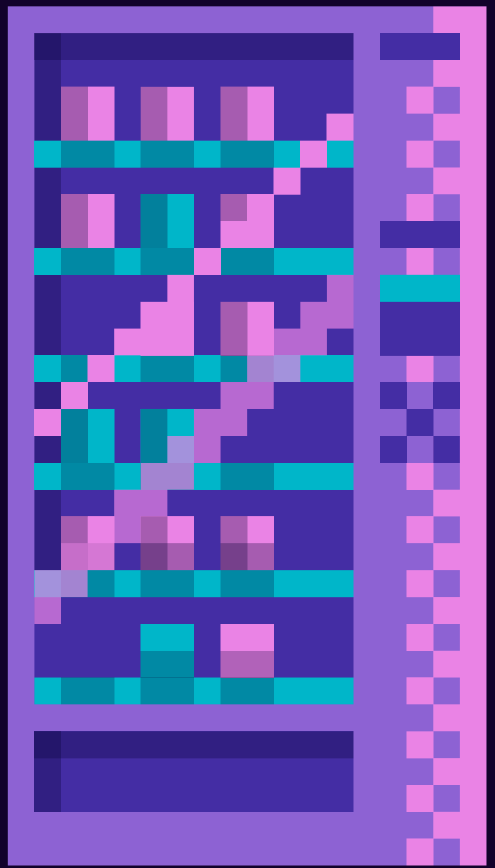
02 UNDERSTANDING THE LOVE 2D FRAMEWORK



```lua
-- Main game loop
local stage = 1
while true do
    print("\n=== Stage " .. stage .. " ===")
    local target = game_base:new("target", math.random(0,2), math.random(0,2), math.random(0,2), math.random(0,2))
    print(string.format("Target (hidden): sweet=%d, salty=%d, sour=%d, bitter=%d", target.sweet, target.salty, target.sour,
    local cleared = false
    for round = 1, 4 do
        print("\nStage " .. stage .. " - Round " .. round)
        local s, sa, so, b = get_user_mix()
        local starter = game_base:new("starter", s, sa, so, b)
        local score = starter:score_against(target)
        print(string.format("Your score: %.1f", score))
        if score > 150 then
            print("Score above 150! Moving to next stage.")
            cleared = true
            break
        end
    end
    if not cleared then
        print("Stage failed. Game over!")
        break
    end
    stage = stage + 1
end
```



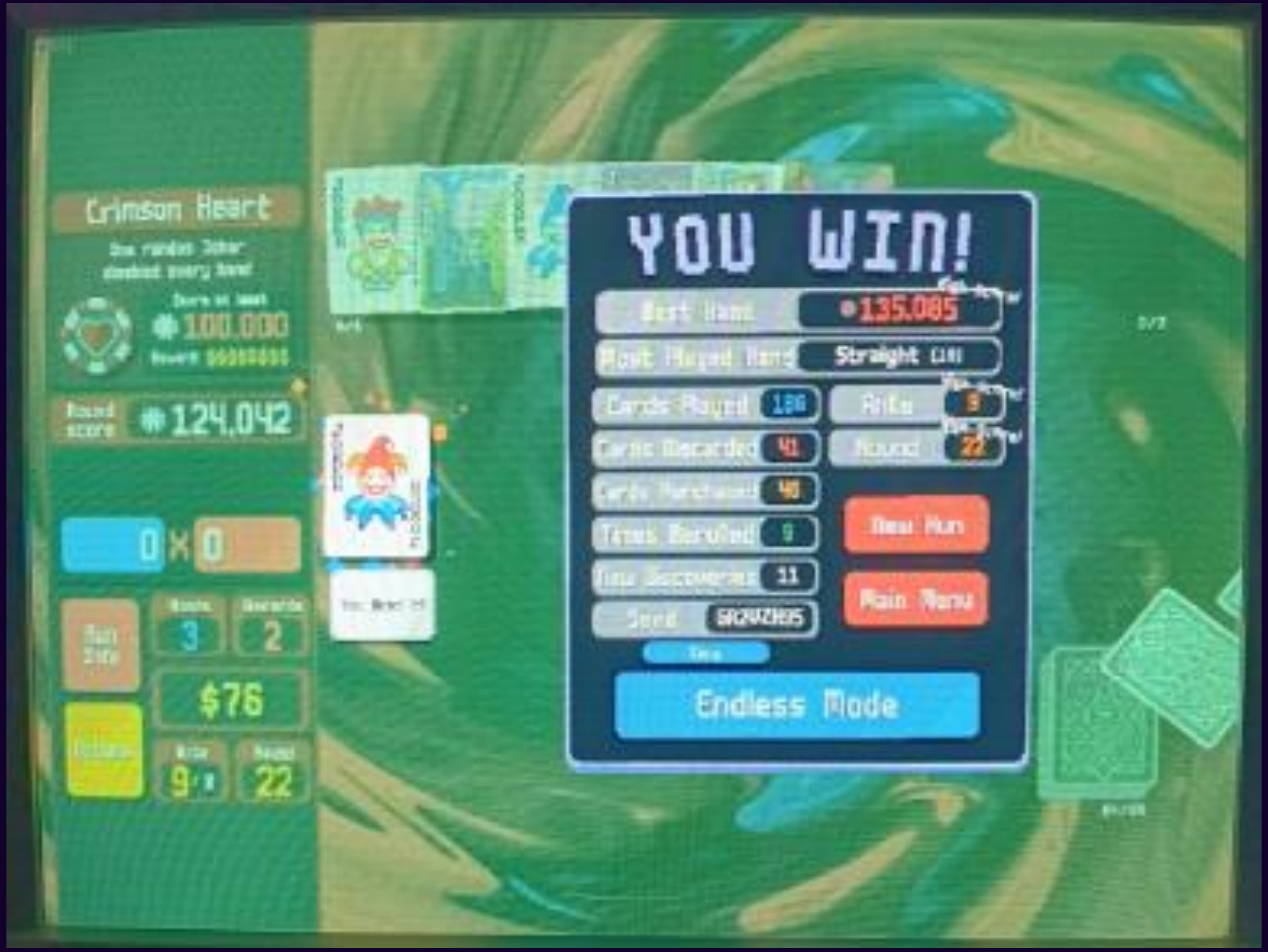
```lua
function serveDrink()
    if rawScore >= targetScore then
        print("You passed the target score!")

        -- Bonus multiplier for leftover tries
        if triesLeft > 0 then
            multiplier = multiplier + triesLeft
            print("Bonus! Multiplier increased by", triesLeft)
        end

        -- Extra bonus if it's a perfect match
        if playerDrink:isExactMatch(targetDrink) then
            multiplier = multiplier + 2
            print("Perfect Match! Extra bonus!")
        end

        -- Move to next target
        targetDrink = generateRandomTarget()
        triesLeft = 4 -- Reset tries
        multiplier = multiplier - 1
    else
        triesLeft = triesLeft - 1
        if triesLeft == 0 then
            print("Out of tries! Reset multiplier!")
            multiplier = 1 -- Reset multiplier
            targetDrink = generateRandomTarget()
            triesLeft = 4
        else
            print("Try again! Tries left:", triesLeft)
        end
    end

    -- Reset drink after every serve
    for attr, _ in pairs(drink) do
        drink[attr] = 0
    end
end
```

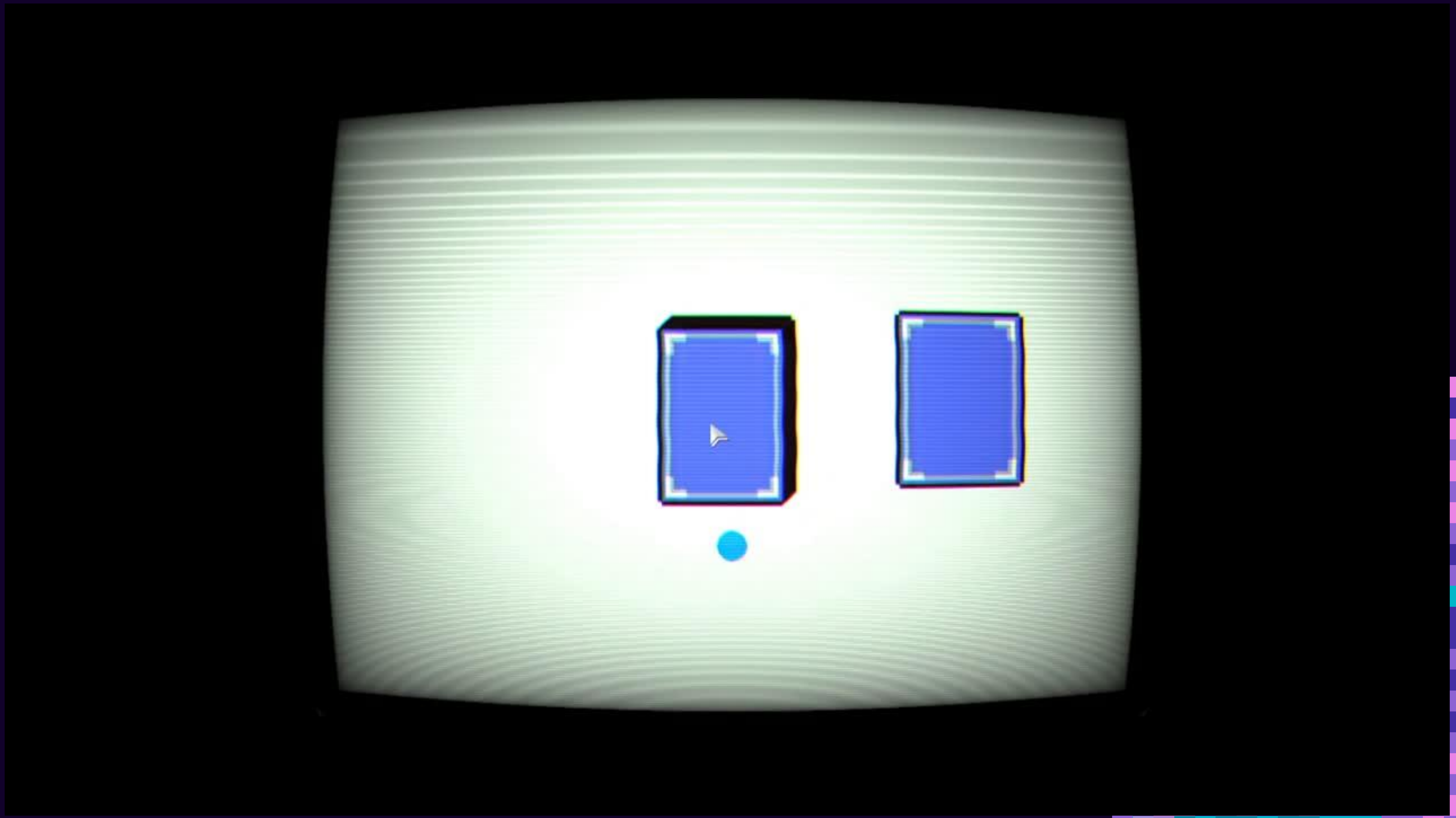


LEARNING AND DEVELOPMENT

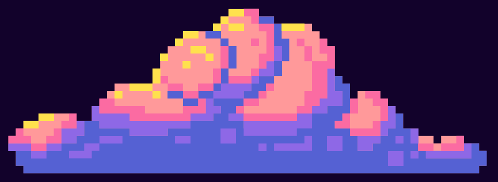
03 BALATRO CARD MOVEMENT

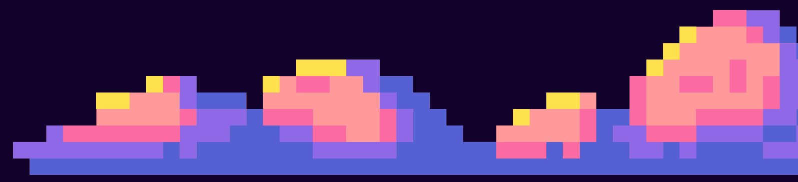
04 UNDERSTANDING THE LOVE 2D FRAMEWORK





GAME MECHANICS





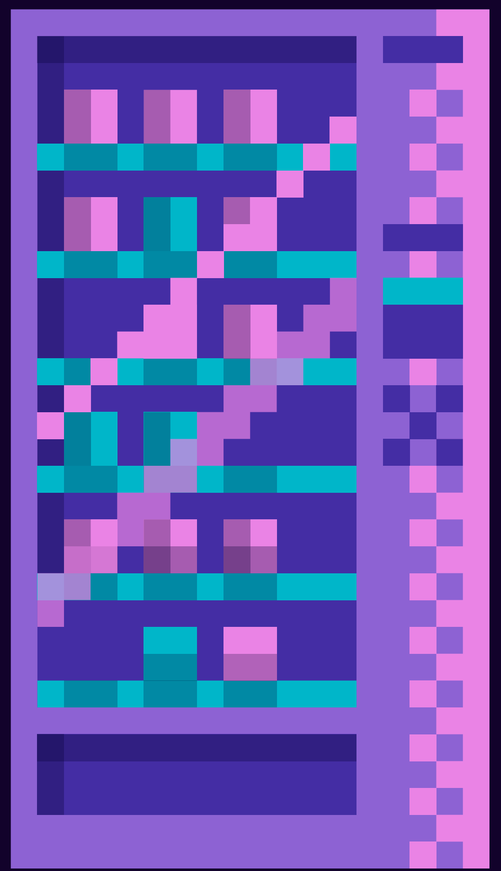
01 GAME TAKES PLAYS IN STAGES WHERE EACH STAGE CONSISTS OF 4 ROUNDS.

02 THE PLAYER HAS 4 CARDS VISIBLE. CARD ELEMENTS INCLUDE SOUR, SALTY, SWEET AND BITTERN.

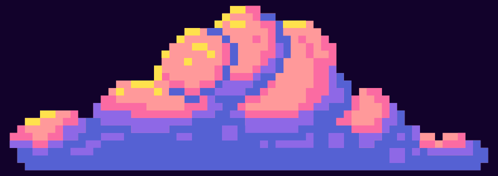
03 THE PLAYER MUST SELECT A LEVEL OF INTENSITY AND OBTAIN A SCORE

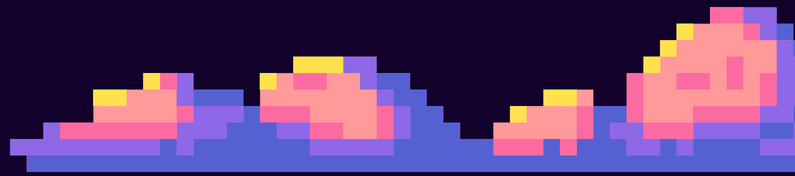
04 PASS THE REQUIRED SCORE AND MOVE ON TO THE NEXT LEVEL

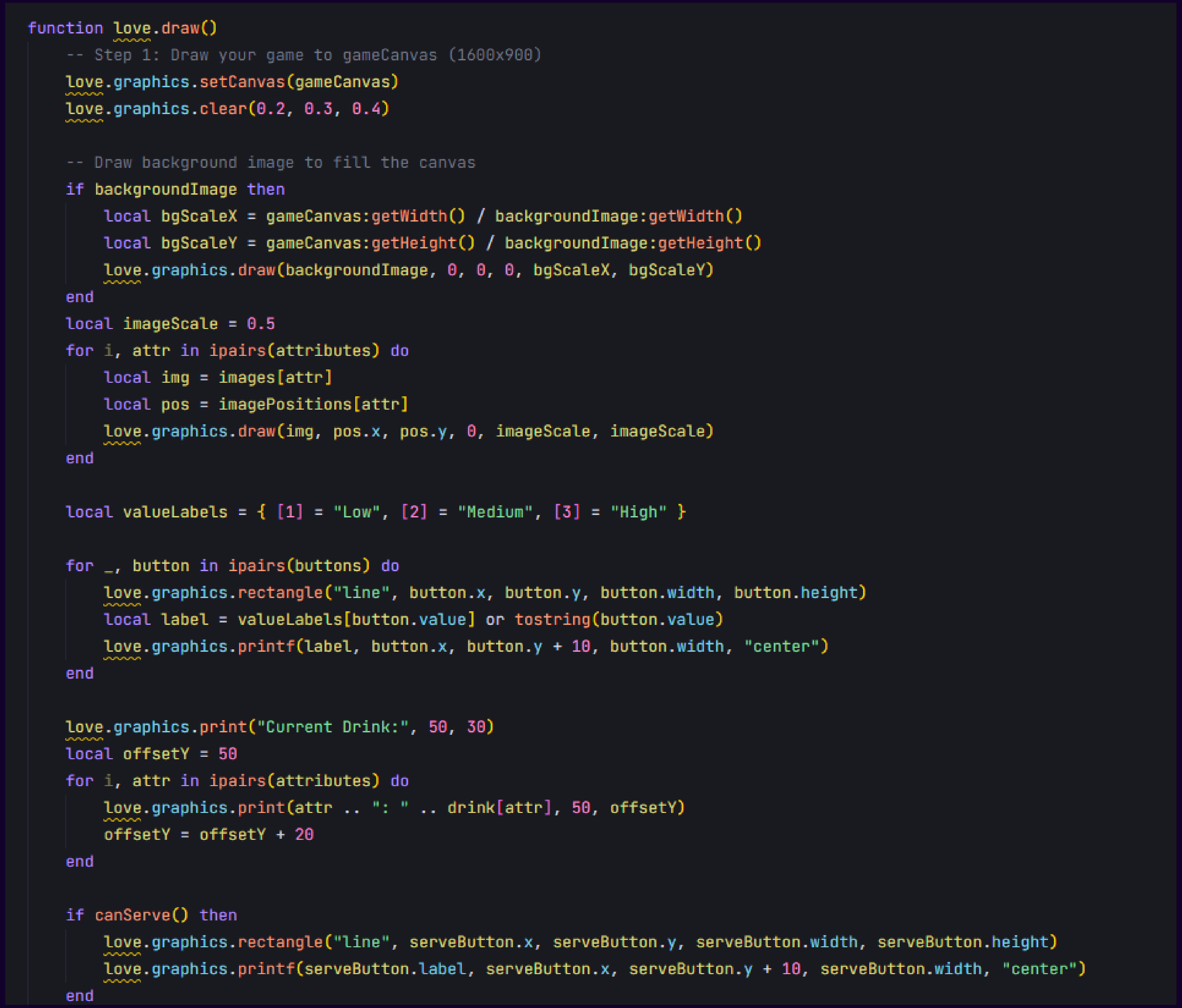
05 FAILURE TO REACH MINIMUM SCORE IN 4 ROUNDS RESULTS IN GAME OVER



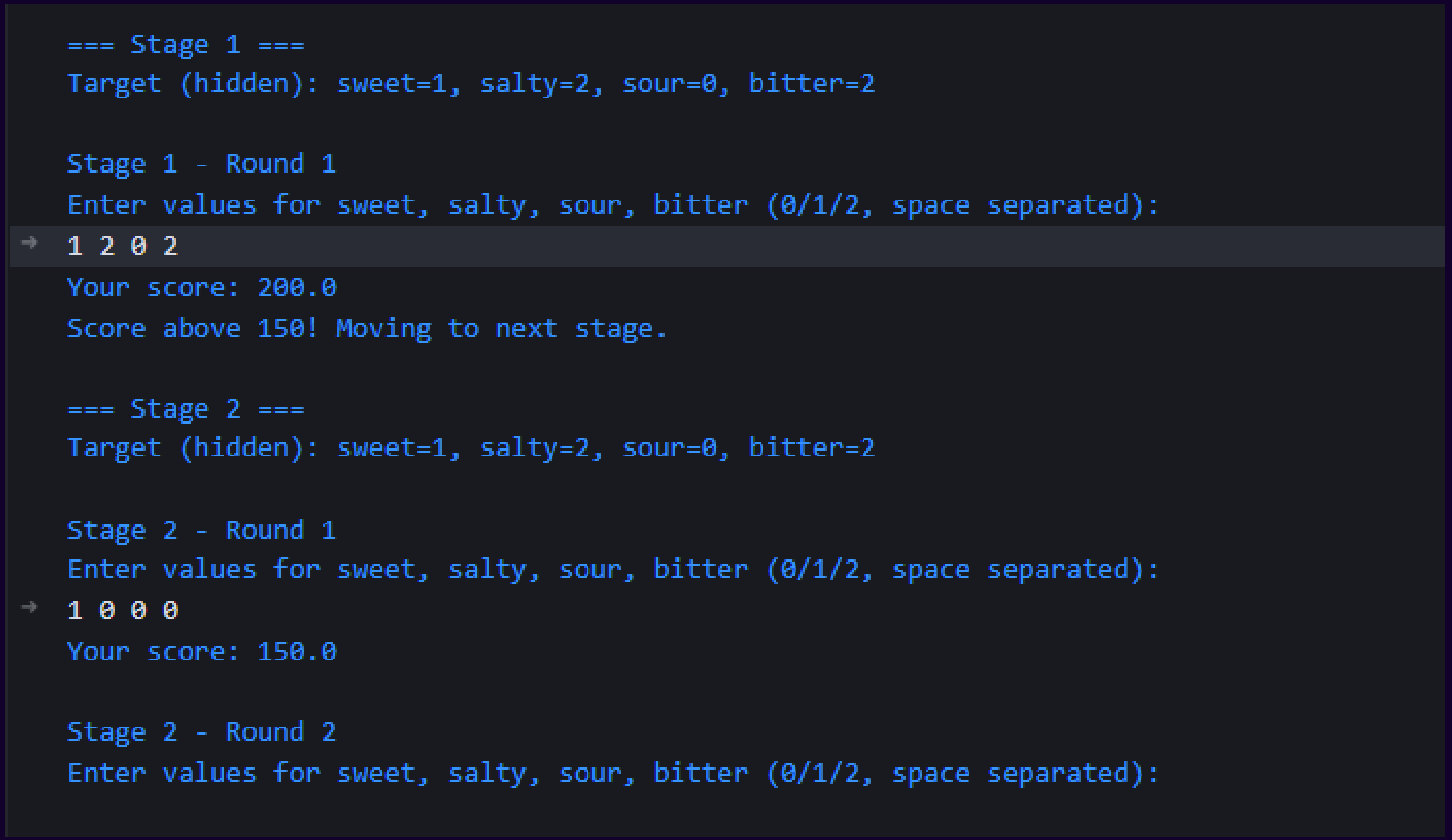
TECHINCAL ACHIEVEMENTS



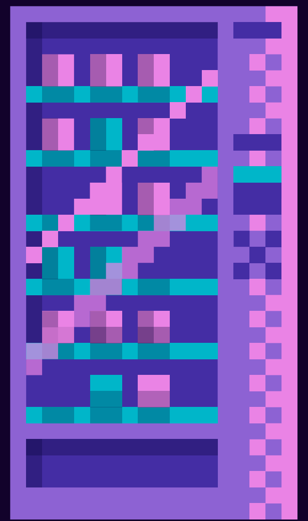




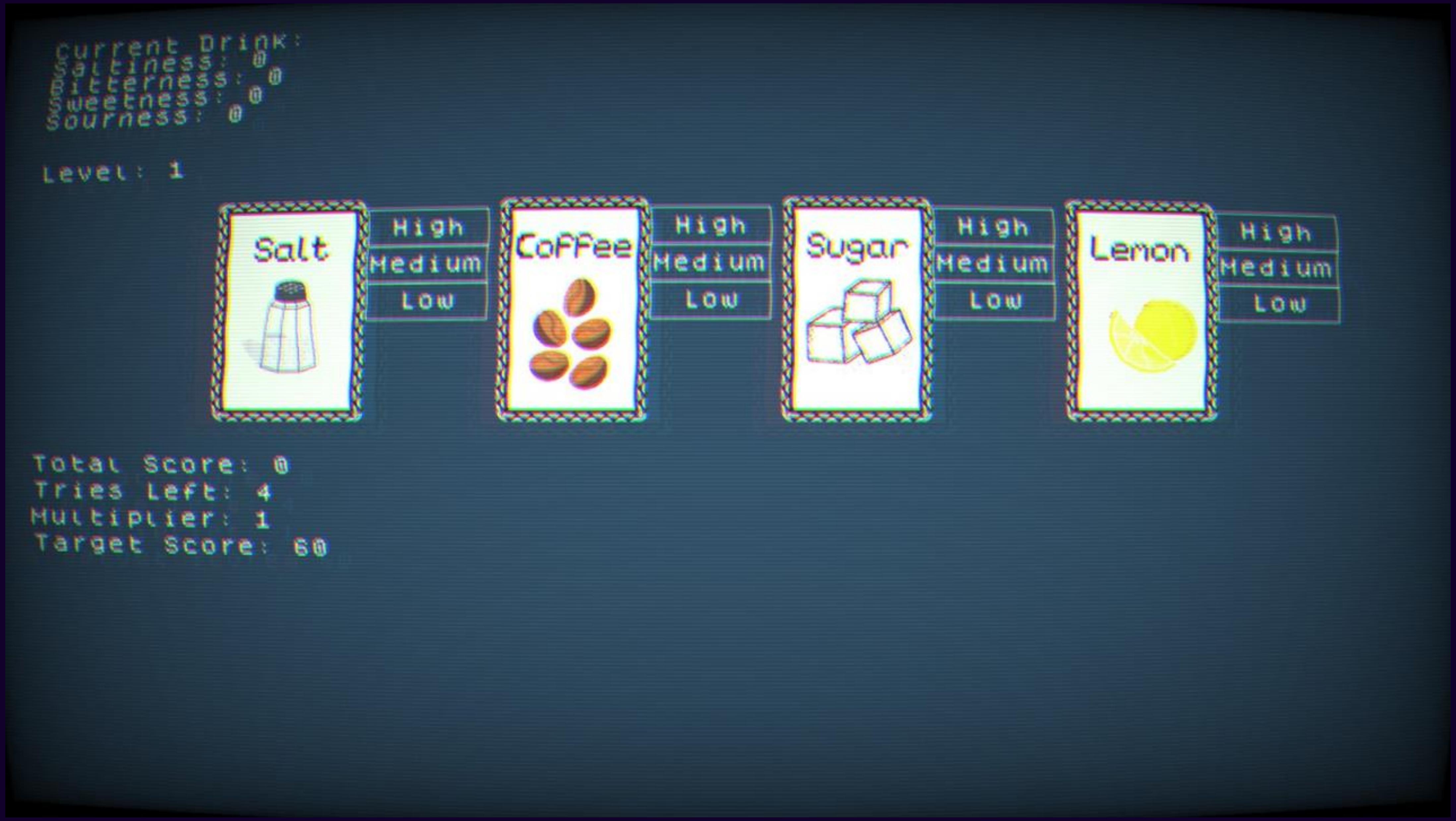
```lua
function love.draw()
    -- Step 1: Draw your game to gameCanvas (1600x900)
    love.graphics.setCanvas(gameCanvas)
    love.graphics.clear(0.2, 0.3, 0.4)

    -- Draw background image to fill the canvas
    if backgroundImage then
        local bgScaleX = gameCanvas:getWidth() / backgroundImage:getWidth()
        local bgScaleY = gameCanvas:getHeight() / backgroundImage:getHeight()
        love.graphics.draw(backgroundImage, 0, 0, 0, bgScaleX, bgScaleY)
    end
    local imageScale = 0.5
    for i, attr in ipairs(attributes) do
        local img = images[attr]
        local pos = imagePositions[attr]
        love.graphics.draw(img, pos.x, pos.y, 0, imageScale, imageScale)
    end

    local valueLabels = { [1] = "Low", [2] = "Medium", [3] = "High" }

    for _, button in ipairs(buttons) do
        love.graphics.rectangle("line", button.x, button.y, button.width, button.height)
        local label = valueLabels[button.value] or tostring(button.value)
        love.graphics.printf(label, button.x, button.y + 10, button.width, "center")
    end

    love.graphics.print("Current Drink:", 50, 30)
    local offsetY = 50
    for i, attr in ipairs(attributes) do
        love.graphics.print(attr .. ": " .. drink[attr], 50, offsetY)
        offsetY = offsetY + 20
    end

    if canServe() then
        love.graphics.rectangle("line", serveButton.x, serveButton.y, serveButton.width, serveButton.height)
        love.graphics.printf(serveButton.label, serveButton.x, serveButton.y + 10, serveButton.width, "center")
    end
```



```
=== Stage 1 ===
Target (hidden): sweet=1, salty=2, sour=0, bitter=2

Stage 1 - Round 1
Enter values for sweet, salty, sour, bitter (0/1/2, space separated):
→ 1 2 0 2
Your score: 200.0
Score above 150! Moving to next stage.

=== Stage 2 ===
Target (hidden): sweet=1, salty=2, sour=0, bitter=2

Stage 2 - Round 1
Enter values for sweet, salty, sour, bitter (0/1/2, space separated):
→ 1 0 0 0
Your score: 150.0

Stage 2 - Round 2
Enter values for sweet, salty, sour, bitter (0/1/2, space separated):
```



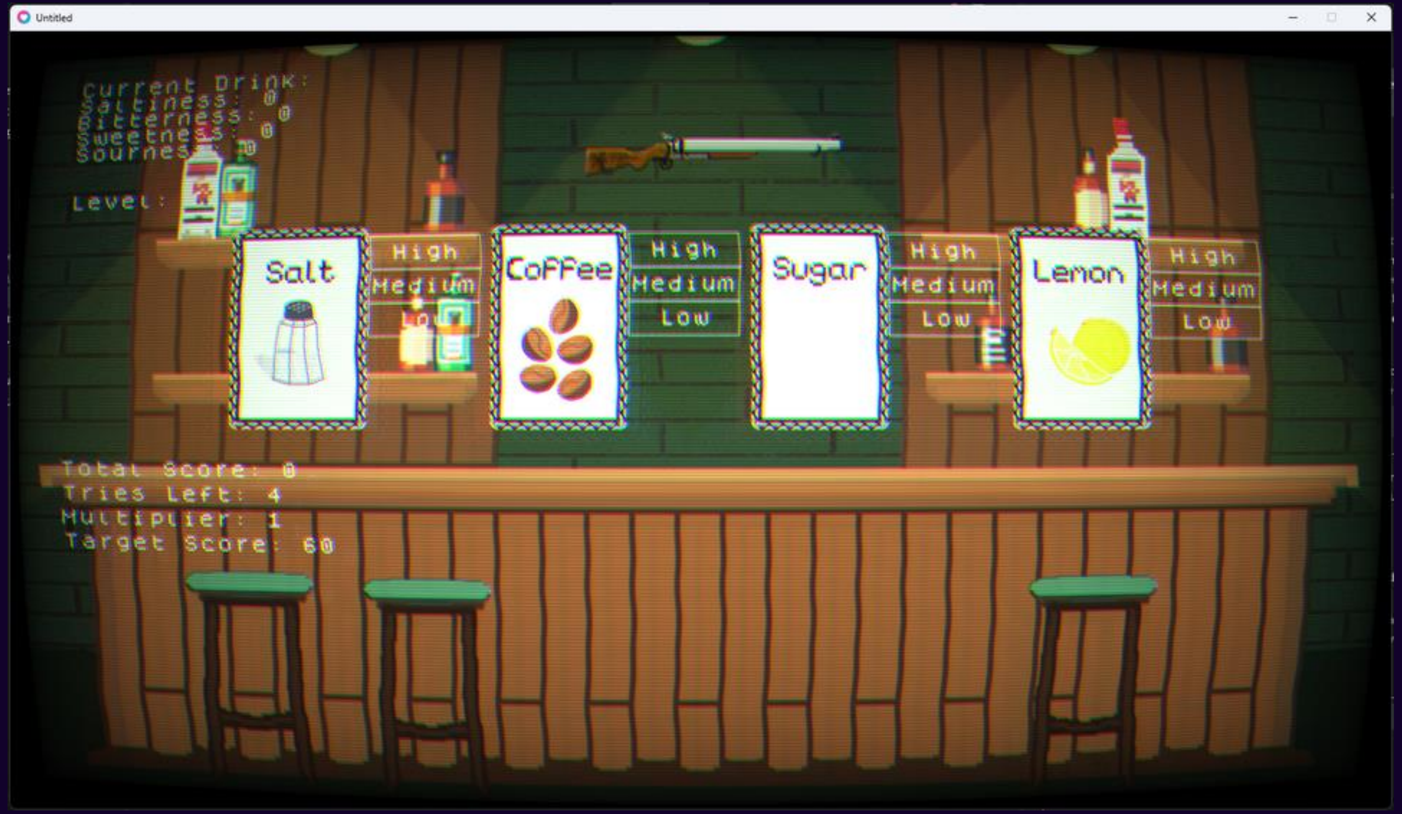
TECHINCAL ACHIEVEMENTS



Current Drink:
Saltiness: 0
Bitterness: 0
Sweetness: 0
Sourness: 0

Level: 1

Total Score: 0
Tries Left: 4
Multiplier: 1
Target Score: 60

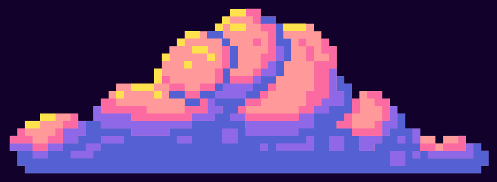


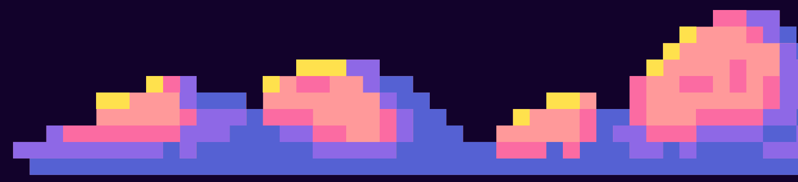
Current Drink:
Saltiness: 0
Bitterness: 0
Sweetness: 0
Sourness: 0

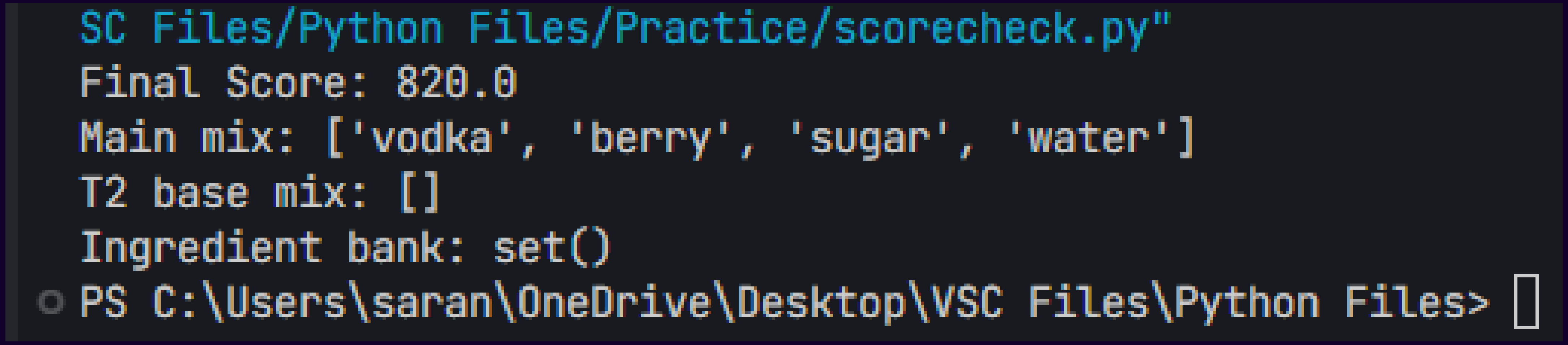
Level:

Total Score: 0
Tries Left: 4
Multiplier: 1
Target Score: 60

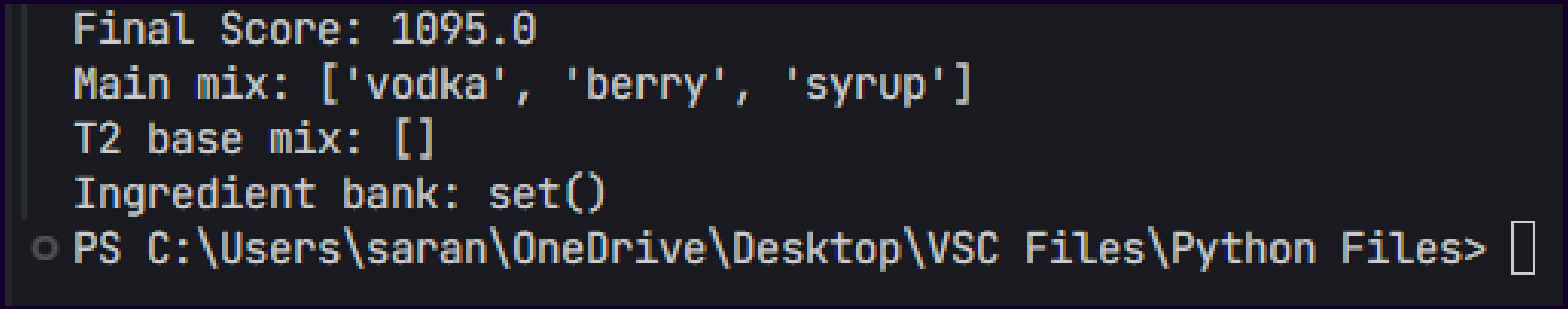
TECHINCAL ACHIEVEMENTS



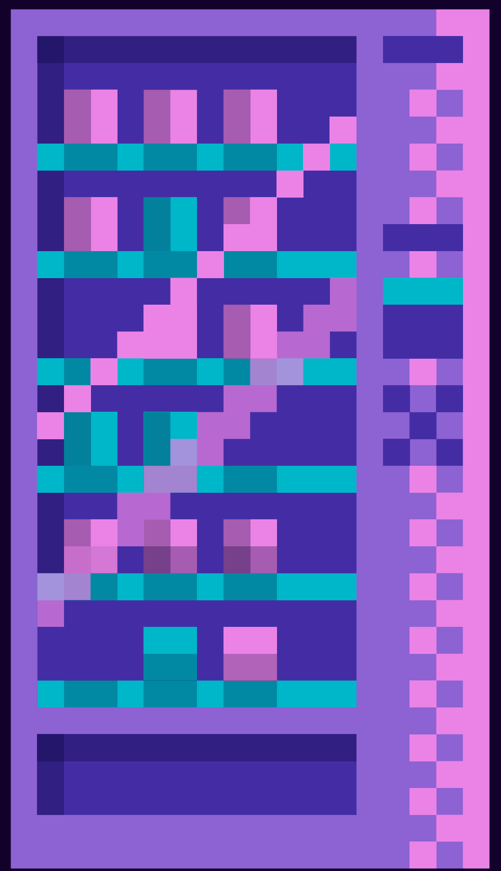




```
SC Files/Python Files/Practice/scorecheck.py"
Final Score: 820.0
Main mix: ['vodka', 'berry', 'sugar', 'water']
T2 base mix: []
Ingredient bank: set()
○ PS C:\Users\saran\OneDrive\Desktop\VSC Files\Python Files>
```



```
Final Score: 1095.0
Main mix: ['vodka', 'berry', 'syrup']
T2 base mix: []
Ingredient bank: set()
○ PS C:\Users\saran\OneDrive\Desktop\VSC Files\Python Files>
```



VISUAL DESIGN

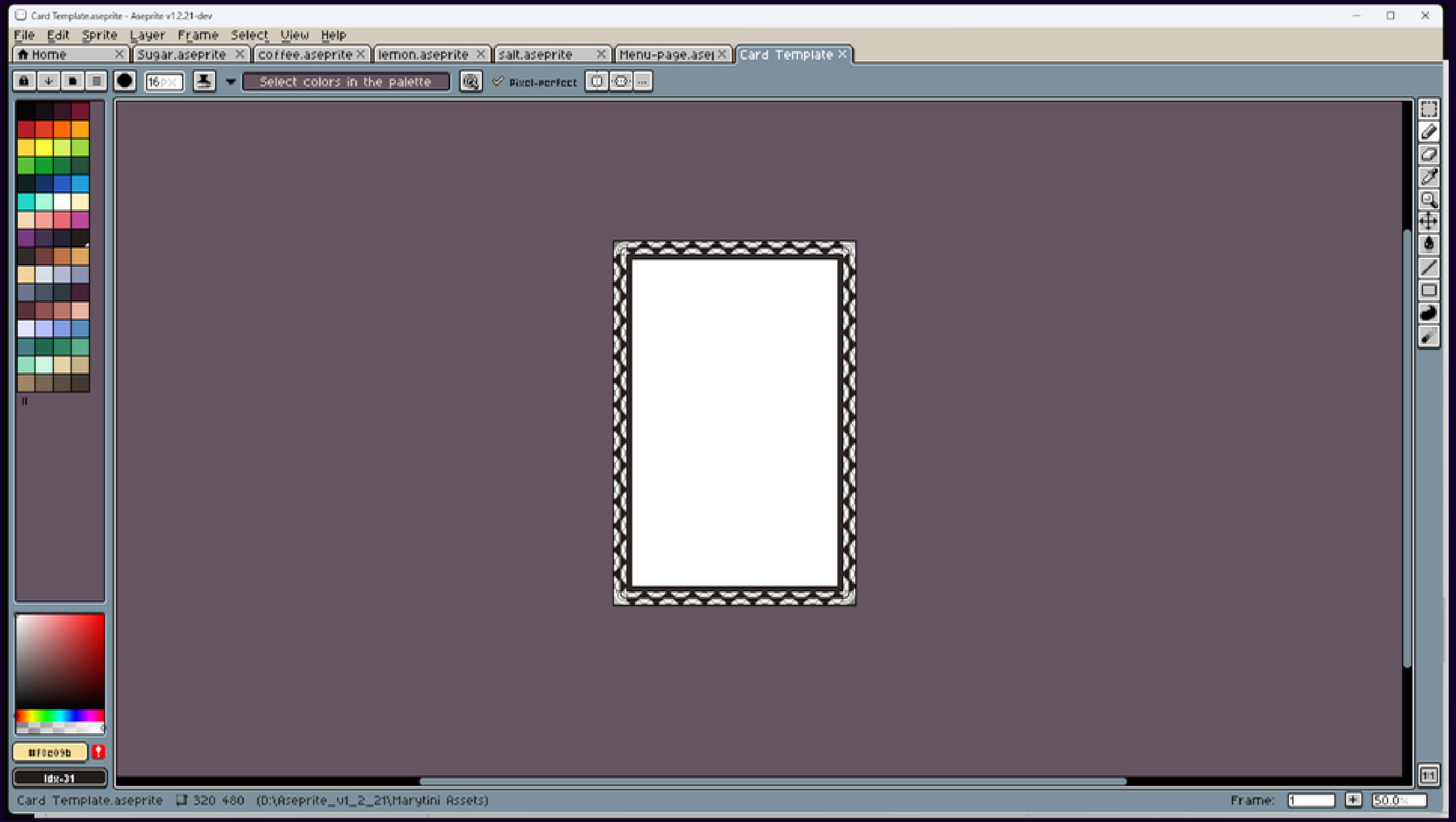
DEVELOPED ON ASPERITE

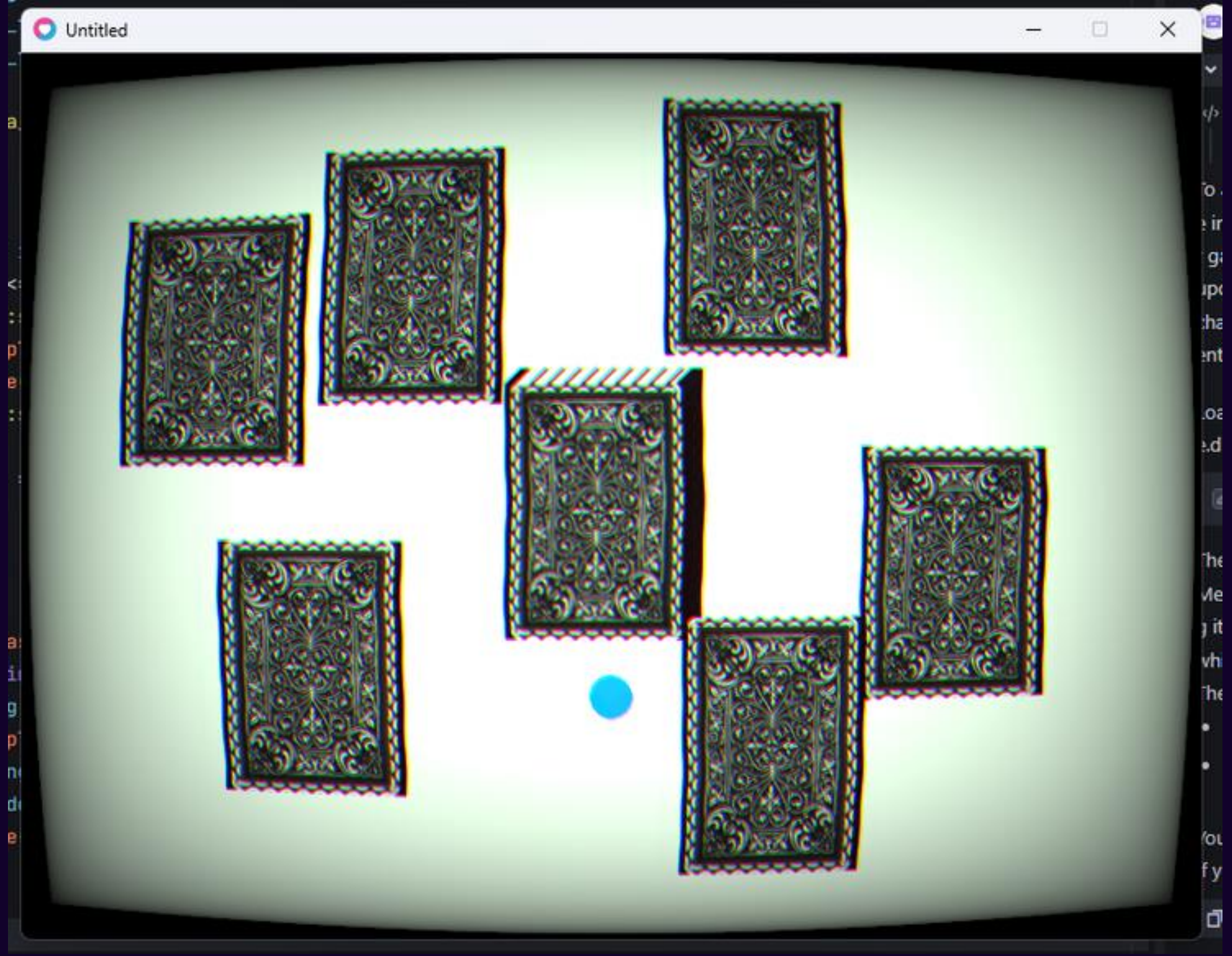
THEME : BAR SETTING,
TAVERN-ESQUE

PRODUCED ASSETS:
CARDS
INGREDIENTS
BAR MAIN MENU

VISUAL DESIGN

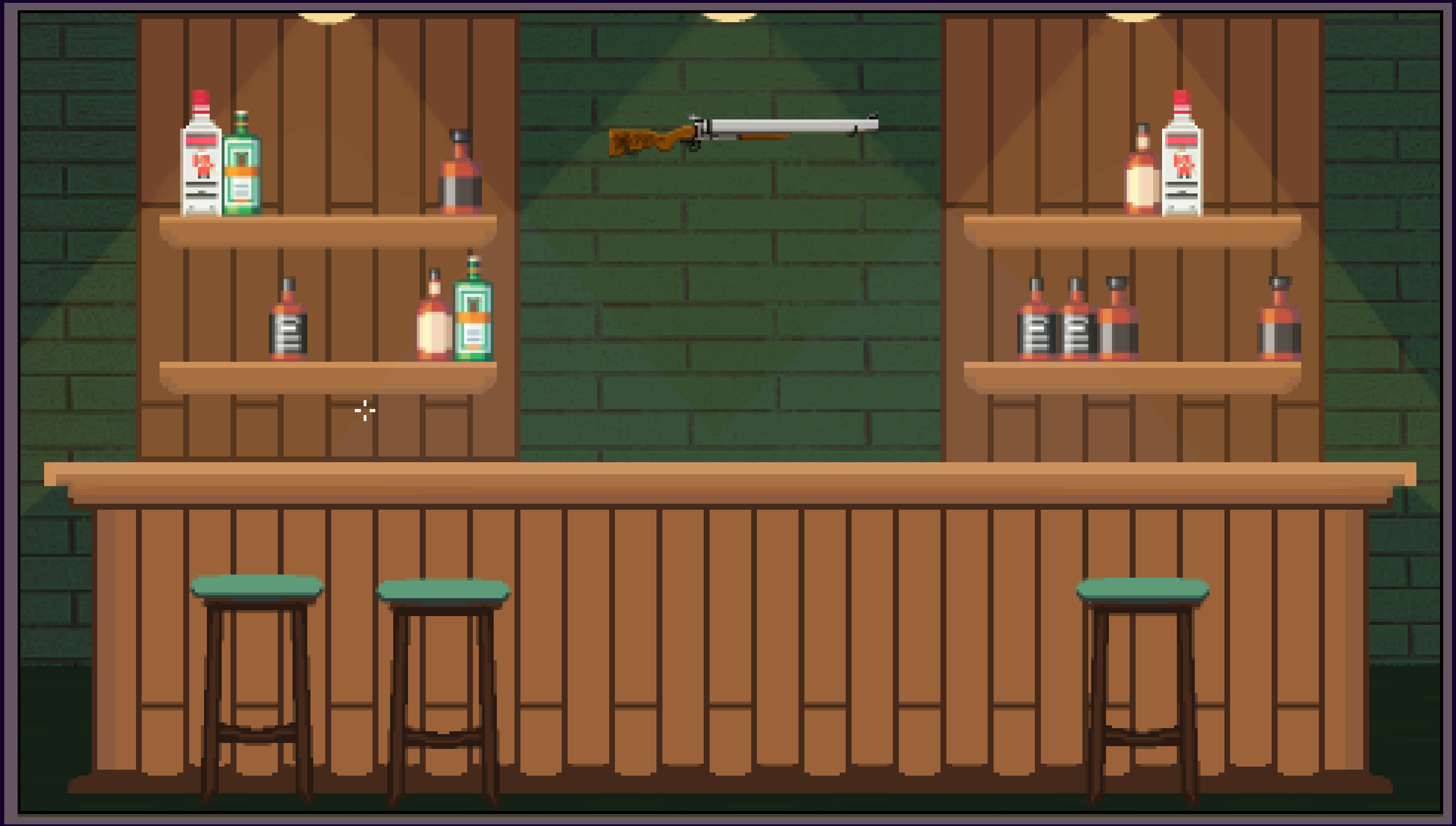
UNDERSTANDING SPRITES AND WORKING WITH ASPERITE


Card Template.aseprite - Aseprite v1.2.21-dev
File Edit Sprite Layer Frame Select View Help
Home
Sugar.aseprite
coffee.aseprite
lemon.aseprite
salt.aseprite
Menu-page.ase
Card Template
Select colors in the palette
Pixel-perfect
Card Template.aseprite 320 480 (D:\Aseprite_v1_2_21\Marytini Assets)
Frame: 1
50.0

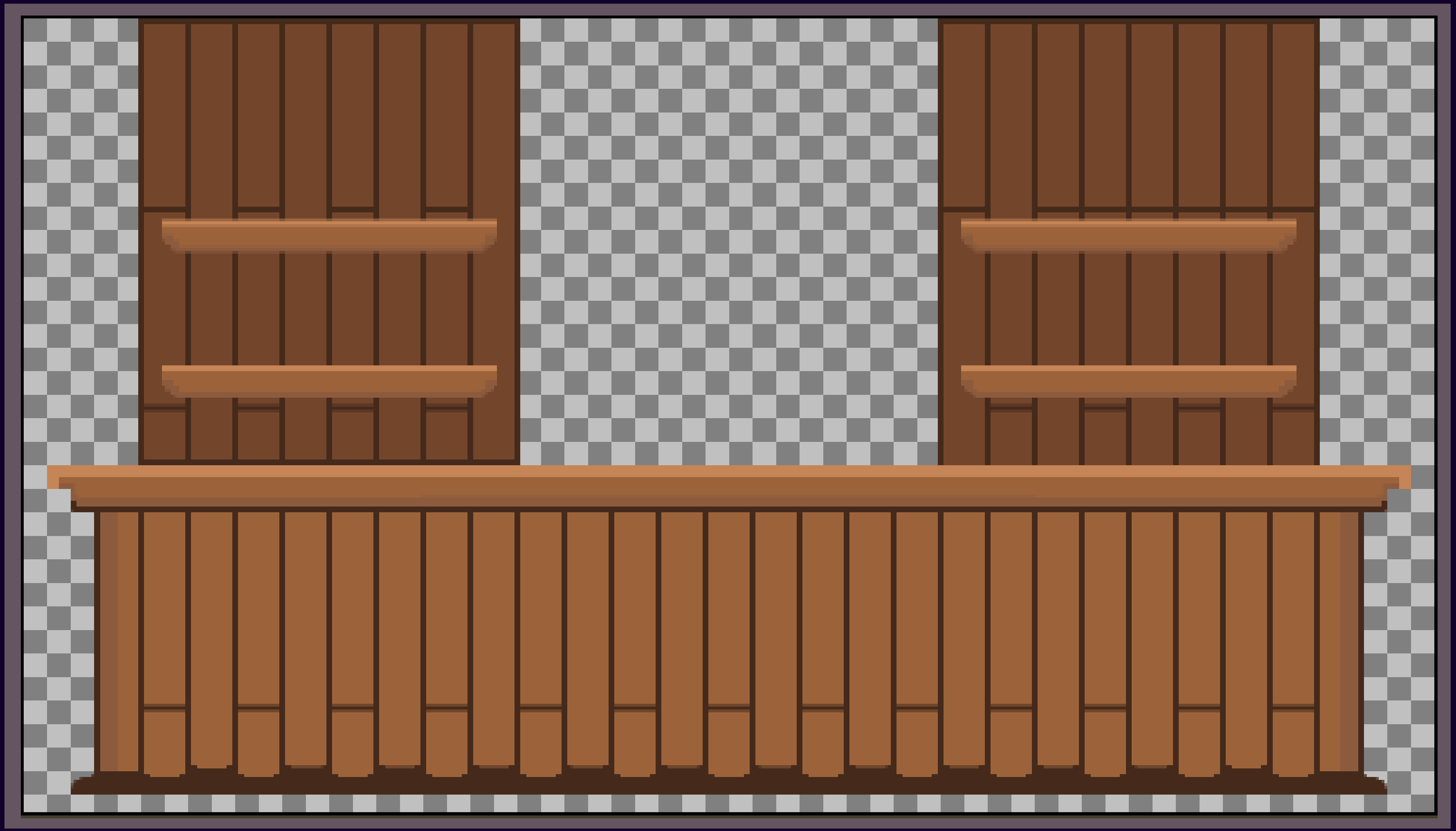


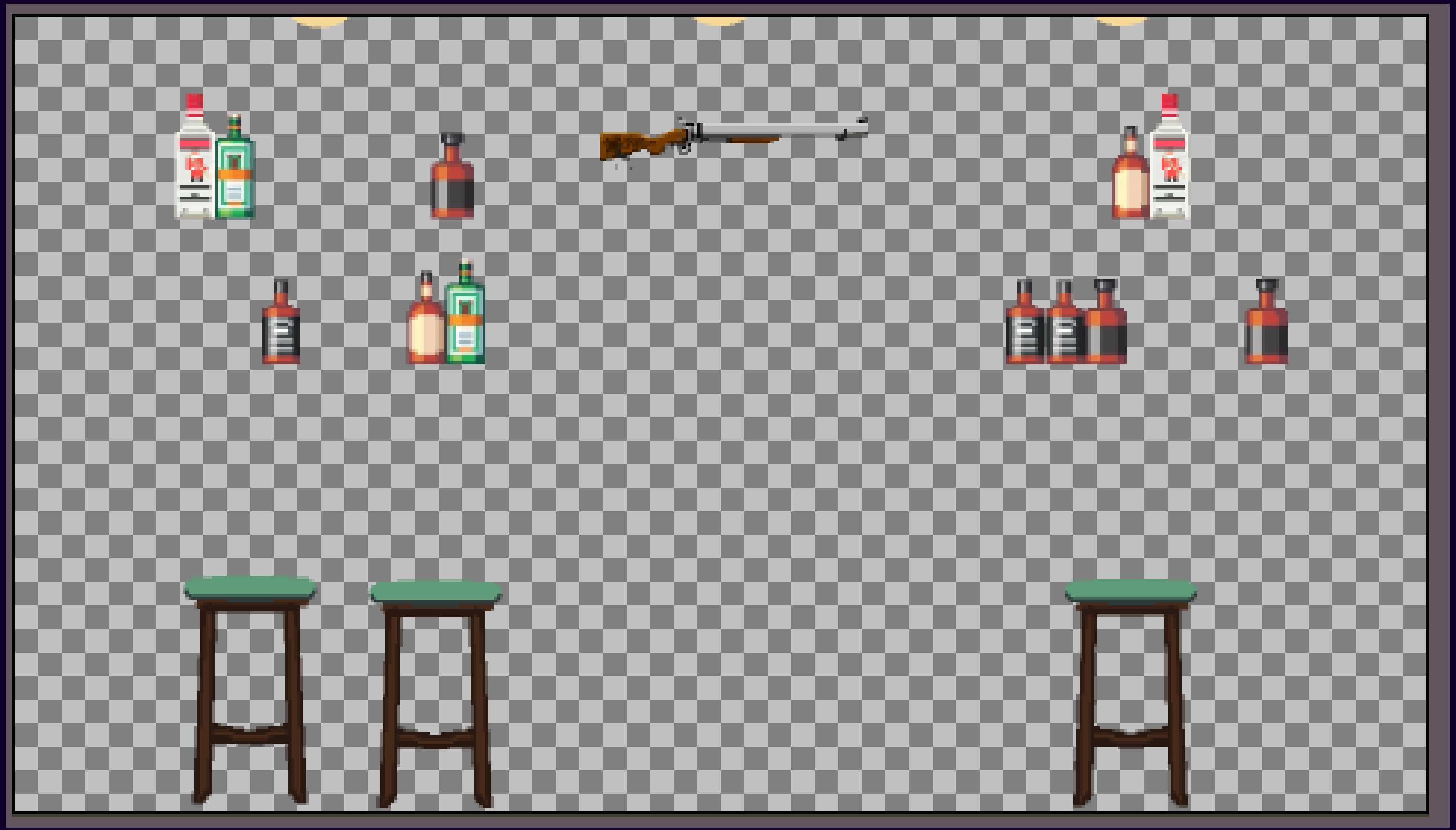
VISUAL DESIGN

VISUAL DESIGN

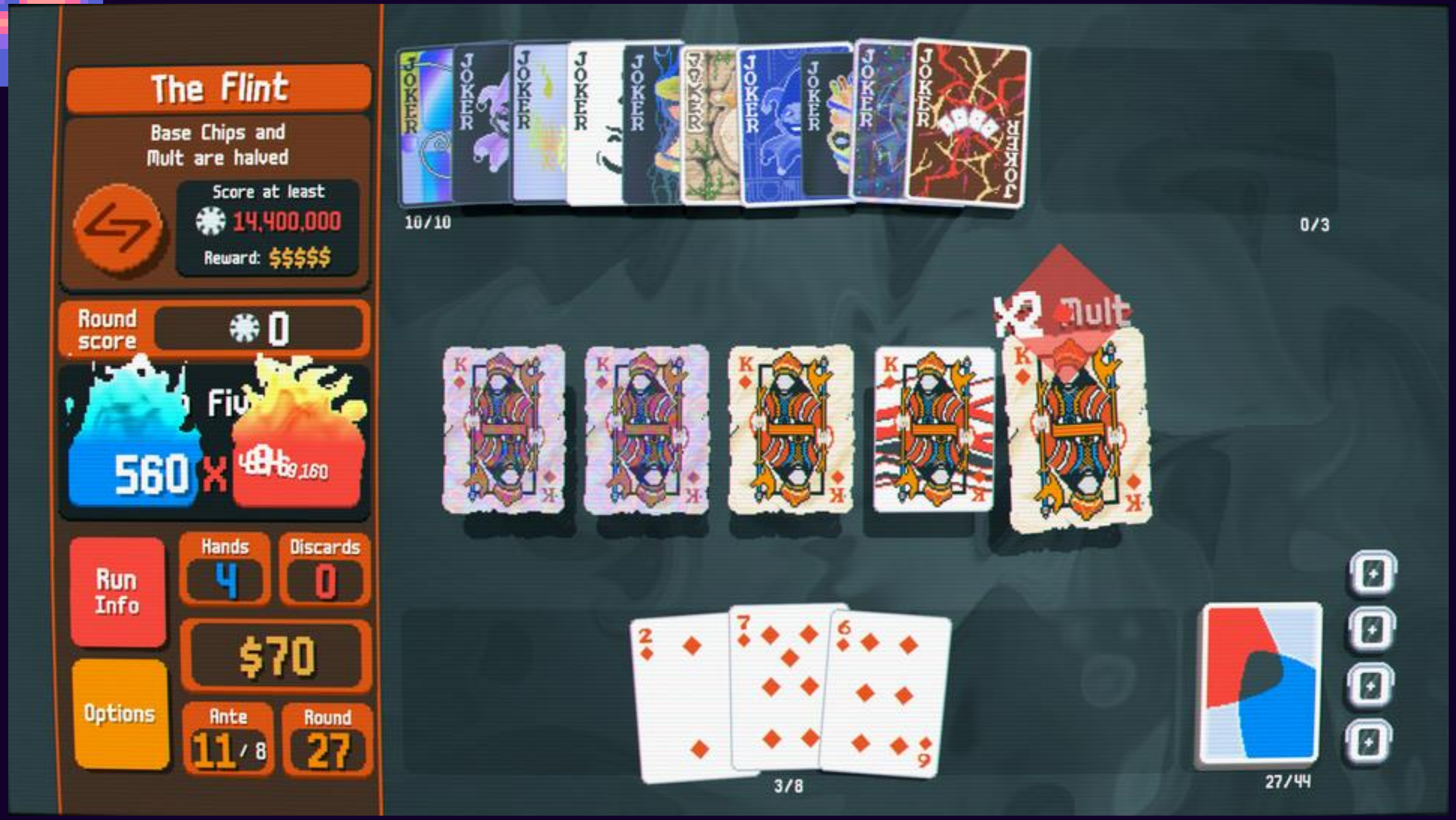


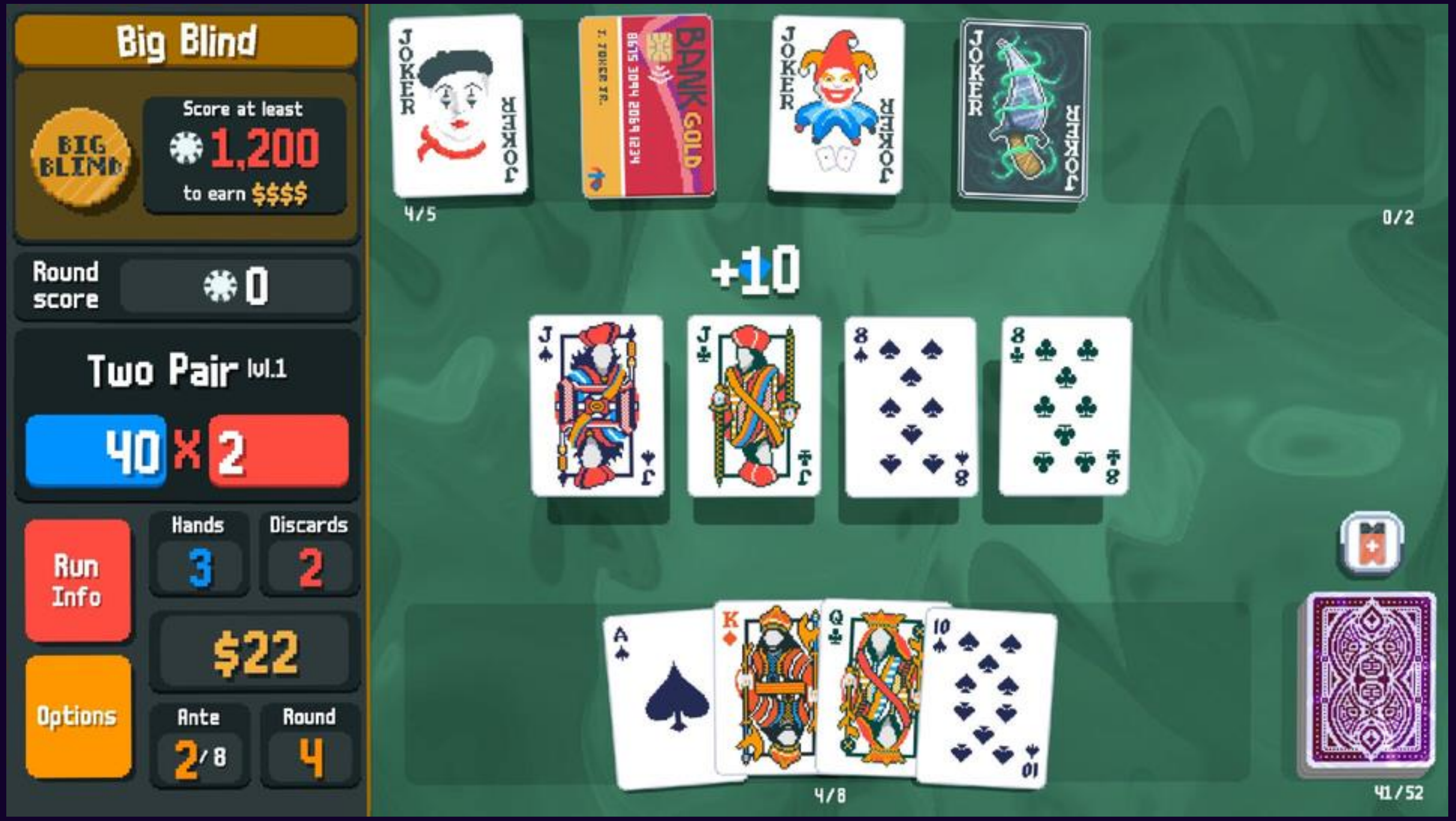
VISUAL DESIGN

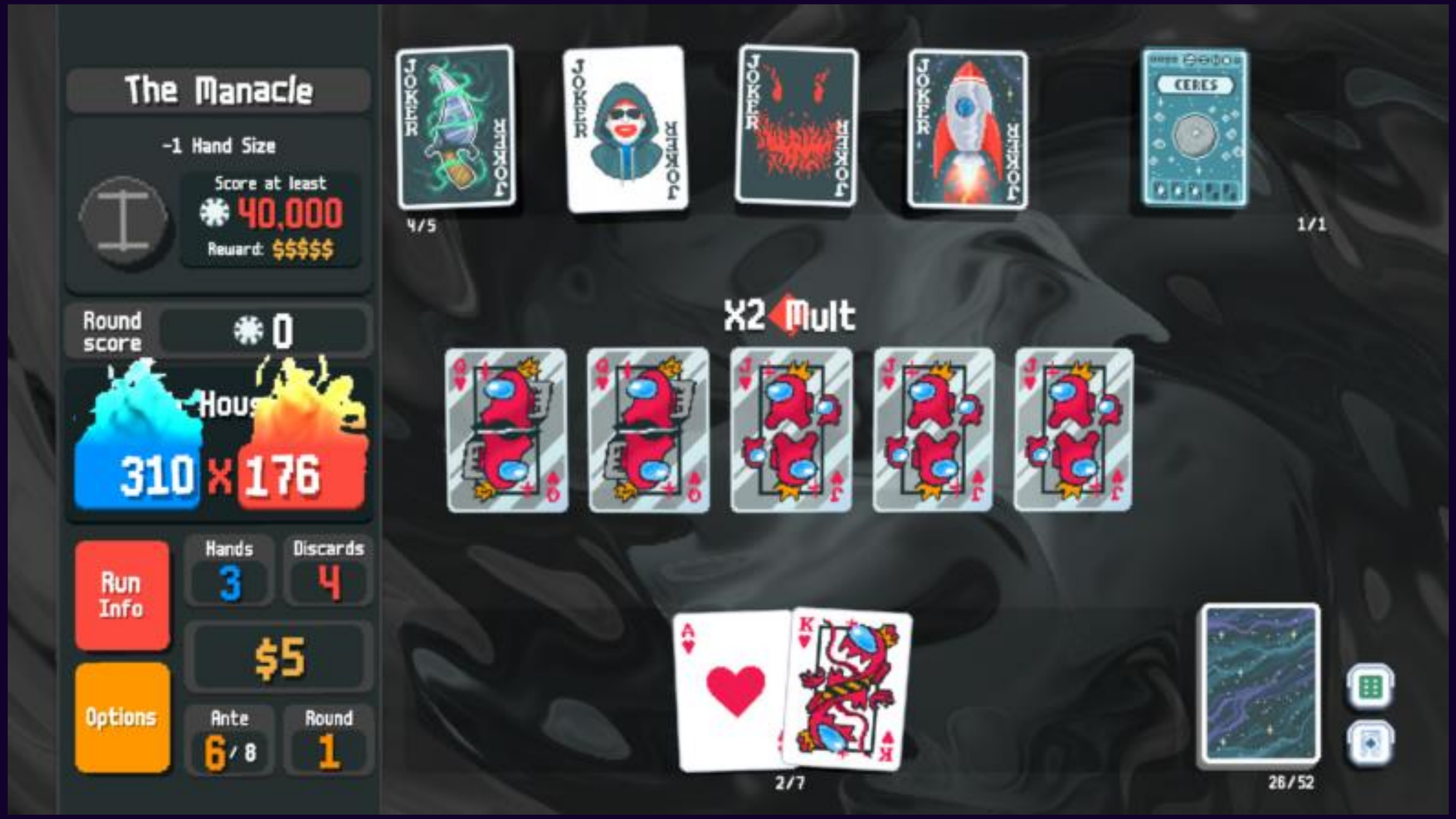


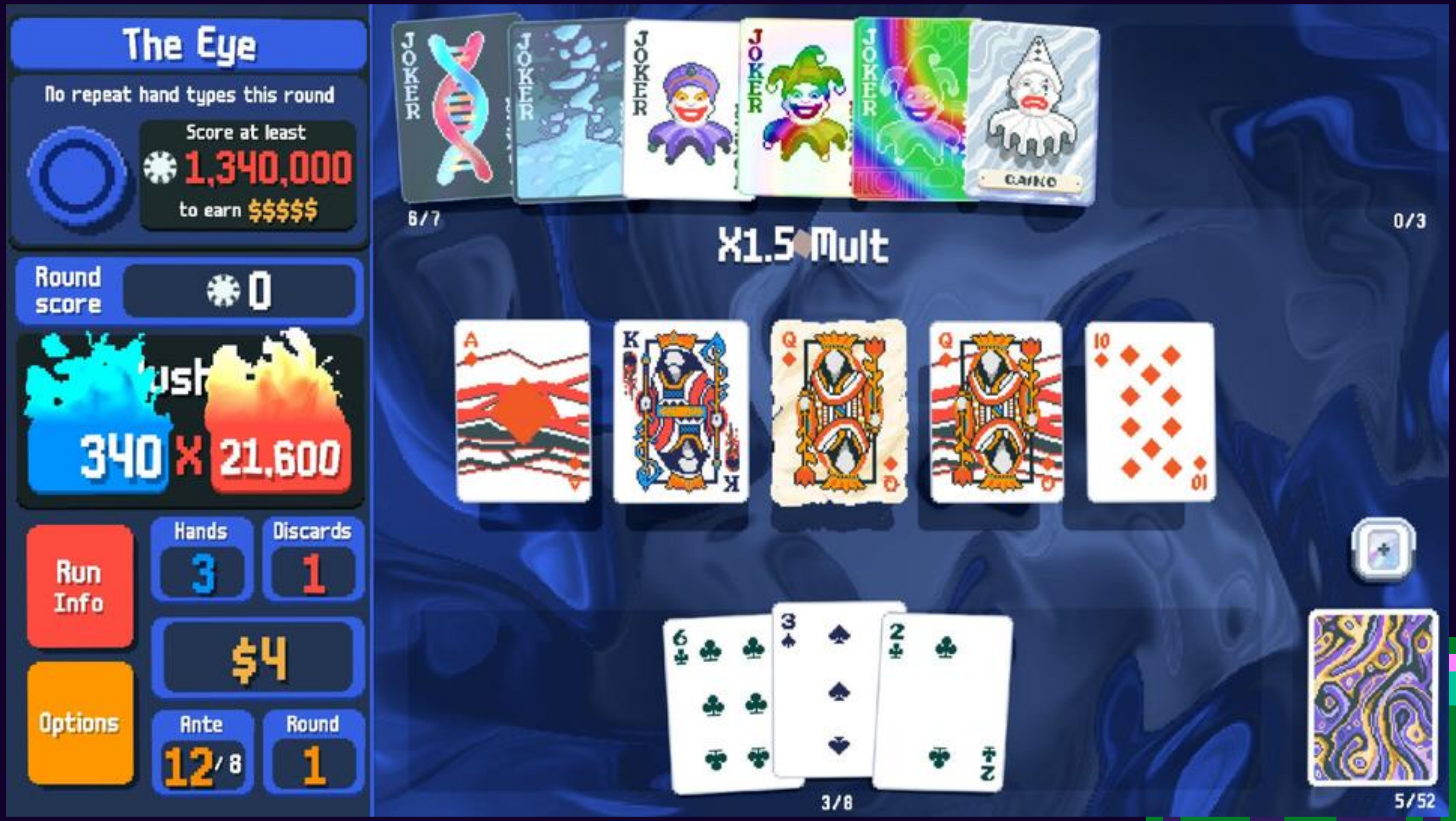


VISUAL DESIGN INSPIRATION









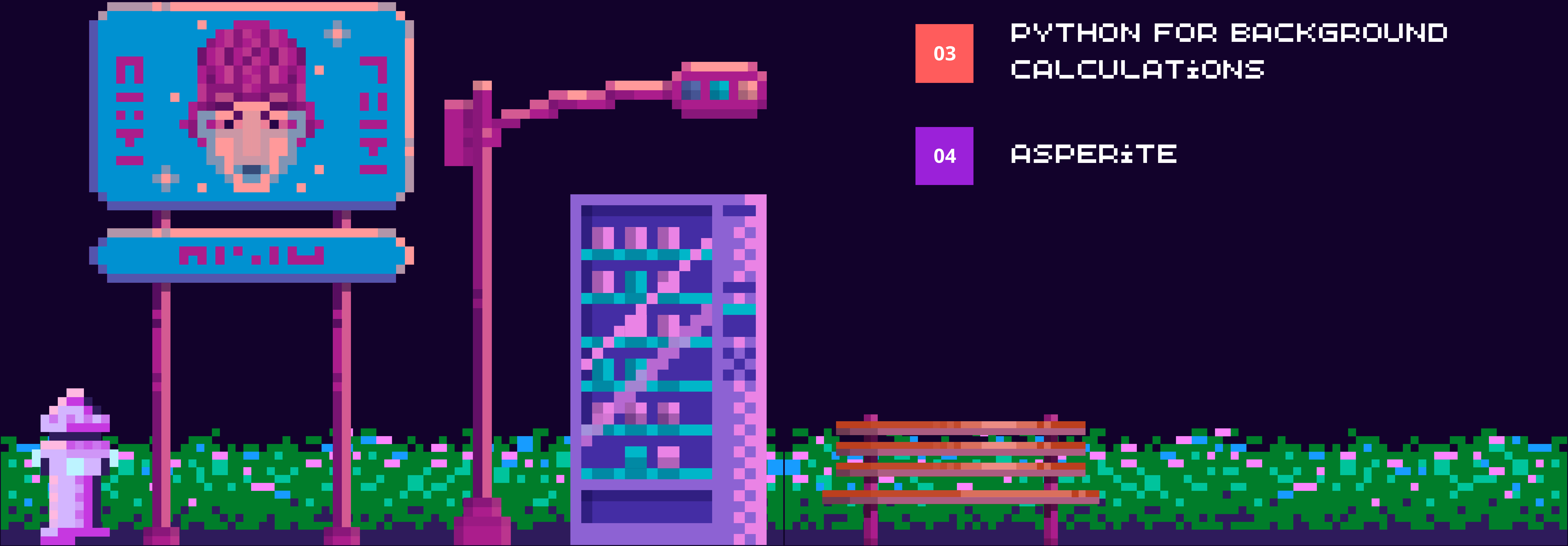
TECHSTACK

01 LÖVE2D FRAMEWORK

02 GAME LOGiC iN LUA

03 PYTHON FOR BACKGROUND CALCULATiONS

04 ASPERiTE



CHALLENGES AND REALIZAITIONS

01 SCORE CALCULATION

02 STRUCTURE OF THE GAME

03 INTEGRATION OF LUA, PYTHON AND ART FILES

04 STORYLINE, OVERESTIMATION OF TIME TO LEARN

FUTURE PLANS

01

COMPLETE LUA, LOVE2D AND ASSET INTEGRATION

02

PLAY TESTING

03

MAINTENANCE AND DEPLOYMENT