

# ***Assignment 2: Terminal App***

***by Clare O'Neil***



# Overview

**01**

## ***What is my App about?***

- Main Features

**02**

## ***Structure of my App***

- Flowchart
- Planning Process

**03**

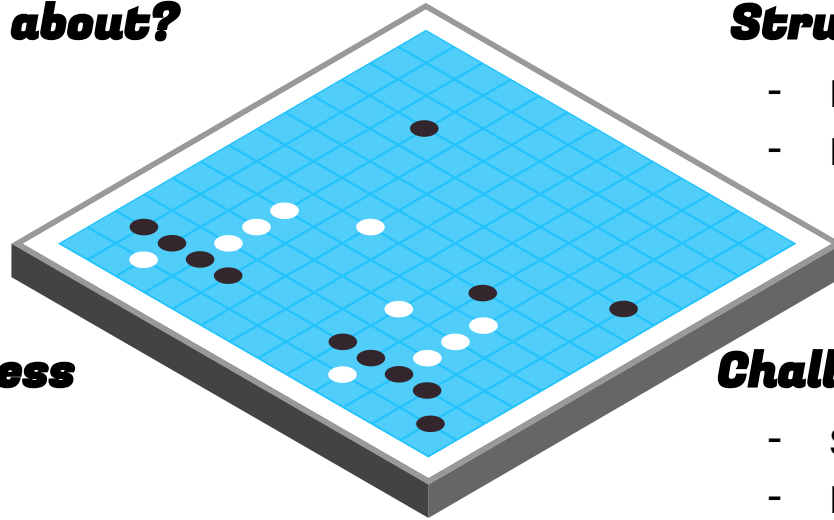
## ***Building Process***

- Gems
- Logic of Code

**04**

## ***Challenges and Issues***

- Struggles
- Favourites
- Ethics

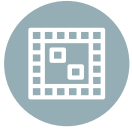




**01**

***What is my  
App about?***

# Game App: Sugoroku

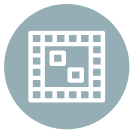


## ***What is sugoroku?***

Board Game originating from Asia where Players roll the dice and move around a board

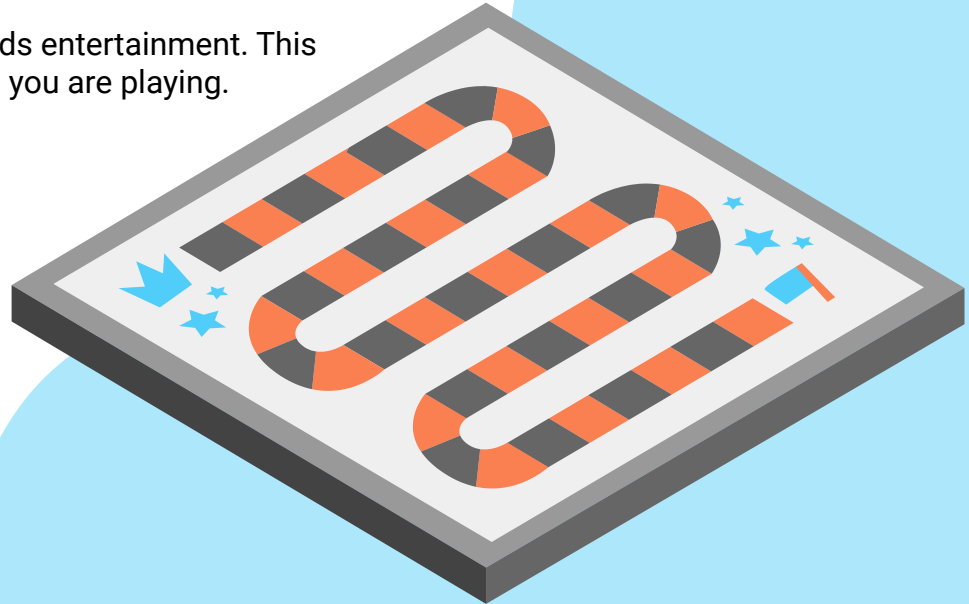
Certain panels have task/activity which adds entertainment. This feature depends on what kind of Sugoroku you are playing.

In most Sugoroku, the Goal is to reach the finish line with fewer number of turns OR with the highest score



## ***Think:***

- Mario party
- Snakes & ladders
- Game of Life
- Monopoly (?)



# ***WHY DID I CODE SUGOROKU?***



***Wanted to build a program  
that has some connection  
to myself (Japanese)***



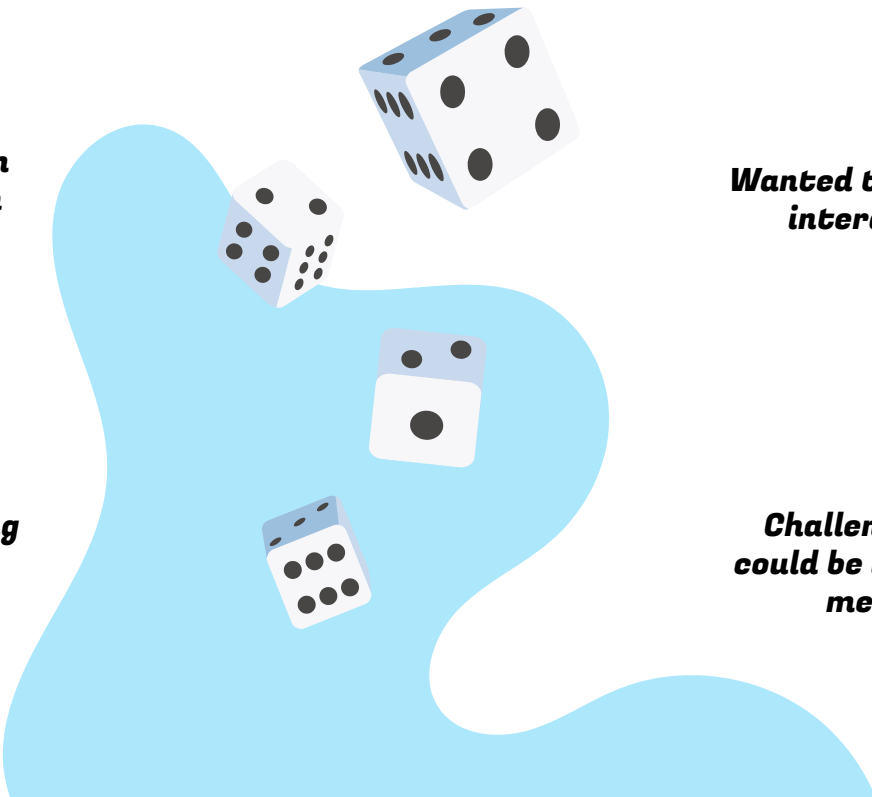
***Wanted to build something  
that was visually  
entertaining***



***Wanted to build something  
interactive and fun***



***Challenging, but felt it  
could be logical enough for  
me to program***



# Features

## 1: How to Play

- Information about Sugoroku
- Game Rules

## 2: Play Sugoroku

- Main feature of this app
- Single player (unfortunately)

## 3: Ranking Board

- Top 100 scores that has been submitted

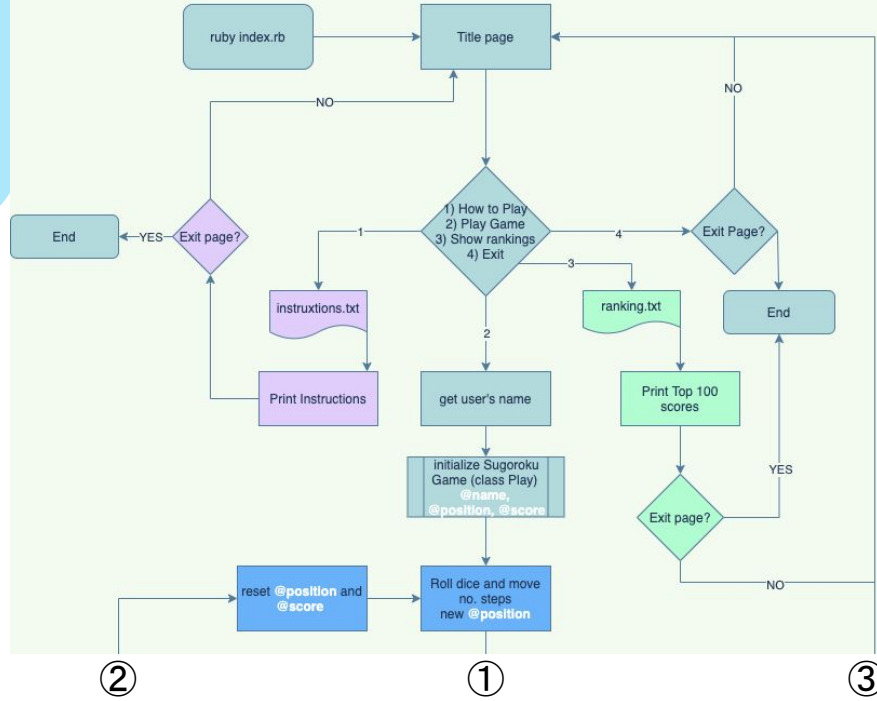




**02**

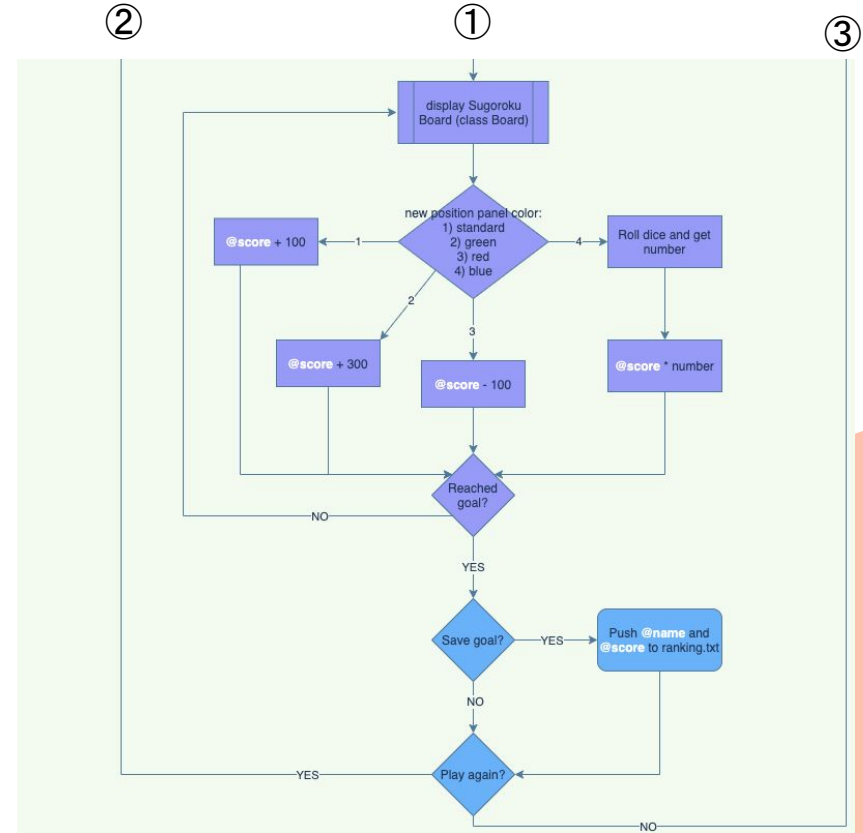
***Structure of  
my Application***

## Terminal Sugoroku Flow



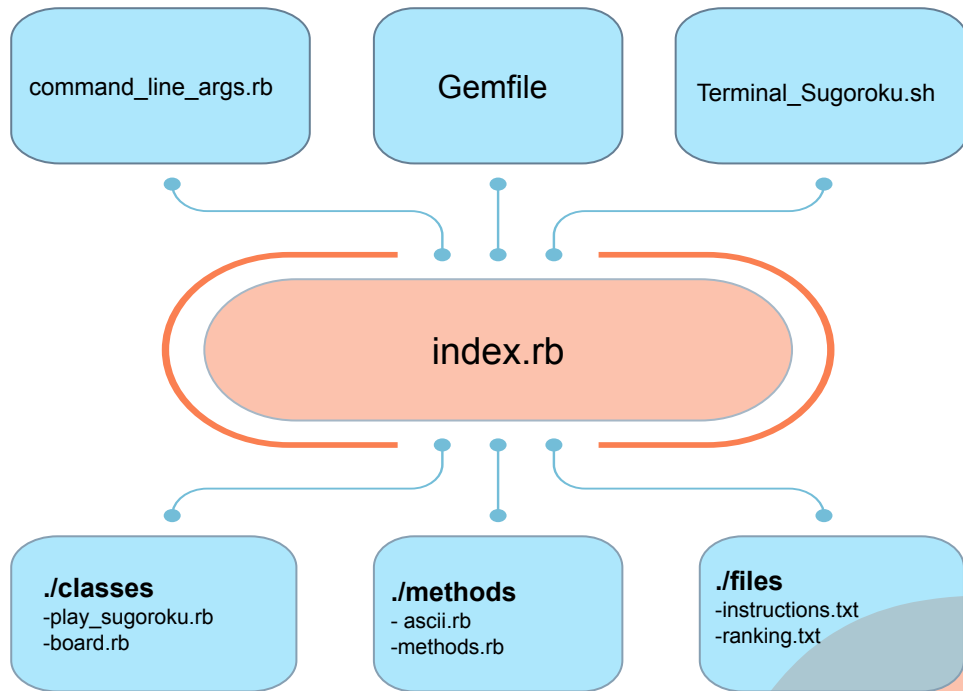
※ could not fit entire flow chart. Please connect matching numbers for correct flow

## Flow Chart





# *File System*



# Programs Used

**01**

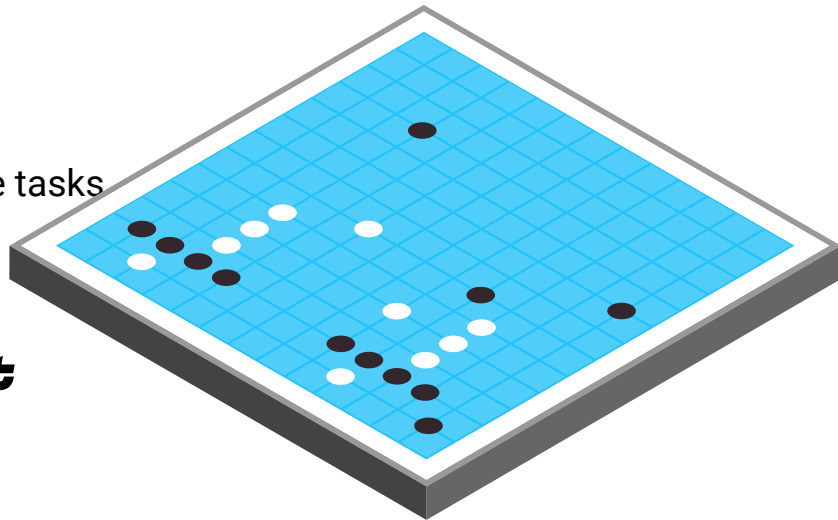
## **Trello**

- Planning
- Checklist
- Donut & Sprinkle tasks

**03**

## **Excel sheet**

- Manual testing



**02**

## **Ruby GEMs**

- Inside bundler
- Created a testing environment (test.rb)

**04**

## **Git & GitHub**

- Regular commits
- Pushing to remote repo

# More Programs Used

## 05

**Google**

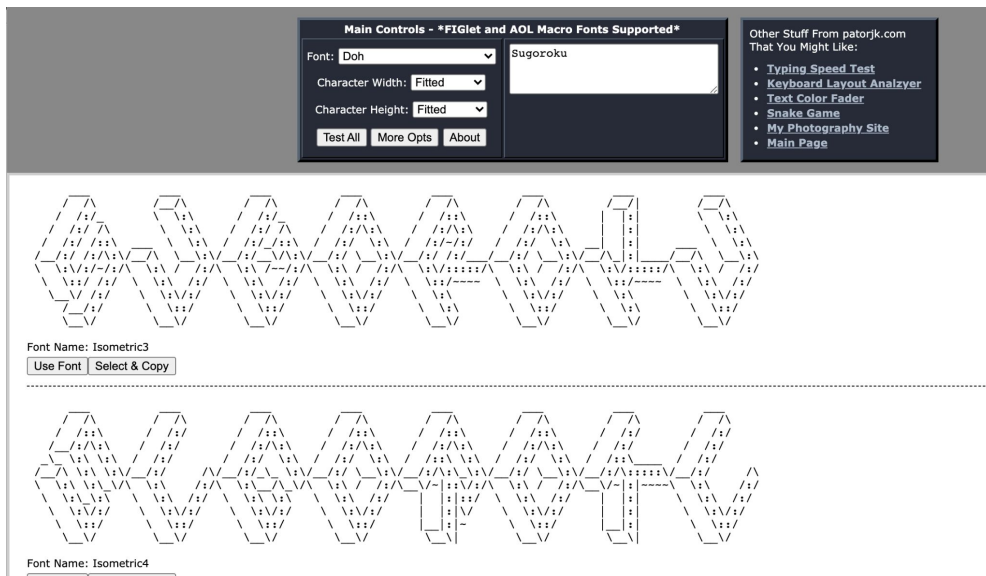
- Research
- Stack overflow
- Example terminal apps
- Ascii text art generator

## 06

**draw.io**

- Drawing flowcharts

patorjk.com



# Manual testing

<b>Sugoroku App: Manual Testing</b>										
File Edit View Insert Format Data Tools Add-ons Help Last edit was 3 hours ago										
J10	A	B	C	D	E	F	G	H	I	J
6	4	1 Sugoroku board	check if sugoroku board can be displayed on terminal	class SugorokuBoard	shows board on terminal	worked as planned	Passed			
7	5	1 whirly gem	check if gem works	whirly.start spinner: "circled_number" do sleep 5 end	show whirly spinner with circled number for 5 seconds	worked as planned	Passed	want to change spinner to custom values. test no longer useful		
8	5	2 whirly gem	check if gem works with custom values	whirly.start spinner: [1, 2, 3, 4, 5, 6] do sleep 5 end	show spinner with 1, 2, 3, 4, 5, 6 as separate frames for 5 seconds	does not work withustom frames.	Critical	shows error. should rescue here even if it works on my computer. test no longer useful		
9	5	3 whirly gem	check if gem works with other default spinners	whirly.start spinner: "boundingBall" do sleep 5 end	show bouncingBall as separate frames for 1.5 seconds		Passed	after research found that whirly gem does not allow for custom frames. resorted to using one of the default spinners. test no longer useful		
10	6	1 tty-spinner gem	check if spinner works with custom frames	spinner = TTY::Spinner.new(frames: ["1", "2", "3", "4", "5", "6"], clear: true)	show numbers 1 to 5 spinning on terminal screen for given amount of time	worked as planned	Passed	easier to use and works the way i want. this is a better gem than whirly.		
11	7	1 play.move	check if @positon and @score will change when Play.move is called	test = Play.new("tester"), test.move(3)	test.position => 3, test.score => 300	position updated but score not added properly	minor	position		
12	8	1 play.move	check if board map icon moves with @position and scores are added properly	username.display.score	icon moves with position and updates each dice roll	icon duplicates as positin moves (fixes the update each time)	minor	instance variable not appropriate? how to "refresh" map each time... no class should be applied?		
13	8	2 play.move	check if board map icon moves with @position and scores are added properly	username.display.score	icon moves with position and updates each dice roll	icon refreshed as planned, but not all the time	minor	when at position 4 icon didn't show.		
14	8	3 play.move	check if board map icon moves with @position and scores are added properly	username.display.score	icon moves with position and updates each dice roll. score updates appropriately	map and position works but ev	minor	case iterator may not be appropriate? try !if/elsif instead		
15	8	4 play.move	check if board map icon moves with @position and scores are added properly	username.display.score. changed iterator from case @position to if @position/elsif @position/else	icon moves with position and updates each dice roll. score updates appropriately	score system works for panels	major			
16	8	5 play.move	check if board map icon moves with	username.display.score. changed back to case statement but with	icon moves with position and updates each dice roll.	works correctly except for when I land on the blue panel	minor	because I roll the dice twice properly,		

Decided to do manual testing because I could keep record of my failed attempts unlike rspec. It was easier to visualise flow process, and took less time compared to rspec TDD method

less time on testing = more time focusing on making the actual code work. It was important for me that I didn't "waste" too much time trying out how rspec works in such a limited amount of time

Link: <https://docs.google.com/spreadsheets/d/1Wi38tJLZNAkXqrJaveeonaHfczbvBRVzld4vyUN3Eac/edit#gid=0>



# ***03*** ***Building Process***

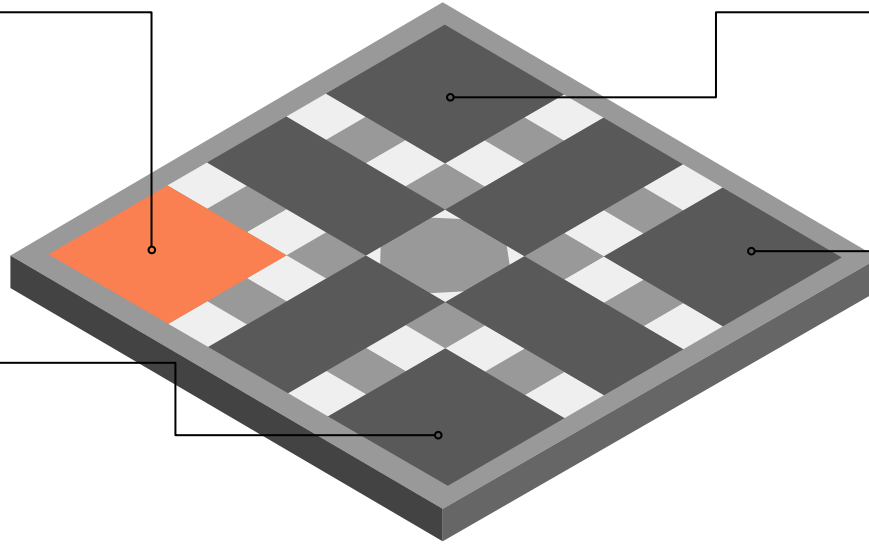
# *Coding Features*

**1: How to Play**

**2: Class Play**

**3: Class  
SugorokuBoard**

**4: Ranking**



# 1: How to Play

```
*****
WHAT IS SUGOROKU?
*****
Sugoroku is a Japanese name for Board Game similar to Snake and ladders or a simplified monopoly.
Usually a multi-player game, it is played by rolling a dice and moving across the number of steps
*****
HOW TERMINAL SUGOROKU WORKS
*****
When you start playin the game, you will see a sugoroku board displayed on your screen.
You will roll a dice and the number that you rolled is the number of steps you can move
You will either land on;
1) standard panel: you will be given 100pts
2) green panel: you will be given 300pts
3) red panel: you will be deducted 100pts
4) blue panel: bonus panel! your point will be multiplied by the number that you roll
The aim is to reach the goal with the highest number of points.
Your score entirely relies upon luck, so...GOOD LUCK!

Do you want to go back to Menu? (Press ↑/↓ arrow to move and Enter to select)
> Yes
No, Exit Program
```

## Ruby gems used:

- **TTY-Prompt**
- **Colorize**

```
#test ID 2
def back_to_menu_or_exit
  prompt = TTY::Prompt.new
  instruction_output = prompt.select("Do you want to go back to Menu?", ["Yes", "No, Exit Program"])
  if instruction_output == "No, Exit Program"
    exit_program
  end
end
```

```
#test ID 1
def open_instructions
  begin
    File.foreach("files/instructions.txt") { |line| puts line }
  rescue
    puts "could not find the file".coloirize(:red)
  end
end
```



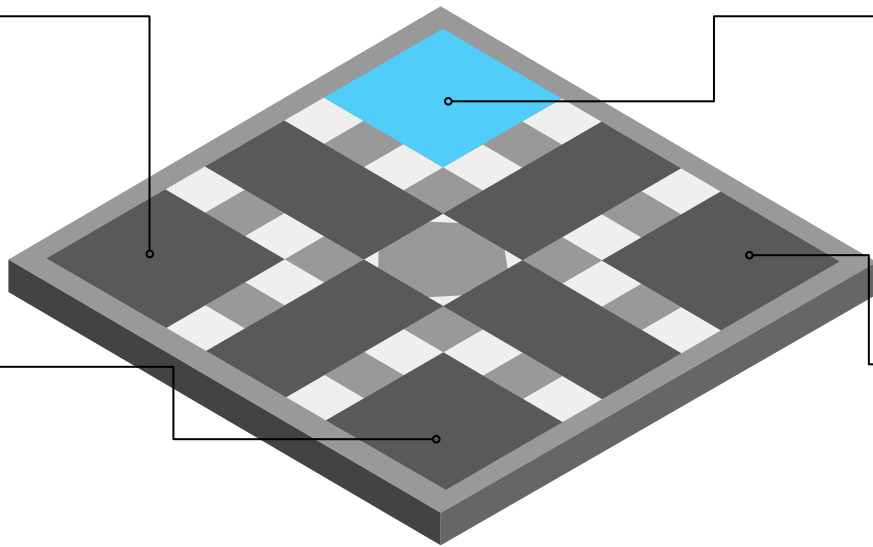
# *Coding Features*

**1: How to Play**

**2: Class Play**

**3: Class  
SugorokuBoard**

**4: Ranking**





# 2: Class Play

## Ruby gems used:

- **TTY-Spinner**  
(Roll dice)
- **Colorize**
- **TTY-table**

```
class Play
  attr_accessor :name
  attr_reader :position, :score

  def initialize(name)
    @name = name
    @position = 0
    @score = 0
  end
end
```



```
def roll_dice
  prompt = TTY::Prompt.new
  prompt.keypress("Press enter to roll the dice and enter again to stop", keys: [:return])
  # start spinner when enter key is pressed
  # test ID 6
  spinner = TTY::Spinner.new(frames: ["1", "2", "3", "4", "5", "6"], clear: true)
  spinner.auto_spin
  # stop spinner by pressing enter
  prompt = TTY::Prompt.new
  prompt.keypress(keys: [:return])
  spinner.stop
  rand(1..6)
end
```



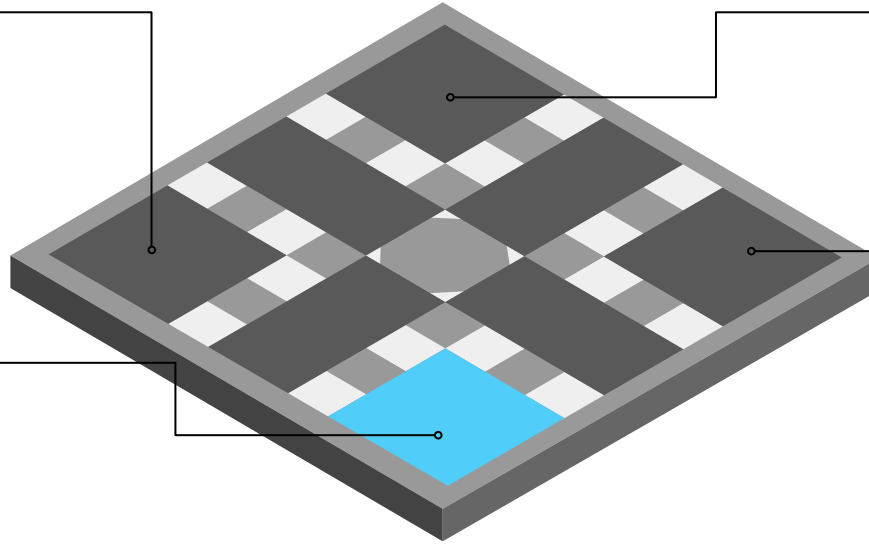
# *Coding Features*

**1: How to Play**

**2: Class Play**

**3: Class  
SugorokuBoard**

**4: Ranking**



## - **Colorize**

***Change in position...***

Name	Total Steps Taken	Current Score
clare	30/33	1300

Press enter to roll the dice and enter again to stop

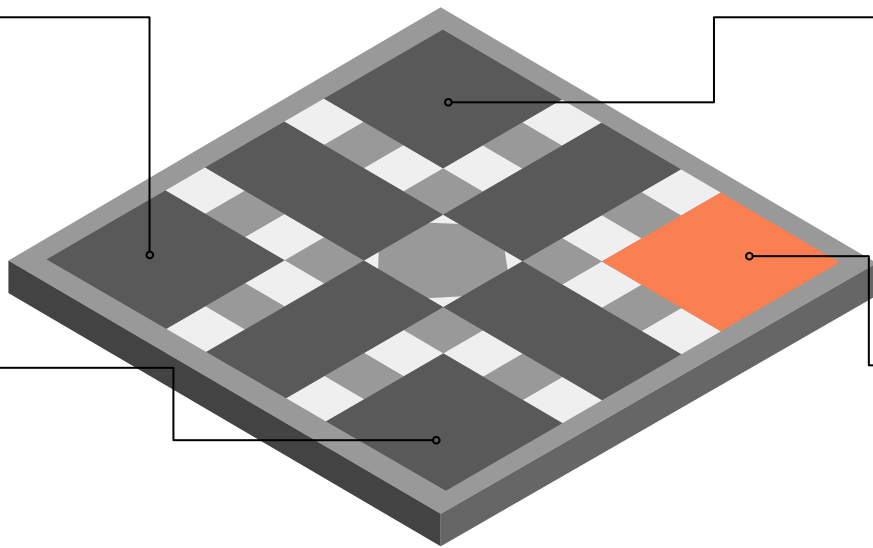
# *Coding Features*

**1: How to Play**

**2: Class Play**

**3: Class  
SugorokuBoard**

**4: Ranking**



# 4-1: Ranking Board

## Ruby gems used:

- **Colorize**
- **TTY-prompt**
- **TTY-spinner**
- **TTY-table**

## In class Play

Name	Position	Score
Clare	27	600

Press enter to roll the dice and enter again to stop

6

**Congratulations! You've reached the Goal!**

Would you like to submit your score to ranking board? (Press ↑/↓ arrow to move and Enter to select)

> Yes

No

```
#test ID 13
def save_data(name, score)
  prompt = TTY::Prompt.new
  save_output = prompt.select("Would you like to submit your score to ranking board?", ["Yes", "No"])
  if save_output == "Yes"
    # test ID 16
    begin
      File.write('files/ranking.txt', "#{name}, #{score}\n", File.size('files/ranking.txt'), mode: 'a')
    rescue
      loading_spinner(2, "Error loading file. Your score could not be added.")
    else
      loading_spinner(2, "Your score has been successfully submitted.")
    end
  elsif save_output == "no"
    puts "alright. navigating you back to menu..."
    hold_and_clear_terminal(0.7)
  end
end
```

**Add @name and @score to  
ranking.txt file**

**Error handling if file could not be  
accessed.**

**Success message if no error**

**If user doesn't wish to  
save file, go back to main  
menu**

# 4-2: Ranking Board

```
files > E ranking.txt
1 oversensitive, 3900
2 tryingit out, -600
3 secondchaaance, 700
4 clare, 1300
5 ponzu, 11000
6 Eos, 3600
7 Lia, 1400
8 dum, 1700
9 dum, 2200
10 dududuuu, 1800
11 Indiana Jones, 1500
12 Marco, 1000
13 Woopwoop, 8600
14 Yaya, 3100
15 Wee, 1000
16 Pommy, 1200
17 letitworkkk, 900
18 Clare, 600
```

- 1) **Convert txt file to hash**
- 2) **Sort hash by value**  
=> converts to array
- 3) **Add rank value**
- 4) **Create table**

```
def display_rankboard
  #convert file data to hash, sort by score and then return in nested array format
  begin
    hash = Hash[File.read('files/ranking.txt').split("\n").map{|i|i.split(', ')}]
  rescue
    puts "failed to load file"
  else
    ranking = hash.map { |k,v| [k, v.to_i]}.sort_by {|key, value| value}.reverse
    # get rid of ranks 101 and over since we only need top 100
    top_hundred = ranking.slice(0, 100)
    # then convert this data into tty table
    rankboard_table(top_hundred)
  end
end

def rankboard_table(ranks)
  # push rank number to each child array
  i = 1
  ranks.each do |one_data|
    one_data.unshift i
    i += 1
  end
  table = TTY::Table.new(["Rank", "Name", "Score"], ranks)
  puts table.render(:unicode)
end
```



Rank	Name	Score
1	ponzu	11000
2	Woopwoop	8600
3	oversensitive	3900
4	Eos	3600
5	Yaya	3100
6	dum	2200
7	dududuuu	1800
8	Indiana Jones	1500
9	Lia	1400
10	clare	1300
11	Pommy	1200
12	Wee	1000
13	Marco	1000
14	letitworkkk	900
15	secondchaaance	700
16	Clare	600
17	tryingit out	-600



# **04** ***Challenges & Issues***

# Biggest Challenge: Getting the Sugoroku Board right!!

START | 1 | 2 | m3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |

| 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 |

| 22 |

| 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | GOAL

Name	Position	Score
Woopwoop	9	1200

Press enter to roll the dice

**What about...**

Colorize after placing icon?

- Code not DRY at all

Colorize after using method?

- When passed values as argument somehow didn't work
- Code not so DRY

Use different board layout?

- What I chose to do

**To place icon in position on board panel..**

```
string.sub({to be replaced} {to replace with})
```

**Only didn't work for 3, 4, 9, 31, 32...**

**WHYYYY?!!**



# More challenges & issues

## Scope of Variables

When you have so many files and variables it gets hard to keep track of how variables are connected  
Passing in as arguments instead of using variable directly in another method/class

## Using code from Google

E.g. Stack Overflow  
Initially Copying without knowing exactly how it works  
Ethically questionable or learning experience? Where is the line?

## Testing

Time consuming  
However extremely useful in keeping track of past trials and how you changed your code

## Keeping code DRY

Moving your code to method often caused errors at first.  
Often DRY up the code and then wonder if I could make it even DRYer  
=> endless loop!  
Pleasing in the end to see files nice and clean

## Took Time to start working

Was Researching gems, coming up with logic in my head, researching examples...etc and not starting on writing code  
Wasted time  
Once I was in the zone, I had a lot of fun coding & could code efficiently

# *Features I want to add in future...*

## ***Multiplayer Mode***

Currently only a single player game  
Would be more fun to play with others  
Can compete by number of steps (i.e. fastest to reach goal wins rather than points)

## ***Selective Board Panels***

One type of panel will get boring after so many plays

Maybe:

- Basic
- Intermediate
- extreme

Depending on what kind of activity there are/length of board

## ***Selective Icons***

Icons that moves around the Sugoroku Board is currently fixed

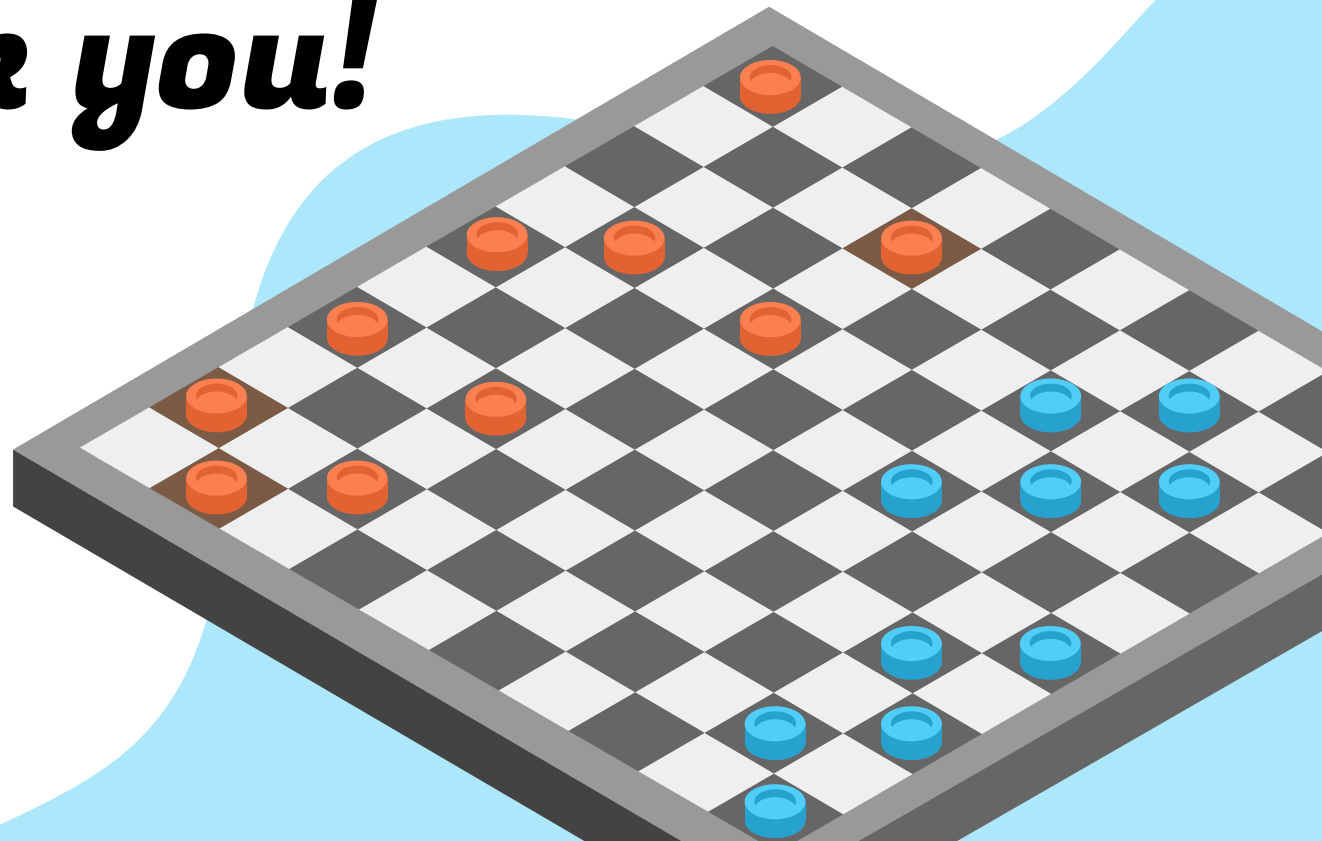
Users could choose from e.g. range of emojis

## ***More panel actions***

Rather than simply adding/subtracting/multiplying, add more activities like;

- "Move back \_\_ steps" (multiplayer)
- "Take next turn off" (multiplayer)
- Mini game to boost score

***Thank you!***



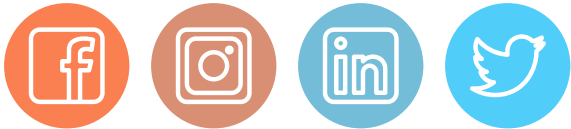
# ***THANKS!***

Do you have any questions?

addyouremail@freepik.com

+91 620 421 838

Yourcompany.com



This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.

**Please keep this slide for attribution.**

