

T1A3 - Terminal Application

Football Shootout!

A simple Ruby game by Simon Tanna

Why a football game?

- They are fun
- Football is a universal language
- My favourite game franchise is the Football Manager / Championship Manager series

The Game

What should a football sim do?

Apart from being fun...

It needs

- Teams
- Players
- A coin toss
- Phases of play
- Goals!
- Extra-time
- A winner!!!

Features & Flow

User welcome and main menu

- a. Rules
- b. Team Information
- c. Begin Game
- d. Exit

Rules

Basic game information - similar to what is in the readme.

Explains what's going to happen in-game.

```
Thank you Simon for choosing Football Shootout
Select an option below to begin your journey to
1. View Rules
2. View Team Details
3. Begin Game
4. Exit Game
Please select an option (1 - 4):_
```

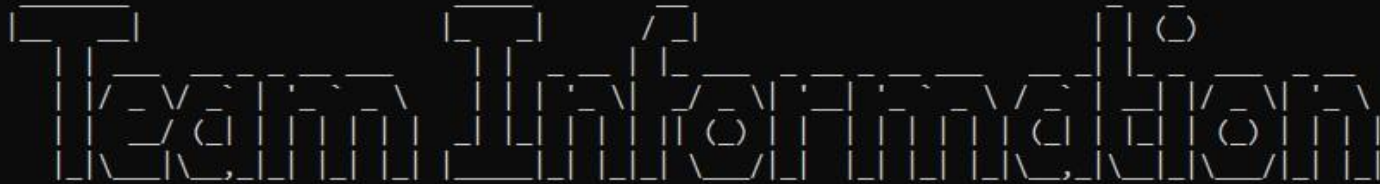
Features & Flow

Team Information

Potential opponents are listed here with randomly populated line-ups through 'faker' Ruby gem.

User can choose to view other teams or return to the main menu.

Error handling with rescue and retry featured in this menu.



Choose a team to view the line-up...(use ↑/↓ arrows on your keyboard) (Press ↑/↓ arrow to move and Enter to select)

☒ Gianluigi Buffon's Barons

Fabian Barthez's Bandits

Andoni Zubizarreta's Zoomers

Mark Schwarzer's Scamps

Features & Flow

Main Game (1)

Create a team of legends

The user will be asked to create a team, give them a name and select 5 legendary players.

Select opposition

Coin Toss

Who kicks off?

If the user wins the toss...

3 attacking and 2 defending phases.



```
WELCOME TO THE GOAT  
Team Name: Simon's GOATS  
Team Captain: Simon  
Players:  
Number: 10 | Name: Pele  
Number: 5 | Name: Franz Beckenbauer  
Number: 1 | Name: Lev Yashin  
Number: 7 | Name: Didier Deschamps  
Number: 2 | Name: Cafu  
Press any key to continue and select your opponent...
```

Features & Flow

Main Game (2)

If the user loses the toss

3 defending and 2 attacking phases.

In each phase there will be a choice for the user to make

- In attack: Shoot or Pass
- In defence: Slide Tackle or Block Tackle

If choice = true then...

- In attack: GOOAAALLL!!!!
- In defence: You win possession

If choice = false then...

- In attack: You lose possession
- In defence: Goal to your opponent

```
Score Simon's GOATS: 1 - 0 :Fabian Barthez's Bandits
Prepare to defend!
Fabian Barthez's Bandits are looking dangerous and are
taking the ball into the final third of the pitch. The
winger is looking to go past you. Do you: (Press ↑/↓ a
rrow to move and Enter to select)
▣ slide tackle
  block tackle_
```


Features & Flow

Main Game (3)

If the user score = bot team score...

EXTRA-TIME!!!

The user will have one last chance to win the game and will have the option to shoot to the left or right of the goalkeeper.

If choice = true then they WIN!

If choice = false they lose...

User will have the option to return to main and create another team.



The Code

index.rb

seed.rb

game.rb

teams.rb

players.rb

Players Class

Simple attributes :

Name and Number

To be used by the Teams class to populate “players” array.

players.rb X

classes > players.rb

You, 2 days ago | 1 author (You)

```
1  class Player
2      attr_accessor :name, :number
3  def initialize(name, number)
4      |   @name = name
5      |   @number = number
6  end
7
8  def to_s
9      |   "Number: #{@number} | Name: #{@name}"
10 end
11
12 def player_name
13     |   "#{@name}"
14 end
15 end
16
```

Teams Class

Attributes:

To be used in the game to determine the truthy or falsey values of game events, the coin toss result and to store the team score.

all_team_info method used in Team Information menu and pre-game.

```
teams.rb M x
classes > teams.rb
You, a day ago | 1 author (You)

1 class Teams
2   attr_reader :captain
3   attr_accessor :name, :players, :toss, :attack_1, :attack_2,
4   def initialize(name, players, score, captain, toss, attack_1
5     @name = name
6     @players = players
7     @score = score
8     @captain = captain
9     @toss = toss
10    @attack_1 = attack_1
11    @attack_2 = attack_2
12    @attack_3 = attack_3
13    @defend_1 = defend_1
14    @defend_2 = defend_2
15    @defend_3 = defend_3
16    @extra = extra
17  end
18
19  def to_s
20    "#{@captain}'s #{@name}"
21  end
22
23  def all_team_info
24    puts "Team Name: #{@captain}'s #{@name}"
25    puts "Team Captain: #{@captain}"
26    puts "Players:"
27    @players.each {|player| player}
28  end
29
```

Game Class

Used to store all teams and players

Seed File

Stores all bot team information

‘Faker’ Ruby gem used to populate player names.

Teams passed into array and then into Game class for use in the index.rb

```
for i in 1..5  
  team_1_players << Player.new(Faker::Sports::Football.unique.player,  
end
```

```
team_1 = Teams.new("Barons", team_1_players, 0, team_1_captain,  
team_2 = Teams.new("Bandits", team_2_players, 0, team_2_captain  
team_3 = Teams.new("Zoomers", team_3_players, 0, team_3_captain  
team_4 = Teams.new("Scamps", team_4_players, 0, team_4_captain,  
# player_team = Teams.new(player_team_name, player_team_players
```

```
teams_array = [team_1, team_2, team_3, team_4]  
game = Game.new("Football Shootout", "5-a-Side Football Sim", t  
return game
```

The Game (index.rb)

Methods for in-game selection options.

- tty-prompt used.

Methods for game phases

- User selects 1 of 2 options

The truthy/falsey value of these variables are determined when the user team is created...

```
def team_select_info(prompt,game)
  team = prompt.select("Choose a team to view")
  return team
end

def team_select_user(prompt,game)
  user_team = prompt.select("Choose your team")
  return user_team
end

def team_defend_3(prompt,game,user_team,bot_team)
  defend_options = ["slide tackle", "block tackle"]
  defend = prompt.select("#{bot_team} are storming into your half")
  return defend
end

def team_extra_time(prompt,game,user_team,bot_team)
  attack_options = ["shoot left", "shoot right"]
  attack = prompt.select("You are one-on-one with the goalkeeper")
  return attack
end
```

The Game (index.rb)

Team Creation

Team name, players and all in-game options are stored here.

In-game prompts variables are set by .sample method. Returns a random element

```
def create_team(prompt,game,player_name,ascii_slant)
  puts ascii_slant.asciify("Time to Create Your Team of Legends!").colorize(
  team_name = ""
  captain = "#{player_name}"
  team_new_players = []
  team_new_toss = ["heads", "tails"].sample
  team_attack_1 = ["pass", "shoot"].sample
  team_attack_2 = ["dribble then shoot", "shoot from range"].sample
  team_attack_3 = ["pass", "shoot"].sample
  team_defend_1 = ["slide tackle", "block tackle"].sample
  team_defend_2 = ["slide tackle", "block tackle"].sample
  team_defend_3 = ["slide tackle", "block tackle"].sample
  team_extra = ["shoot left", "shoot right"].sample
  choice_1 = Player.new("Pele", 10), Player.new("Diego Maradona", 10)
  choice_2 = Player.new("Bobby Moore", 6), Player.new("Franz Beckenbauer", 5)
  choice_3 = Player.new("Lev Yashin", 1), Player.new("Dino Zoff", 1)
  choice_4 = Player.new("Didier Deschamps", 7), Player.new("Paul Breitner", 4)
  choice_5 = Player.new("Cafu", 2), Player.new("Marco Tardelli", 14)
  puts "What would you like to call your team?"
  team_name = gets.chomp
  puts "Select your legends"
  team_new_players << prompt.select("Choose your star striker: ", choice_1)
  team_new_players << prompt.select("Choose your defensive rock: ", choice_2)
  team_new_players << prompt.select("Choose your iron-gloved goalkeeper: ", choice_3)
  team_new_players << prompt.select("Choose your midfield general: ", choice_4)
  team_new_players << prompt.select("Choose your rampaging full-back: ", choice_5)
  team_user = Teams.new(team_name, team_new_players, 0, captain, team_new_toss)
end
```

You, 2 days ago - created player team class, not sure if it will

The Game (index.rb)

The Match

User input is passed into this method.

Return of score value is passed into the user or bot team variable.

- Loops used:
 - if-else
 - if-elsif-else
 - while

```
if toss == user_team.toss
  puts ascii.asciify("You have won the toss!").colorize(:green)
  35.times do
    sleep(0.03)
    kickoff_bar.advance
  end
end
system "clear"
15.times {puts ascii_slant.asciify("PEEP!!!").colorize(:color => :blue)}
sleep(0.1)
system "clear"
puts ascii_slant.asciify("PEEP!!!").colorize(:color => :white, :bold => true)
sleep(0.1)
system "clear"
puts "The referee blows the whistle to start the game. You are
attack_choice_1 = team_attack_1(prompt,game,user_team)
if attack_choice_1 == user_team.attack_1
  user_team.score += 1
  puts "GOAL"
  puts "Score #{user_team}: #{user_team.score} - #{bot_team.score}"
else
  puts "You have given away position to the opposition..."
end
```

The Game (index.rb)

Extra-time

Nested if-else loop determines the winner if the scores are level after 5 phases of play.

```
if user_team.score == bot_team.score
  puts "The scores are as follows..."
  puts "#{user_team}: #{user_team.score}"
  puts "#{bot_team}: #{bot_team.score}"
  puts "The scores are level... Extra-Time"
  puts "The referee decides that #{user_team} have been the better"
  puts "PEEP! The referee's whistle blows and your team is immediately"
  extra_time_choice = team_extra_time(prompt,game,user_team,bot_team)
  if extra_time_choice == user_team.extra
    user_team.score += 1
    puts "GOAL!!!"
    puts "#{user_team} WINS"
    puts "Congratulations"
  else
    bot_team.score += 1
    puts "The goalkeeper saves!"
    puts "He spots his striker unmarked upfield and boots the ball..."
    puts "The opposition striker controls the ball...."
    puts "He spots your goalkeeper off his line and takes a long shot..."
    puts "GOAL! #{bot_team} WINS"
    puts "Better luck next time"
  end
end
```

The Game (index.rb)

Gems Used

```
require 'tty-prompt'  
require 'colorize'  
require 'artii'  
require 'tty-progressbar'  
require 'tty-font'  
require 'faker'
```

Build Process

- Control Flow and Data Flow
 - Useful when stuck on elements of the code
- Trello
 - Really really useful when trying to decide “what to do next!?!”
- Ethical Issue
 - Using real football player names
 - If this game was to be released commercially, permission would be needed.
- Favourite Part
 - The result
 - Building something that began as an idea through to a real and tangible thing!
 - Being able to create a football team of legends



The Result