Ultimate Foosball

Canyon · Minsun · Brandon · Kydon
PLTW P.O.E - 4B

TABLE OF CONTENTS

01

PROJECT OVERVIEW

This slide will provide a brief
overview of this project

02

DESIGN BRIEF & PROJECT PLAN

This section highlights the
planning stage of our project

03

BRAINSTORMING SKETCHES & DECISION MATRIX

This section highlights the brainstorming stage
and selecting a potential solution

04

FINAL SKETCHES & PRODUCT

This section will demonstrate our
final product (code and physical
product)

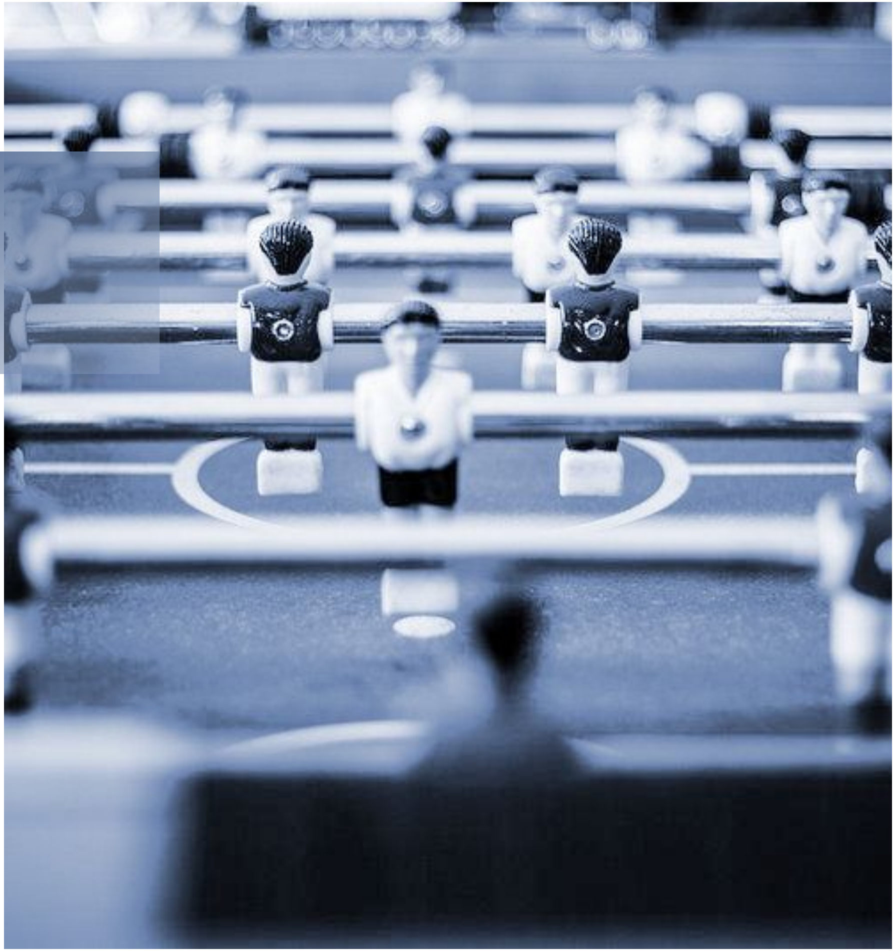
05

THANK YOU

We are grateful for your interest in this
project and its development

PROJECT OVERVIEW

1

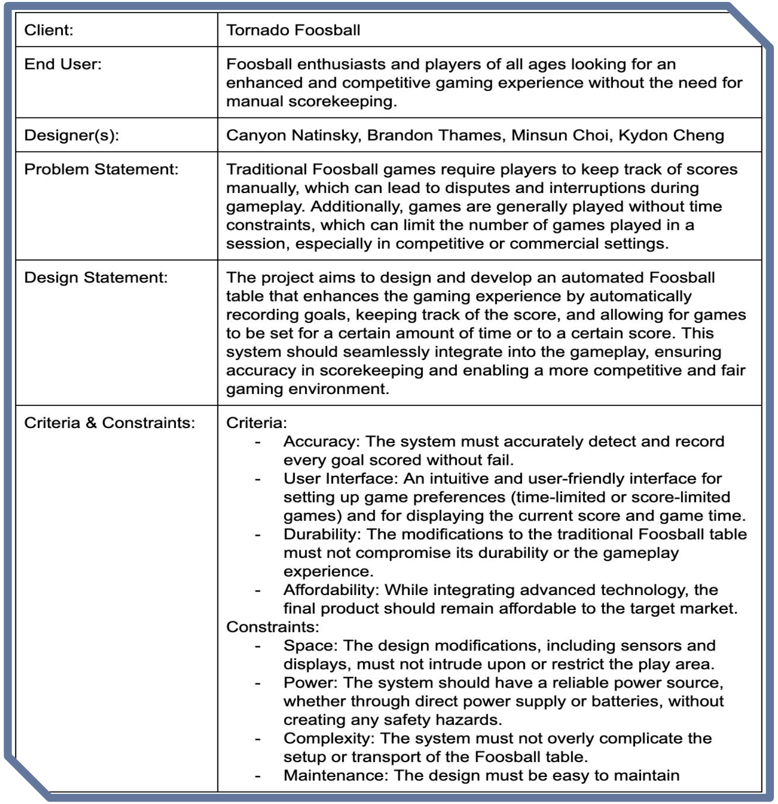


PROJECT OVERVIEW

Our project, Ultimate Foosball, modernizes the traditional foosball table with an automated scoring system that enhances gameplay through technology. Players can customize their game by choosing a target score or a time limit, with the system automatically recording scores and determining winners. An overtime feature resolves ties, ensuring every match is competitive. This integration of mechanical, electrical, and software engineering principles elevates the foosball experience.

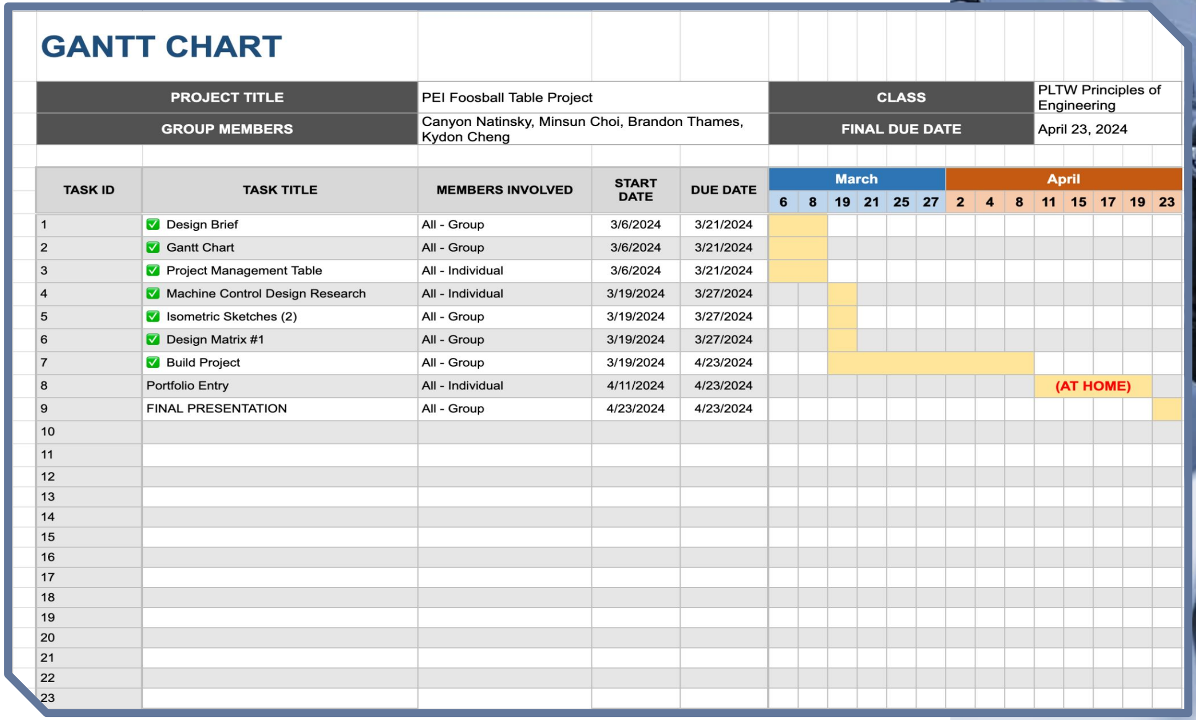
DESIGN BRIEF & PROJECT PLAN

Design Brief



Client:	Tornado Foosball
End User:	Foosball enthusiasts and players of all ages looking for an enhanced and competitive gaming experience without the need for manual scorekeeping.
Designer(s):	Canyon Natinsky, Brandon Thames, Minsun Choi, Kydon Cheng
Problem Statement:	Traditional Foosball games require players to keep track of scores manually, which can lead to disputes and interruptions during gameplay. Additionally, games are generally played without time constraints, which can limit the number of games played in a session, especially in competitive or commercial settings.
Design Statement:	The project aims to design and develop an automated Foosball table that enhances the gaming experience by automatically recording goals, keeping track of the score, and allowing for games to be set for a certain amount of time or to a certain score. This system should seamlessly integrate into the gameplay, ensuring accuracy in scorekeeping and enabling a more competitive and fair gaming environment.
Criteria & Constraints:	Criteria: - Accuracy: The system must accurately detect and record every goal scored without fail. - User Interface: An intuitive and user-friendly interface for setting up game preferences (time-limited or score-limited games) and for displaying the current score and game time. - Durability: The modifications to the traditional Foosball table must not compromise its durability or the gameplay experience. - Affordability: While integrating advanced technology, the final product should remain affordable to the target market. Constraints: - Space: The design modifications, including sensors and displays, must not intrude upon or restrict the play area. - Power: The system should have a reliable power source, whether through direct power supply or batteries, without creating any safety hazards. - Complexity: The system must not overly complicate the setup or transport of the Foosball table. - Maintenance: The design must be easy to maintain

Gantt Chart



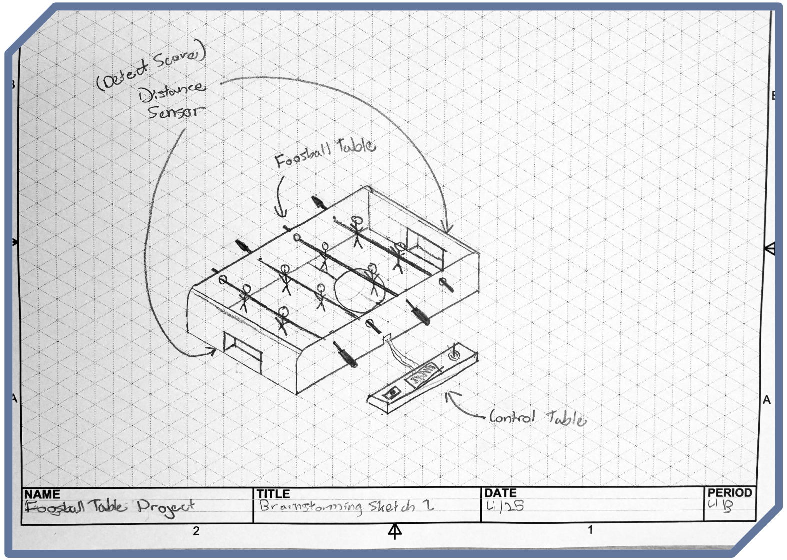
GANTT CHART

PROJECT TITLE	PEI Foosball Table Project	CLASS	PLTW Principles of Engineering
GROUP MEMBERS	Canyon Natinsky, Minsun Choi, Brandon Thames, Kydon Cheng	FINAL DUE DATE	April 23, 2024

TASK ID	TASK TITLE	MEMBERS INVOLVED	START DATE	DUE DATE	March						April							
					6	8	19	21	25	27	2	4	8	11	15	17	19	23
1	✅ Design Brief	All - Group	3/6/2024	3/21/2024	🟨	🟨												
2	✅ Gantt Chart	All - Group	3/6/2024	3/21/2024	🟨	🟨												
3	✅ Project Management Table	All - Individual	3/6/2024	3/21/2024	🟨	🟨												
4	✅ Machine Control Design Research	All - Individual	3/19/2024	3/27/2024			🟨											
5	✅ Isometric Sketches (2)	All - Group	3/19/2024	3/27/2024			🟨											
6	✅ Design Matrix #1	All - Group	3/19/2024	3/27/2024			🟨											
7	✅ Build Project	All - Group	3/19/2024	4/23/2024			🟨	🟨	🟨	🟨	🟨	🟨	🟨					
8	Portfolio Entry	All - Individual	4/11/2024	4/23/2024										🟨 (AT HOME)	🟨	🟨	🟨	
9	FINAL PRESENTATION	All - Group	4/23/2024	4/23/2024														🟨
10																		
11																		
12																		
13																		
14																		
15																		
16																		
17																		
18																		
19																		
20																		
21																		
22																		
23																		

BRAINSTORMING SKETCHES & DECISION MATRIX

Potential Physical Sketches



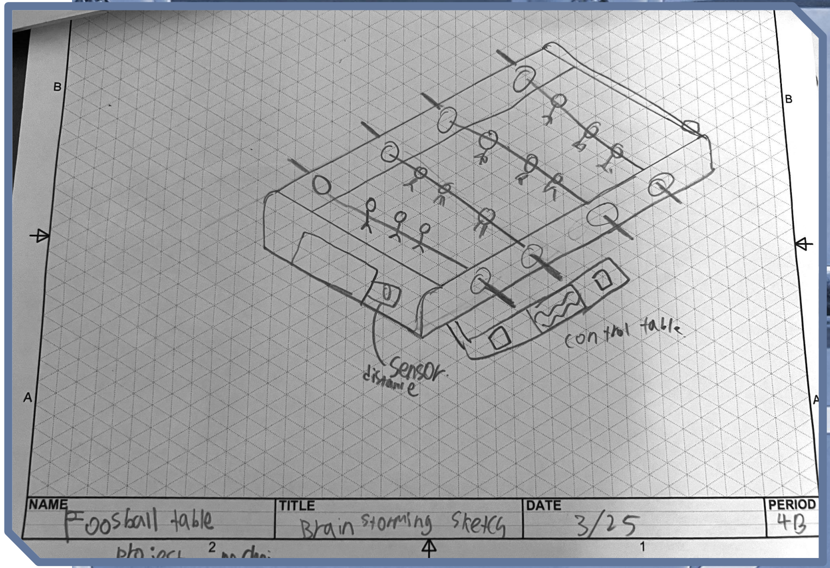
NAME Foosball Table Project

TITLE Brainstorming Sketch 1

DATE 4/25

PERIOD 4B

1



NAME Foosball table project

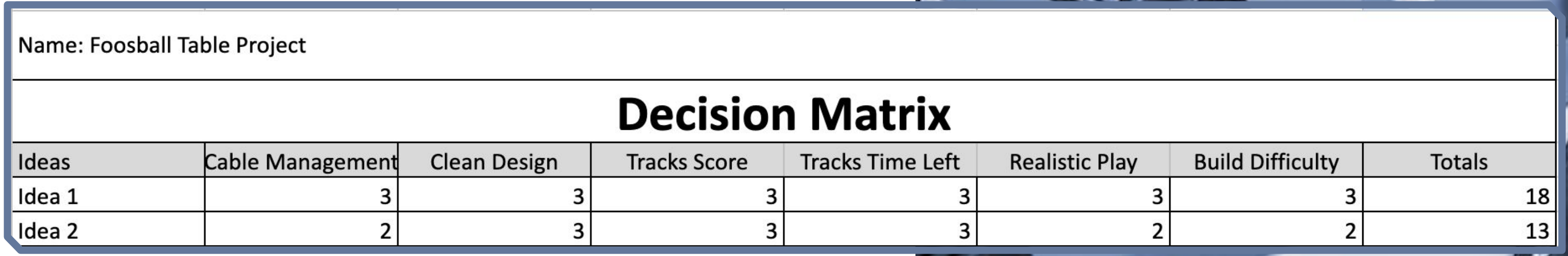
TITLE Brain storming Sketch

DATE 3/25

PERIOD 4B

2

Decision Matrix

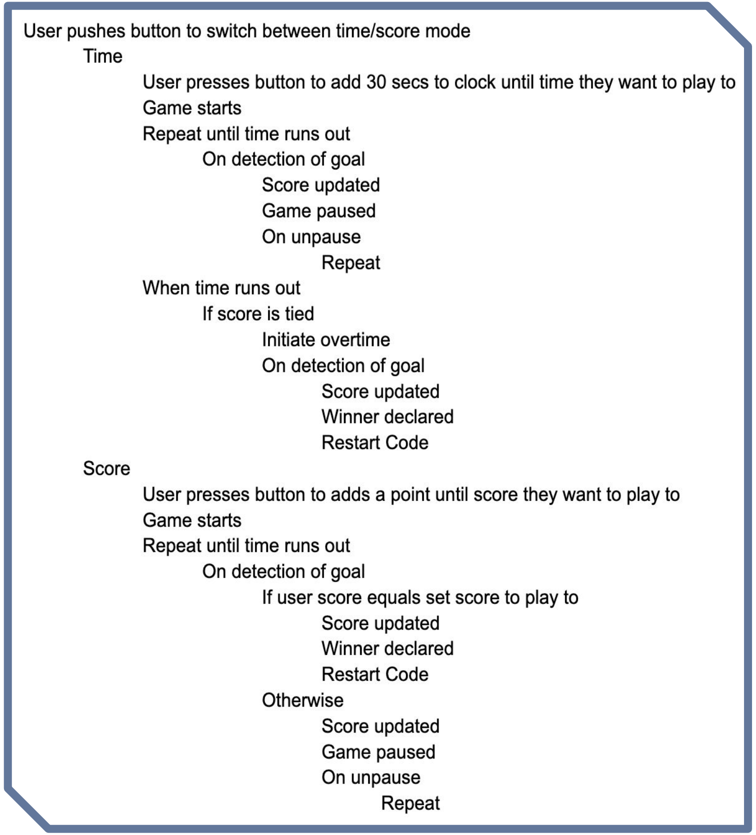


Name: Foosball Table Project

Decision Matrix

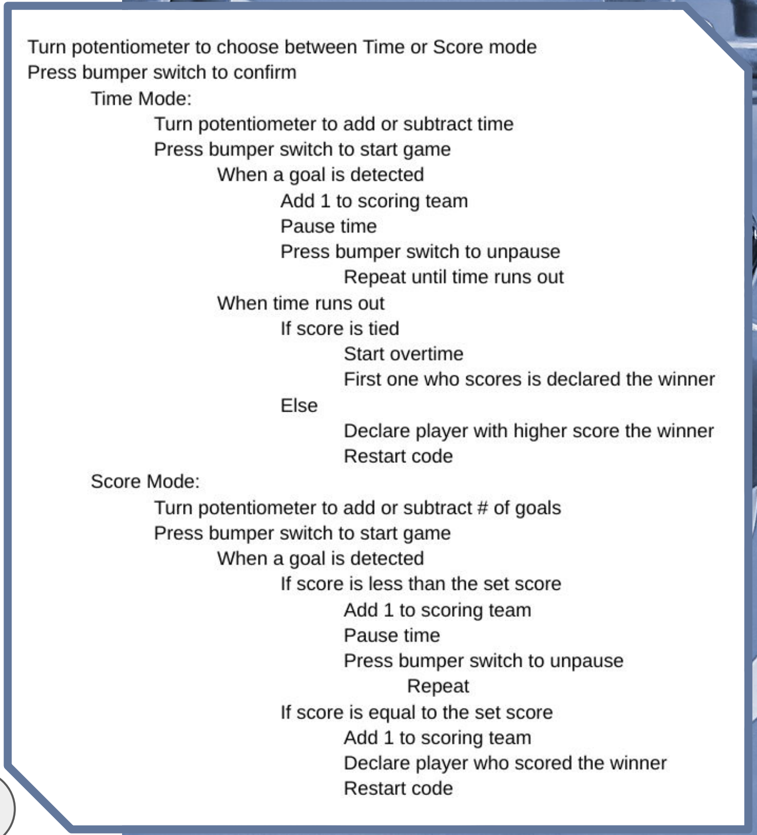
Ideas	Cable Management	Clean Design	Tracks Score	Tracks Time Left	Realistic Play	Build Difficulty	Totals
Idea 1	3	3	3	3	3	3	18
Idea 2	2	3	3	3	2	2	13

Potential Program Sketches



```
User pushes button to switch between time/score mode
    Time
        User presses button to add 30 secs to clock until time they want to play to
        Game starts
        Repeat until time runs out
            On detection of goal
                Score updated
                Game paused
                On unpause
                    Repeat
        When time runs out
            If score is tied
                Initiate overtime
                On detection of goal
                    Score updated
                    Winner declared
                    Restart Code
    Score
        User presses button to adds a point until score they want to play to
        Game starts
        Repeat until time runs out
            On detection of goal
                If user score equals set score to play to
                    Score updated
                    Winner declared
                    Restart Code
                Otherwise
                    Score updated
                    Game paused
                    On unpause
                        Repeat
```

1

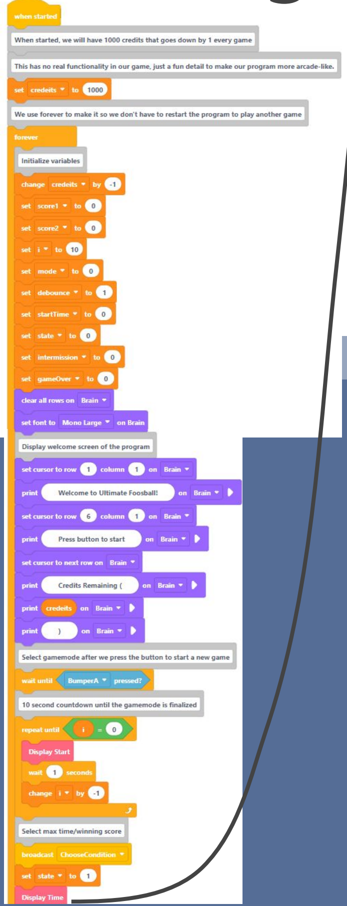


```
Turn potentiometer to choose between Time or Score mode
Press bumper switch to confirm
    Time Mode:
        Turn potentiometer to add or subtract time
        Press bumper switch to start game
            When a goal is detected
                Add 1 to scoring team
                Pause time
                Press bumper switch to unpause
                    Repeat until time runs out
            When time runs out
                If score is tied
                    Start overtime
                    First one who scores is declared the winner
                Else
                    Declare player with higher score the winner
                    Restart code
    Score Mode:
        Turn potentiometer to add or subtract # of goals
        Press bumper switch to start game
            When a goal is detected
                If score is less than the set score
                    Add 1 to scoring team
                    Pause time
                    Press bumper switch to unpause
                        Repeat
                If score is equal to the set score
                    Add 1 to scoring team
                    Declare player who scored the winner
                    Restart code
```

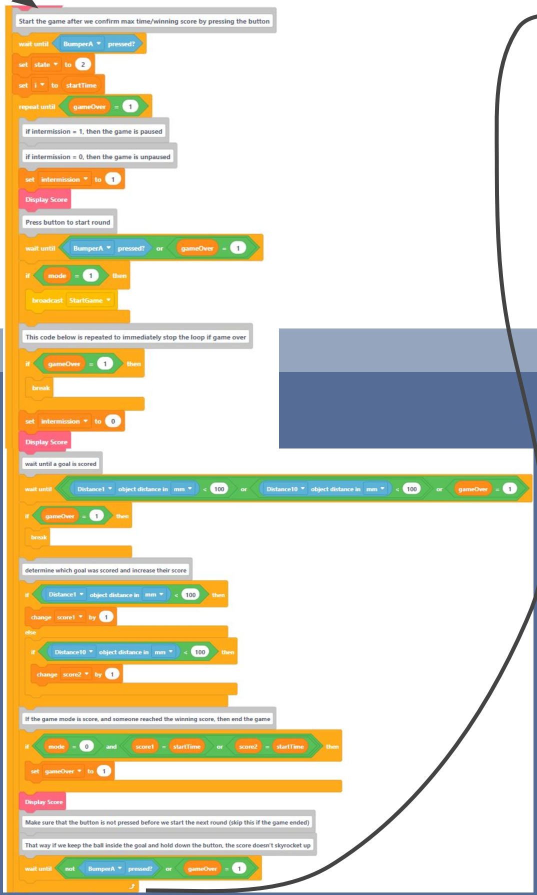
2

FINAL SKETCHES & PRODUCT

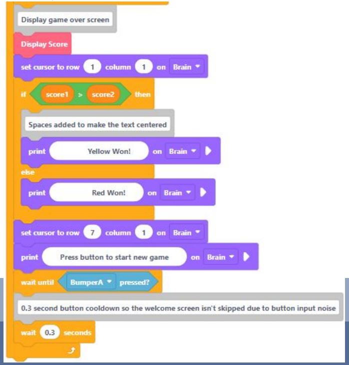
Final Program w/ Descriptions



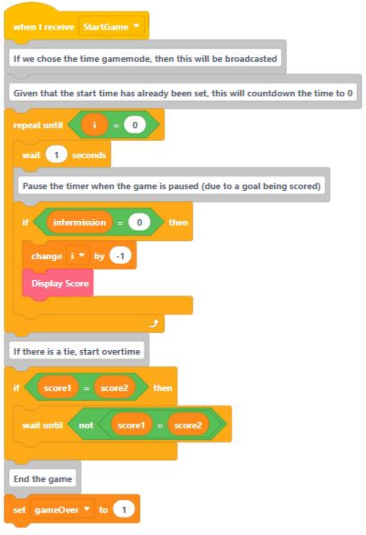
```
when started
When started, we will have 1000 credits that goes down by 1 every game
This has no real functionality in our game, just a fun detail to make our program more arcade-like.
set credeits to 1000
We use forever to make it so we don't have to restart the program to play another game
forever
    Initialize variables
    change credeits by -1
    set score1 to 0
    set score2 to 0
    set i to 10
    set mode to 0
    set debounce to 1
    set startTime to 0
    set state to 0
    set intermission to 0
    set gameOver to 0
    clear all rows on Brain
    set font to Mono Large on Brain
    Display welcome screen of the program
    set cursor to row 1 column 1 on Brain
    print Welcome to Ultimate Foosball! on Brain
    set cursor to row 6 column 1 on Brain
    print Press button to start on Brain
    set cursor to next row on Brain
    print Credits Remaining ( on Brain
    print credeits on Brain
    print ) on Brain
    Select gamemode after we press the button to start a new game
    wait until BumperA pressed?
    10 second countdown until the gamemode is finalized
    repeat until i = 0
        Display Start
        wait 1 seconds
        change i by -1
    Select max time/winning score
    broadcast ChooseCondition
    set state to 1
    Display Time
```



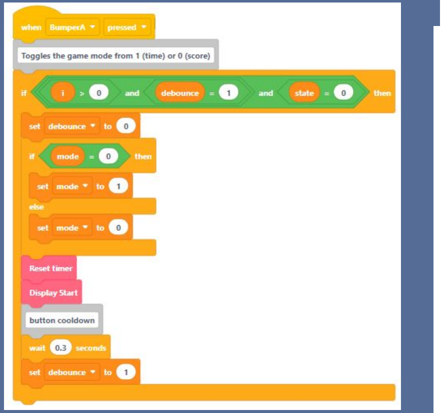
```
    Start the game after we confirm max time/winning score by pressing the button
    wait until BumperA pressed?
    set state to 2
    set i to startTime
    repeat until gameOver = 1
        if intermission = 1, then the game is paused
        if intermission = 0, then the game is unpaused
        set intermission to 1
        Display Score
        Press button to start round
        wait until BumperA pressed? or gameOver = 1
        if mode = 1 then
            broadcast StartGame
        This code below is repeated to immediately stop the loop if game over
        if gameOver = 1 then
            break
        set intermission to 0
        Display Score
        wait until a goal is scored
        wait until Distance1 object distance in mm < 100 or Distance10 object distance in mm < 100 or gameOver = 1
        if gameOver = 1 then
            break
        determine which goal was scored and increase their score
        if Distance1 object distance in mm < 100 then
            change score1 by 1
        else
            if Distance10 object distance in mm < 100 then
                change score2 by 1
        If the game mode is score, and someone reached the winning score, then end the game
        if mode = 0 and score1 = startTime or score2 = startTime then
            set gameOver to 1
        Display Score
        Make sure that the button is not pressed before we start the next round (skip this if the game ended)
        That way if we keep the ball inside the goal and hold down the button, the score doesn't skyrocket up
        wait until not BumperA pressed? or gameOver = 1
```



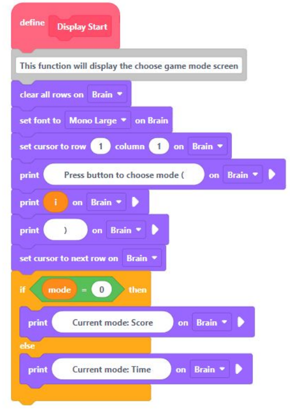
```
    Display game over screen
    Display Score
    set cursor to row 1 column 1 on Brain
    if score1 > score2 then
        Spaces added to make the text centered
        print Yellow Won! on Brain
    else
        print Red Won! on Brain
    set cursor to row 7 column 1 on Brain
    print Press button to start new game on Brain
    wait until BumperA pressed?
    0.3 second button cooldown so the welcome screen isn't skipped due to button input noise
    wait 0.3 seconds
```



```
when I receive StartGame
If we chose the time gamemode, then this will be broadcasted
Given that the start time has already been set, this will countdown the time to 0
repeat until i = 0
    wait 1 seconds
    Pause the timer when the game is paused (due to a goal being scored)
    if intermission = 0 then
        change i by -1
        Display Score
If there is a tie, start overtime
if score1 = score2 then
    wait until not score1 = score2
End the game
set gameOver to 1
```

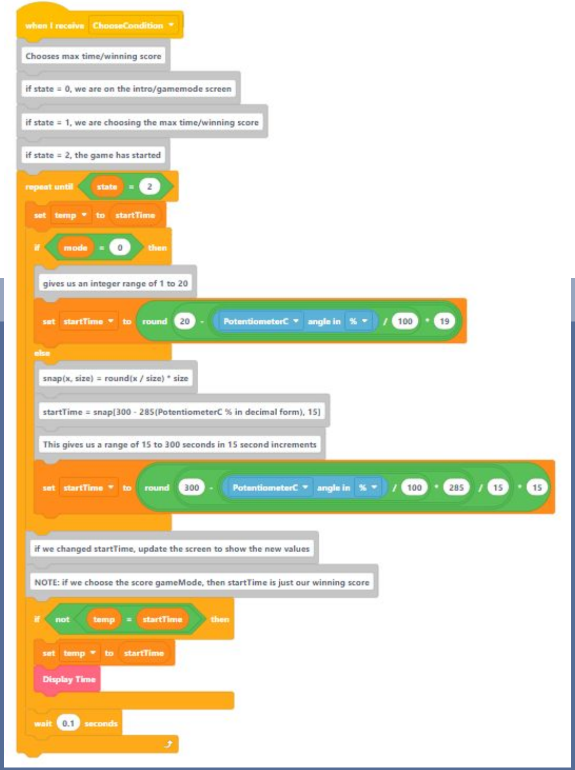


```
when BumperA pressed
Toggles the game mode from 1 (time) or 0 (score)
if i > 0 and debounce = 1 and state = 0 then
    set debounce to 0
    if mode = 0 then
        set mode to 1
    else
        set mode to 0
    Reset timer
    Display Start
    button cooldown
    wait 0.3 seconds
    set debounce to 1
```

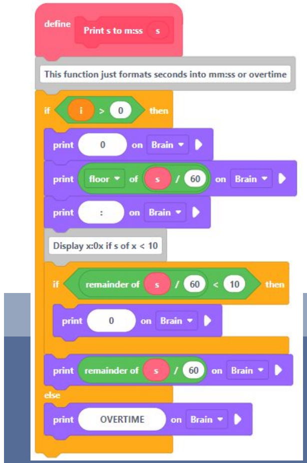


```
define Display Start
This function will display the choose game mode screen
clear all rows on Brain
set font to Mono Large on Brain
set cursor to row 1 column 1 on Brain
print Press button to choose mode ( on Brain
print i on Brain
print ) on Brain
set cursor to next row on Brain
if mode = 0 then
    print Current mode: Score on Brain
else
    print Current mode: Time on Brain
```

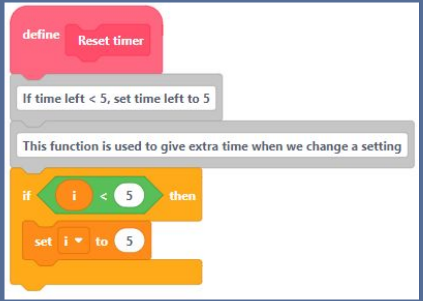
Final Program w/ Descriptions



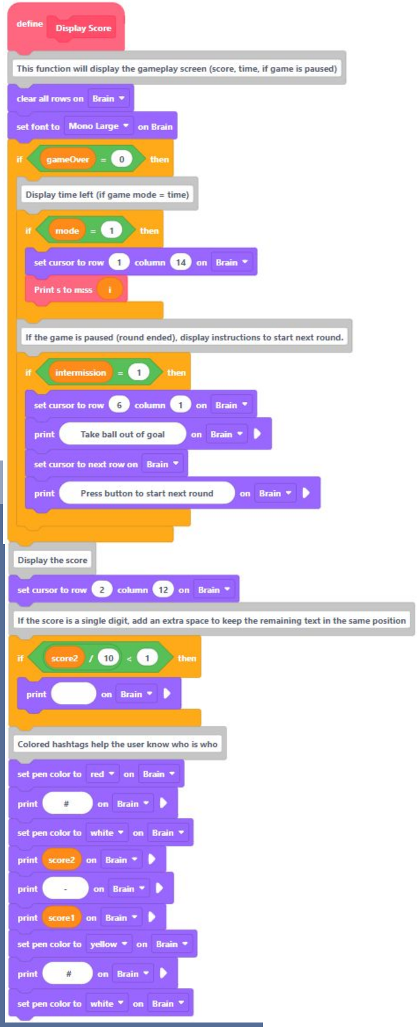
```
when I receive ChooseCondition
Chooses max time/winning score
if state = 0, we are on the intro/gamemode screen
if state = 1, we are choosing the max time/winning score
if state = 2, the game has started
repeat until state = 2
    set temp to startTime
    if mode = 0 then
        gives us an integer range of 1 to 20
        set startTime to round 20 - PotentiometerC angle in % / 100 * 19
    else
        snap(x, size) = round(x / size) * size
        startTime = snap[300 - 285(PotentiometerC % in decimal form), 15]
        This gives us a range of 15 to 300 seconds in 15 second increments
        set startTime to round 300 - PotentiometerC angle in % / 100 * 285 / 15 * 15
    if we changed startTime, update the screen to show the new values
    NOTE: if we choose the score gameMode, then startTime is just our winning score
    if not temp = startTime then
        set temp to startTime
        Display Time
    wait 0.1 seconds
```



```
define Print s to m:ss s
This function just formats seconds into mm:ss or overtime
if i > 0 then
    print 0 on Brain
    print floor of s / 60 on Brain
    print : on Brain
    Display x:0x if s of x < 10
    if remainder of s / 60 < 10 then
        print 0 on Brain
    print remainder of s / 60 on Brain
else
    print OVERTIME on Brain
```

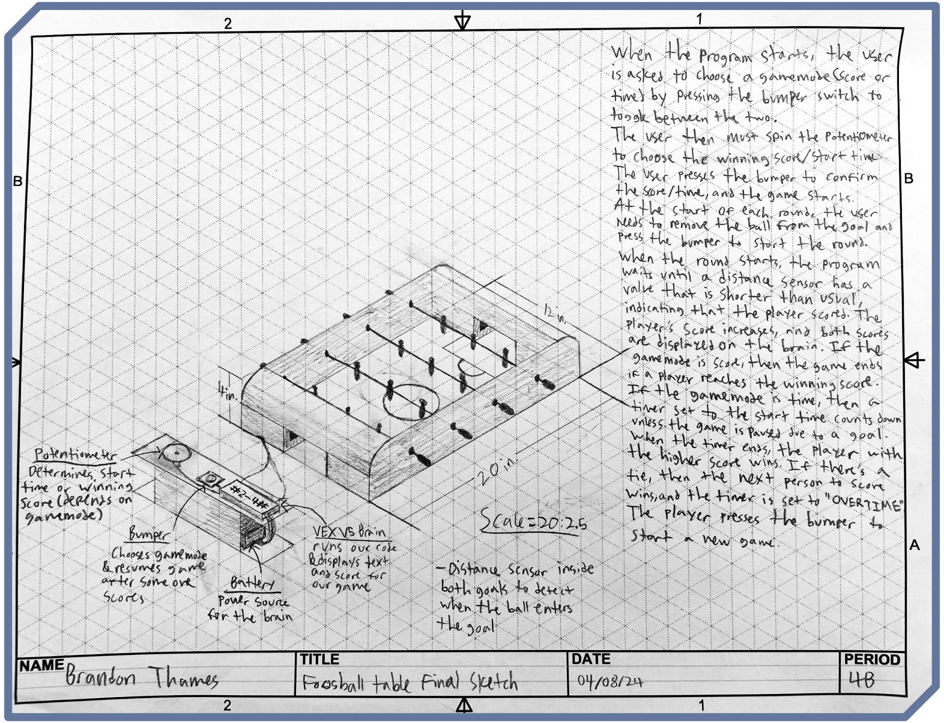


```
define Reset timer
If time left < 5, set time left to 5
This function is used to give extra time when we change a setting
if i < 5 then
    set i to 5
```



```
define Display Score
This function will display the gameplay screen (score, time, if game is paused)
clear all rows on Brain
set font to Mono Large on Brain
if gameOver = 0 then
    Display time left (if game mode = time)
    if mode = 1 then
        set cursor to row 1 column 14 on Brain
        Print s to m:ss i
    If the game is paused (round ended), display instructions to start next round.
    if intermission = 1 then
        set cursor to row 6 column 1 on Brain
        print Take ball out of goal on Brain
        set cursor to next row on Brain
        print Press button to start next round on Brain
Display the score
set cursor to row 2 column 12 on Brain
If the score is a single digit, add an extra space to keep the remaining text in the same position
if score2 / 10 < 1 then
    print   on Brain
Colored hashtags help the user know who is who
set pen color to red on Brain
print # on Brain
set pen color to white on Brain
print score2 on Brain
print - on Brain
print score1 on Brain
set pen color to yellow on Brain
print # on Brain
set pen color to white on Brain
```

Final Physical Sketch



When the program starts, the user is asked to choose a gamemode (score or time) by pressing the bumper switch to toggle between the two.
The user then must spin the potentiometer to choose the winning score/start time.
The user presses the bumper to confirm the score/time, and the game starts.
At the start of each round, the user needs to remove the ball from the goal and press the bumper to start the round.
When the round starts, the program waits until a distance sensor has a value that is shorter than usual, indicating that the player scored. The player's score increases, and both scores are displayed on the brain. If the gamemode is score, then the game ends if a player reaches the winning score. If the gamemode is time, then a timer set to the start time counts down unless the game is paused due to a goal. When the timer ends, the player with the higher score wins. If there's a tie, then the next person to score wins, and the timer is set to "OVERTIME". The player presses the bumper to start a new game.

Potentiometer
Determines start time or winning score (depends on gamemode)

Bumper
Chooses gamemode & resumes game after some one scores

Battery
Power source for the brain

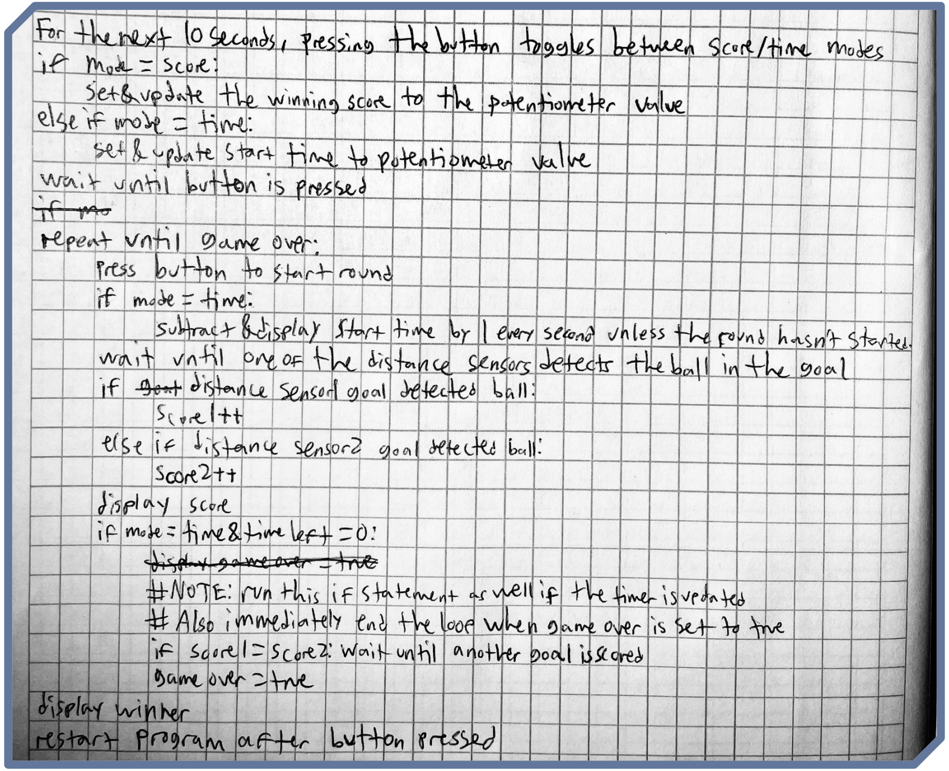
VEX V5 Brain
runs our code & displays text and score for our game

Scale = 20:2.5

- Distance sensor inside both goals to detect when the ball enters the goal

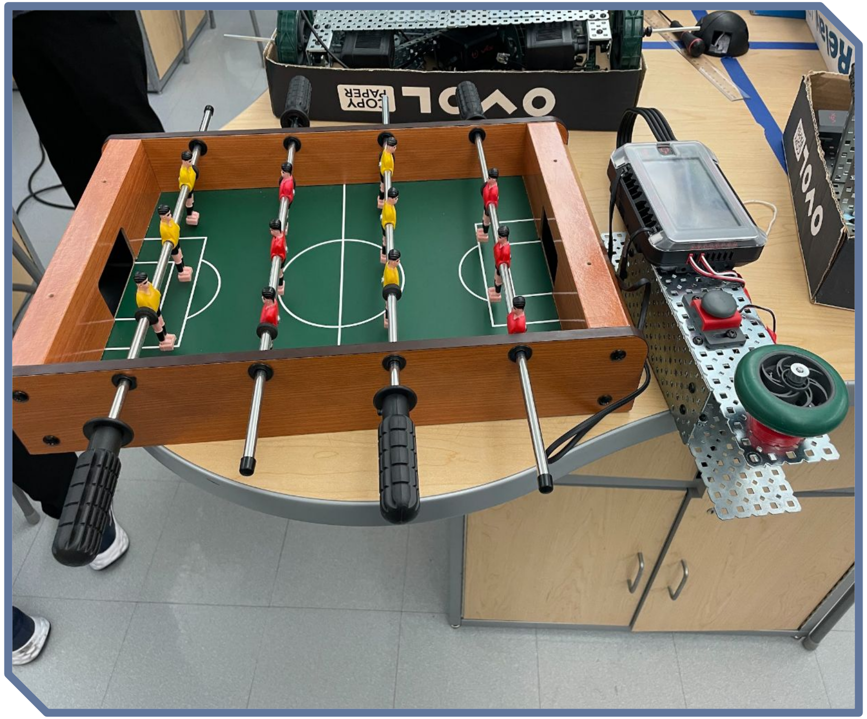
NAME	TITLE	DATE	PERIOD
Brandon Thames	Foosball table Final Sketch	04/08/24	4B

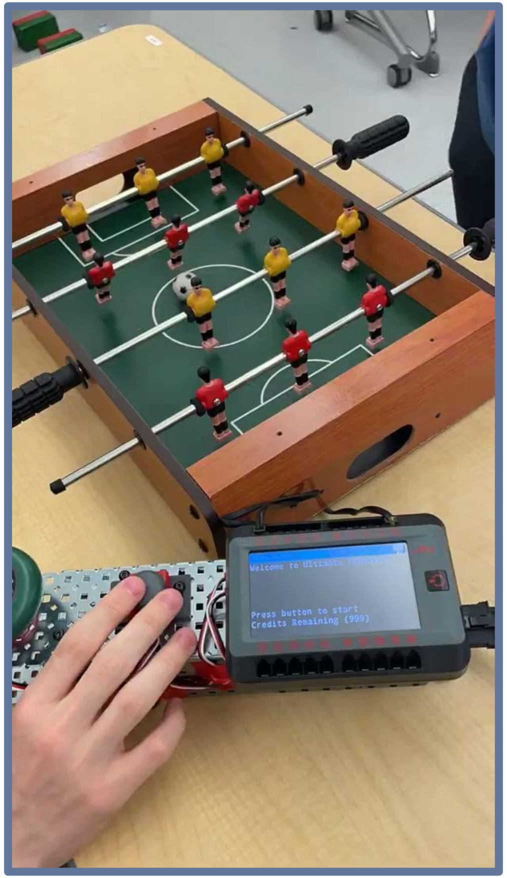
Final Program Sketch



```
For the next 10 seconds, pressing the button toggles between score/time modes
if Mode = score:
    set & update the winning score to the potentiometer value
else if mode = time:
    set & update start time to potentiometer value
wait until button is pressed
~~if mo~~
repeat until game over:
    press button to start round
    if mode = time:
        subtract & display start time by 1 every second unless the round hasn't started.
    wait until one of the distance sensors detects the ball in the goal
    if ~~Goal~~ distance sensor1 goal detected ball:
        Score1++
    else if distance sensor2 goal detected ball:
        Score2++
    display score
    if mode = time & time left = 0:
        ~~display game over = true~~
        #NOTE: run this if statement as well if the timer is updated
        # Also immediately end the loop when game over is set to true
        if score1 = score2: wait until another goal is scored
        game over = true
display winner
restart program after button pressed
```

Final Solution





```
Welcome to Ultimate [illegible]

Press button to start
Credits Remaining (999)
```

THANK YOU