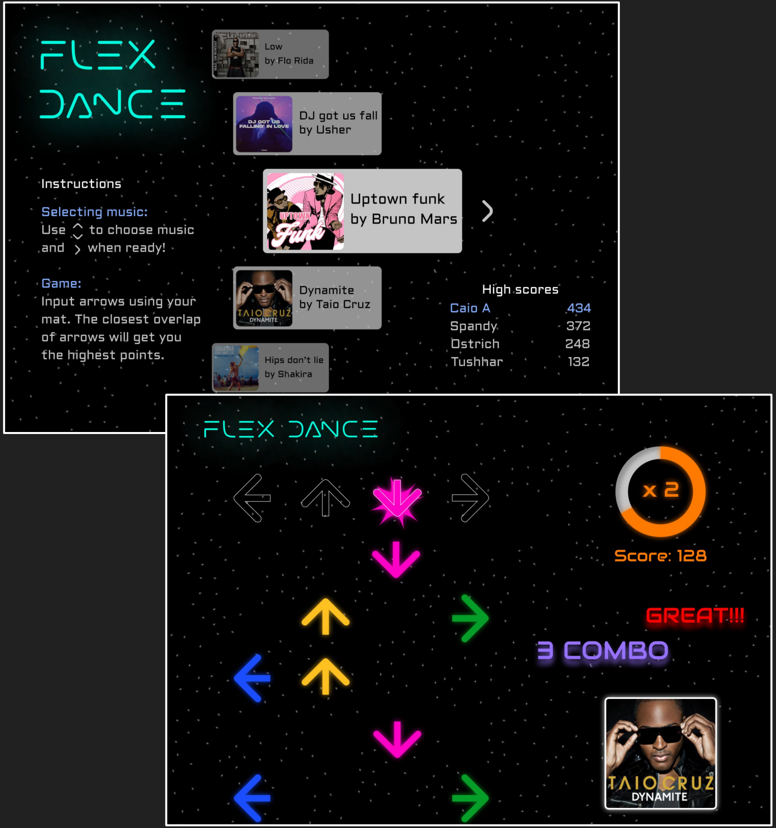
Team A3 - Flex Dance

Spandan Sharma, Caio Araujo, Tushhar Saha

Contents:

- Use Case / Application
- Use-Case Requirements, especially quantitative
- Solution Approach (include Design Requirements here)
- System Specification / Block Diagram
- Implementation Plan
- Test, Verification and Validation Plans
- Project Management



FLEX DANCE

Low
by Flo Rida

DJ got us fall
by Usher

Uptown funk
by Bruno Mars

Dynamite
by Taio Cruz

Hips don't lie
by Shakira

Instructions

Selecting music:
Use ⌃⌄ to choose music
and › when ready!

Game:
Input arrows using your mat. The closest overlap of arrows will get you the highest points.

High scores

Caio A	434
Spandy	372
Ostrich	248
Tushhar	132

FLEX DANCE

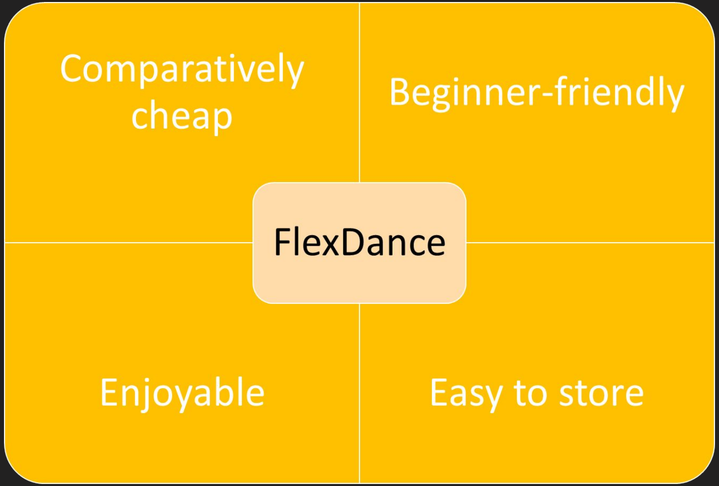
x 2

Score: 128

GREAT!!!

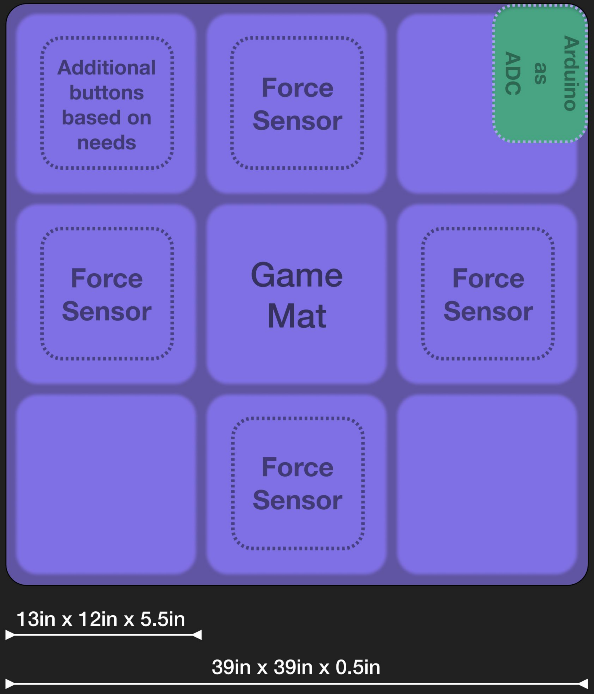
3 COMBO

Use Case and Requirements



Requirement	Why?
Purchase cost < $200	Affordable; current alternatives start from $300
Storage size: Folded size < 13in x 12in x 5.5in Unfolded size < 39i x 39in	Easy to store; Size of average drawer; Living room space
Easy installation: connect to display through HDMI cable	Accessibility: game should be able to be installed by a child or older people
Force detection threshold ~10lbs	Resting foot weight

Use Case and Requirements



Requirement	Why?
Easy to start the game: Game screen should be 3 clicks away	Beginner-friendly interface
Differentiate between pressing and holding buttons	Necessary for navigation between screens and gameplay
Stimulating while respecting visual weight: follow 60-30-10 rule	Keep user engaged while not overwhelming or confusing them
Lasts ~650 sessions	Assuming exercising 4 days a week and expected 3 years of use

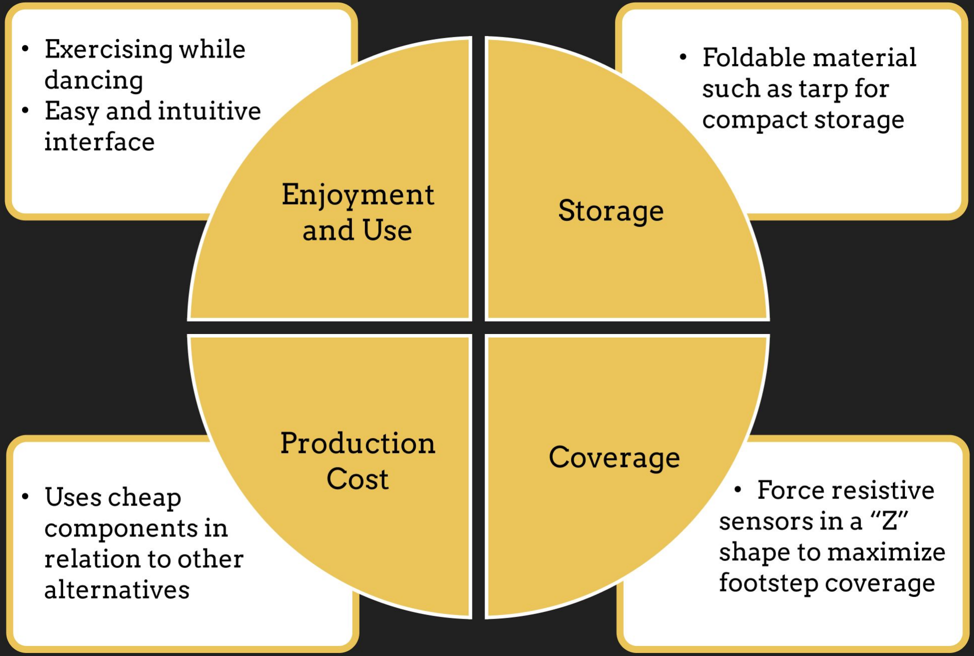
Use Case and Requirements

Perfect player should ace 1 in 4 games (25%). If a 3 minute song track has an average of 200 arrows (a little more than 1 arrow/second), we get:

- ➔ Probability of scoring every arrow correctly = 0.25
- ➔ $(1\text{-error rate})^{200} = 0.25$
- ➔ Error rate ~= 1%

Requirement	Why?
Error rate < 1%	See calculation on the side
Latency of signal between Arduino to Raspberry Pi < 100ms	Humans perceive images in 1/10th of a second
Arrow button 360° coverage	User can press the buttons in any feet orientation
Linear scoring scale	Users expect to receive points even if they don't time their step perfectly

Solution Approach



Enjoyment and Use

- Exercising while dancing
- Easy and intuitive interface

Storage

- Foldable material such as tarp for compact storage

Production Cost

- Uses cheap components in relation to other alternatives

Coverage

- Force resistive sensors in a "Z" shape to maximize footstep coverage

Foldable Dance Dance Revolution (DDR) Kit

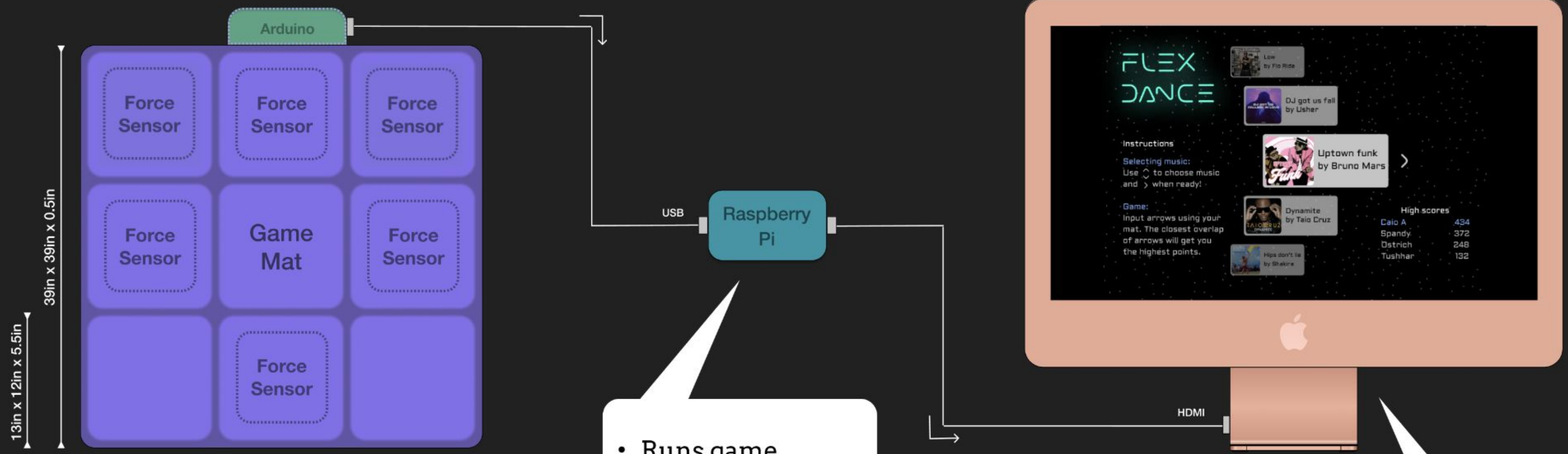
○ Raspberry Pi

○ Force Sensitive Resistors

○ Arduino as an ADC

○ Pygame

System Specification: Components



Instructions

Selecting music:
Use ⌃⌄ to choose music and › when ready!

Game:
Input arrows using your mat. The closest overlap of arrows will get you the highest points.

High scores	
Caio A	434
Spandy	372
Ostrich	248
Tushhar	132

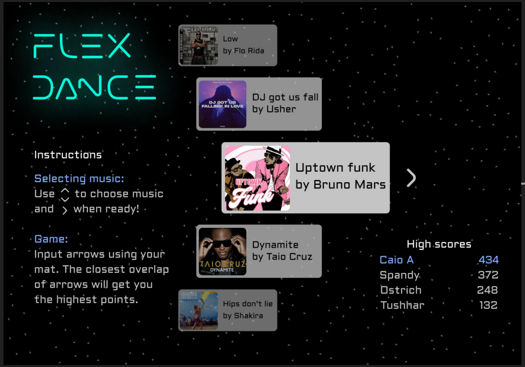
- ADC
- Detects input from mat

- Runs game
- Stores music files

- Any HDMI display
- Shows game interface

FLEX DANCE

System Specification: Game interface

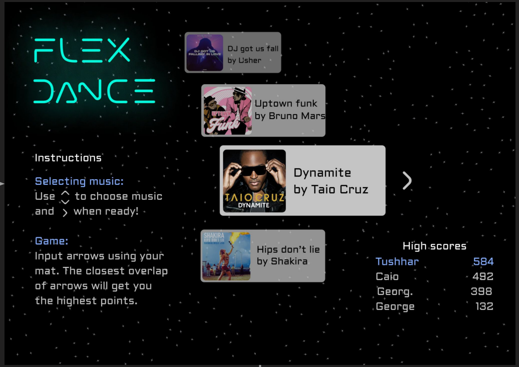


Instructions

Selecting music:
Use ⌃⌄ to choose music
and › when ready!

Game:
Input arrows using your mat. The closest overlap of arrows will get you the highest points.

High scores	
Caio A	434
Spandy	372
Ostrich	248
Tushhar	132

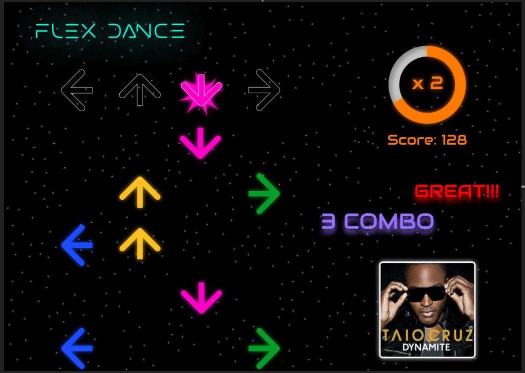


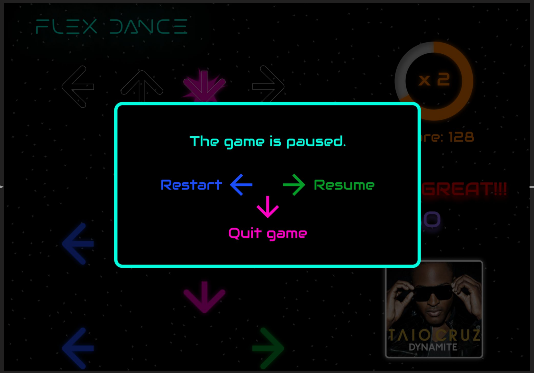
Instructions

Selecting music:
Use ⌃⌄ to choose music
and › when ready!

Game:
Input arrows using your mat. The closest overlap of arrows will get you the highest points.

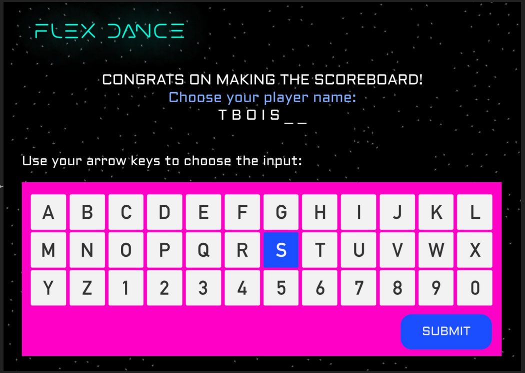
High scores	
Tushhar	584
Caio	492
Georg.	398
George	132





The game is paused.

Restart ← → Resume
↓
Quit game



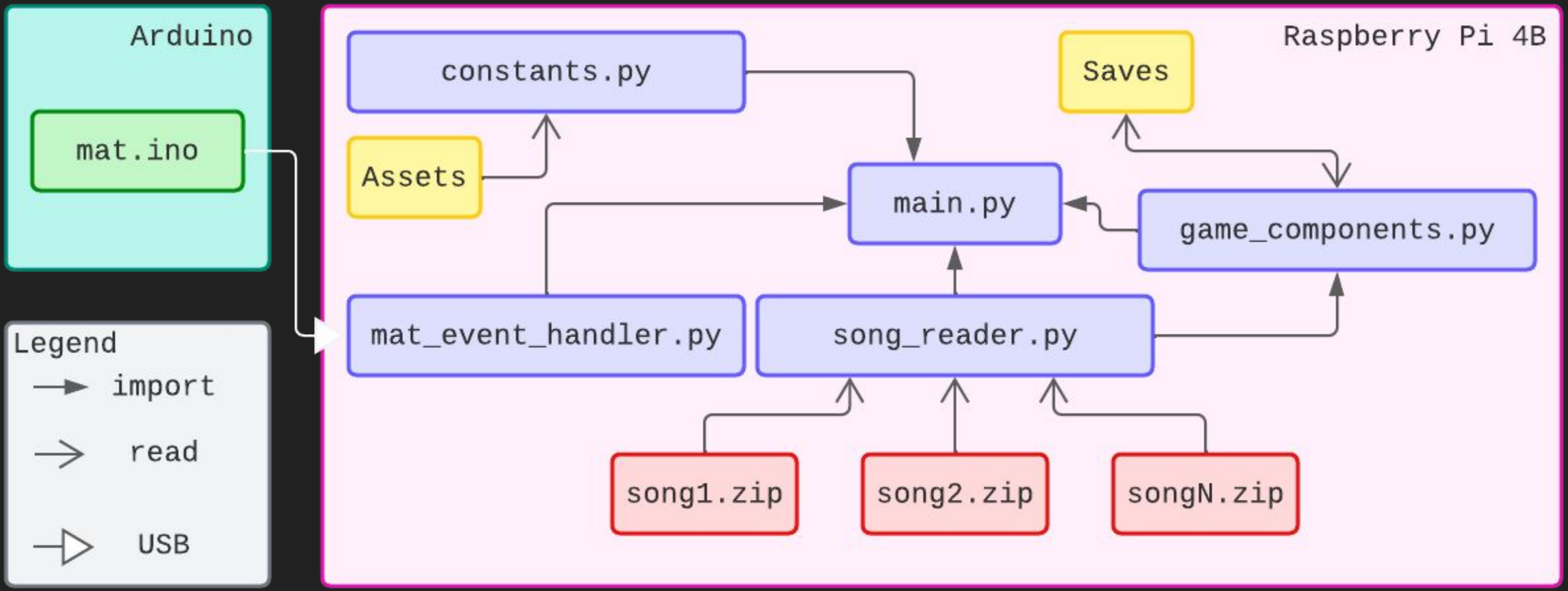
CONGRATS ON MAKING THE SCOREBOARD!
Choose your player name:
TBOIS_ _

Use your arrow keys to choose the input:

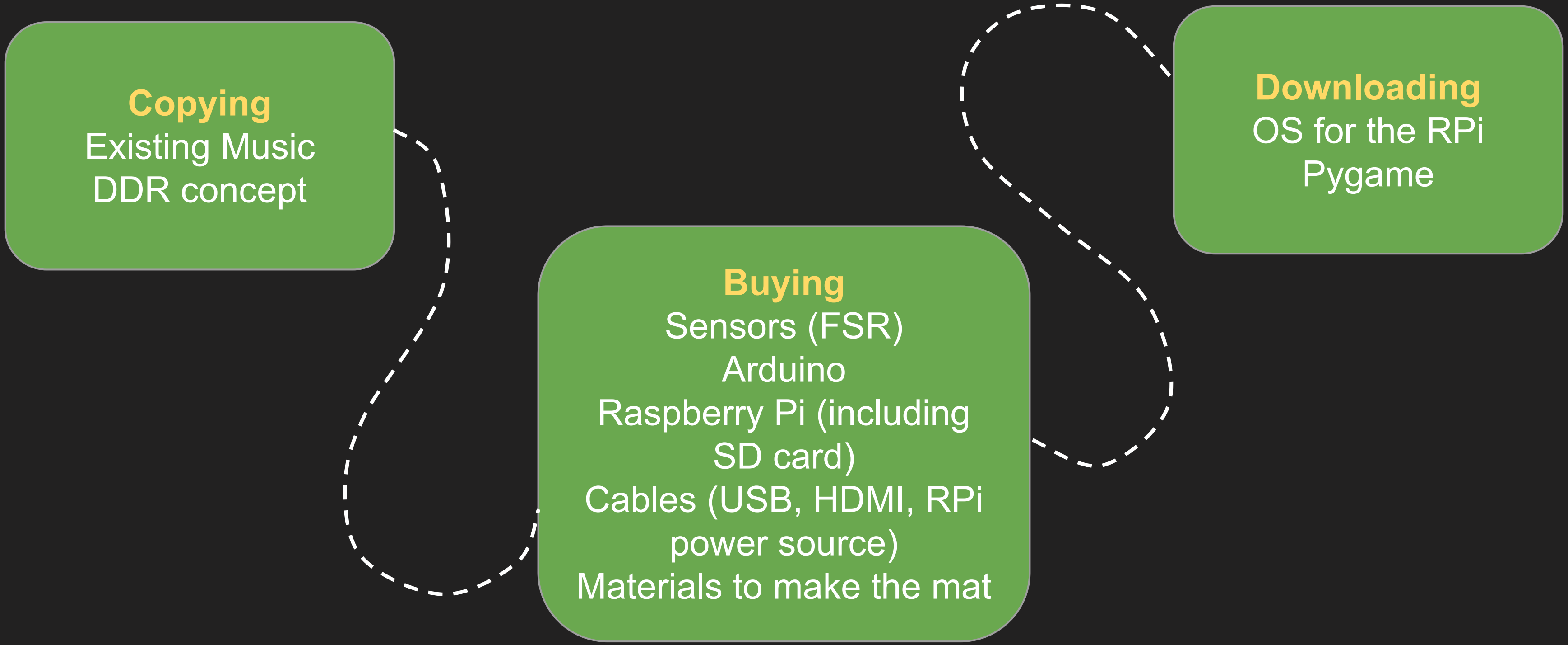
A	B	C	D	E	F	G	H	I	J	K	L
M	N	O	P	Q	R	S	T	U	V	W	X
Y	Z	1	2	3	4	5	6	7	8	9	0

SUBMIT

System Specification: Code structure



Implementation Plan



Copying
Existing Music
DDR concept

Buying
Sensors (FSR)
Arduino
Raspberry Pi (including SD card)
Cables (USB, HDMI, RPi power source)
Materials to make the mat

Downloading
OS for the RPi
Pygame

Implementation Plan

Assembling	Designing and Developing on our own
Combining all the components for the mat: ● FSRs ● Wires ● Tarp ● Arduino	Creating our own arrow sequences (choreo) for the music GUI Game software Plastic box for Arduino (with a hole for wires)

Testing, Verification, and Validation - Mat

Requirement	How to measure	Goal	Mitigation
Folded size	Measuring tape	<= 13 in x 12 in x 5.5 in	Reduce size of individual squares
Unfolded size	Measuring tape	<= 39 in x 39 in	" "
Minimum force detected	Arduino serial monitor and force gauge	~ 10 lbs	Change thresholds in Arduino code
Arrow button coverage	Step on the mat in different orientations	Circular shape	Specify foot position that is acceptable

Testing, Verification, and Validation - Game

Requirement	How to measure	Goal	Mitigation
Latency	Measure time between sending and receiving signal using python	<= 0.1 s	Less external assets in the game
Error rate	Step on the mat and register successful detections	<= 1%	Change thresholds in Arduino code
Fair scoring scale	Have a few people play the game and state if they get frustrated	<= 25% people frustrated	More forgiving scores
Cost	Components' cost	<= $180	Raspberry Pi 3B and cheaper mat

Schedule and division of labor

	Feb 21 - 27	Feb 28 - Mar 6	Mar 7 - 13	Mar 14 - 20	Mar 21 - 27	Mar 28 - Apr 3	Apr 4 - 10	Apr 11 - 17	Apr 18 - 24	Apr 25 - May 1
Caio	Find good OS Game screen	Implement game screen		Implement game screen	Implement game screen					
Tushhar	Test sensors	Read Arduino from RPi		Construct physical mat		Test & calibrate sensors				
Spandan	Implement menu screen	Implement menu screen	Spring break	Implement menu screen	Implement menu screen	Make song playable with arrow keys			Buffer (Debugging)	
Caio & Spandan	Plan custom track file format					Interim Demo				
All		DR Report			Construct physical mat	Integrate game and mat	Integrate game and mat		Final presentation	Peer review