

RHYTHM RUSH



Created by **NEXUS**

Aidan Ryder • Amarie Fasoro • Ikem Enebeli • Ruairidh Murdoch • Yousuf Shahabuddin

Human Centred Design • DE1

CONTEXT: RESEARCH SUMMARY

How might we design a gamified experience that organically encourages social interactions in event queues while maintaining flow and engagement?



WHY A GAMIFIED EXPERIENCE?

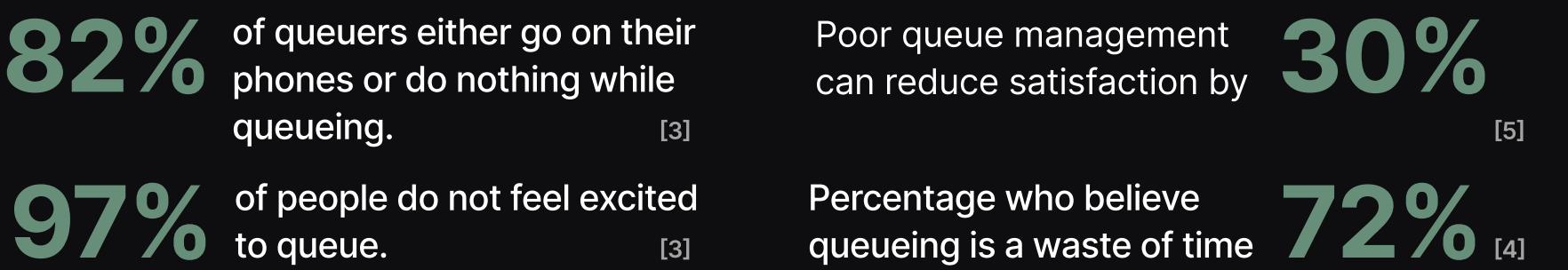
Our previous research showed that a gamified experience lowers the social barrier by offering an easy, shared activity that feels less risky and more rewarding than a typical conversation with a stranger. [1]

DESK RESEARCH: GENERAL OPINION



WHY A QUEUE?

Using mixed methodologies, our research showed queues as underused social spaces where most people disengage, making them ideal for playful interaction.



OUR DEMOGRAPHIC

We are designing for 18–35 year olds queuing at large events like concerts, sports games, conventions, and theme parks. These queues often happen in the evening at busy venues, especially in London, where people share common interests, making them great spaces for social interaction and playful experiences.

Through our second survey with 34 people from our demographic, we found that most prefer to receive information through a mix of text and images, with a stronger preference for visuals.

Our prior observations at existing spaces supported this data, also suggesting people can get disinterested if there are extra steps to interact with a product such as QR codes or a downloadable app.

KEY INSIGHTS: INTERVIEWS



People are hesitant to engage with strangers due to safety concerns. From 24 interviews, we found that shared interests, like attending an event, can help lower social barriers.

- This shared context fosters familiarity and trust, making interactions feel more natural and organic, reducing the discomfort of engaging with others.

KEY INSIGHTS: EXPERIMENTS



In social gatherings it takes just one person/group to break the surface tension.

- Once a space has been deemed ‘social’, people are happy to join in. This herd mentality results in more people talking.

Gamified interactions are more effective for socialising.

- In contrast, conventional conversation starters lack the same compelling draw.
- This indicates that structured play creates a psychological invitation perceived as lower risk and higher benefit compared to open-ended conversation.

KEY INSIGHTS: OBSERVATIONS



Phones act as social shields in queues.

- When people are on their phones they are already engaged with familiar activity, making them less receptive to new connections.
- While technology keeps them socially engaged, it isolated them from their immediate surroundings.

HMWS → DIRECTIONS

How might we create an interactive experience that adapts to the pace of a moving queue without causing delays or disruptions?

- Design Direction:** Test how the interaction between different users and the product affects queue flow. If issues are consistent, design a feature(s) that adjust to movement in real time.

How might we design engaging, low-pressure activities that allow both active participants and spectators to feel included in the experience?

- Design Direction:** A focus could also be put on viewing areas to encourage social interactions among spectators.

How might we make the process of joining conversations feel more inviting and less intimidating for others so individuals take the first step in overcoming social tension?

- Design Direction:** Explore different ways to make the concept more approachable (i.e. providing optional participation).

At its core, the problem isn't about queue boredom, it's about creating moments that feel safe, social, and worth engaging in, even for those who wouldn't normally participate.

SOLUTION STATEMENT

We aim to create a gamified experience that invites optional, low-pressure participation to naturally encourage social interaction in event queues, whether the user is playing, spectating, or simply in the proximity of it. The experience should foster a shared social atmosphere, to act as a conversation aid while maintaining queue flow.

IDEATION: CONCEPT SELECTION

CONCEPTS

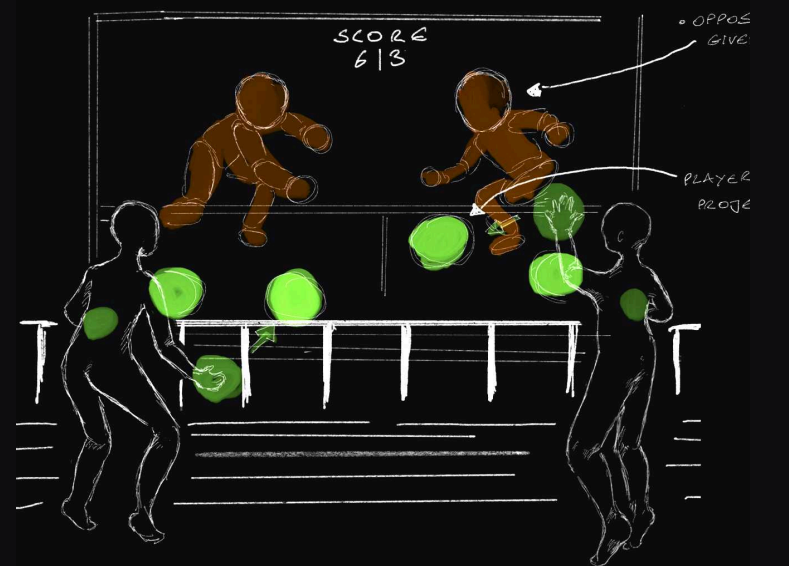
DRESS-UP MIRRORS



The screen acts as a mirror using a camera to show players with **real-time digital clothing** based on their selections from a control interface.

(Conventions & Concerts)

KINETIC PORTALS



Using **two screens** and body-tracking, up to **four players can compete in games**. The screens act as **portals between queue areas**, with matches adapting to player numbers.

(All Events)

RHYTHM RUSH



A motion-tracked **dance battle for up to six players**, on a large screen. Players compete based on **enthusiasm** and **creativity**.

(Concerts)

USER CENTRED EVALUATION

HARRIS: QUANTITATIVE^[7]

While choosing the final concept, we used a **Harris Profile**: a decision matrix where users scored each idea against key **user-centred criteria**. The data was gathered through discussions with university students.

	Dress Up Mirrors				Kinetic Portals				Rhythm Rush			
	-2	-1	+1	+2	-2	-1	+1	+2	-2	-1	+1	+2
Enjoyment			■				■	■			■	■
Social Engagement			■				■	■			■	
Replayability	■	■					■	■			■	■
Pace Fit			■		■						■	■
Atmosphere Fit		■					■				■	■
Learnability			■	■	■				■			

Rhythm Rush was the **highest-ranked** concept with 8 points, ahead of Kinetic Portals (6 points) and Dress-Up Mirrors (1 point).

HARRIS: QUALITATIVE

- Learnability** - users emphasised the need for the game to be intuitive and accessible to players of all skill levels
- Pace-Fit** - The game's practicality in a queue setting was also questioned. People asked things like, "Will my spot be saved?" and "If the queue moves quickly, do I miss my turn?"
- Re-playability** - Greater song variety was also requested amongst participants to increase re-playability.

HARRIS: CONCLUSION

The focus of user testing will be on **reducing the skill barrier** to make the game more accessible to players of all levels, and improving learnability so that users can **quickly understand how to engage with the system**. At the same time, testing will ensure that the product integrates more smoothly with the **natural pace and flow of the queue**.

EMOTIONAL ANALYSIS

PREMO: QUANTITATIVE^[7]

We asked **six people** to evaluate **all three concepts using emotion cards** to express how they'd feel encountering each one in a queue, both with friends and alone. **PREMO** (Product Emotion Measurement Tool) uses illustrated cards to capture emotional responses. Participants **could choose multiple emotions**, allowing for instinctive and unbiased feedback.



Emotion	Rhythm (F)	Rhythm (S)	Kinetic (F)	Kinetic (S)	Dress-up (F)	Dress-up (S)
Excited	5	2	4	2	4	0
Curious	4	1	3	2	3	1
Amazed	2	1	1	0	1	0
Proud	0	0	1	0	0	0
Shy	2	1	0	2	2	0
Offput	3	3	0	2	2	3
Bored	1	2	0	2	1	2
Refusal	0	3	0	0	1	3

(F) = with friends (S) = solo

EVIDENCE: APPENDIX B: [1]

PREMO: QUALITATIVE



"Excited but I'd need friends to push me"

"I'd enjoy seeing other people take part"

"I don't want to play bad in front of people"

"I'm too shy and would find it awkward, I don't want to be a weirdo"



PREMO: CONCLUSION

All three concepts were well-received with friends but became more **intimidating when experienced alone**. This shows that we need to **improve the concepts approachability**. This concludes concept collection, with the chosen concept being Rhythm Rush

USER RESEARCH: STAKEHOLDER

REASONING FOR STAKEHOLDER TEST

To ensure our final concept for Rhythm Rush **met real user needs**, we needed to conduct a role-play session to help us **explore different perspective**, find potential issues, and refine the concept based on their concerns.



SET-UP

Using a set of **four university students** we conducted a **role-play** session to step into the shoes of different stakeholders. Each participant **selected one random stakeholder from a stack of six role cards**, created before the session based on the people affected by Rhythm Rush.

ROLE-PLAY CARDS SELECTED

Stakeholder 2: Casual Partygoer

Name: Sue Spence

Background: You're at a concert or nightclub with friends. You like to have fun but you're not super into dancing or being the center of attention. You usually go along with what the group's doing.

Your Role Today: You're in the queue and you notice people playing Rhythm Rush, a motion-tracked dance battle for up to six players, shown on a large screen in the queue. You're curious but unsure.

Think About: What would make you want to join in (or not)? What would make it feel fun or comfortable? What might make you walk away or avoid it?



Stakeholder 3: Competitive Dancer

Name: Groove Lee

Background: You love rhythm games, freestyle dance, and showing off a bit. You're confident, energetic, and you're always up for a challenge in public spaces, especially if there's music involved.

Your Role Today: You see Rhythm Rush, a motion-tracked dance battle for up to six players, shown on a large screen in the queue and you're ready to dominate it.

Think About: What would make this feel like a real competition for you? What features or rewards would you expect? What would annoy you or make it feel too basic?



Stakeholder 4: Parent Guest

Name: Neil Down

Background: You're attending a family-friendly event with two kids. You're trying to keep them entertained while waiting in line. You want them to have fun but stay safe and feel confident.

Your Role Today: You see Rhythm Rush, a motion-tracked dance battle for up to six players, shown on a large screen, playing in the queue and you're wondering if your kids should try it.

Think About: What would make you feel comfortable letting your kids join in? Would you want to join in or just watch? What might be a concern for you?



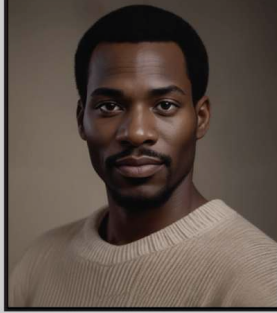
Stakeholder 6: Security Supervisor

Name: Pat Roll

Background: Your job is to make sure the space stays orderly, people move safely, and nothing disrupts the flow or creates risks. You work closely with event organisers, but you're focused on practical logistics and public safety.

Your Role Today: You're evaluating whether Rhythm Rush, a motion-tracked dance battle for up to six players, shown on a large screen in the queue area, is something that helps or hinders your job.

Think About: Does this kind of activity create safety risks or help reduce boredom and tension in the queue? Would it make your job easier or harder? What would you need to see for it to be allowed or supported?



On each stakeholder card It had a brief background on each person as well as some of their role in the session and some **prompting questions** to help make an engaging discussion without any participant being confused.

RESULTS & INSIGHTS

CAPACITY: PLAYER LIMIT

Too many players could cause crowding and increase costs, while too few might **exclude others**. Hence we set the **player limit to 4** to reduce costs while accommodating typical families or small friend groups.

"If too few people are allowed to play, there might be people left out."

MOTIVATION: REWARDS

Solo and group users mentioned that having visible rewards, like **scores or leader boards, would increase motivation and engagement**, making the experience more exciting.

MUSIC: THEME

It was collectively agreed that using **music from the concert line-up will not be used** as it could spoil the live experience for attendees.

INCLUSIVITY: SKILL

Parents want family-friendly music and **simpler moves** to join in with kids, while teens may feel **too self-conscious** to participate. higher-skilled users, however, prefer harder moves for a challenge

"I would want to dance with my kids and like holding hands or whatever and having fun"

SUPPORT SYSTEMS: SOCIAL COMFORT

Group participants said they'd only **join if their friends did**, and suggested that intermittent tutorials or **mirrored moves** for each player could help **ease social awkwardness**.

"If you had a tutorial, like someone to mirror... intermittently freestyle sections"



TRANSCRIPT: APPENDIX C

PACE FIT: QUEUE POSITION

Participants hesitated to play Rhythm Rush due to fear of losing their spot in the queue, creating **tension between fun and progress**. A solution must be made to let users play, watch, or opt out without affecting queue position.

"What if I want to leave the game halfway through, I'm not super into dancing"

PROTOTYPING: TECHNICAL

BODY TRACKING PROGRAM💡

The Rhythm Rush game requires **tracking users' movements** to display them on screen and **calculate a dance score**. We needed proof of concept to ensure body tracking was accurate enough for dancing. This was defined as:

- Having a **high frame rate** of detection
- Having enough precision to detect **large limb movements**, not hand/face movements
- Being able to **detect depth** for better tracking and positioning

CODING PROCESS

We used **Google's MediaPipe** system to implement the body tracking, and **Python** for the coding. A webcam feed gets inputted into a program and it tracks a persons movements by setting **'landmark' points** on the limbs of their body. These points are then connected by lines to give a representation of what the person looks like. These can then be sent to a game engine to **animate our digital characters**. [6]

```
body.py

class PoseWorkerThread(threading.Thread):
    def __init__(self, input_queue, output_queue, pid):
        super().__init__()
        self.input_queue = input_queue
        self.output_queue = output_queue
        self.pid = pid
        self.pose = mp.solutions.pose.Pose(
            min_detection_confidence=0.5,
            min_tracking_confidence=0.5,
            model_complexity=global_vars.MODEL_COMPLEXITY
        )

    def run(self):
        while not global_vars.KILL_THREADS:
            try:
                # Wait for frame half to process
                frame_half = self.input_queue.get(timeout=0.1)
            except queue.Empty:
                continue

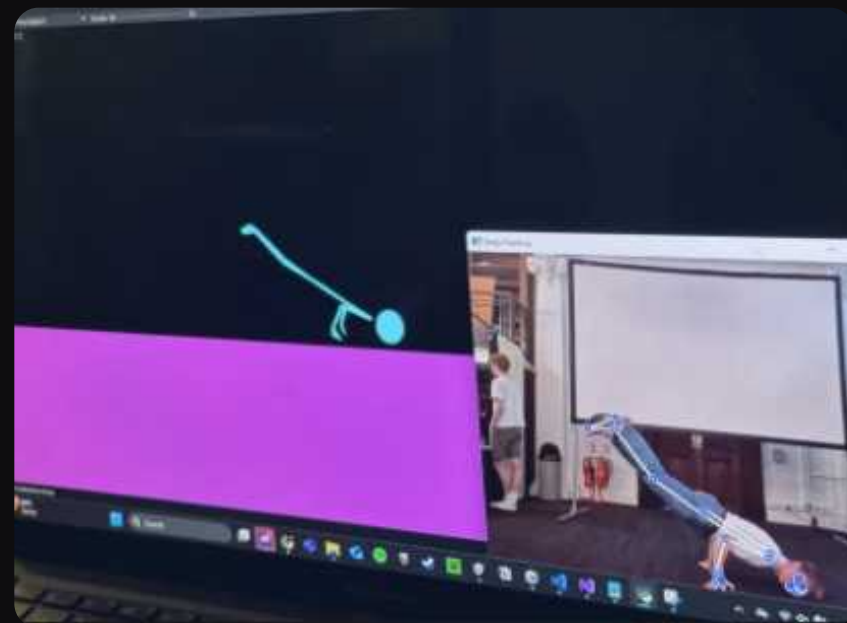
            if frame_half is None:
                break # Shutdown signal

            results = self.pose.process(frame_half)
            self.output_queue.put((self.pid, results))
        ...
```

FULL CODE: APPENDIX D

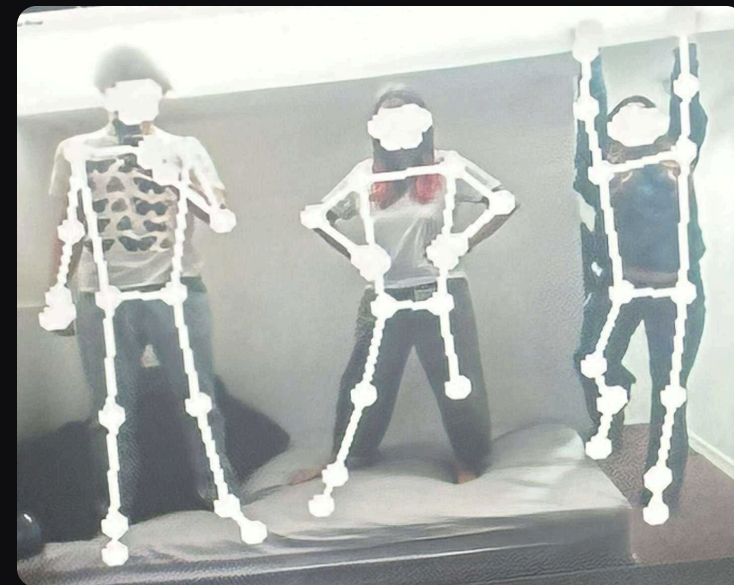
TESTING PROCESS

Each progression in the code was accompanied by a limits test. These tests were conducted with a **small group of volunteers**. While demographic variation was not a key factor, **we included participants of different body sizes** to ensure the system could handle varied proportions.



TESTING: DRAMATIC POSES

We tested more **unique poses, jumps and flips** in the cases that the user would be more dramatic in their dances, as breaking the immersion would ruin the enjoyment. **The program needed to be adapted multiple times** because of these as the person originally had to be grounded all the time.



TESTING: MULTIPLE PEOPLE

This test was conducted with three people in frame. **The people could be detected** using a similar system but accuracy was reduced and the program would **confuse each person with another**, so any scoring would not be user specific.



TESTING: ENJOYMENT

During the testing, even without music, **people were dancing** to see the limitations of the body tracking without any prompting.

"Wait, this is so cool... look it's like actually tracking everything"

EVIDENCE: APPENDIX B: [2]

IMPROVEMENTS

We learned that it is feasible to track human body movement using accessible and relatively lightweight tools like *MediaPipe*. However, **accuracy and performance** during testing are influenced by a few variables:

- Lighting
- Camera resolution
- Model complexity
- The number of individuals in the frame



And **any delays or inaccuracies cause frustration**, break immersion, and reduce engagement.

- To reduce glitches, the program temporarily uses **the last known position if a person is undetected** for a split second so that it looks smoother.
- Tested low, medium, and high model complexities; and **chose medium for best balance** of accuracy and speed.
- Users need to be well lit, so the screen setup should include proper lighting.

SPLIT-SCREEN TRACKING



To fix the issues with multiple people tracking, **the program divides the camera in half** and checks for one person in each half. This means that the **people will need to stand on specific points** in our area to make sure the program does not confused with other people.

DISTANCE FROM CAMERA

The people will need to stand at an appropriate distance from the camera for the best view, and the camera needs to be at a specific height. The measured distance was around **2m away** and the **camera would sit just above 1m height**.

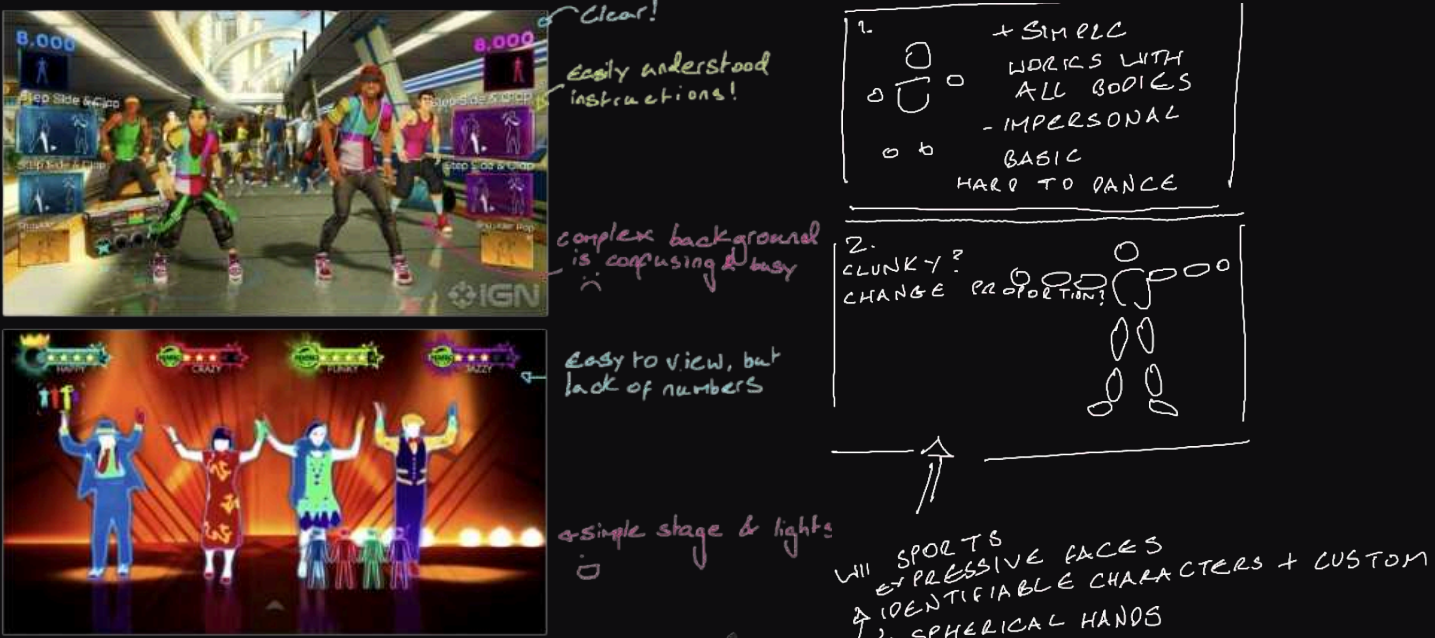
PROTOTYPING: GAME ENVIRONMENT & CHARACTER

REASONING

A player's engagement begins with their first impression of the attraction. **Engaging lights and sound are more likely to attract attention** and curiosity from a crowd, which in turn drives a shared desire to interact through play. Discussing the **aesthetics of the digital elements with stakeholders**, who act as co-designers, allows us to determine the attributes of an appealing attraction.

APPROACH

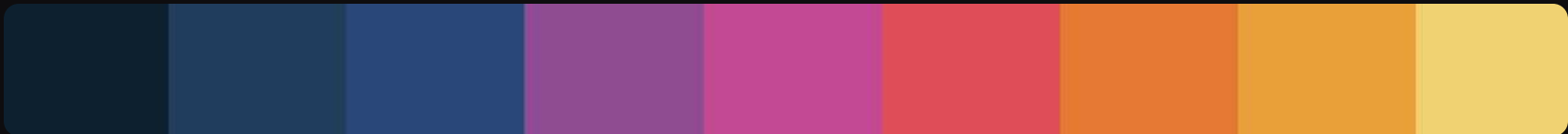
There was an **initial analysis of dance based games** to determine what works well, and why these design choices are made - **these design decisions were factored into our own process**.



Multiple semi-structured and guided testing sessions were held with the stakeholders, showcasing alternate versions of the **base character and the surrounding environment**, this was to find the character proportions, the level of realism and the most preferred colour palettes for the theme.

ENVIRONMENT DESIGN

Stakeholders strongly agreed that **a dark, evening-toned colour palette supports the concert and party atmosphere** of the queue. Bright, colourful neon lights were seen as a strong visual contrast that complements this setting well. In comparison, relying only on spotlights was considered too plain and visually unengaging.



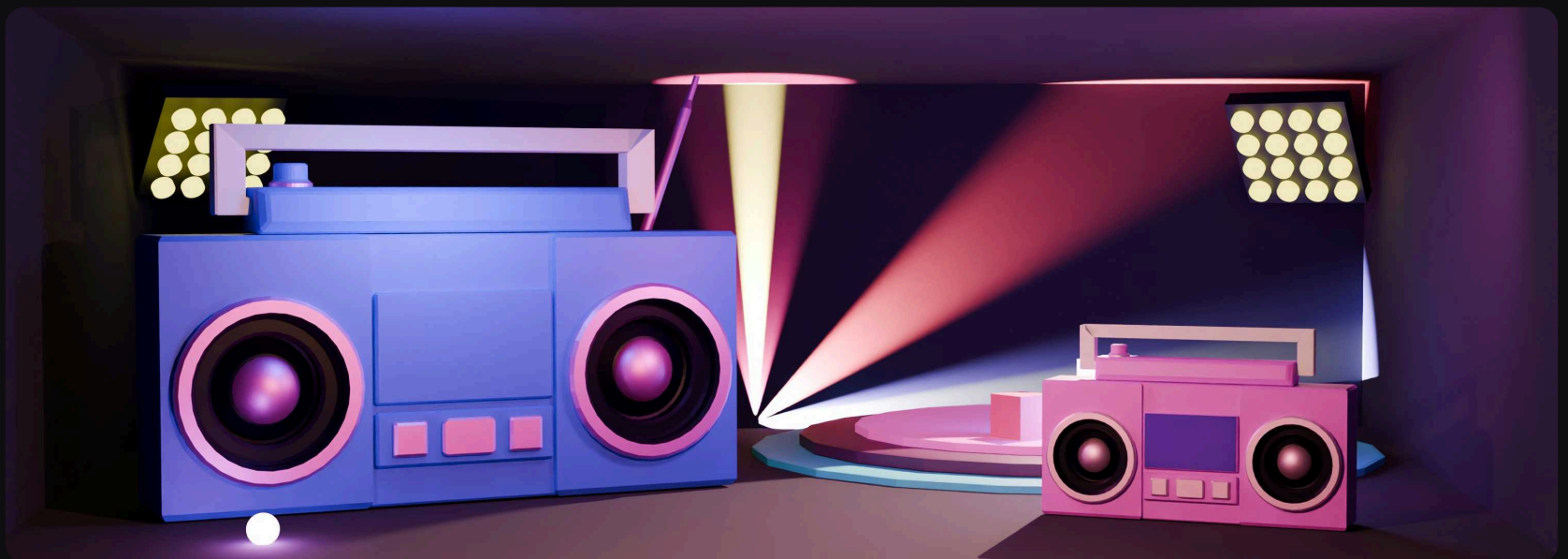
BACKGROUND: LIGHTING

Simple background lights were preferred to complex and colourful ones, **as they may be confusing**.



BACKGROUND: OBJECTS

Only a small number of environmental objects is preferred. Enough to **suggest a setting** but not so many that they distract from the gameplay.



This environment was combined with the character designs to **create a unified visual scene** that felt lively, readable, and thematically consistent with the concert-inspired tone of the experience.

CHARACTER DESIGN

INITIAL EXPERIMENTATION

Simplified limbs provide both visual and practical advantages:

- **Visual:** Uniquely simple style.
- **Practical:** Mitigates the need to track individual fingers and other complex joints.

Unrealistic, childlike proportions give a **sense of cuteness**. Stakeholders enjoyed the unique 'low-poly' style.



ADDING: LIGHT

Light makes the characters look 3D, appearing **more tangible and relatable** to users.

Purple, blue & yellow lighting was found to atmosphere coherent with concerts.



ADDING: CHARACTER

Each character is given a **near-monochrome colour palette** for unique identification from users.

Individual hairstyles were added to give 'character'.



PROTOTYPING: VISUAL GAME ELEMENTS

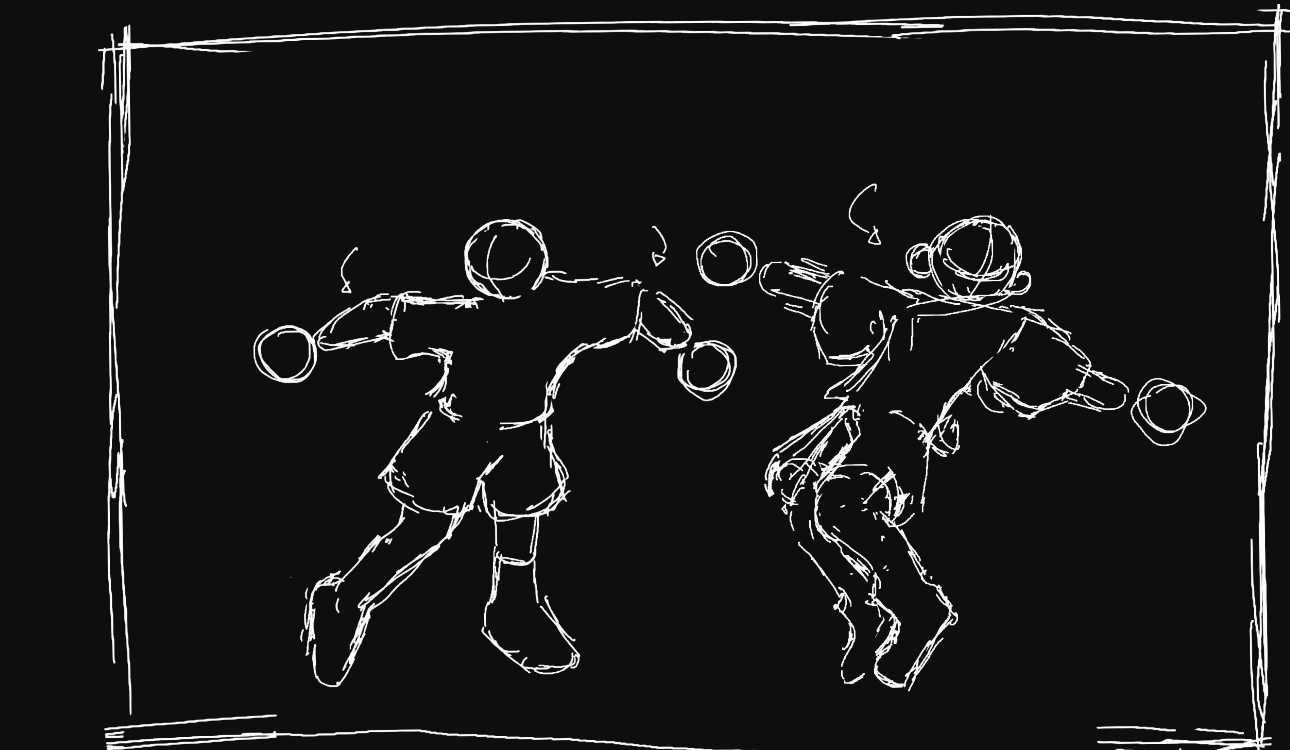
REASONING

The UI needs to be **intuitive for first-time users**, reducing the need for explanation or prior experience.

Feedback from PREMO, Harris profiles, and stakeholder sessions made it clear that **learnability is essential** to get people engaged, **especially in a queue setting**.

CO-DESIGN

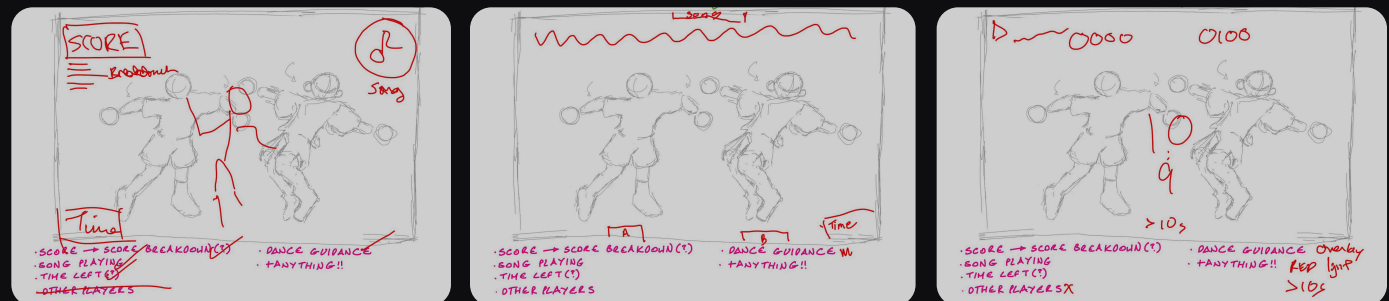
A baseline sketch of the games screen was created, along with a list of UI components **we deemed necessary**.



- SCORE → SCORE BREAKDOWN(?)
- DANCE GUIDANCE
- SONG PLAYING
- +ANYTHING!!
- TIME LEFT(?)
- OTHER PLAYERS

Members of our user group were briefed, given the baseline sketch, and were guided to:

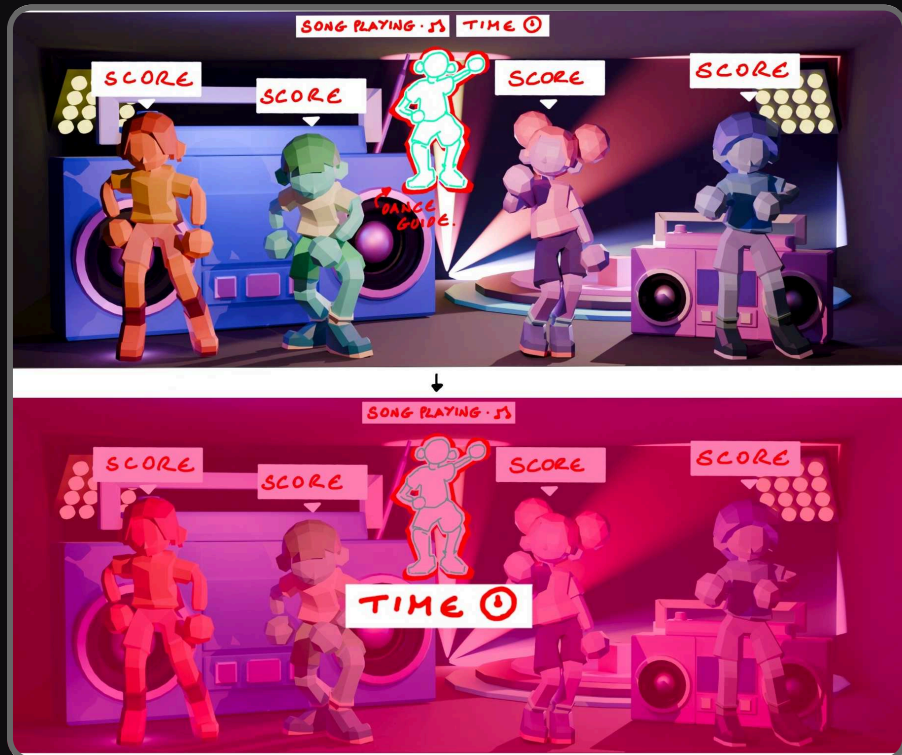
- **Sketch out their interpretation** of UI component placement.
- **Add or remove any components** they thought would be unnecessary or confusing.



(Selected examples of stakeholder sketches) EVIDENCE: APPENDIX A

USER FEEDBACK & LO-FI DESIGNS

We gathered inspiration from initial sketches and **discussed them with users to gain deeper insight**. Using this feedback, we developed **three lo-fi UI concepts** for comparison and testing. These informed the creation of our final UI design.



[Concept A] (3 votes)

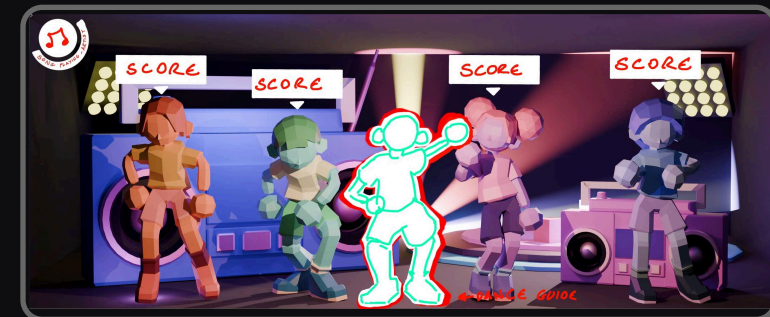
Emphasis on end of song **prompting users to leave** or switch out:

'It's important to see what each person's scores are'

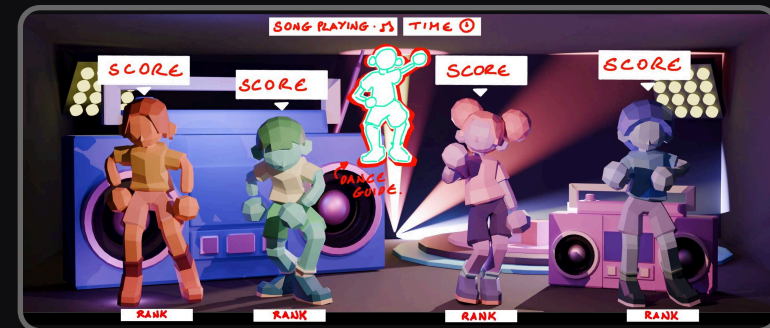
'I like that you can see when time's done'

Mentioned Positives:

- **Easy to view** all elements from any position on-stage.
- Clear **indication of time**.
- Minimalistic UI.



[Concept B] (1 vote)



[Concept C] (1 vote)

Larger dance guide and **decreased emphasis on song**:

'I think ... the dance guide is too big...'

Score is averaged into a **letter rank** to improve legibility:

'Those kind of ranks would be effective if players had a short time and will only play it once'

FINAL DESIGN

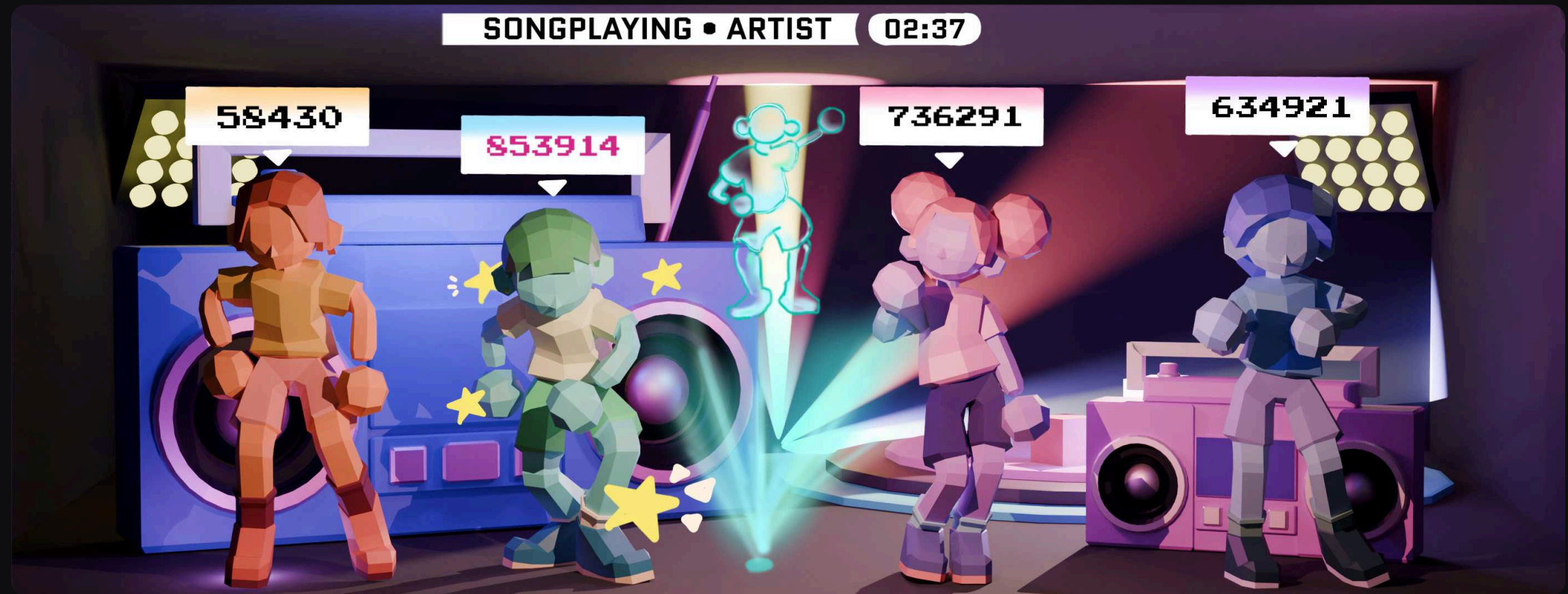
Concept A was the most popular - the final UI is based upon this.

An **idle screen will be used** to make the system feel approachable, and to signal to users that they can step in.



The screen would show the **dance guide moving**, with translucent player avatars raising their hands at calibration points. This would guide users on how to start the game, and **help the program detect them**.

This was based on findings from our *visual vs. text survey* where **animated prompts were preferred over static instructions**.^[1]



Players noted they feel more encouraged by seeing their own personal score, as opposed to a ranking. To maintain some competition, **the winning player gains a unique visual effect and highlighted score**.
:☆☆☆☆

The final design takes inspiration from similar early **2000's dance games**. Pixelated numbers and the stylisation of the **dance guide as a hologram** retain the effect of **similar nostalgic dance games**.

PROTOTYPING: SCREEN

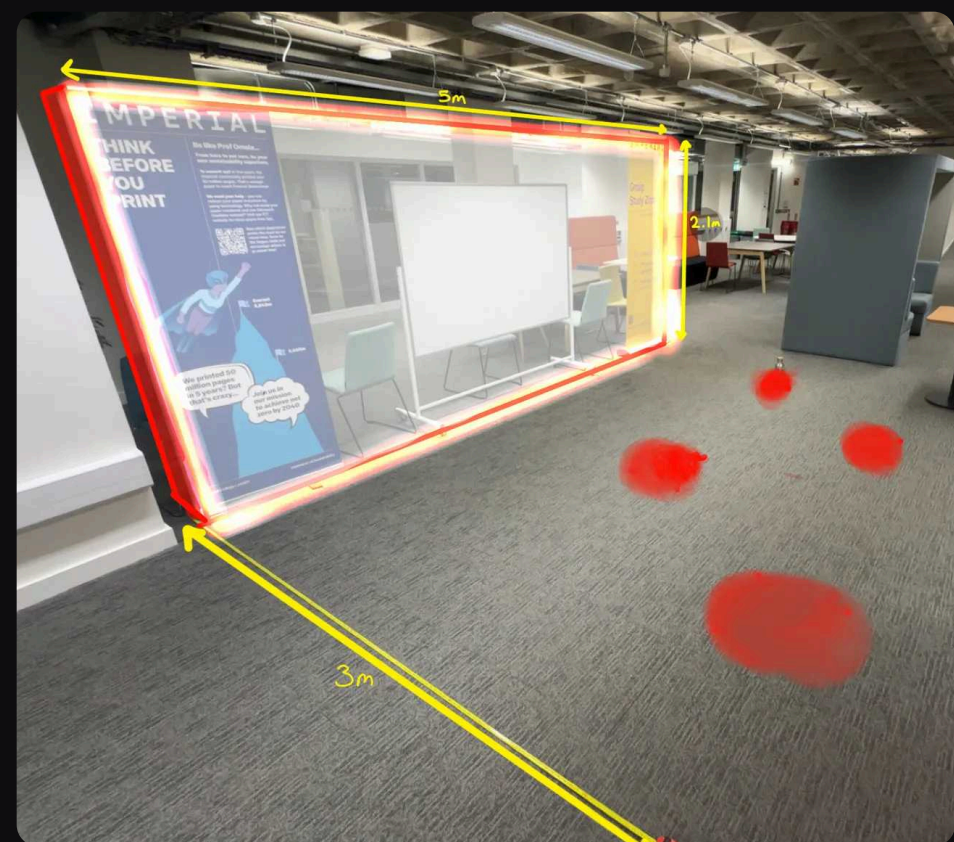
REASONING

PREMO research showed the game **felt intimidating and unclear**, so the setup needed to be more welcoming and visually reflect the game's theme. The technical prototyping highlighted the **need for proper lighting, player distance, a large screen, and speakers**. The final goal was a setup that was inviting, clear, and functionally sound.



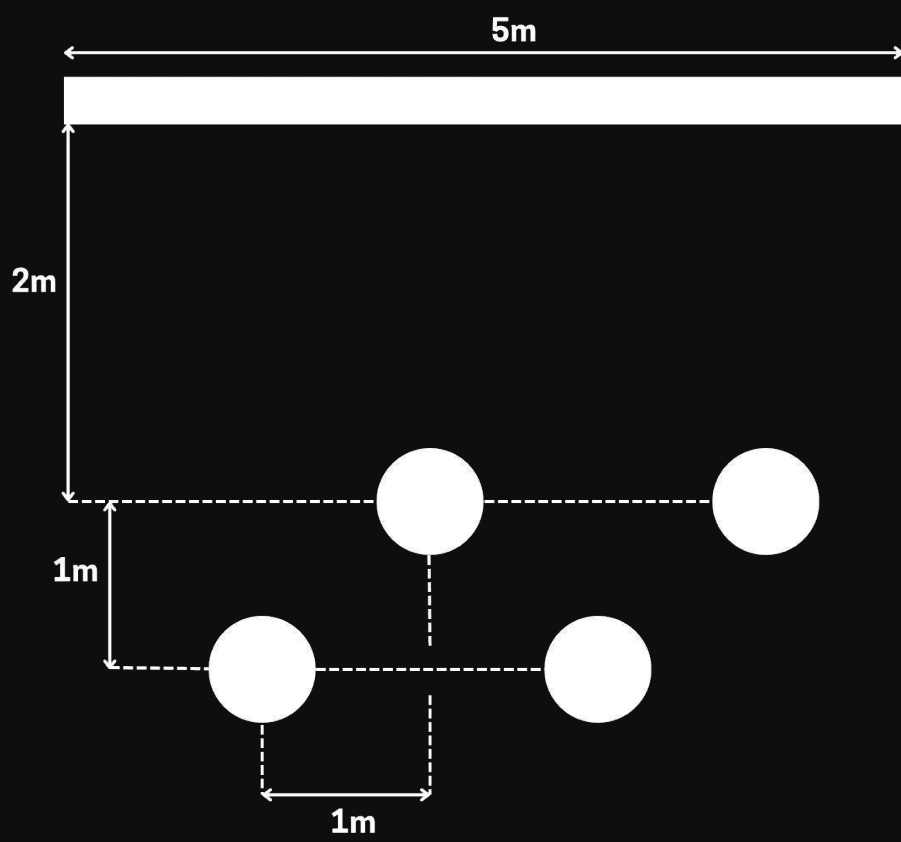
FULL SCALE SIZING

Using **technical tests** and **anthropometric data** (98 cm radius per person), we defined safe personal space. **Players were staggered to reduce screen width** and improve interaction. A full-size mock-up (with banners and whiteboards for the screen and placeholder items for people) helped test layout and UI visibility.



EVIDENCE: APPENDIX B: [3]

- We learned how to **arrange the players** so each person has **enough space to move** without bumping into others during dancing.
- We also made sure the **screen and dance guide were placed so all players could see them clearly**.
- We designed the space so **players could interact with each other without feeling crowded or uncomfortable**.



MODELS

MODEL: CARDBOARD LO-FI

We started with low-fidelity prototyping, to **explore a range of different screens using cardboard mock-ups**. These were used to quickly iterate through different component placements.



MODEL: CAD TESTING

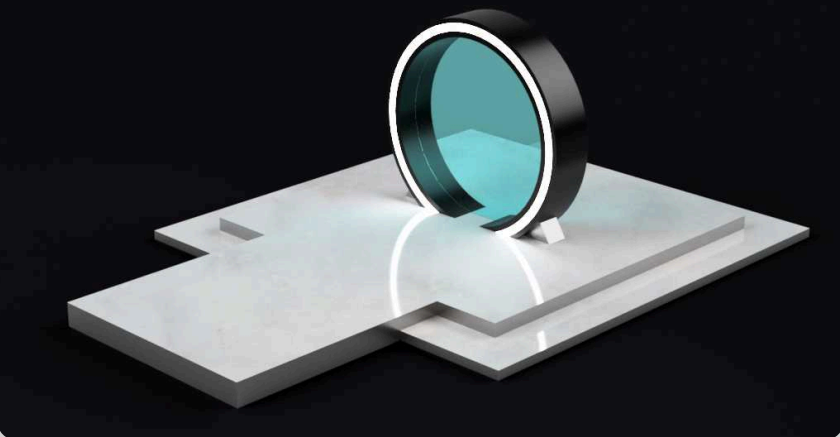


Once we had a clearer sense of which ideas resonated with us as the designers, we then refined the selected prototypes in CAD software to produce a more realistic and professional visual.

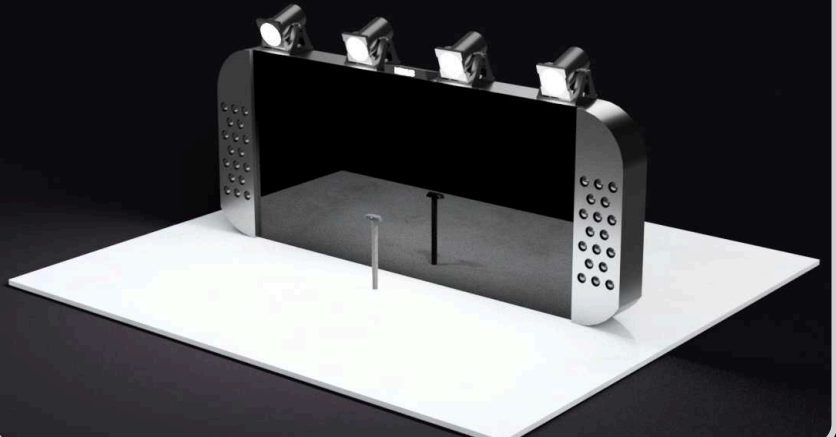
BOX TV



PORTAL



STAGE



These models were then **tested with users** to gather feedback on the **visual impact, usability and themes**.

MODEL: FINAL DESIGN

The final design is a refined combination of the **strongest elements from the three initial concepts**:

- From the 'Stage-Lighting' design, we kept the clear rectangular shape, which made it easy to recognise as a screen.
- From the 'Portal' design, we kept the framed lighting and placed the curved frame into the ground
- From the 'BOX TV' we took the simple style; we did this by placing the speakers under the frame and added an under-display camera.



PROTOTYPING: QUEUE INTERACTION

REASONS FOR TESTING

Both the **Harris profile** and **stakeholder** exercise **highlighted concerns** about the **queue's pacing** and how well our product fits in the scene. This prompted us to simulate a queue scenario to better understand:

- How a flow of a queue works with our design.
- Whether a people feel more comfortable with a dance guide
- If users feel rushed or if they want to skip the experience entirely

"I would hog it because I love dancing. I would want to hold my place in the queue though."

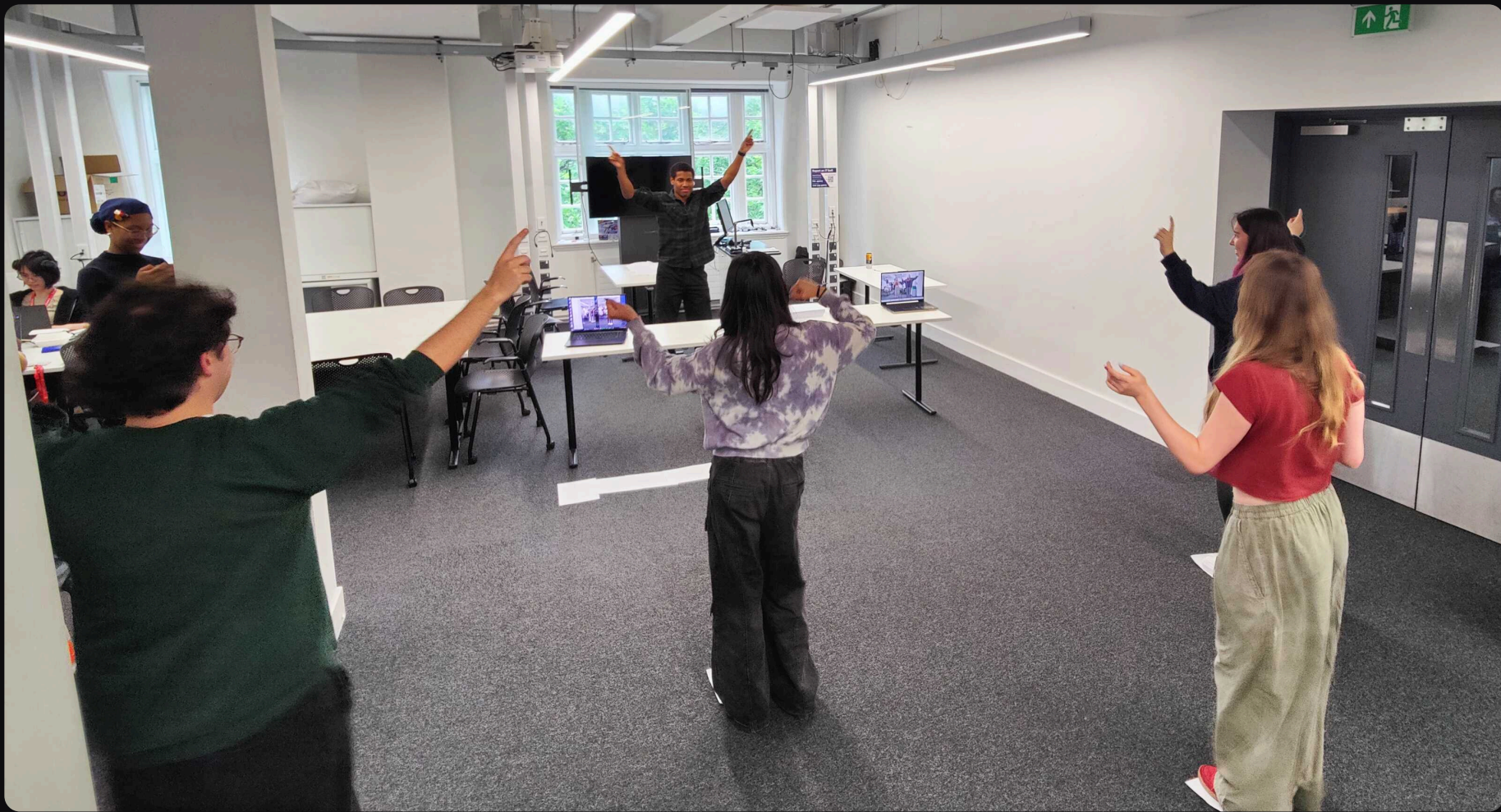
"Would it slow down the queue significantly because what if my kids just want to dance"

"What if I want to leave the game halfway through, I'm not super into dancing"



EXPERIMENT SETUP

We simulated our queue using laptops as stand-ins for the screens, running the program. **Distances between each user and the screen matched the intended spacing of the final setup.** A dance guide and upbeat music created the environment. Various methods were used to produce a realistic delay to the queue for natural flow.



RESULTS

APPROACHABILITY: DANCE GUIDE

We learned that people were initially curious but hesitant to dance. When a dance guide was introduced, they felt **much more comfortable**. Seeing one person dance also encouraged others to join in, like a herd mentality.

PACE FIT: RUSHED

Participants **felt rushed when moved along before finishing their dance**, especially when enjoying the experience. This disrupted group dynamics and caused congestion. If the fourth person left, others had to walk across the space, **leading to tracking issues**.

FEEDBACK: SOCIAL

We found that the experience significantly **boosted people's energy and mood**, making them feel 'hyped up'. However, **being asked to leave while enjoying it dampened that mood**. Participants also **enjoyed chatting** during the experience, so music volume should allow for conversation.

SCORING: ENGAGEMENT

Participants appeared to enjoy the experience more when they could dance freely, using expressive movements. They were **most engaged when moving in time with the music and experimenting with speed and style**.

EVIDENCE: APPENDIX B: [4]

IMPROVEMENTS

Junction System Introduced:

- **Participants can choose** to enter the game, join the spectator space, or bypass the experience and continue queueing.

Spectator Zone Added:

- Provides a **low-pressure way to engage**, with the freedom to cheer or join when ready.

Reduced Congestion:

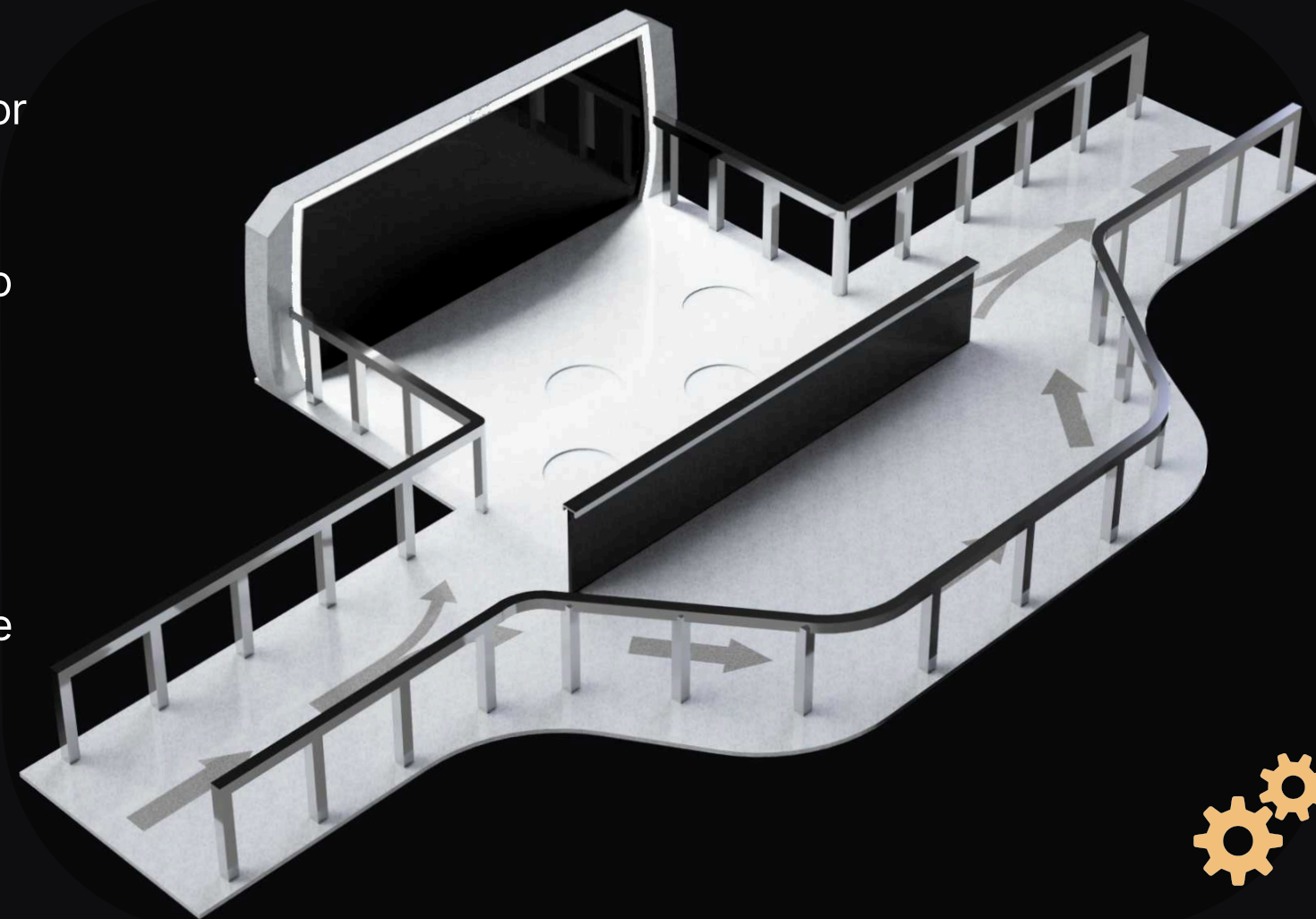
- A wider spectator area helps prevent bottlenecks and **maintain flow**.

Flexible Participation:

- Users can leave at any time, with a **natural exit prompt** at the end of each song.

Scoring System:

- Adapted so that it rewards **creativity, speed, and beat-matching** instead of strict accuracy.



FINAL DESIGN: VALIDATION

ANNOTATED FINAL DESIGN



STORYBOARD



VALIDATION REQUIREMENTS

The experience should feel **approachable**

Participation **should be optional** and flexible

The system and interface should be **easy to understand**

The setup should **let users feel safe** and at ease

Users should feel **encouraged to be creatively**

The experience should **work with the flow of the queue**

Spectators should feel **included and engaged**

EVIDENCE

Re-using PREMO showed that participants found the design more friendly and approachable, helped by the playful avatars, minimal UI, and dance guide.
EVIDENCE: APPENDIX B: [6]

The junction system was designed to offer users the choice to play, spectate, or bypass the experience entirely. While not directly tested, it is expected to support flexible participation.
EVIDENCE: QUEUE INTERACTION

The interface was tested with our user group, who responded positively and described it as clear and intuitive. Users understood the experience without additional instruction, suggesting strong learnability
EVIDENCE: APPENDIX B: [5]

Testing confirmed that staggered player positions reduced physical crowding. Spectators were positioned behind players and faced the same way, reducing feelings of being watched.
EVIDENCE: SCREEN

Stakeholders responded positively to the scoring system, which focused on creativity, beat-matching, and speed rather than technical accuracy, encouraging, non-judgemental play.
EVIDENCE: VISUAL GAME ELEMENTS

Queue simulation suggested that the junction and song-end exit points allow players to enter and leave without disrupting flow. While not fully tested, the setup was designed for pacing.
EVIDENCE: QUEUE INTERACTION

Queue interaction testing showed that non-players enjoyed watching others dance, with some expressing increased interest in joining.
EVIDENCE: QUEUE INTERACTION

FINAL DESIGN: CONCLUSION

Rhythm Rush transforms queue spaces into moments of **playful social interaction**. Its **creative**, low-pressure gameplay encourages self-expression while **allowing users to engage** however they feel comfortable. By drawing on the shared energy of the concert environment, it helps spark connection between people and **builds a sense of community** among attendees before they even enter the venue.

REFERENCES

- ## SUPPLEMENTARY SKETCHES

STAKEHOLDER: SKETCHES

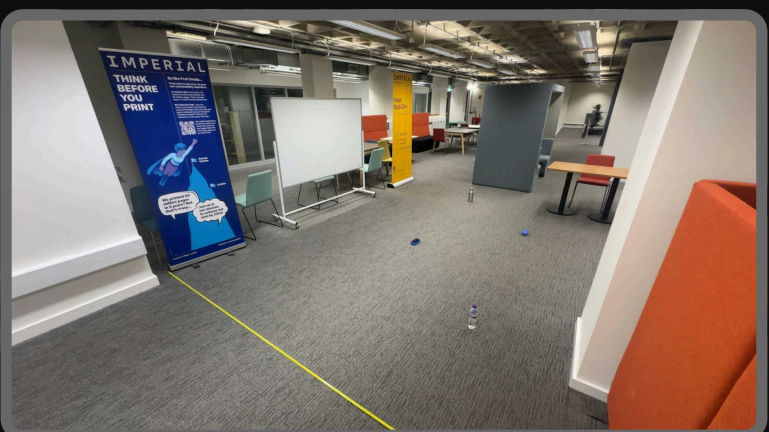


APPENDIX B: RESEARCH EVIDENCE

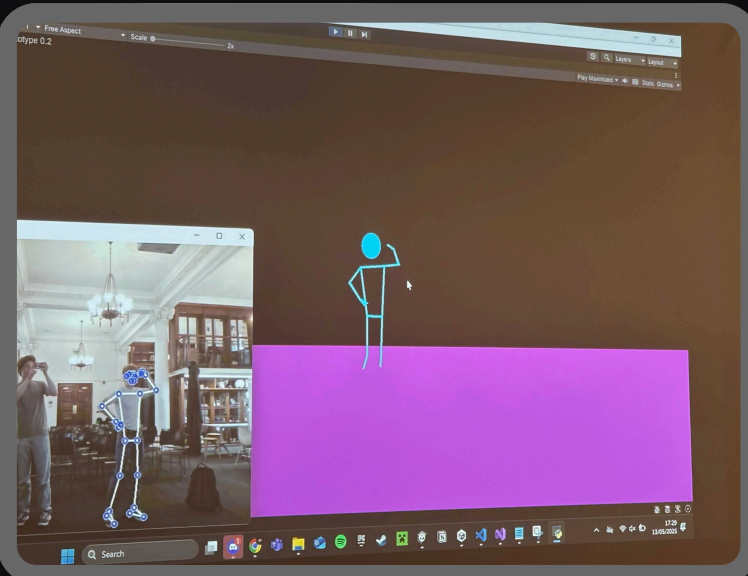
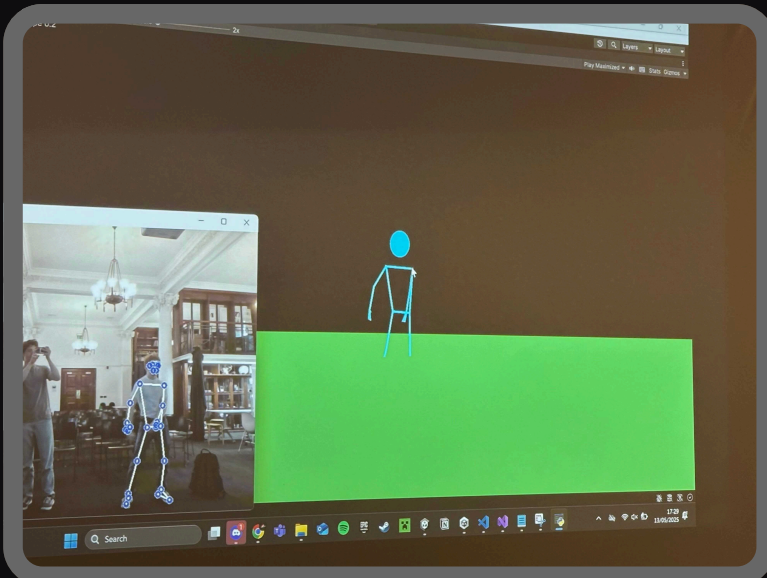
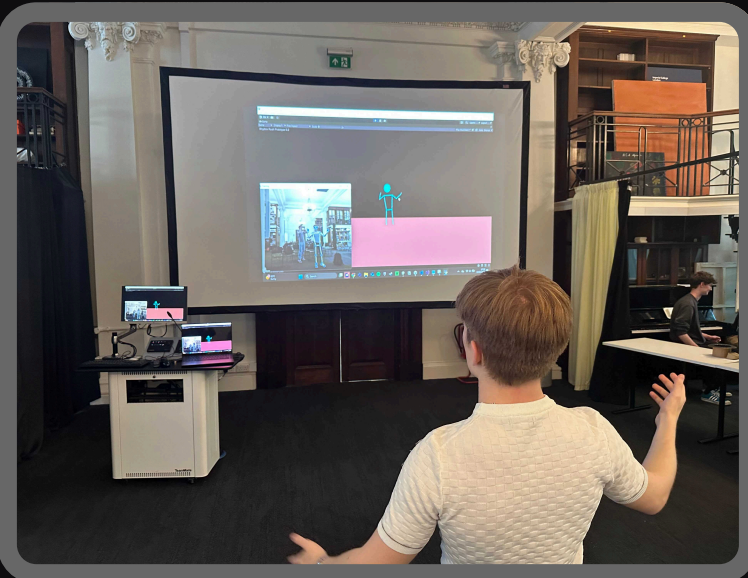
[1] PREMO: ANALYSIS



[3] SCREEN: SIZING



[2] CODE: TESTING



[4] QUEUING : INTERACTION



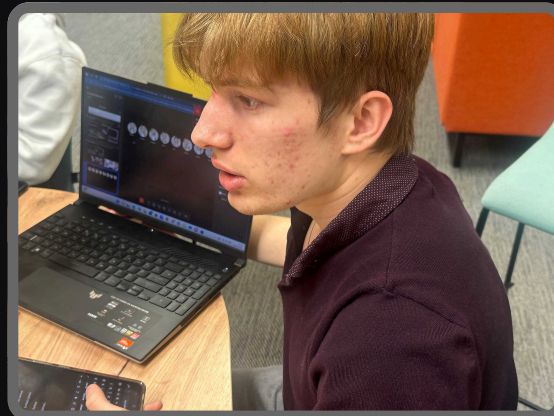
[5] VALIDATION: VISUALS



Notable Quotes:

- "It just made sense...It is self-
- "It feels like a game I've played before, but way
- "Even if you've never done it before, you'd figure it out ... I knew where to look straight
- "I don't usually play games, but I still got what I had to do pretty
- "I feel like if I was dancing, the UI wouldn't get in the way"

[6] VALIDATION: PREMO



With friends

- Excited (5)
- Curious(5)
- Amazed (3)
- proud(1)
- Shy(2)

Without friends

- Excited (3)
- Curious (4)
- Amazed(1)
- Shy(2)
- offput(1)
- bored(1)

APPENDIX C: TRANSCRIPT

STAKEHOLDER ROLE-PLAY

Ikem: "Rhythm Rush is a dance battle game for up to six players."
Noah: "Six is great - I could play with all my friends."
Ikem: "Would six be too many people?"
Will: "I like having a lot of people involved."
Tim: "If there are too few people, then others might be left out."
Joe: "I think six is a good number. As a family man, I'd be fine with my kids doing it because I'd be there. I'd want to dance with my kids and have fun, like holding hands or whatever."
Ikem: "Would you want the dances to be shown on screen?"
Will: "I'd prefer if we could choose from different dances."
Will: "I want a difficult dance."
Ikem: "Is there any way we could make it more fun for you and your friends?"
Noah: "I'm fine with it - but that's because I'm very drunk."
Ikem: "What about spacing and safety in the dance zone?"
Tim: "There needs to be a way to space people out so no one's bumping into each other. There should be an indicator for entering the dance zone. Otherwise, I'd have to stop people from entering. Maybe a gate system - I'd be worried someone could get injured."
Ikem: "What happens if someone wants to stop mid-game?"
Noah: "What if I want to leave the game halfway through? I'm not super into dancing."
Ikem: "Would this slow down the queue?"
Joe: "Would it slow down the queue significantly? What if my kids just want to keep dancing?"
Will: "I'd hog it - I love dancing. But I'd move along if the queue was moving."
Tim: "If the queue is moving quickly, people might get less time to dance, and that could cause conflict."
Joe: "If people could move along with the queue, that would be best."
Tim: "You'd also need a system for how long people are spending on it - otherwise someone might hog it."
Ikem: "Would you personally approach it?"
Noah: "I'd approach it if my friends did -herd mentality."
Will: "I'd approach it."
Joe: "I'd approach it because of my kids."
Tim: "There could be conflict if there are separate queues.""
Ikem: "Who do you think would actually engage with the game?"
Joe: "Teenagers might feel too uncomfortable. Kids under 12 and adults would engage more."
Ikem: "What kind of music would you want?"
Will: "Something with a strong beat."
Joe: "Family-friendly music."
Will: "Something inspired by Just Dance."
Tim: "Nothing that would make people get too boisterous."
Ikem: "Would you want music from the concert line-up or something else?"
Noah: "I'd rather not have music from the concert -it would ruin the live experience."
Ikem: "What kind of visuals would make it appealing?"
Tim: "I want cool visuals."
Ikem: "If you're shy, what would help you feel more comfortable?"
Joe: "I'd want a guide - I don't want to do improv dancing."
Noah: "Hints would help. Just Dance is good. At the start, have someone you can mirror."
Tim: "Or include freestyle sections."
Noah: "Yeah, or a tutorial you can mirror. Then it goes into freestyle."
Tim: "And maybe you could skip the tutorial just for yourself."

PARTICIPANTS

- Joe (Stakeholder 4)
- Noah (Stakeholder 2)
- Will (Stakeholder 3)
- Tim (Stakeholder 6)

Facilitator: Ikem

APPENDIX D: PROGRAM & MEETINGS

BODY.PY

```
import time
import threading
import queue
import cv2
import mediapipe as mp
import global_vars

cv2.setUseOptimized(True)
cv2.setNumThreads(4)

class VideoCaptureThread(threading.Thread):
    def __init__(self):
        super().__init__()
        self.cap = None
        self.frame_lock = threading.Lock()
        self.frame = None
        self.is_active = False

    def run(self):
        self.cap = cv2.VideoCapture(global_vars.WEBCAM_INDEX)
        if global_vars.USE_CUSTOM_CAM_SETTINGS:
            self.cap.set(cv2.CAP_PROP_FPS, global_vars.FPS)

        time.sleep(1)
        fps = self.cap.get(cv2.CAP_PROP_FPS)
        print(f"Camera started at {fps:.1f} FPS")

        while not global_vars.KILL_THREADS:
            ret, frame = self.cap.read()
            if not ret:
                print("Failed to capture frame")
                break

            frame = cv2.flip(frame, 1)

            with self.frame_lock:
                self.frame = frame
                self.is_active = True

        self.cap.release()

class PoseWorkerThread(threading.Thread):
    def __init__(self, input_queue, output_queue, pid):
        super().__init__()
        self.input_queue = input_queue
        self.output_queue = output_queue
        self.pid = pid
        self.pose = mp.solutions.pose.Pose(
            min_detection_confidence=0.5,
            min_tracking_confidence=0.5,
            model_complexity=global_vars.MODEL_COMPLEXITY
        )

    def run(self):
        while not global_vars.KILL_THREADS:
            try:
                # Wait for frame half to process
                frame_half = self.input_queue.get(timeout=0.1)
            except queue.Empty:
                continue

            if frame_half is None:
                break # Shutdown signal

            results = self.pose.process(frame_half)
            self.output_queue.put((self.pid, results))

class DrawingThread(threading.Thread):
    def __init__(self):
        super().__init__()
        self.frames = queue.Queue(maxsize=2)
        self.stop_flag = threading.Event()

    def submit_frames(self, left_frame, right_frame):
        try:
            self.frames.put_nowait((left_frame, right_frame))
        except queue.Full:
            pass

    def run(self):
        while not self.stop_flag.is_set():
            try:
                left_small, right_small = self.frames.get(timeout=0.1)
                cv2.imshow("Left Person", left_small)
                cv2.imshow("Right Person", right_small)
                if cv2.waitKey(1) & 0xFF == ord('q'):
                    global_vars.KILL_THREADS = True
                    break
            except queue.Empty:
                continue

cv2.destroyAllWindows()
```

```
...
class BodyTrackingThread(threading.Thread):
    def __init__(self):
        super().__init__()
        self.last_connection_attempt = 0

    def run(self):
        capture_thread = VideoCaptureThread()
        capture_thread.start()

        left_input_queue = queue.Queue(maxsize=1)
        right_input_queue = queue.Queue(maxsize=1)
        output_queue = queue.Queue(maxsize=2)

        left_worker = PoseWorkerThread(left_input_queue, output_queue, pid=0)
        right_worker = PoseWorkerThread(right_input_queue, output_queue, pid=1)
        left_worker.start()
        right_worker.start()

        if global_vars.DEBUG:
            drawing_thread = DrawingThread()
            drawing_thread.start()

        drawing_utils = mp.solutions.drawing_utils
        mp_pose = mp.solutions.pose

        last_valid_world_landmarks = [None, None]

        while not global_vars.KILL_THREADS:
            with capture_thread.frame_lock:
                frame = capture_thread.frame.copy() if capture_thread.frame is not None else None

                if frame is None:
                    time.sleep(0.01)
                    continue

                height, width, _ = frame.shape
                mid = width // 2
                left_frame = frame[:, :mid]
                right_frame = frame[:, mid:]

                target_width = 640

                scale_left = target_width / left_frame.shape[1]
                target_height_left = int(left_frame.shape[0] * scale_left)
                left_small = cv2.resize(left_frame, (target_width, target_height_left))

                scale_right = target_width / right_frame.shape[1]
                target_height_right = int(right_frame.shape[0] * scale_right)
                right_small = cv2.resize(right_frame, (target_width, target_height_right))

            try:
                if left_input_queue.empty():
                    left_input_queue.put_nowait(left_small)
                if right_input_queue.empty():
                    right_input_queue.put_nowait(right_small)
            except queue.Full:
                pass

            results = [None, None]
            count = 0
            while count < 2:
                try:
                    pid, res = output_queue.get(timeout=0.05)
                    results[pid] = res
                    if res and res.pose_world_landmarks:
                        last_valid_world_landmarks[pid] = res.pose_world_landmarks
                    count += 1
                except queue.Empty:
                    break

            if global_vars.DEBUG:
                if results[0] and results[0].pose_landmarks:
                    drawing_utils.draw_landmarks(left_small, results[0].pose_landmarks, mp_pose.POSE_CONNECTIONS)
                else:
                    cv2.putText(left_small, "", (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 0.6, (0, 0, 255), 2)

                if results[1] and results[1].pose_landmarks:
                    drawing_utils.draw_landmarks(right_small, results[1].pose_landmarks, mp_pose.POSE_CONNECTIONS)
                else:
                    cv2.putText(right_small, "", (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 0.6, (0, 0, 255), 2)

            drawing_thread.submit_frames(left_small, right_small)

        left_input_queue.put(None)
        right_input_queue.put(None)
        left_worker.join()
        right_worker.join()

    if global_vars.DEBUG:
        drawing_thread.stop_flag.set()
        drawing_thread.join()
```

MAIN.PY

```
# Simple pipe server runner script

from body import BodyTrackingThread
import time
import struct
import global_vars
import sys

def main():
    # Start the body thread
    body_thread = BodyTrackingThread()
    body_thread.start()

    # Wait for user input in terminal to trigger shutdown
    i = input()
    print("Shutting down...")

    # Signal other parts of the system to stop
    global_vars.KILL_THREADS = True

    # Allow a little time for threads to clean up
    time.sleep(0.5)

    sys.exit()

if __name__ == "__main__":
    main()
```

MEETING MINUTES

https://imperiallondon-my.sharepoint.com/:w:/g/personal/ie124_ic_ac_uk/EdRI7JRt3-xGi4cJvqk5HJ0BnhRxn_k5tdcnafIREkTq1w?e=ZHYrl4

GLOBAL_VARS.PY

```
#Do not touch
KILL_THREADS = False

#Shows webcam and pipes
DEBUG = True

#To switch cameras, (Default: 0)
WEBCAM_INDEX = 0

#For custom settings.
USE_CUSTOM_CAM_SETTINGS = False
FPS = 60
WIDTH = 320
HEIGHT = 240

MODEL_COMPLEXITY = 1
```