🧠 Objective

Participants must build a system that listens to audio files and identifies time intervals when it's safe to "move" (i.e., when "Green Light" is said). Using pre-trained audio models and basic Python, they simulate a player moving at a fixed rate through increasingly noisy and deceptive audio environments.

🛠️ Tools & Requirements:

- **Pretrained models** (recommended: Hugging Face Wav2Vec2, Whisper, etc.)
- **Basic Python** (data parsing, audio processing)
- **Audio processing** (librosa, torchaudio, scipy, or built-in tools)
- **CSV writing**

🗂️ Files Provided:

- Audio clip for each level
- Starter code for loading audio
- Sample outputs and formats (JSON/CSV)
- Distance and movement rate

🧪 Evaluation Criteria:

Each level's submission is run through an evaluation script:

- Simulates player movement based on predicted "Green Light" time intervals
- Penalizes moving during "Red Light" intervals (Basically checking if they get shot)
- Checks whether the player completes the distance in time

🔁 The Fun Stuff (Level Progression & Tips):

🟢 Level 1 (100 pts)

Audio: 15 min, one speaker, clear commands
Noise: No background noise whatsoever
Output: CSV (start of green light, start of red light) for "Green Light" periods in seconds!
Goal: Identify “Green Light” and “Red Light” start times

Tips for participants:

- Use speech recognition with forced alignment or timestamp detection
- Use a threshold confidence or match for “Green Light”/“Red Light” Try to get the start of the green light audio and start of the red light audio for most accuracy

🟢 Level 2 (200 pts)

Audio: 30 min, one speaker
Noise: Nonsense gibberish in the background inserted
Output: CSV (start of green light, start of red light) for "Green Light" periods in seconds!
Rate: 1.5 m/s
Distance: 1000m

Tips:

- Classifier must ignore nonsense phrases
- Can use keyword spotting or fuzzy string matching

🔴 Level 3 (300 pts)

Audio: 30 min
Noise: Screaming + Gunshots in the background
Output: CSV (start of green light, start of red light) for "Green Light" periods in seconds!
Rate: 1.5 m/s
Distance: 1200m

Tips:

- Denoising filters or attention-based models can help

- Look into pretrained models trained on noisy datasets
- Might need to pre process to get the noise out

🔴 Level 4 (400 pts)

Audio: 30 min
Noise: Screaming + multilingual commands + gunshots in the background
Output: CSV (start of green light, start of red light) for "Green Light" periods
Rate: 1.5 m/s
Distance: 1500m

Tips:

- Model must detect multilingual commands

- Whisper or XLS-R models from Hugging Face support multiple languages

🔴 Level 5 (500 pts)

Audio: 30 min
Noise: Screaming + impersonators saying red light green light + gunshots in the background
Output:CSV (start of green light, start of red light) for "Green Light" periods
Rate: 1.5 m/s
Distance: 1200m

Tips:

- Use speaker diarization or speaker recognition

- Participants must identify **the correct speaker**(the first voice that is used is the correct speaker)