# Project VI Additional Feature

## Section 1: General Concept of "Game Mode" Feature

### Overview

The "Game Mode" feature introduces an engaging and interactive game element to the elevator project, enhancing the user experience with an entertaining twist. This feature is designed to provide users with a fun and immersive experience while maintaining the practicality and functionality of the elevator control system.

### Practical and Fun Aspects

Adding a "Game Mode" is an excellent fit for this project for several reasons:

1. **Engagement**: It transforms a basic function into an entertaining activity, encouraging users to interact more with our system.
2. **Innovation**: It showcases creativity and technological innovation by integrating game mechanics with an elevator control system.
3. **Practical Application**: It demonstrates the versatility of our elevator system and its capability to support additional features beyond basic functionality.

## Section 2: User Experience

### Game Trigger and Initiation

The game begins when the user clicks on the website's hidden game icon (easter egg). This action triggers the start of the game audio, creating an immediate sense of immersion.

### User Interaction

Once the game is initiated, a new user interface (UI) appears, allowing the user to start interacting with the game. The UI is designed to be intuitive and visually appealing, guiding the user through the game seamlessly.

### Game Phases

1. **Start**: The user discovers and clicks the hidden game icon, triggering the game.
2. **Gameplay**: The user guesses which floor the elevator will stop at. The elevator randomly generates a floor number from 1 to 3 and moves to it.
3. **Guess Outcome**: Upon reaching the floor:
    - **Correct Guess**: The elevator speaker plays a celebratory audio trigger.
    - **Incorrect Guess**: The elevator speaker plays an incorrect guess audio trigger.

4. **Lives and Score**: The user has 3 lives. Incorrect guesses reduce lives, while correct guesses are tracked and displayed.
5. **Endgame**: When out of lives, the user can enter their score into a leaderboard if it is among the top 10 scores.
6. **Post-Game Options**: The user can choose to exit or replay the game.

### Audio Triggers

- **Main Game Audio**: Arcade-themed songs play on the user's PC during gameplay.
- **Correct Guess Audio**: Specific audio clips play from the elevator speaker upon correct guesses.
- **Incorrect Guess Audio**: Unique audio clips play from the elevator speaker upon incorrect guesses.
- **User Guess Audio**: Immediate audio feedback plays from the elevator speaker as soon as the user enters a guess.

## Section 3: Technical Details

### Game Mechanics

- **Random Floor Generation**: The system uses a random number generator to select a floor number from 1 to 3 each time the user makes a guess.
- **Lives and Score Tracking**: The system keeps track of the user's remaining lives and the number of correct guesses.
- **Leaderboard Integration**: Upon game completion, the user's score is checked against the top 10 scores, and 3 characters can be entered into a leaderboard if applicable.

### Audio Management

- **Game Start Audio**: "66MHz" - Waveshaper
- **Main Game Playlist**:
  - "Arcade Summer" - FM84
  - "The Algorithm" - Droid Bishop
  - "Da funk" - Daft Punk (loops if needed)
- **Correct Guess Audio**:
  - Random: "Lotto 649 - Winner, Gagnon!", "Crowd Clapping Audio", "Mario - yahoo!"
  - Specific: "He's heating up!" (3 correct in a row), "The best" - Pokemon theme (5 correct in a row), "We are the champions" - Queen (7 correct in a row)
- **Incorrect Guess Audio**:
  - 1st: "Price Is Right Losing Horn"
  - 2nd: "Pacman death noise"
  - 3rd: "Super Mario lose a life (game over)"
- **User Guess Audio**: "Wheel of Fortune - Puzzle Reveal"

### Code Implementation

- **Random Number Generation**: Implemented using JavaScript's Math.random() function.
- **Audio Playback**: Managed through HTML5 audio elements and JavaScript event listeners.
- **UI and Game Logic**: Developed using a combination of HTML, CSS, and JavaScript, ensuring responsive design and seamless user experience.

## Section 4: Team Responsibilities

### Isai Torres Garcia - Web Developer and User Interface Specialist

- **UI Design**: Responsible for designing and developing the user interface for the game mode.
- **Frontend Development**: Implementing the interactive elements and ensuring a smooth user experience.
- **Audio Integration**: Managing the integration and playback of audio files on the user's PC and elevator speaker.

### Brandon Hauck - Implementation Specialist

- **Backend Development**: Developing the logic for random floor generation and lives/score tracking.
- **Elevator Integration**: Ensuring the elevator control system interacts correctly with the game mode.
- **Leaderboard Functionality**: Implementing the leaderboard system and managing score submissions and display.

## Section 5: Future Deliverables and Developments (if time allows)

### Testing and Quality Assurance

- **Test Plan**: Outline the testing strategy to ensure the game mode operates correctly without bugs.
- **User Testing**: Conduct user testing sessions to gather feedback and make necessary adjustments.

### Future Enhancements

- **Additional Features**: Potential future updates such as new game modes, additional audio triggers, or enhanced graphics.
- **User Feedback Integration**: Plan for integrating user feedback to improve and expand the game mode.