

Enemy Escape – README

1. Introduction

Enemy Escape is an interactive action-logic game where the player must avoid dynamic enemies while progressing through levels. It uses Prolog-based decision logic, AI-driven behaviors, and a responsive UI.

2. System Overview

Backend

- Serves static game files.
- APIs:
 - /get-speed → enemy speed per level.
 - /can-level-up → checks score for level-up.
- Loads Prolog rules.
- Lightweight and scalable.

Frontend

- Full game logic in HTML, CSS, JS.
- Enemy AI, movement, player controls.
- Timers, levels, achievements.
- Local browser storage.
- Responsive SPA design.

3. Cross-Device Compatibility

Optimized for mobile, tablets, desktops with adaptive UI, smooth animations, and intuitive controls.

4. Gameplay Flow

1. Start game.
2. Select level.
3. Prolog sets enemy behavior.
4. Player avoids enemies.
5. Game ends if player is hit, time ends, or player escapes.

5. Enemy Logic (Prolog)

- Level 1: slow enemies

- Level 2: faster enemies

- Level 3: bird enemies

- Level 4: UFO enemies

- Level 5: dragon boss

6. AI Decision Theory

- Moves toward player

- Predicts movement

- Adjusts speed per level

7. Timer System

- Level 1 → 30s

- Level 2 → 25s

- Level 3 → 20s

- Level 4 → 15s

- Level 5 → 10s

8. Scoring System

- Survival

- Quick escape

- No damage

- Enemy defeat (battle mode)

9. Word Journal

Tracks unlocked enemy types, behaviors, speed levels, and future weaknesses.

10. Offline Features

- Fully offline gameplay

- Local Prolog engine

- Local achievements storage

11. Data Persistence

- Highest unlocked level
- Best time
- Enemy journal
- Game mode

12. Future Enhancements

- Multiplayer
- Online Leaderboard
- Themes
- Smarter AI
- Boss battles
- Cloud save

Conclusion

Enemy Escape merges Prolog logic with modern frontend tech, delivering a smooth, intelligent, and responsive gaming experience across all devices.