

# Visual Recognition Game for Autistic Children

## Project Design & Technical Documentation

---

### 1. Project Overview

This project is a **visual recognition and cognitive training game** designed for autistic children. The game focuses on improving: - Visual recognition - Categorization skills - Shape and object differentiation - Attention and response accuracy

The game progressively increases difficulty by: - Introducing more objects per round - Using visually similar objects (apple vs ball) - Reducing display time - Introducing distractor items

The game will be built in **Unity** and deployed using **WebGL** so it can run inside a web page.

---

### 2. Gameplay Flow

#### 2.1 Core Gameplay Loop

1. Show the target object (example: Ball).
2. After a short delay, show multiple objects on screen.
3. User selects the correct object.
4. Provide positive feedback (sound, animation, icon).
5. Difficulty increases over time.

#### 2.2 Difficulty Progression

- **Level 1:** 2 choices, different categories (ball vs car).
  - **Level 2:** 3-4 choices, simple differences.
  - **Level 3:** Similar objects introduced (apple vs ball).
  - **Level 4:** 5-8 choices.
  - **Level 5:** Similar objects + randomized orientation.
  - **Level 6:** Timed rounds.
- 

### 3. Core Features

#### 3.1 Visual Stimulus System

- Displays object images.
- Supports PNG/JPG assets.
- Animations for showing/hiding items.

### **3.2 Target Object Display**

- Shows the object to be identified.
- Optional: voice-over that names the object.

### **3.3 Object Selection Grid**

- Dynamically generates a grid of objects depending on difficulty.
- Random placement of correct answer.
- Supports 2-12 items on screen.

### **3.4 User Input / Interaction**

- Click/tap selection.
- Highlight on hover (desktop).
- Touch-friendly hitboxes.

### **3.5 Feedback System**

- Positive reinforcement animations.
  - Sound cues (optional mute).
  - Non-negative failure response.
- 

## **4. Accessibility Features (Important for Autism)**

- Minimal distraction UI.
  - Simple colors and high-contrast backgrounds.
  - No sudden flashes or harsh animations.
  - Slow-paced transitions.
  - Optional sound and voice-over.
  - Customizable difficulty.
  - Optional parental controls.
- 

## **5. Tools & Technologies Required**

### **5.1 Unity Engine Components**

- **Canvas UI System** (buttons, images, layout groups)
- **Animator** (feedback animations)
- **ScriptableObjects** (store object data)
- **EventSystem** (mouse/touch input)
- **WebGL Build Tools**

## 5.2 Scripting Tools

- C# scripts for:
- Game manager
- Difficulty controller
- UI controller
- Object loader
- User progression tracker

## 5.3 Audio Tools

- Simple positive tone sounds
- Optional narration

## 5.4 Graphics Tools (Optional)

- Photoshop/Figma for creating shape images.
- Free asset packs (if needed).

---

# 6. Unity Project Structure

```
Assets/
    └── Scripts/
        ├── GameManager.cs
        ├── DifficultyController.cs
        ├── ObjectItem.cs
        ├── UIManager.cs
        └── FeedbackController.cs

    └── Images/
        ├── TargetObjects/
        └── DistractorObjects/

    └── Audio/
        ├── PositiveFeedback/
        └── VoiceOvers/

    └── Prefabs/
        ├── ObjectButton.prefab
        └── FeedbackEffect.prefab

    └── Scenes/
        └── MainMenu.unity
```

```
|   |   └── Game.unity  
|   └── Results.unity
```

---

## 7. WebGL Build Requirements

- Enable WebGL in Unity Hub.
  - Lightweight UI (avoid heavy particle effects).
  - Texture compression enabled.
  - Keep memory usage low (WebGL is limited).
  - Full-screen browser support.
  - Mobile browser compatibility.
- 

## 8. Data & Progress Tracking

### Optional but useful:

- Track correct/incorrect answers.
  - Store progress in browser local storage.
  - Allow session continuation.
  - Parental dashboard overview.
- 

## 9. Future Expansion Ideas

- Add levels for counting, sorting, or matching.
  - Add customizable difficulty settings.
  - Add multiple categories (animals, fruits, vehicles).
  - Add reward system (stars, badges).
  - Add voice recognition for children who want to verbalize the answer.
  - Multiplayer/teacher-assisted mode.
- 

## 10. Conclusion

This document outlines the required tools, features, and structure needed for a Unity-based WebGL visual recognition game tailored for autistic children. Its focus on accessibility, progressive difficulty, and minimal distractions ensures a calming and educational experience.