Interactive Educational Puzzle Platform

Requirements Document

Gal Azoulay 323859967, Or Reshef S 324064849

Stakeholders

Children (Primary Users): Young learners who engage with the platform to improve their cognitive abilities, visual recognition skills, and problem-solving capabilities through interactive puzzle-solving activities.

Adults (Supervisors): Parents, teachers or therapists who monitor children's progress and guide the learning experience.

Functional Requirements

This section details the specific behaviors and capabilities of the system, organized by the user interface screens and core functionalities. Each requirement is assigned a priority level from 1 (Critical) to 4 (Low/Optional) to guide implementation planning within time constraints.

A. User Authentication & Management

1. **User Registration (Priority 2: High)**
   Users can create an account by providing their name, email address, username, secure password and selecting an avatar for their profile picture via the "Sign Up" button. Essential for personalized experience but guest mode provides alternative access.

2. **User Authentication (Priority 2: High)**
   Registered users can log in to their personalized accounts using their credentials (email/username and password). The system authenticates credentials and grants access to saved history, progress tracking and customized settings. Core security requirement for accessing personalized features.

3. **Guest Mode Access (Priority 2: High)**
   Users who wish to try the platform without creating an account can select "Play as Guest" for immediate access to the game. Guest sessions allow full gameplay functionality but do not save long-term progress or history. Important for lowering barriers to entry and allowing trial without commitment.

4. **User Authorization (Priority 2: High)**
   Users are authorized to access all core gameplay features, including all difficulty levels, navigation between pages (Home page, Game page, Progress page), avatar customization and complete access to the game interface.

B. Game Setup & Configuration

5. **Image Selection via Custom Search (Priority 1: Critical)**
   Users can choose specific puzzle topics (e.g., "Animals", "Space", "Cars") using the search bar. The system connects to the Unsplash API to fetch high-quality, child-appropriate images matching the entered keyword, ensuring personalized and relevant content. Core functionality enabling personalized, relevant educational content.

6. **Random Image Discovery (Priority 1: Critical)**
Users can select the "Surprise Me" button to receive a randomly selected image from a curated collection of safe, engaging categories (such as Candy, Nature, or Toys), adding variety and spontaneity to the learning experience. Enhances variety and engagement.

7. **Scalable Difficulty Selection (Priority 2: High)**
Users can adjust puzzle complexity by selecting grid size, ranging from a simple 2-piece split for beginners up to a detailed 32-piece grid for advanced players. Additionally, users can choose the number of missing pieces to progressively increase spatial reasoning challenges. Important for progressive learning but system could function with fixed difficulty initially.

8. **Game Initialization (Priority 1: Critical)**
Once the topic and difficulty are configured, clicking the "Start Game" button initiates the session. The system downloads the selected image, algorithmically processes it into puzzle fragments and masks specific regions with black squares (by using an algorithm we will write), generates decoy pieces, and transitions to the active puzzle interface.
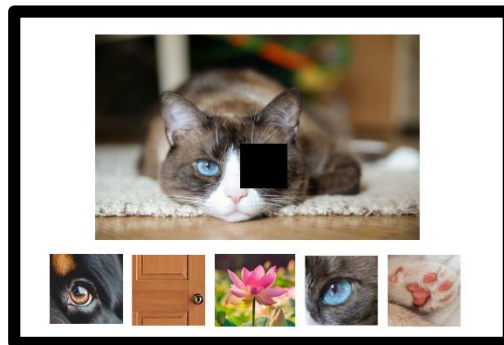
C. Puzzle Gameplay Interface

9. **Puzzle Interaction (Priority 1: Critical)**
The interface displays an incomplete image on the main canvas with one or more regions replaced by black squares. Below the canvas, a "Piece Tray" contains the correct missing fragment(s) alongside several algorithmically-generated decoy pieces (using an algorithm we will write). Users interact by dragging and dropping pieces from the tray into empty slots on the canvas. Core user interface component - the primary interaction mechanism.

10. **Visual Validation (Priority 1: Critical)**
When the user clicks the "Check My Answer" button, the system captures the visual data of the combined image and sends it to the backend for comprehensive analysis. The validation employs a dual-layer approach:

    - **Computer Vision Layer:** Analyses edge continuity, color compatibility, and texture consistency at piece boundaries.
    - **Deep Learning Layer:** Uses deep learning libraries and feature extraction to evaluate semantic context, ensuring the piece makes logical sense within the image (e.g., a cat's eye belongs on the face, not the tail).

11. **Immediate Sensory Feedback (Priority 1: Critical)**
Based on the validation analysis, the system provides immediate visual feedback. A successful match triggers a celebratory animation (confetti, stars). An incorrect placement triggers a gentle "mistake" animation with encouraging prompts like "Try Again!" to maintain motivation and resilience.

12. **Hint System (Priority 3: Medium)**
    Users can request hints by clicking the "Hint" button. The system analyses the overall color distribution of the visible image by computing the average color values across all pixels. Based on this color analysis, the system provides intelligent guidance by recommending the dominant color that should appear in the missing piece (e.g., "The missing piece is probably blue" for an ocean scene, or "Look for green pieces" for a forest image). This color-based hint helps guide the user's selection without directly revealing which specific piece is correct. Helpful learning support feature but not required for core functionality.

D. User Profile & Progress Tracking

13. **Recent Activity History (Priority 3: Medium)**
    The user profile displays a scrollable list of the last 10 (or more) puzzles completed, showing thumbnails of images and difficulty levels. This allows users to track their progress and review favorite topics.

14. **Avatar Customization (Priority 4: Low)**
    Users can change their profile avatar by selecting from a collection of predefined images, personalizing their account appearance. Enhances personalization and engagement but purely cosmetic.

15. **Password Management (Priority 4: Low)**
    Users can securely change their account password through the profile settings.

16. **Account Logout (Priority 3: Medium)**
    Users can securely log out of their account through a clearly marked "Logout" button, terminating the current session and protecting account security.

17. **User Account Deletion Button (Priority 4: Low)**
    Users can request complete account deletion through their profile settings. Upon deletion request, all associated user data (including account information, game history, and progress records) is permanently removed from the database, ensuring user privacy and data protection compliance.

Non-Functional Requirements

This section defines the quality attributes, performance standards, and operational constraints of the system.

Performance & Responsiveness

1. **Image Fetching Speed**
   New puzzle images fetched from the Unsplash API must load and get to the server within 5 seconds max.

2. **Puzzle Image Generation Processing Time**
   The puzzle generation algorithm (image grid division, piece extraction, and pixel masking) must complete within 3 seconds after the server got the image from the API. This ensures immediate transition from setup to gameplay without noticeable delay.

3. **Hint System Response Time**
   When the user requests a hint by clicking the "Hint" button, the system must analyze boundary colors and generate contextual guidance within 3 seconds. This rapid response ensures the hint feature remains helpful without disrupting gameplay flow or causing frustration during challenging puzzles.

4. **Validation Response Time**
The Computer Vision validation process (triggered by "Check My Answer") must return results within 15 seconds to maintain child engagement. This requires optimized algorithm implementation and efficient server-side processing.

API Constraints & Resource Management

5. **Unsplash API Rate Limit Compliance**
The Unsplash Free Tier allows 50 requests per hour. For this project this limitation is ok, but if we want to comply with this limitation, the system will implement a user-based rate limit of 50 puzzle requests per hour per user. The specific implementation approach will be finalized during the detailed design phase.

6. **Image Processing Size Limit**
Images processed by the Computer Vision algorithms are automatically resized to a maximum width of 1024 pixels to prevent server overload from processing high-resolution 4K images, ensuring stable system performance.

Data Management & Storage

7. **User Progress History Storage**
The system stores the most recent 10 (or more) completed puzzles per user, including difficulty level, completion time, and success/failure status. This limited history provides meaningful progress tracking while maintaining efficient database usage.

8. **Guest User Data Management**
Guest user sessions do not persist any data beyond the current browser session. When a guest user closes the browser or navigates away, all gameplay history and progress are cleared, encouraging account creation for users who wish to track long-term progress.

9. **External Image Reference Policy**
The system stores only image URLs from Unsplash rather than downloading and storing actual image files. This approach minimizes database storage requirements and ensures compliance with Unsplash's terms of service regarding image hosting.

10. **Avatar Image Storage**
User-selected avatars are stored as references to predefined avatar image IDs rather than uploading custom image files. This approach ensures consistent file sizes, prevents inappropriate content uploads, and maintains efficient storage usage suitable for a children's educational platform.

Security & Protection

11. **Brute Force Login Protection**
The system locks user accounts for 15 minutes after 5 consecutive failed login attempts to prevent unauthorized access through brute force attacks. Users receive clear notification of the lockout and when they can retry. The specific implementation approach will be finalized later on this project.

12. **Input Sanitization & Injection Prevention**
All user inputs, particularly the search bar and registration forms, must be sanitized to prevent SQL Injection and Cross-Site Scripting (XSS) attacks. This includes escaping special characters and validating input formats before processing. The specific implementation approach will be finalized later on this project.

13. **Child-Safe Content Filtering**
    All Unsplash API queries include the "content filter" = high parameter to ensure only age-appropriate, child-safe images are returned.

14. **Password Security**
    User passwords must be hashed using bcrypt or a similar strong hashing algorithm with appropriate salt before storage in the database. Passwords are never stored in plain text. The specific implementation approach will be finalized later on this project.

Portability & Compatibility

15. **Browser Compatibility**
    The application must function correctly on modern versions of Chrome, Edge and Firefox.

Usability & Accessibility

16. **Child-Friendly Interface Design**
    The user interface must feature large, colorful buttons, clear visual icons, and minimal text to accommodate young children with developing reading skills. Navigation should be intuitive and require minimal adult assistance.

17. **Feedback Clarity**
    All system feedback (success animations, error messages, hints) must use simple, age-appropriate language and rely heavily on visual cues (colors, animations, icons) rather than text to ensure comprehension by young users.

Use Cases

This section provides detailed scenarios describing how users interact with the system to accomplish specific goals.

Use Case 1: Solving a Puzzle with Visual Validation

Goal in Context	A child engages in cognitive practice by selecting puzzle pieces and receiving immediate visual feedback on their placement.
Scope	Interactive educational puzzle platform.
Level	Primary task.
Preconditions	The child has selected a topic and difficulty level, and the system has generated the puzzle board.
Success End Condition	The child has successfully placed the correct piece, the system has validated the visual continuity and provided positive feedback (success animation), the success data is updated in the database.
Failed End Condition	The child places an incorrect piece, the system detects a visual mismatch and provides corrective feedback ("Try Again" prompt), the specific attempt is not recorded as a success in the database.
Primary Actor	Child user.
Trigger	The child enters the game interface to solve a generated puzzle.

Main Success Scenario:

1. The system displays the incomplete image on the main canvas and a tray of possible puzzle pieces below it.

2. The child identifies a missing section and drags a piece from the tray into the empty slot.

3. The child clicks the "Check My Answer" button to verify their move.

4. The system captures the visual data of the combined image and analyzes the boundaries using Computer Vision algorithms.

5. The system confirms the visual continuity and provides positive feedback (success animation).

6. The system records the success in the user's history and displays a "Next Puzzle" or "Play Again" option.

Sub-Variations:

3. a. The child decides the piece is wrong before checking and drags it back to the tray to swap it.

5. a. The system detects a visual mismatch (edges or colors do not align).

- 5.a.1. The system triggers a "mistake" animation and a gentle corrective prompt.

- 5.a.2. The child tries a different piece from the tray (return to step 2).

Use Case 2: Customizing Game Experience

Goal in Context	The user configures the puzzle session by selecting a specific topic of interest and adjusting the difficulty level to suit the child's skills.
Scope	Interactive Educational Puzzle Platform.
Level	User Goal.
Preconditions	The user is on the Home Dashboard.
Success End Condition	The system successfully retrieves an image matching the criteria, segments it according to the chosen difficulty, and launches the game interface.
Failed End Condition	The system fails to fetch an image (API error) or cannot generate the requested grid size, returning an error message to the dashboard.
Primary Actor	Child or Adult user.
Trigger	The user decides to start a new game session.

Main Success Scenario:

1. The user interacts with the "Topic Selection" interface (either typing in the Search Bar or clicking "Surprise Me").

2. The system validates the input and prepares the Unsplash API query.

3. The user adjusts the difficulty to the desired level (e.g., 32 pieces / 2 missing pieces).

4. The user clicks the "Start Game" button.

5. The system fetches the image, processes the segmentation, and transitions to the Game Interface.

Sub-Variations:

1.a. The user selects "Surprise Me".

- 1.a.1. The system automatically selects a random category from the safe-list.

5.a. The Unsplash API returns no results for the search term.

- 5.a.1. The system displays a "No images found" message and asks for a new keyword.