BloomBrain: A Modular Language Learning Quiz Game

GoogleDoc LINK: CLICK HERE

Overview

BloomBrain is an interactive language-learning quiz game developed in Unity. It showcases a modular level progression system with difficulty scaling, adhering to best practices such as the Single Responsibility Principle (SRP) and leveraging Unity's ScriptableObjects. Designed for scalability and reusability, BloomBrain aligns with the goal of creating educational games that grow more challenging as players progress.

Key Features

1. **Level Progression System**
    - Tracks player progress through levels, with clear XP thresholds.
    - Provides visual feedback via a dynamic XP progress bar.
2. **Dynamic Difficulty Adjustment**
    - Implements difficulty scaling using a `DifficultyAdjuster` component.
    - Adjusts question complexity and time limits as players advance, categorized into Easy, Medium, and Hard modes.
3. **Configurable System**
    - Uses ScriptableObjects to store and manage:
        - XP thresholds
        - Level definitions
        - Difficulty parameters
    - Allows easy customization without modifying underlying code.
4. **Engaging Gameplay**
    - Features a variety of language-related questions, including translations, fill-in-the-blank, and sentence matching.
    - Provides immediate feedback with animations for correct/incorrect answers.
    - Celebrates milestones with level-up screens and difficulty previews.
5. **Reusability and Scalability**
    - Packaged as a Unity Package Manager (UPM) package, ensuring seamless integration into other projects.
    - Developed with adherence to design patterns like Strategy and Observer for maintainability.

System Components

XP System

- **Responsibility:** Tracks player XP and manages level thresholds.
- **Key Methods:**
  - `AddXP(int amount)` – Adds XP based on question outcomes.
  - `CheckLevelUp()` – Validates if a level-up condition is met.

Level Manager

- **Responsibility:** Manages level progression and triggers difficulty adjustments.
- **Key Methods:**
  - `LoadNextLevel()` – Progresses to the next level and adjusts difficulty.
  - `SetDifficulty(string difficulty)` – Adjusts parameters for Easy, Medium, or Hard levels.

Difficulty Adjuster

- **Responsibility:** Dynamically modifies game settings.
- **Adjustments Include:**
  - Time limits for answering questions.
  - Complexity of questions presented.

Progress Bar

- **Responsibility:** Provides a visual representation of XP earned.
- **Key Features:**
  - Animations for XP gain.
  - Dynamic updates during gameplay.

ScriptableObject Configurations

- **Data Stored:**
  - XP thresholds
  - Question pools categorized by difficulty
  - Level parameters
- **Benefits:** Enhances usability for game designers and facilitates runtime adjustments.

Development Approach

Adherence to Design Patterns

- **SRP:** Each component has a single, well-defined responsibility.
- **Observer Pattern:** Ensures XP and level updates propagate to UI elements.
- **Strategy Pattern:** Dynamically switches difficulty algorithms.

UPM Package

- **Structure:**
  - Organized folders for Scripts, ScriptableObjects, and Resources.
  - Metadata files for seamless integration.
- **Ease of Use:** Designed to allow import/export as a package for reuse.

Playtesting and Polishing

- Conducted rigorous testing to ensure smooth transitions and accurate difficulty scaling.
- Added animations and sound effects to enhance user engagement.

How BloomBrain Meets Assignment Requirements

Requirement	Implementation
SRP and Design Patterns	Modular components adhering to SRP, using Strategy and Observer patterns.
UPM Package	Packaged as a Unity UPM package for easy integration.
ScriptableObjects for Configs	Used for storing thresholds, difficulty parameters, and questions.
XP System	Tracks XP and triggers level progression.
Difficulty Adjuster	Dynamically scales gameplay based on player progression.

Conclusion

BloomBrain is more than just an educational game; it is a testament to scalable, reusable, and engaging game development. The system's adherence to best practices and robust architecture ensures it can be easily adapted for future projects or used as a framework for other educational tools.