Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the Play, Learn, and Protect Platform. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, operational parameters, and constraints, acting as a reference for developers, testers, project managers, and involved parties during development.

1.2 Document Conventions

Functional Requirements	Prefixed with FR (e.g., FR-Requirement)
Non-Functional Requirements	Prefixed with NFR (e.g., NFR-Requirement)
Specifications	Prefixed with SP (e.g., SP-Specification)
User Stories	Prefixed with US (e.g., US-Story)
Essential terminology and section titles	Appear in **Bold**
Critical annotations or explanations	Appear in *Italic*
Importance tiers/ Prioritization	Categorized as Must, Should, or Could

1.4 Product Scope

The software being produced is called the Play, Learn, and Protect Platform. The system will encourage creativity and social skills through playful games that prioritizes Egyptian culture, and ensure digital safety. This creates a cultural first, engaging, and protected online environment for Egyptian children aged 3–12.

The PLP system will allow a user to register as a parent or a teacher. The parent will be allowed to create a child account, and set the privacy and permission options. A parent will have the option to freeze a child's account, and overview screen-time, digital behavior, and performance through their dashboard. The platform will allow children users to open the dashboard and select a game from a wide variety of age-appropriate games, a child will only be able to select or access games that are within the permissions set by their parent and appropriate for their age. The platform will allow children users to receive points according to their accomplishment and unlock achievement badges, and view where their rank amongst their peers (if allowed by the parent). The platform will allow teachers to apply for a teachers account, provided that they work for an educational institution, which will be confirmed by the admin before account creation. The platform will allow teachers to recommend games or modules for children under their supervision, all children under a teachers' supervision must be confirmed by their parents. Teachers will also be allowed to overview screen-time, digital behavior, performance of students that are under their supervision. The admin will be allowed to review and accept/reject teacher proposals. The admin can ban any user, specifying a reason. The admin can add/remove games/modules.

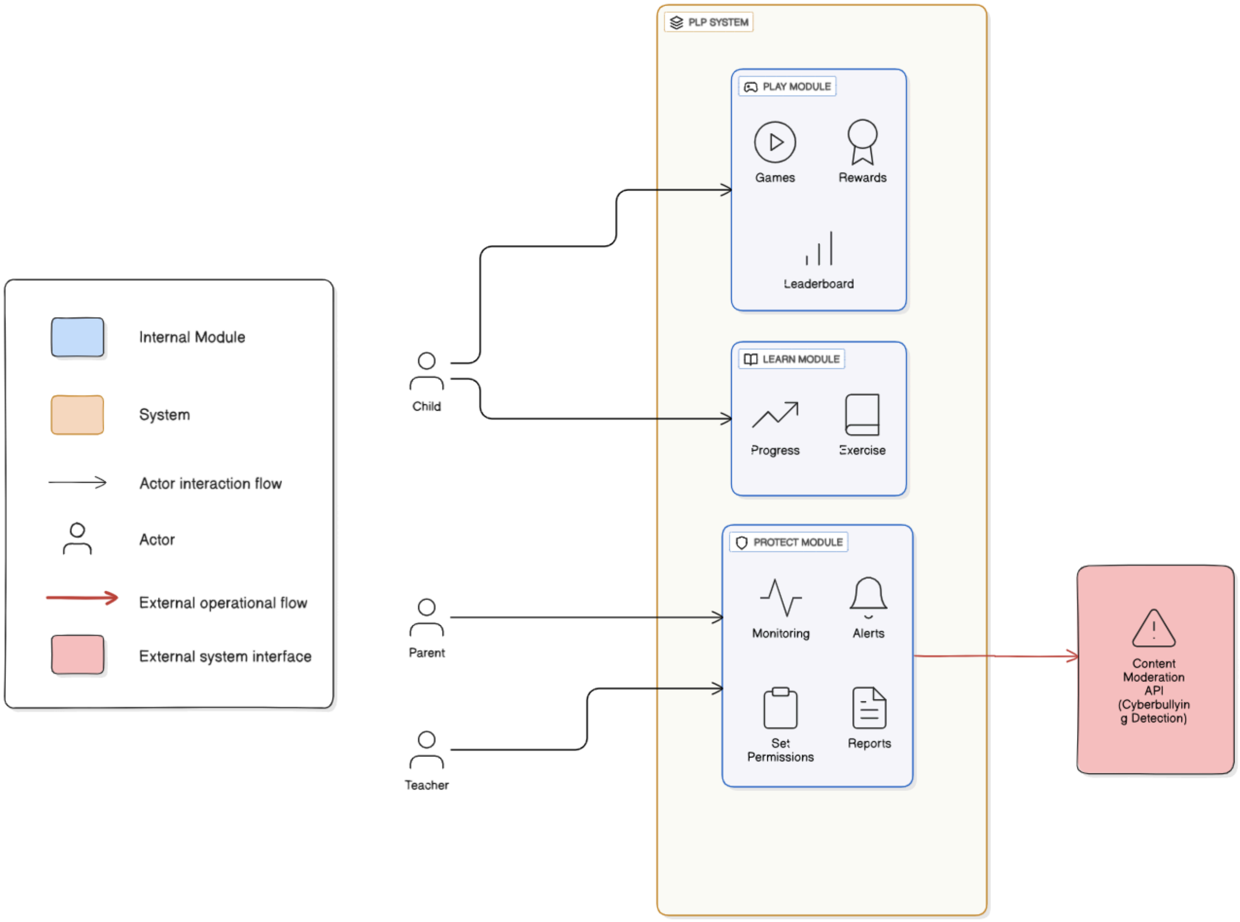
The system has multiple constraints on what it can provide. The system will not provide educational material in the form of videos. It will not allow parents or teachers to overview children that are not under their supervision. The platform will not allow children to create accounts, or take actions that exceed their allowed permissions set by their parents. Teachers will not be allowed to recommend content to students that isn't appropriate to their age. There will only be one admin that is hard-coded by the system.

1.5 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

2. Overall Description

2.1 Product Perspective



In the diagram above the fully integrated PLP system is illustrated. The system has 3 active actors, each interacting with their respective modules (shown via black arrows). Interactions with external systems are shown via red arrows. External systems are placed inside a red box. All internal modules are placed inside the orange box (labelled as PLP system). Internal modules are placed in blue boxes within the PLP system.

2.2 Product Functions

Blue=> reserved

FR-ID	**Description**	**Priority**
FR-1	Allow children to select and launch age-appropriate educational games.	Must
FR-2	Track and reward points for completing games.	Must
FR-3	Display a leaderboard to show top learners per age group.	Should
FR-4	Apply learned knowledge through a gamified experience in.	Must
FR-5	Allow parents/teachers to monitor a child’s screen time	Must
FR-6	Allow parents/teachers to monitor a child’s digital behavior	Must
FR-7	Allow parents/teachers to monitor a child’s performance.	Must
FR-8	Detect potential cyberbullying keywords/phrases and alert parents/teachers.	Must
FR-9	Allow system administrators to upload and manage educational content.	Must
FR-10	Allow multi-role login (Child, Parent, Teacher, Admin).	Must

FR-11	Allow children to save and resume progress in learning modules.	Must
FR-12	Let parents set daily screen time limits per child account.	Must
FR-13	Provide help within games when a child is stuck.	Could
FR-14	Allow parental accounts to manage multiple child profiles.	Must
FR-15	Include culturally relevant themes games.	Must
FR-16	Notify parents when a safety alert is triggered.	Must

2.3 User Classes and Characteristics

There are 4 types of users that will interact with our platform. Child users (aged 3-12) are assumed to have mixed varying digital literacy and short attention spans. Parent users are assumed to have potentially restricted technical skills, are safety-conscious, and want visibility into a child’s activity, and performance. Teacher users are curriculum experts, they oversee multiple students, and overview their reports. System administrators are responsible for account governance, content curation, and infrastructure upkeep.

2.5 Design and Implementation Constraints

There are constraints to be considered when developers are developing the platform. All content must be delivered primarily in Arabic, and reflect Egyptian culture. Easy to use interface for children such as large touch targets and intuitive navigation. Parental consent is required for data collection on their children.
Add implementation Constraints

2.7 Assumptions and Dependencies

The system assumes children will have access to an internet-connected device, and parents and teachers will possess basic digital literacy. It depends on stable cloud infrastructure for hosting, integration with third-party content moderation APIs for safety features, and ongoing educator collaboration to validate curriculum alignment and cultural relevance.

4-System Features

FR-1: Allow children to select and launch age-appropriate educational games (3–12 years).

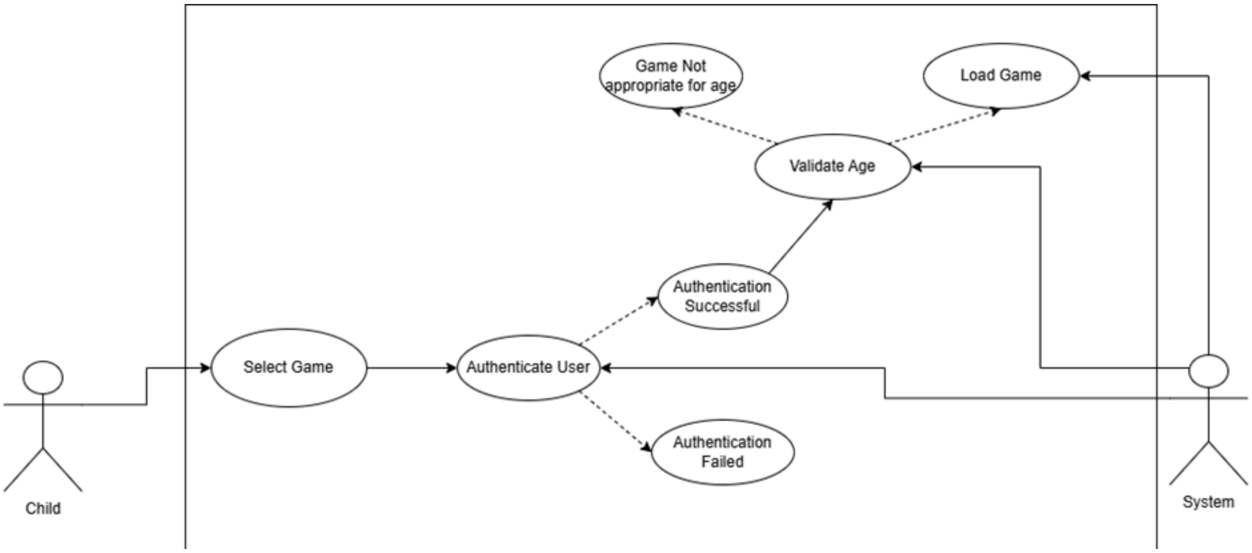
User Story

As a child user, I want to be able to choose and start games that are right for my age group, so that I can learn and play safely without seeing content that's too easy or too hard for me.

Detailed Specifications

tion	
m shall authenticate the child before allowing the game library.	ation via username/password or parent-managed
's age group shall be retrieved from their profile n.	ps: 3–5, 6–8, 9–12.
e library shall be filtered dynamically based on the e group.	tside the age range shall not appear in the main
e tile shall display: title, age badge, learning and estimated playtime.	licators for age appropriateness (e.g., color-coded
e selection, the system shall validate that the ge range matches the child's age group.	ch, show a friendly error: "This game is for a ge."
nch shall initialize within 3 seconds after	creen with playful animation should be displayed.
shall be able to exit the game and return to the any time.	' button shall be visible but not prominently placed ccidental clicks.
launch attempts shall be logged for parental	des: timestamp, game name, session duration.

UC-01



Acceptance Criteria

ıce Criterion	ario
child is logged in, when they view the game library, then only · their age group are displayed.	6-year-old → see only 6–8
hild selects a game, when the game's age range does not match ile, then an age-restriction message is shown and the game does ı.	l tries to launch a 9–12 game ıessage appears.
alid game selection, when the child clicks "Play", then the game ıin 3 seconds.	m click to game start screen ≤ .
game is playing, when the child clicks "Exit", then they are o the game library.	n functions and returns to ·iew.
/ game launch attempt, when completed, then an entry is recorded ·vity log with correct details.	ıhboard shows game played, ·tion.

child is not authenticated, when they try to access the game en they are redirected to the login screen.	L access without login → login.
child’s profile age is updated (e.g., birthday), when they next n the game library reflects the new age group.	ted from 8 to 9 → library now 12 games.

FR-2: Track and reward points for completing learning modules and mini-games.

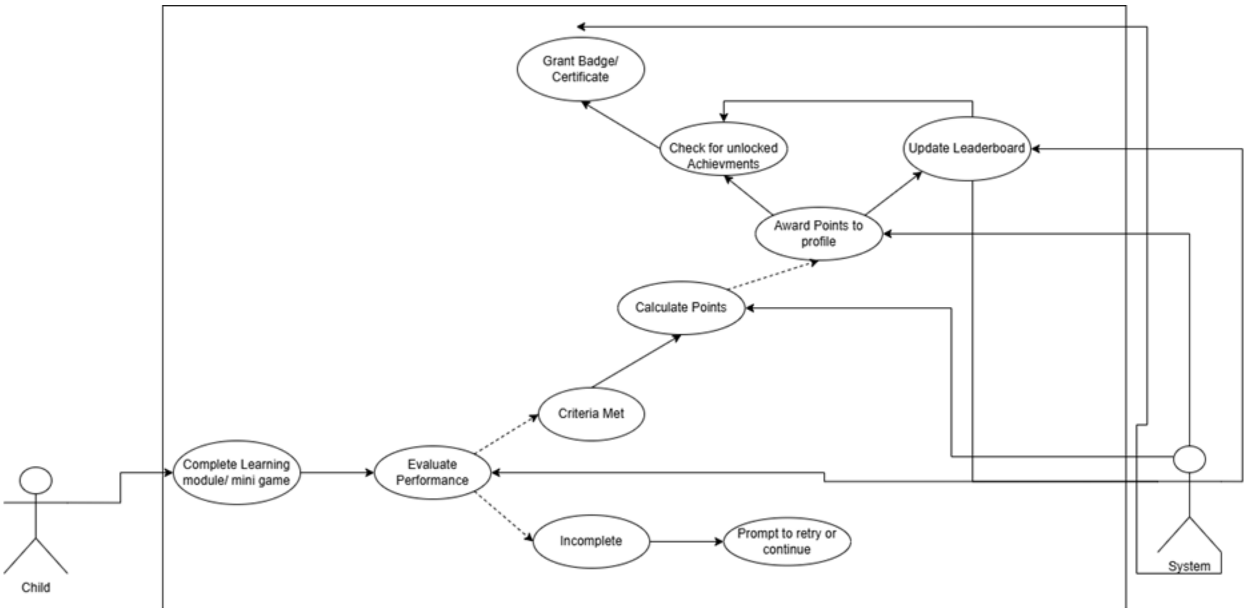
User Story

As a child user, I want to earn points when I finish learning activities and mini-games, so that I feel motivated to keep learning and can see my progress grow over time.

Detailed Specifications

ID	**Specification**
SP-2.1	The system shall track completion of each learning module and mini-game.
SP-2.2	Points shall be awarded based on: · Answer Accuracy · Completion Time · Module Difficulty
SP-2.3	The point breakdown shall be displayed to the child after module completion.
SP-2.4	Points shall be added in the child’s profile and visible in their dashboard.
SP-2.5	Achievements/badges shall be unlocked when point milestones are reached.

SP-2.6	Leaderboard rankings shall update automatically based on total points (per age group).
SP-2.7	Parents/teachers shall receive notifications for major achievements.



Acceptance Criteria

ce Criterion	ario
hild completes a learning module, when they finish, then calculated and displayed within 5 seconds.	a math game → see points n popup.

nts are awarded, when the child views their profile, then oints are updated correctly.	ofile before/after earning 50 points → ases by 50.
hild earns 100 total points, when the milestone is reached, Century Club" badge is unlocked.	from 99 to 100 → badge appears in ents.
hild is in the 6–8 age group, when they earn points, then updates on the age-group leaderboard.	ts → check leaderboard → see rank.
hild completes the same module twice within 24 hours, finish again, then no additional points are awarded.	me game → points remain d.
arent has notifications enabled, when their child unlocks a ievement, then the parent receives an alert.	ns "Science Star" → parent gets tification.
system experiences a failure during point awarding, when , points are not duplicated or lost.	crash after module completion → ints awarded once.

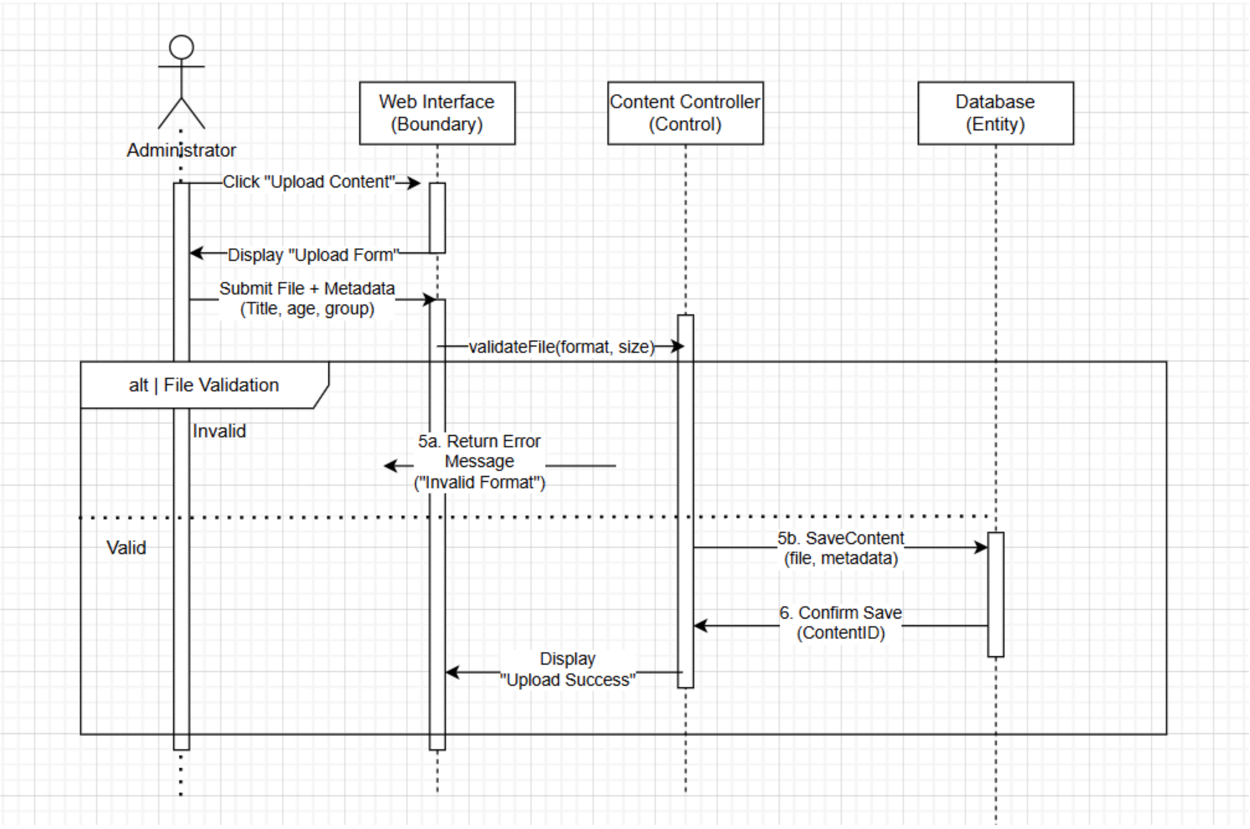
FR-9: Allow system administrators to upload and manage educational content

User Story

As a system administrator,

I want to upload and manage/control educational games and videos. In other words, I want to have the ability to upload, edit, and delete educational games and learning materials in order for the system to be updated with new and relevant content geared towards children.

ID	Specifications
SP-9.1	In the platform, there should exist a dashboard only to a person with the “Administrator” role.
SP-9.2	The Administrator should have the ability to upload game assets with file size cap(e.g., 1GB) in certain formats (.zip, .mp4, .html5).
SP-9.3	When uploading, the application will need the administrator to add metadata for every upload. For instance, he/she will need to input a description, title, targeted age group(3-5, 6-8, 9-12), and subject category (English, History, etc).
SP-9.4	While uploading, the application will check if the file format fits the desired/supported format. In other words, the file will get approved if it is in the supported format and it will be rejected if it has unsupported extensions.
SP-9.5	The administrator has the power to “Delete” or “Unpublish” content which will make it unavailable/unseen on the child’s dashboard.


Administrator
Web Interface (Boundary)
Content Controller (Control)
Database (Entity)
Click "Upload Content"
Display "Upload Form"
Submit File + Metadata (Title, age, group)
validateFile(format, size)
alt | File Validation
Invalid
5a. Return Error Message ("Invalid Format")
Valid
5b. SaveContent (file, metadata)
6. Confirm Save (ContentID)
Display "Upload Success"

Acceptance Criteria

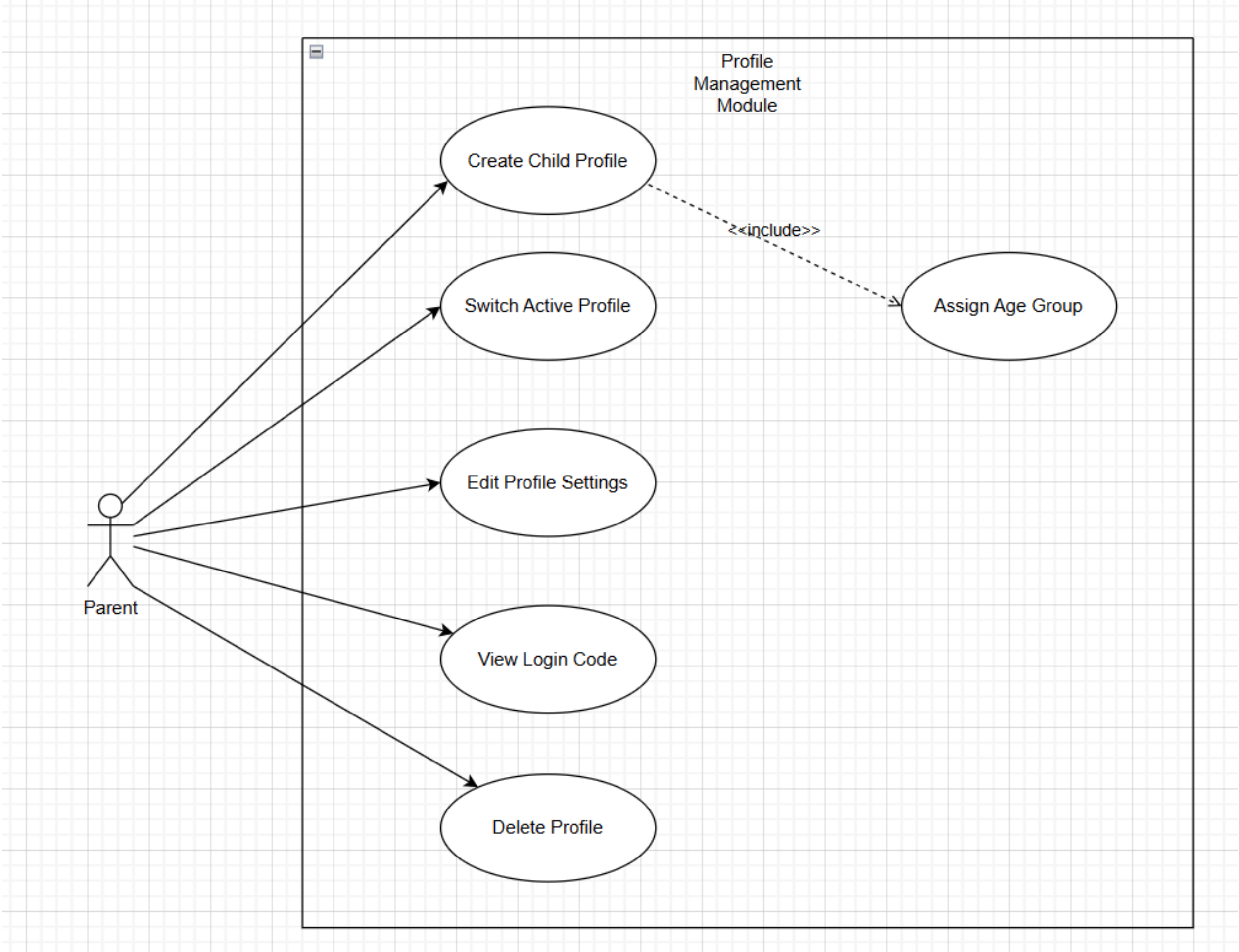
Acceptance Criteria	**Scenario**
Successful Content Upload	Since the administrator is already on the upload screen, when the uploaded file and metadata are all done, the system should then save the content and display “Upload Successful”.
Invalid File Type Rejection	After attempting to upload the file, the system denies the upload and displays an error “Invalid File Type”.
Content Removal	If the game is still working and the admin deletes the game, then the game should be removed from the database and it shouldn’t be displayed in the child’s search results.

FR-14: Allow parental accounts to manage multiple child profiles

User Story

As a Parent, I want to be able to add several child profiles to my account so I can control their settings and see the progress for all of my children from one login.

ID	**Specification**
SP-14.1	The parent should have the ability to create their child’s profile by clicking “Add Child” from their(parent’s) dashboard.
SP-14.2	In order for the parent to create their child’s profile, the system will require for the parents to add their name, age, and avatar preference.
SP-14.3	By analyzing the child’s information from the parent’s inputs, the application will assign a child to their age group.
SP-14.4	The parent will be able to switch between their children' s profiles to view each child’s reports without logging out.
SP-14.5	Each child profile will have a simple username or a unique login code which is generated by the application.



Acceptance Criteria

Acceptance Criterion	**Scenario**
Child Login Redirect	If a child user logs into the application, he/she will be taken to the Game Library page.
Parent Login Redirect	If a parent user logs into the application, he/she will be taken to the parent dashboard.
Invalid Login	If a user enters the wrong password when they are logging in, the application will stay on the login page and it will display “Invalid Username or Password.”

1. External Interface Requirements

1.1 User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>

1.2 Software Interfaces

Supabase, a backend-as-a-service (BaaS), will be primarily used as the backend. It offers multiple services that will help speed the development of the project:

- A REST API (via PostgREST): used to communicate with the PostgreSQL database.
- Auth: Supubase uses JWT as its authentication strategy, and already provides a read-to-use register and login functionality that follows security best practices.
- PostgreSQL database: Supabase automatically handles the deployment of the database.
- File storage: Supabase has a file storage implementation, which eliminates having to use an external file storage service.
- Realtime support: updates to the database are immediately sent to subscribed clients.

All communication is done via HTTPS (except for Supabase Realtime, it uses Websockets), and request/response bodies are in JSON format.

We will use two methods for content moderation:

1. Google’s Perspective API: used to detect cyberbullying and inappropriate content
    - Six detection categories of toxicity: (toxicity, severe toxicity, identity attack, insult, threat, profanity)
    - Provides multilingual support (for Arabic).
2. Our own custom-built time tracker to detect excessive gaming.

2. Functional Requirements

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

2.1 System Feature 1

<Don't really say "System Feature 1." State the feature name in just a few words.>

4.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

4.1.2 Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

4.1.3 Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use "TBD" as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-1:
REQ-2:

FR-01
User story:

Specification:

Diagram:
Use case or activity or sequence

Acceptance criteria:

2.2 System Feature 2 (and so on)

3. Nonfunctional Requirements

3.1 Performance Requirements

In order to provide an engaging experience, the platform must provide responsive and fluid interactions across all user pages. Below are requirements that ensure the system performs fluidly and responsibly under various scenarios.

NFR-ID	Description
NFR-01	The system must be able to handle at least 1,000 concurrent users without any noticeable delays
NFR-02	Under normal conditions, all webpages and dashboards must load in under three seconds
NFR-04	API responses must be under 100 milliseconds
NFR-06	Realtime updates must propagate to clients under 1 second

3.2 Safety Requirements

The system must protect children from digital harm. Below are requirements that ensure users are protected and secure.

NFR-ID	Description
NFR-07	All user‑generated content, chat, and comments must be processed in real time for Cyberbullying, hate speech, and inappropriate material using Google's Perspective APIs
NFR-08	Parents must be able to set daily time limits for their child's account

NFR-09	Upon reaching a limit, the platform must automatically pause the session
NFR-10	Data regarding users name, age, location must not be visible to anyone without parental approval
NFR-12	An "Account Freeze" function must be available to parents to immediately suspend their child's access

3.3 Security Requirements

NFR-ID	Description
NFR-14	Passwords should have a number of conditions that are set in place
NFR-15	Multi-factor authentication for parent and teacher accounts
NFR-16	Child accounts can only be created through a parent
NFR-17	Teacher accounts require verification via a valid institutional email and approval by the administrator
NFR-18	Sensitive data at must be encrypted using AES-256
NFR-19	Use a role-based access control with the following roles: Child, Parent, Teacher, Administrator

3.4 Usability

NFR-ID	Description
NFR-20	A child must be able to navigate to and launch a game within 3 interactions, without external help.

NFR-21	A parent or teacher with basic digital literacy must be able to navigate the application with ease without requiring user manuals or support.

3.5 Reliability & Availability:

NFR-ID	Description
NFR-22	The platform must maintain 99% uptime during core operational hours (6:00 AM - 10:00 PM local time).
NFR-23	Scheduled maintenance must occur outside of peak child usage hours (e.g., 2:00 AM - 6:00 AM)
NFR-24	System failures must fail gracefully