Babu Media

MVP Product Specification

Version 1.0
December 2025

Confidential

Table of Contents

1. Executive Summary

Babu Media is a next-generation "Edutainment" platform that transforms passive screen time into active, creative bonding sessions between parents and children (ages 5–12). The MVP delivers the core experience: personalized character creation and AI-powered story generation.

1.1 MVP Scope

The Minimum Viable Product includes six core modules:

1. **Authentication** — Google Sign-In via Supabase Auth
2. **Onboarding** — Child profile creation with age, name, and preferences
3. **Style Preferences** — Selection of favorite children's content styles
4. **Fusion Lab** — AI character creation from animal selection or custom prompt
5. **Character Storage** — Persistent character database with anchor images
6. **Plot World** — Story generation with text and AI-generated images

1.2 Core Technology Stack

Component	Technology
Frontend	React Native (Expo)
Backend	Supabase (PostgreSQL + Edge Functions)
Authentication	Supabase Auth (Google OAuth)
LLM Engine	Claude 3.5 Sonnet (Anthropic API)
Character Generation	Nano Banana Pro
Scene Generation	Flux PuLID (via Fal.ai)

2. Product Vision & Philosophy

2.1 The IKEA Effect Strategy

Psychological research demonstrates that people value things they create themselves disproportionately more than things they buy. In the current state, children consume pre-made content (Paw Patrol, Bluey, Minions) with no ownership. Babu Media enables children to build their own characters and write their own stories, creating significantly higher emotional attachment and retention.

2.2 The Sidekick Model

Unlike apps designed to "babysit" children, Babu Media facilitates active parenting through a three-way collaboration:

- **The AI** handles the heavy lifting: drawing, animating, and providing imagination suggestions
- **The Parent** provides creative direction: prompting, reading, and guiding the experience
- **The Child** provides the imagination: the wild ideas that power the creation

2.3 Target Audience

Primary User: Children ages 5–12

Secondary User: Parents (especially fathers as the "sidekick")

Session Duration: 10–15 minutes of quality bonding time

3. MVP Feature Specifications

3.1 Feature 1: Authentication

Purpose: Secure user identification and session management

Requirements

- Google OAuth 2.0 integration via Supabase Auth
- Automatic profile creation on first sign-in
- Secure session token management
- Sign-out functionality
- Persistent login across app restarts

User Flow

1. User launches app and sees welcome screen
2. User taps "Sign in with Google" button
3. Google OAuth popup/redirect appears
4. User authenticates with Google credentials
5. System creates/retrieves user profile
6. User is redirected to Onboarding (new) or Home (returning)

3.2 Feature 2: Onboarding

Purpose: Collect essential child information for personalized experience

Data Collection Fields

Field	Type	Description
Child Name	Text (required)	First name for personalization
Age	Number (5-12)	Age-appropriate content filtering
Parent Name	Text (optional)	For personalized prompts
Avatar	Image selection	Profile avatar from preset options

3.3 Feature 3: Style Preferences

Purpose: Determine visual style preferences for character generation

Available Style Options

Users select their top 3 favorite styles from popular children's content:

Content	Visual Style Keywords
Minions	Bright yellow, round shapes, playful, 3D Illumination style
Paw Patrol	Bold colors, heroic poses, rescue themes, clean 3D CGI
Pokemon	Anime style, creature design, vibrant, action poses
Avatar (Airbender)	Anime-inspired, elemental magic, martial arts, epic landscapes
K-Pop Demon Hunters	Stylish, K-drama aesthetic, supernatural, modern urban
Pixar/Disney	Warm lighting, emotional expressions, high-quality 3D render

Content	Visual Style Keywords
Bluey	Soft colors, family-friendly, Australian wildlife, 2.5D style

Style Integration

Selected styles are stored in the user profile and automatically appended to character generation prompts. The system combines the child's style preferences with their character description to create consistent, personalized visuals.

3.4 Feature 4: Fusion Lab (Character Creation)

Purpose: Enable children to create personalized AI characters

Creation Methods

Method A: Animal Selection

Users choose from a curated list of animal archetypes:

- Fox, Raccoon, Bunny, Wolf, Bear, Cat, Dog, Owl, Lion, Dragon
- Each animal comes with default personality suggestions
- Users can customize with clothing and accessories

Method B: Custom Prompt

Users describe their character in their own words:

- Free-text input field for character description
- AI parses and extracts key visual elements
- Example: "A cyber-punk raccoon with neon goggles"

Character Attributes

Attribute	Input Type	Example Values
Name	Text input	"Rocky", "Luna", "Captain Whiskers"
Archetype	Selection/Text	"Fox", "Cyber-Raccoon", "Space Bunny"
Clothing	Selection	"Blue Hoodie", "Detective Coat", "Space Suit"
Personality	Multi-select	["brave", "silly", "curious", "kind"]
Style	Auto-applied	Derived from Style Preferences selection

Generation Process

1. User completes character form (name, archetype, clothing, personality)
2. System constructs detailed prompt combining user input + style preferences
3. Nano Banana Pro generates high-fidelity 3D character image
4. User previews result with fun loading messages ("Mixing DNA...", "Sewing costume...")
5. User confirms or regenerates
6. Confirmed character becomes the "Anchor" image for future stories

3.5 Feature 5: Character Storage

Purpose: Persist created characters for reuse across stories

Storage Requirements

- Each character saved with unique ID and anchor image URL
- Characters linked to user profile
- Images stored in Supabase Storage bucket
- Metadata includes creation timestamp and usage count
- Characters appear in "My Characters" gallery

3.6 Feature 6: Plot World (Story Generation)

Purpose: Create personalized bedtime stories with the child's character

Story Creation Flow

1. User selects a character from their gallery
2. User enters a plot idea (e.g., "goes to the moon", "finds a treasure")
3. Claude API generates a 5-scene story with text and visual prompts
4. Flux PuLID generates images for each scene using the anchor image
5. Story is saved and displayed in the interactive reader

Story Structure

Each story consists of 5 scenes (pages):

Scene	Narrative Role	Description
1	Introduction	Introduce character and setting
2	Rising Action	Character encounters the adventure/challenge
3	Development	Character works through the challenge
4	Near Resolution	Character approaches the solution
5	Climax & Resolution	Happy ending with lesson learned (is_climax: true)

Reader UI Layout

The story reader uses a split-screen layout optimized for tablets:

- **Left Side (40%):** Story text in large, readable font
- **Right Side (60%):** Full-bleed generated image
- **Navigation:** Swipe left/right to turn pages
- **Page Indicator:** Dots at bottom showing current position

4. Technical Architecture

4.1 System Overview

Babu Media follows a serverless, AI-native architecture. The React Native frontend is intentionally "thin" — it primarily displays content generated by backend services. All business logic, API orchestration, and AI integration happens in Supabase Edge Functions.

4.2 Data Flow

1. **Authentication:** User → Google OAuth → Supabase Auth → JWT Token
2. **Character Creation:** App → Edge Function → Nano Banana Pro → Supabase Storage → Database
3. **Story Generation:** App → Edge Function → Claude API → Flux PuLID → Database
4. **Reading:** App → Database → Display (images served from Supabase Storage CDN)

4.3 API Key Management

All API keys are stored as Supabase secrets, never exposed to the client:

- ANTHROPIC_API_KEY — Claude 3.5 Sonnet access
- NANO_BANANA_API_KEY — Character generation
- FAL_API_KEY — Flux PuLID scene generation

5. Database Schema

The PostgreSQL database (hosted on Supabase) consists of five core tables.

5.1 Table: profiles

Purpose: Extended user data (linked to Supabase Auth)

Column	Type	Description
id	UUID (PK)	References auth.users
email	TEXT	User email from Google
child_name	TEXT	Child's first name
child_age	INT	Age (5-12)
parent_name	TEXT	Parent's name (optional)
style_preferences	TEXT[]	Array of style keys
avatar_url	TEXT	Profile avatar image
credits	INT	Freemium credits (default: 5)
onboarding_complete	BOOLEAN	Onboarding status flag
created_at	TIMESTAMPTZ	Account creation timestamp

5.2 Table: characters

Purpose: Store created characters ("Anchors")

Column	Type	Description
id	UUID (PK)	Auto-generated unique ID
user_id	UUID (FK)	References profiles.id
name	TEXT	Character's given name
archetype	TEXT	Base animal/creature type
personality_traits	TEXT[]	Array of traits
description	TEXT	Full prompt used for generation
anchor_image_url	TEXT	Master reference image URL
created_at	TIMESTAMPTZ	Creation timestamp

5.3 Table: stories

Purpose: Store story metadata

Column	Type	Description
id	UUID (PK)	Auto-generated unique ID
user_id	UUID (FK)	References profiles.id
character_id	UUID (FK)	References characters.id
title	TEXT	Story title (AI-generated)
plot_summary	TEXT	User's original plot idea
created_at	TIMESTAMPTZ	Creation timestamp

5.4 Table: scenes

Purpose: Store individual story pages

Column	Type	Description
id	UUID (PK)	Auto-generated unique ID
story_id	UUID (FK)	References stories.id
page_number	INT	Scene order (1-5)
text_content	TEXT	Story text for this page
image_prompt	TEXT	Prompt sent to Flux PuLID
image_url	TEXT	Generated scene image URL
is_climax	BOOLEAN	Marks scene 5 as climax

6. API Specifications (Edge Functions)

6.1 create-character

Trigger: User clicks "Create" in Fusion Lab

Request Body

```
{ name, archetype, clothing, personality_traits, custom_prompt? }
```

Workflow

1. Construct prompt combining archetype + clothing + style preferences
2. Call Nano Banana Pro API for high-fidelity character generation
3. Upload generated image to Supabase Storage
4. Insert character record into database
5. Return character object with anchor_image_url

Prompt Template

```
A cute 3D render of a {archetype} wearing {clothing}, {style} style, clean
white background, front view, character sheet aesthetic, high detail, Pixar
quality
```

6.2 generate-story

Trigger: User clicks "Write Story" in Plot World

Request Body

```
{ character_id, plot_idea }
```

Workflow

- Fetch character data from database (name, archetype, anchor_image_url)
- Call Claude 3.5 Sonnet with story system prompt
- Parse JSON response into 5 scenes
- For each scene, call Flux PuLID with anchor image + visual prompt
- Upload scene images to Supabase Storage
- Insert story and scenes into database

Flux PuLID Parameters

- **reference_image_url:** Character's anchor_image_url
- **prompt:** Visual description from Claude's response
- **id_weight:** 0.85 (preserves face while allowing body movement)

7. UI/UX Specifications

7.1 Design Principles

- **Tablet First:** Optimized for iPad landscape mode
- **Large Touch Targets:** Minimum 44pt tap areas for small fingers
- **Cinematic Feel:** Dark theme with vibrant accent colors
- **Minimal Text:** Icon-driven navigation where possible
- **Parent Supervision Indicators:** Clear visual cues when parental input is needed

7.2 Screen Specifications

Screen 1: Login

- Full-screen animated background with floating characters
- Babu Media logo centered
- "Sign in with Google" button prominently displayed

Screen 2: Onboarding (3-step flow)

- **Step 1:** Child name + age input with playful animations
- **Step 2:** Style preference selection (grid of movie posters)
- **Step 3:** Avatar selection + confirmation

Screen 3: Home / Library

- Bookshelf visual metaphor
- Created stories displayed as book covers
- "+ New Story" and "+ New Character" prominent CTAs
- Bottom navigation: Home, Characters, Settings

Screen 4: Fusion Lab

- Split layout: Form inputs (left) + Preview area (right)
- Animal selection carousel
- Clothing/accessory picker
- Personality trait toggle chips
- Loading state with fun messages: "Mixing DNA...", "Sewing costume..."

Screen 5: Plot World (Story Creator)

- Character selector at top
- Large text input for plot idea
- Suggestion chips for plot inspiration
- Generation progress indicator with scene previews

Screen 6: Story Reader

- Split layout: Text (left 40%) + Image (right 60%)
- Swipe navigation between pages
- Page indicator dots at bottom
- Close button returns to Library

8. AI Integration Details

8.1 Nano Banana Pro (Character Generation)

Use Case: Initial character creation in Fusion Lab

- High-fidelity 3D character rendering
- Consistent style matching user preferences
- Output becomes the "anchor" reference for all future generations

8.2 Claude 3.5 Sonnet (Story Generation)

Use Case: Generate structured 5-scene stories

- Strict JSON output format
- Age-appropriate content filtering
- Visual prompts optimized for image generation
- 15-word maximum per scene text (bedtime reading)

8.3 Flux PuLID (Scene Generation)

Use Case: Generate story scene images with character consistency

- Reference image maintains character identity
- id_weight parameter balances identity vs. pose flexibility
- Produces consistent character across all 5 story scenes

9. Cost Analysis

9.1 Per-Story Unit Economics

Action	Provider	Est. Cost
Create Character	Nano Banana Pro	$0.04
Write Script	Claude 3.5 Sonnet	$0.01
Generate 5 Scenes	Flux PuLID	$0.15
TOTAL PER STORY		**$0.20**

9.2 Break-Even Analysis

Assuming a subscription price of $9.99/month:

- **Break-even:** ~50 stories per user per month
- **Expected usage:** 4-8 stories per month (based on bedtime frequency)
- **Gross margin:** ~80-85% at expected usage levels

10. Development Roadmap

10.1 MVP Development Timeline

Day	Focus	Deliverables
1	Infrastructure	Supabase project, database tables, storage buckets
2	Auth + Backend	Google OAuth, create-character Edge Function
3	Fusion Lab UI	React Native character creation screen
4	Story Backend	generate-story Edge Function (Claude + Flux)
5	Reader UI	Story reader screen with swipe navigation
6-7	Onboarding	3-step onboarding flow with style preferences
8-10	Polish	Bug fixes, animations, loading states, testing

10.2 Post-MVP Phases

- **Phase 2 (Months 6-9):** Communication Rooms, animated video loops
- **Phase 3 (Months 10-14):** Classroom modules, Playground games
- **Phase 4 (Month 15+):** Physical book printing, video calls with characters

11. Appendix: Prompt Engineering

11.1 Claude Story Engine System Prompt

The following system prompt is sent to Claude 3.5 Sonnet for story generation:

```
You are the "Story Engine" for a children's app.
Your Goal: Create a 5-page bedtime story based on a user's prompt.
Output Format: STRICT JSON only. No markdown, no chatting.
```

Expected JSON Structure

```
{
  "title": "Story Title",
  "scenes": [
    {
      "page": 1,
      "text": "Max 15 words for child to read.",
      "visual_prompt": "IMG person doing [action]...",
      "is_climax": false
    }
  ]
}
```

Critical Rules

- visual_prompt MUST start with "IMG person" (trigger word for character)
- visual_prompt must be descriptive of background and setting
- Scene 5 must always set is_climax: true
- Keep tone whimsical, safe, and educational

11.2 Character Generation Prompt Template

```
A cute 3D render of a {archetype} wearing {clothing},
{style_preferences} style, clean white background,
front view, character sheet aesthetic, high detail
```

— End of Document —