StoryBond: Complete 6-Week AI MVP Delivery Plan

Where Every Child Becomes the Hero of Their Story

Project Duration: 6 Weeks (February 3 - March 14, 2026)
Demo Day: March 28, 2026 (5-7 minute interactive presentation)
Team Size: 6 members × 5 hours/week = ~30 hours/week total capacity
Target: Urban Indian parents (mothers 28-42) in Bangalore, Mumbai, Delhi, Hyderabad

---

Executive Summary

This plan delivers a **demonstrable, production-ready MVP** of StoryBond in 6 weeks with a 6-person part-time team. The MVP focuses on **3 core experiences** that create maximum wow-factor for Demo Day:

- **Personalized Story Generation**: AI-powered narratives in English, Hindi, and Kannada where the child is the hero
- **Hero Image Creation**: Custom AI-generated illustration featuring the child's characteristics
- **Digital Story Experience**: Beautiful mobile-first reading interface with shareable stories

Critical Constraint: Team members work part-time (5 hours/week each = 30 hours/week total). This plan is optimized for **asynchronous work**, **minimal meetings**, and **maximum impact per hour invested**.

PART 1: TECHNICAL FOUNDATION

A. Tech Stack (AIGF Course-Aligned)

Frontend Stack (Google Antigravity + Stitch)

- **Primary Tool:** Google Antigravity IDE with Stitch Skills (agentic development from Google)
- **Why:** AI-powered design-to-code in minutes, Gemini-native, AIGF-aligned, full-stack automation
- **Design Layer:** Stitch (stitch.withgoogle.com) for UI generation from text prompts
- **Deployment:** Firebase Hosting (Google ecosystem) or Vercel
- **Backup:** v0 by Vercel if team prefers standalone tool

Backend Stack (Supabase)

- **Database:** Supabase PostgreSQL (from AIGF curriculum)
- **Authentication:** Supabase Auth (email/password)
- **Storage:** Supabase Storage (hero images)
- **Edge Functions:** Supabase Edge Functions (story generation pipeline)
- **Why:** All-in-one backend, free tier, taught in AIGF

AI Stack (OpenAI)

- **Story Generation:** GPT-4o (balance of quality and cost)
  - Cost: $2.50/1M input + $10/1M output tokens
  - Target: ~800 words/story = ~$0.05/story
- **Image Generation:** DALL-E 3
  - Cost: $0.04/image (standard 1024×1024)
- **Safety:** OpenAI Moderation API (free)

Domain & Hosting

- **Domain:** storybond.in (₹800/year on GoDaddy/Namecheap)
- **Frontend:** Lovable hosting (included) or Vercel (free tier)
- **Backend:** Supabase free tier (up to 500MB database, 50K users)

B. Complete Database Schema (Supabase)

The database schema includes four core tables with Row Level Security:

1. Profiles Table - Extends Supabase auth.users with additional user information

2. Children Table - Stores child profiles with age (2-8 years), language preferences (English, Hindi, Kannada), and interests

3. Stories Table - Contains generated stories with title, content, hero image URL, language, and reading level

4. Product Mockups Table - Manages keepsake orders (storybook PDF, poster, mug) with status tracking

Row Level Security policies ensure users can only access their own data, with parent_id filtering for children and stories tables.

C. Cost Breakdown

Monthly Costs (MVP Phase)

Item	Cost	Notes
OpenAI API	$10-30	50-150 stories during dev + demo
Supabase	$0	Free tier (sufficient for MVP)
Domain	₹67/month	storybond.in annual ÷ 12
Hosting	$0	Lovable/Vercel free tier
Total	~$15-35/month	~₹1,200-2,800/month

One-Time Setup Costs

- Domain registration: ₹800 (annual)
- OpenAI API credit: $5 initial deposit
- **Total:** ~₹1,200 (~$15)

PART 2: TEAM ROLES & RESPONSIBILITIES

Role Assignments (6 Members × 5 Hours/Week)

Role 1: Product & Demo Lead (Binu Chacko)

Weekly Time: 5 hours

Core Duties:

- Define MVP scope and feature priorities
- Write demo script and scenarios
- Test user flows and gather feedback
- Own Demo Day presentation

Detailed Weekly Tasks:

- **Week 1 (5h):** Finalize scope document, write 3 demo scenarios, create success metrics
- **Week 2 (5h):** Review 10 test stories for quality, document issues, refine demo personas
- **Week 3 (5h):** Test mobile UX on 3 devices, invite 2 friends for feedback, update requirements
- **Week 4 (5h):** Define keepsake catalog, write product descriptions, plan mockup specs
- **Week 5 (5h):** Analyze 30 test stories, prioritize top 5 fixes, draft final demo script
- **Week 6 (5h):** Rehearse demo 3× with team, finalize slides, prepare Q&A responses

---

Role 2: AI & Prompt Engineering (Rajesh K Agarwal)

Weekly Time: 5 hours

Core Duties:

- Design GPT-4o prompts for story generation

- Integrate OpenAI APIs (GPT-4o, DALL-E 3, Moderation)
- Implement Supabase Edge Function for story pipeline
- Ensure content safety and multilingual quality

Detailed Weekly Tasks:

- **Week 1 (5h):** Research GPT-4o children's storytelling, design 3 system prompts (English, Hindi, Kannada) for ages 2-8
- **Week 2 (5h):** Write Supabase Edge Function generate-story, integrate GPT-4o API, test 5 stories/language
- **Week 3 (5h):** Add DALL-E 3 integration to Edge Function, optimize parallel API calls, test 10 hero images
- **Week 4 (5h):** Tune image prompts for print quality, assist Design Lead with mockup requirements
- **Week 5 (5h):** Iterate prompts based on feedback (3 refinement rounds), add theme filtering logic
- **Week 6 (5h):** Generate 5 "perfect" demo stories, document AI architecture for Q&A, prepare explainer slide

Key Deliverables:

- 3 system prompt templates (1 per language)
- Working Edge Function with GPT-4o + DALL-E 3
- 30+ test stories generated across languages

Role 3: Design & Content Strategy (Abhigyan Srivatsava)

Weekly Time: 5 hours

Core Duties:

- Design UI wireframes and visual brand
- Create content guidelines (age-appropriate, culturally sensitive)
- Design hero image style guides
- Create Demo Day slides and mockups

Detailed Weekly Tasks:

- **Week 1 (5h):** Create Figma wireframes for 4 screens, define color palette + typography, write content guidelines
- **Week 2 (5h):** Design child profile form, story prompt UI, loading animations
- **Week 3 (5h):** Design Reading Room layout, story detail page, mobile responsiveness specs
- **Week 4 (5h):** Create 3D product mockups (book, poster, mug) using Photoshop/Canva, design Gift Shop UI
- **Week 5 (5h):** Design feedback UI (thumbs up/down), polish visual consistency, create brand assets
- **Week 6 (5h):** Create Demo Day presentation deck (10-12 slides), prepare demo flow screenshots

Key Deliverables:

- Complete UI design system in Figma

- 3D product mockups for 3 keepsake types
- Demo Day slide deck

Role 4: Backend & Supabase (Kumar LR)

Weekly Time: 5 hours

Core Duties:

- Set up Supabase project and database schema
- Implement Row Level Security policies
- Configure storage bucket for images
- Support AI Engineer with Edge Function deployment

Detailed Weekly Tasks:

- **Week 1 (5h):** Create Supabase project, run SQL schema, configure auth, set up storage bucket stories-images
- **Week 2 (5h):** Test auth flow end-to-end, verify RLS policies, seed test data (5 users, 10 children)
- **Week 3 (5h):** Test image upload/download, verify public URLs work, monitor storage usage
- **Week 4 (5h):** Create product_mockups table, write CSV export script for manual orders
- **Week 5 (5h):** Add database indexes for performance, optimize queries, document schema
- **Week 6 (5h):** Deploy final schema to production Supabase, write deployment runbook, backup database

Key Deliverables:

- Supabase project fully configured
- Database schema with RLS
- Manual order export script

Role 5: Frontend & Antigravity (Neev Bafna)

Weekly Time: 5 hours

Core Duties:

- Build all 4 screens using Google Antigravity + Stitch
- Connect frontend to Supabase backend via Antigravity agents
- Implement responsive mobile-first UI with Stitch-generated code
- Test user flows and fix bugs with AI assistance

Detailed Weekly Tasks:

- **Week 1 (5h):** Set up Antigravity IDE, install Stitch Skills MCP, authenticate Google Cloud, generate skeleton for 4 screens using Stitch prompts (see Appendix A)
- **Week 2 (5h):** Use Stitch to design auth screens (sign up, login), let Antigravity connect to Supabase auth automatically, build child profile form with validation

- **Week 3 (5h):** Design story generation flow in Stitch (form → loading → result), use Antigravity agents to integrate Edge Function, display hero image + story text with responsive layout
- **Week 4 (5h):** Generate Gift Shop screen with Stitch (keepsake selection UI), implement order confirmation page, connect to Supabase product_mockups table
- **Week 5 (5h):** Add feedback UI (thumbs up/down) via Stitch, improve error states with Antigravity debugging agents, test on mobile devices (iOS + Android Chrome)
- **Week 6 (5h):** Polish animations using Stitch refinement, test demo flow 10× with Antigravity preview mode, export production build to Firebase Hosting

Key Deliverables:

- 4 fully functional screens generated by Antigravity + Stitch
- Mobile-responsive UI (auto-generated by Stitch)
- Connected to Supabase backend (agent-assisted integration)

Role 6: QA, DevOps & Integration (Vishawajeet Sarkar + Zaheer Ul Islam)

Weekly Time: 5 hours (split between 2 people: 2.5h each)

Core Duties:

- Test all features end-to-end
- Monitor Supabase and API usage
- Coordinate team integration
- Prepare Demo Day checklist and backups

Detailed Weekly Tasks:

- **Week 1 (5h):** Set up GitHub project board, create testing spreadsheet template, document setup instructions
- **Week 2 (5h):** Test auth flow on staging, verify API key security, load test story generation (5 concurrent)
- **Week 3 (5h):** Test image upload/retrieval, verify mobile responsiveness on 3 devices, document bugs
- **Week 4 (5h):** Test keepsake selection flow, verify CSV export, create manual fulfillment guide
- **Week 5 (5h):** Organize 30-story testing campaign, track results in spreadsheet, file GitHub issues for bugs
- **Week 6 (5h):** Final QA (test every feature), record demo video as backup, prepare Demo Day tech checklist

Key Deliverables:

- Testing documentation
- Demo Day backup plan (video, screenshots)
- Bug tracking and resolution

PART 3: WEEK-BY-WEEK EXECUTION PLAN

Meeting Cadence (Part-Time Optimized)

Async Daily Check-In (5 mins/person via WhatsApp)

Format:

- Yesterday: One specific thing completed
- Today: One specific thing doing
- Blocker: Any? If yes, tag who can help

Weekly Video Sync (60 mins, Sundays 8 PM IST)

Agenda:

1. **Demo last week's work** (10 min each role = 60 min total) - Screen share what you built, live test if possible
2. **Identify blockers** (10 min)
3. **Plan next week** (10 min) - Confirm tasks from plan below, adjust if needed

Ad-Hoc Pairing (as needed)

- Backend + AI: 1-2 sessions for Edge Function integration
- Frontend + Design: 1-2 sessions for UI implementation
- Use Google Meet or Zoom, record sessions

Week 1: Foundations & Scope Lock

Dates: Feb 3-9, 2026
Goal: Align team, lock scope, set up technical infrastructure

Objectives

- Finalize MVP scope (no changes after Week 1)
- Set up Supabase project and database
- Create Lovable project skeleton
- Design 3 system prompt templates
- Write 3 demo scenarios

Success Criteria

- All 6 team members can access Supabase + Antigravity IDE (Google Cloud auth complete)
- Database schema created with test data
- 3 demo child personas documented
- Team aligned on what's in/out of MVP
- Stitch Skills installed and tested (generate 1 sample screen as proof)

Deliverables by Role

Role	Hours	Deliverables
Product Lead	5h	Scope doc, 3 demo scenarios, success metrics
AI Engineer	5h	3 system prompt templates (English, Hindi, Kannada)
Design Lead	5h	Figma wireframes (4 screens), color palette, typography
Backend	5h	Supabase project, database schema, RLS policies, storage bucket
Frontend	5h	Lovable project with 4 empty screens + routing
QA/DevOps	5h	GitHub project board, testing template, setup documentation

Key Decisions This Week

- **Languages:** English, Hindi, Kannada (no more for MVP)
- **Age bands:** 2-4, 5-8 (not 3 bands to save time)
- **Keepsakes:** 3 types only (book, poster, mug)
- **No payments** in MVP (demo only)

Week 2: Core Engine (Story + Auth)

Dates: Feb 10-16, 2026
Goal: Working story generation end-to-end

Objectives

- User can sign up, log in, create child profile
- Story generation works for all 3 languages
- 15 test stories generated (5 per language)
- Content safety moderation working

Success Criteria

- Auth flow works on staging
- Story generation API returns valid JSON
- 15 stories reviewed and approved for quality
- API response time less than 15 seconds per story

Technical Milestone

Edge Function deployed: generate-story

- Input: child_id, prompt
- Output: title, story_text, reading_level
- Integrated with OpenAI GPT-4o + Moderation API

Week 3: Images & Reading Experience

Dates: Feb 17-23, 2026
Goal: Complete core UX with hero images

Objectives

- DALL-E 3 integration working
- Hero images stored in Supabase Storage
- Reading Room displays stories with images
- Mobile UX polished

Success Criteria

- Story generation produces text + image in less than 30 seconds
- 10 hero images generated and pass quality review
- Mobile UX smooth on iPhone and Android
- 3 friendly users provide positive feedback

Technical Milestone

- Edge Function extended: story + image generation
- Images stored in stories-images bucket
- Public URLs returned and displayed in frontend

Week 4: Keepsakes (Mockup)

Dates: Feb 24 - Mar 2, 2026
Goal: Demonstrate physical keepsake vision

Objectives

- Gift Shop UI complete with 3 product options
- Order creation flow working (database only)
- 3D mockups created for demo
- Manual export workflow documented

Success Criteria

- Keepsake selection looks fully functional
- Order creation stores data in database
- 3D mockups look professional and shareable
- Team can demo full flow including "order"

Demo Strategy

- Show keepsake selection (looks real)
- Display order confirmation (feels complete)
- Show 3D mockups (generates excitement)
- Explain: "Production will integrate with Printful"

Week 5: Polish & Testing

Dates: Mar 3-9, 2026
Goal: Refine quality, fix bugs, iterate

Objectives

- 30 test stories generated by team
- Story quality improved based on feedback
- All critical bugs fixed
- Content filtering working

Success Criteria

- 30 stories generated with 80%+ positive feedback
- Zero critical bugs in staging
- Content filtering prevents inappropriate themes
- Team confident for Demo Day

Testing Campaign

Each team member generates 5 stories and tracks in shared spreadsheet:

- Story ID, prompt, language, quality rating (1-5), issues
- File GitHub issues for bugs found
- Verify all fixes deployed to staging

Week 6: Demo Day Prep

Dates: Mar 10-14, 2026
Goal: Polish, rehearse, prepare for Demo Day

Objectives

- Demo rehearsed 3+ times without issues
- 3 perfect demo stories pre-generated
- Presentation deck finalized
- Backup plans prepared

Success Criteria

- Bug-free demo flow tested 10+ times
- Team can present without hesitation
- Backup video recorded
- Confidence level: 9/10

Demo Day Rehearsal Schedule

- **March 22 (Saturday):** Full team rehearsal 1 (2 hours)
- **March 25 (Tuesday):** Full team rehearsal 2 (1.5 hours)
- **March 27 (Thursday):** Final rehearsal 3 (1 hour)

PART 4: DEMO DAY EXECUTION PLAN

Demo Day: March 28, 2026

Presentation Structure (7 Minutes)

0:00-0:30 Hook (Product Lead)

"Think of the stories you heard as a child. What language were they in? Who was the hero? Today, most children still grow up in someone else's story, in someone else's language. What if we could change that in just 30 seconds?"

0:30-1:15 Problem (Product Lead)

- 22 official Indian languages, yet 99% of kids' content is English-only
- Custom stories cost ₹5,000 + 3 weeks
- Parents give up on personalized content
- First stories (ages 2-8) shape lifelong identity

1:15-1:45 Solution (Product Lead)

"StoryBond uses AI to create personalized stories where YOUR child is the hero, in their language, in minutes, not weeks."

1:45-5:00 Live Demo (AI Engineer + Frontend Lead)

1. **Show Nursery** (0:15): "Here are stories created by parents"
2. **Create child profile** (0:30): "Meet Aarav, 6 years old, loves space and dosa, speaks Kannada"
3. **Generate story live** (1:00): Enter prompt, show loading, reveal story + hero image
4. **Show Reading Room** (0:45): Scroll through story, highlight Aarav as hero
5. **Show Gift Shop** (0:45): Display keepsake mockups (book, mug)

5:00-5:45 Traction (Product Lead)

- 30 stories generated during testing
- 85% positive feedback from early users
- 3 languages supported (English, Hindi, Kannada)
- Mobile-first, works on any device

5:45-6:30 Vision (Product Lead)

- Scale to 10 languages
- Full POD integration (automatic fulfillment)
- 10,000 families in first year
- Partnerships with schools and libraries

6:30-7:00 Ask & Close (Product Lead)

"We're looking for feedback, pilot users, and potential partners. Stories shape identity. Let's ensure the next generation grows up in their own story."

Demo Scenarios (Pre-Prepared)

Scenario 1: Live Demo (Primary)

- **Child:** Aarav, 6 years old, loves space and dosa, Kannada
- **Prompt:** "A bedtime adventure where Aarav saves the space station with his dosa-powered rocket"
- **Status:** Will generate live during demo
- **Backup:** If live generation fails, switch to Scenario 2

Scenario 2: Pre-Generated (Backup 1)

- **Child:** Priya, 5 years old, loves animals, Hindi
- **Prompt:** "A brave girl who saves a forest from pollution"
- **Status:** Already generated, tested, approved
- **Use if:** Live demo API fails or is too slow

Scenario 3: Pre-Generated (Backup 2)

- **Child:** Rohan, 8 years old, loves science, English
- **Prompt:** "A curious boy who discovers a new planet"
- **Status:** Already generated, tested, approved
- **Use if:** Both primary scenarios fail

Demo Day Checklist

One Week Before (March 21)

- Record full demo video as backup (7 minutes)
- Create screenshot slide deck (Plan C)
- Print 3D mockups if possible (show physical)
- Prepare demo device (laptop + phone backup)

One Day Before (March 27)

- Test production environment (all 3 scenarios)
- Verify OpenAI API key funded ($20+ balance)
- Load 3 pre-generated stories in database
- Charge all devices (laptop, phone, tablet)
- Download backup video locally

- Test on presentation equipment (projector, clicker)

Day Of (March 28, Morning)

- Test internet connection + backup hotspot
- Open all tabs (production app, slides, backup video)
- Clear browser cache, log out of personal accounts
- Set up "demo user" account with clean stories
- Do final run-through (30 minutes before)
- Set phone to airplane mode

Contingency Plans

If API Fails (Live Demo)

1. Stay calm, say: "Let me show you one we prepared earlier"
2. Click on pre-generated story in Nursery
3. Continue demo smoothly

If Internet Fails

1. Switch to mobile hotspot (test beforehand)
2. If still fails, play backup video
3. Narrate over video live

If UI Breaks

1. Switch to screenshot slide deck
2. Walk through flow using slides
3. Show video clip of working demo

If Technical Questions Get Hard

1. AI Engineer steps in to answer
2. Product Lead handles business questions
3. Defer deep technical questions to post-demo discussion

PART 5: RISK ASSESSMENT & MITIGATION

Critical Risks (High Likelihood × High Impact)

Risk 1: Story Quality Inconsistent

Likelihood: High | **Impact:** High

Description: GPT-4o generates stories that are too long, lack emotion, or culturally inappropriate.

Mitigation:

- Test 30+ stories across languages/ages in Week 5

- Iterate prompts 3-5 times based on feedback
- Add content filtering (blacklist scary/violent keywords)
- Pre-generate 5 "perfect" demo stories by Week 6
- During demo, use pre-generated story if live one is poor

Early Signals:

- Test stories rated less than 3/5 by team
- Stories flagged by Moderation API
- Feedback mentions "not age-appropriate"

Risk 2: API Latency or Failures During Demo

Likelihood: Medium | **Impact:** High

Description: OpenAI API slow (more than 30s) or fails during live demo.

Mitigation:

- Test API reliability 10+ times in Week 6
- Pre-generate 3 backup stories and load in database
- Have backup video ready to play
- Practice smooth transition: "Let me show you one we prepared earlier"
- Fund OpenAI account with $20+ before demo

Early Signals:

- API response times more than 20 seconds in testing
- Rate limit errors during Week 5 testing
- OpenAI status page shows issues

Risk 3: Team Bandwidth (Part-Time Constraint)

Likelihood: High | **Impact:** Medium

Description: Team members can't commit 5 hours/week due to work/life pressures.

Mitigation:

- Set expectations upfront: 5 hours is non-negotiable
- Track hours in shared spreadsheet weekly
- If someone less than 3 hours for 2 consecutive weeks, reassign critical tasks
- Pair less available members on lower-priority tasks
- Buffer 1 week between MVP completion (Week 6) and Demo Day (March 28)

Early Signals:

- Missed daily check-ins (more than 2 days silent)
- Tasks not completed by Sunday meeting
- "I'll catch up next week" repeated statements

Risk 4: Supabase Free Tier Limits Exceeded

Likelihood: Low | **Impact:** Medium

Description: Exceed 500MB database or 50K monthly active users during testing.

Mitigation:

- Monitor Supabase dashboard weekly
- Limit test story count (30-50 max before demo)
- Delete old test data regularly
- If approaching limit, upgrade to Pro ($25/month)
- Use separate staging vs production projects

Early Signals:

- Database size more than 300MB
- Storage warnings in Supabase dashboard
- API rate limit errors

Risk 5: Multilingual Quality (Hindi or Kannada)

Likelihood: Medium | **Impact:** Medium

Description: GPT-4o English is great, but Hindi/Kannada stories have grammar issues or sound unnatural.

Mitigation:

- Have native speakers (team or friends) review 5 stories per language
- Iterate prompts specifically for Hindi/Kannada (add "natural, conversational tone")
- If quality poor after 3 iterations, reduce MVP to English only (still valid)
- Document known limitations transparently for Demo Day

Early Signals:

- Native speaker feedback: "This sounds translated, not natural"
- Incorrect word choices or grammar errors
- Cultural references that don't make sense

Medium Risks (Medium Likelihood × Medium Impact)

Risk 6: Antigravity/Stitch Learning Curve

Likelihood: Low | **Impact:** Medium

Description: Team unfamiliar with Antigravity IDE or Stitch Skills workflow may slow Week 1 progress.

Mitigation:

- Complete Pre-Week 0 setup (Google Cloud auth, Antigravity install) before Feb 3
- Frontend Lead (Neev) watches 3 tutorial videos (see Appendix G) before Week 1

- Pair with AI Engineer (Rajesh) for first Stitch screen generation (Week 1, 30-min session)
- If major blockers persist, use Stitch standalone (stitch.withgoogle.com) and export code manually to Antigravity
- Fallback: Use v0 by Vercel (similar AI-generated UI approach)

Risk 7: Hero Image Quality Poor

Likelihood: Medium | **Impact:** Medium

Description: DALL-E 3 images don't look child-friendly or are inconsistent.

Mitigation:

- Test 20+ image prompts in Week 3
- Iterate prompt template (add "vibrant, child-friendly, storybook art style")
- Use image post-processing if needed (crop, brighten)
- For demo, manually select best-looking images

Risk 8: Demo Day Presentation Nerves

Likelihood: Medium | **Impact:** Low

Description: Presenter forgets script or gets flustered during demo.

Mitigation:

- Rehearse 3+ times (March 22, 25, 27)
- Have co-presenter ready to jump in
- Print speaker notes (bullets only)
- Practice transitions between speakers
- Remember: judges care about problem-solution, not perfect delivery

PART 6: SUCCESS METRICS & VALIDATION

Demo Day Success Criteria

Must-Achieve (Blockers)

- Complete 7-minute demo without technical failure
- Show live or pre-generated story in Reading Room
- Display hero image with story text
- Demonstrate keepsake mockup (UI)
- Answer 3 Q&A questions confidently
- Team presents cohesively (no confusion about who speaks)

Nice-to-Have (Bonus)

- Generate story live during demo (not pre-generated)
- Show multilingual capability (switch between English/Hindi/Kannada)
- Show physical mockup sample (printed book or mug)
- Get positive feedback from 3+ judges/attendees
- Interest from 2+ potential pilot users or partners

MVP Quality Metrics (Internal)

Story Generation Quality

- **Target:** 80%+ of test stories rated 4-5/5 by team
- **Measure:** Feedback collected during Week 5 testing campaign
- **Threshold:** If less than 60%, iterate prompts urgently in Week 5

API Performance

- **Target:** Less than 20 seconds for story + image generation
- **Measure:** Average response time across 30 test generations
- **Threshold:** If more than 30 seconds, optimize parallel API calls or pre-generate for demo

Mobile UX

- **Target:** All 4 screens usable on iPhone (iOS 17) and Android (Android 13+)
- **Measure:** Manual testing by 3 team members on their devices
- **Threshold:** If major layout breaks, fix in Week 5

Multilingual Coverage

- **Target:** 5+ quality stories in each language (English, Hindi, Kannada)
- **Measure:** Native speaker review in Week 5
- **Threshold:** If 1 language consistently poor, reduce to 2 languages for MVP

PART 7: POST-DEMO ROADMAP

Immediate Next Steps (Post-Demo Day)

Week 7-8: Iterate Based on Feedback

- Incorporate Demo Day feedback
- Fix any issues discovered during presentation
- Generate 10 more test stories for quality validation
- Document lessons learned

Phase 2 (Months 2-3): Production Readiness

- Full Printful/Printify API integration
- Razorpay payment integration
- Audio narration (ElevenLabs)
- Email/SMS notifications
- Expand to 5 languages (add Tamil, Telugu)

Phase 3 (Months 4-6): Scale

- Pilot with 100 families (₹299-599/story pricing)
- B2B partnerships (schools, libraries)
- Referral program (invite friends, get free story)
- Analytics dashboard (track engagement, story themes)

PART 8: APPENDICES

Appendix A: Antigravity + Stitch Prompts for 4 Screens

How to use these prompts:

1. Open Antigravity IDE → New Project → Name: "storybond-mvp"
2. Connect GitHub: Settings → GitHub → Link your storybond repository
3. Activate Stitch: Type /stitch in chat panel or Cmd+K → "Design with Stitch"
4. Paste prompt below → Antigravity calls Stitch → Generates UI mockup + React/Tailwind code
5. Review in live preview → Iterate with follow-up prompts → Deploy to Firebase

Screen 1: The Nursery (Dashboard)

Antigravity + Stitch Prompt:

Using Stitch skills, design "The Nursery" dashboard for StoryBond MVP:

CONTEXT:

- Target: Urban Indian parents (28-42yo mothers) using mobile-first
- Purpose: Landing page after login showing child's story collection

UI REQUIREMENTS:

- Header: "Welcome, [parent name]" with small avatar icon (top-left)
- Large primary CTA: "Create New Story" button (center, warm orange #FF8B60)
- Below header: Responsive grid of story cards
  - Mobile: 2 columns
  - Desktop: 3 columns
  - Card content: child name (bold), story title (h3), hero image thumbnail (16:9), created date (small gray text)
  - Each card: Two action buttons - "Read Story" (primary) and "Preview Gift" (secondary outline)

- Empty state: If no stories exist, show child-friendly illustration + "No stories yet. Let's write your child's first hero story!"

TECHNICAL INTEGRATION:

- Connect to Supabase 'stories' table
- Filter: WHERE parent_id = auth.user.id (Row Level Security)
- Use React hooks for data fetching
- Tailwind CSS for styling
- Mobile-responsive breakpoints (sm, md, lg)
- Loading skeleton while fetching
- Error state if Supabase connection fails

DESIGN STYLE:

- Colors: Warm pastels (#FFB6C1 pink, #98D8C8 mint, #FFF4E6 cream background)
- Typography: Inter for UI, Poppins for headings
- Child-friendly rounded corners (8px cards, 12px buttons)
- Smooth transitions (0.2s ease)
- Support Kannada/Devanagari fonts (Noto Sans family)

Generate full React component with Supabase integration, then show live preview.

Expected output: Antigravity generates Nursery.jsx with imports, useState/useEffect for Supabase queries, styled components, and live preview in 15-30 seconds.[web:91][web:103]

Screen 2: The Story Lab (Creation Wizard)

Antigravity + **Stitch Prompt:**

Using Stitch skills, design "The Story Lab" multi-step creation wizard for StoryBond:

CONTEXT:

- Purpose: Guide parents through generating a personalized AI story
- Flow: Select child → Enter story idea → Generate (calls OpenAI via Supabase Edge Function)

UI REQUIREMENTS:

- Step 1: Child Selection
  - Dropdown populated from Supabase 'children' table (WHERE parent_id = auth.user.id)
  - Display: child name, age, language (English/Hindi/Kannada)
  - "Add New Child" button below dropdown (opens inline form)
    - Inline form fields: Name (text), Age (number slider 2-8), Language (radio buttons), Interests (multiselect chips: Space, Animals, Food, Sports, Science, Art)
    - Save button → INSERT into 'children' table → refresh dropdown
- Step 2: Story Idea Input
  - Large textarea (3 rows): "Describe an adventure for [child name]..."
  - Character count: 10-100 characters
  - Quick suggestion chips below: "Space adventure", "Forest rescue", "Dosa-powered rocket", "Visit to grandparents"

- Clicking chip populates textarea with template
- Generate Button:
  - Large rounded button: "✨ Generate [Child Name]'s Story"
  - On click: Call Supabase Edge Function 'generate-story' (POST /functions/v1/generate-story)
  - Request body: { child_id, prompt, language }
- Loading State:
  - Full-screen overlay with animated illustration (child reading book)
  - Text: "Crafting [child name]'s story... This takes 20-30 seconds"
  - Spinner animation
- Success: Redirect to /reading-room/[story_id]
- Error: Toast notification "Oops! Let's try that again" with retry button

TECHNICAL:

- Use React useState for form steps, useRouter for navigation
- Supabase Edge Function call with error handling
- Form validation: Require child selection, min 10 char prompt
- Mobile-responsive (vertical layout on small screens)

DESIGN:

- Warm encouraging colors (orange CTA, soft green chips)
- Progress indicator: Step 1 of 2
- Friendly error messages (no technical jargon)

Generate full component with Supabase queries and Edge Function integration.

Pro tip: After Antigravity generates code, test Edge Function separately: /debug edge-function generate-story → Antigravity runs test calls with mock data.[web:92]

Screen 3: The Reading Room (Story Viewer)

Antigravity + Stitch Prompt:

Using Stitch skills, design "The Reading Room" story viewer for StoryBond:

CONTEXT:

- Purpose: Beautiful reading experience for generated stories
- Audience: Parents reading to children ages 2-8 on mobile devices

UI REQUIREMENTS:

- Dynamic Route: /reading-room/[story_id]
- Data Fetch: Supabase 'stories' table WHERE id = story_id AND parent_id = auth.user.id (RLS)
- Layout (Vertical scroll):
  1. Story Title: Large serif heading (32px), dark navy color
  2. Subtitle: "A story for [child name], age [age]" (gray, 14px)
  3. Hero Image: Full-width responsive (16:9 aspect ratio), source: hero_image_url from Supabase Storage
     - Alt text: "Illustration for [story title]"
     - Lazy loading for performance

4. Story Text:
- Serif font (Merriweather or Lora), 18px, line-height 1.8
- Generous paragraph spacing (24px margin)
- Max-width 600px (centered on desktop)
- White background with subtle shadow
- Preserve line breaks from database content

- Sticky Bottom Bar (fixed position):
  - Left: "← Back to Nursery" button (gray outline)
  - Right: " Preview Gift" button (primary orange)
  - Background: white with top border
  - Mobile-safe padding (avoid notches)
- Loading State: Skeleton screen with animated pulse
  - Text: "Opening your storybook..."
- Error State: "Story not found" with friendly illustration + "Back to Nursery" button

TECHNICAL:

- Next.js dynamic routing or React Router with useParams
- Supabase query with error handling
- Image optimization (next/image or lazy loading)
- SEO meta tags (title, og:image from hero image)

DESIGN:

- Reading experience priority (no distracting elements)
- High contrast text for readability (WCAG AA)
- Mobile-first (80% of traffic)
- Smooth scroll behavior
- Print-friendly CSS (optional: allow browser print to PDF)

Generate full component with routing, data fetching, and responsive layout.

Iteration tip: After initial generation, refine typography: "Make story text more readable for bedtime reading - increase line-height to 2.0 and add slight cream tint to background" [web:103]

Screen 4: The Gift Shop (Keepsake Preview)

Antigravity + Stitch Prompt:

Using Stitch skills, design "The Gift Shop" keepsake preview screen for StoryBond MVP:

CONTEXT:

- Purpose: Show parents what physical keepsakes would look like (demo only, no payment)
- Flow: Display hero image on product mockups → Select type → Save preference

UI REQUIREMENTS:

- Input: story_id from URL or passed as prop
- Data Fetch: Supabase 'stories' table (title, hero_image_url)
- Top Section (60% viewport height):
  - Large 3D mockup preview area

- Default: Hero image displayed on storybook cover mockup
- Rotate/scale animation on load (subtle 3D effect)
- Use CSS transforms or simple canvas rendering
- Middle Section: Keepsake Type Selector
  - Three pill buttons (horizontal layout on mobile, row on desktop):
    1. " Storybook PDF" (default selected)
    2. " Poster Print"
    3. " Magic Mug"
  - On click: Switch mockup visualization
    - Book: Hero image on book cover with story title
    - Poster: Hero image in frame (A3 size mockup)
    - Mug: Hero image wrapped around white ceramic mug
  - Active state: Orange border, white background
  - Inactive: Gray outline
- Bottom Sticky Bar:
  - Info text: " This is a preview of your child's keepsake"
  - "Confirm Keepsake" button (large, primary orange)
  - On click:
    - INSERT into 'product_mockups' table: { story_id, keepsake_type, status: 'preview', created_at: NOW() }
    - Show success modal: "We've saved this keepsake! In production, this will be sent to print "
    - "Back to Reading Room" button in modal
- NO payment UI needed (demo mode only)

TECHNICAL:

- Supabase INSERT with error handling
- State management for selected keepsake type (useState)
- CSS animations for mockup transitions
- Mobile-responsive (stack vertically on small screens)
- Image preloading for smooth transitions

DESIGN:

- Mockup visualization: Use CSS 3D transforms or simple image compositing
- Warm exciting colors (gift-giving vibe)
- Clear "demo mode" indicator (avoid confusion about pricing)
- Accessible button labels and focus states

Generate full component with mockup switching logic and Supabase integration.

Week 4 tip: Design Lead (Abhigyan) provides 3D mockup images (book.png, poster.png, mug.png) → Frontend Lead imports into Antigravity project → Stitch overlays hero image using CSS positioning[web:92][web:103]

Appendix B: GPT-4o System Prompt Template

English, Ages 2-4

You are a master children's storyteller writing for children aged 2-4 years.

CORE VALUES:

- Independence: The child is the protagonist and makes choices/takes actions alone
- Respect: Kind behavior toward friends, family, nature is celebrated
- Sensory Detail: Use colors, textures, sounds, movements
- Simplicity: Use 5-10 word sentences, repeat key words, avoid complex grammar

STORY PARAMETERS:

- Child's Name: child_name
- Age: age_years years old
- Interests: interests
- Story Length: 200 words (for ages 2-4)

STORY STRUCTURE:

1. Hero Introduction (50 words): Introduce child_name in a relatable, everyday activity
2. Problem/Challenge (75 words): A simple, age-appropriate challenge they solve independently
3. Respectful Action (50 words): child_name shows kindness to someone/something
4. Happy Ending (25 words): Celebrate child_name's pride and new skill learned

AVOID:

- Supernatural elements (dragons, magic for this age)
- Scary situations (monsters, darkness, being alone)
- Violence or conflict
- Gendered stereotypes

OUTPUT FORMAT (valid JSON):

- title: Story Title (max 5 words)
- content: Full story text (200 words)
- dalle_prompt: Detailed DALL-E 3 prompt for hero image (toddler-friendly, cartoon style, bright colors)
- reading_level: Ages 2-4
- moral: Simple moral lesson (1 sentence)

Hindi, Ages 5-8 (Example)

आप 5-8 साल के बच्चों के लिए एक बेहतरीन कहानीकार हैं।

मुख्य मूल्य:

- स्वतंत्रता: बच्चा नायक है और अपने आप निर्णय लेता है
- सम्मान: दोस्तों, परिवार, प्रकृति के प्रति दयालु व्यवहार
- इंद्रिय विवरण: रंग, बनावट, ध्वनि, गति का उपयोग करें

- सरलता: 8-12 शब्दों के वाक्य, सरल व्याकरण

कहानी पैरामीटर:

- बच्चे का नाम: child_name
- उम्र: age_years साल
- रुचियाँ: interests
- कहानी की लंबाई: 350 शब्द (5-8 साल के लिए)

आउटपुट प्रारूप (वैध JSON):

- title: कहानी का शीर्षक (अधिकतम 5 शब्द)
- content: पूरी कहानी (350 शब्द)
- dalle_prompt: DALL-E 3 के लिए विस्तृत प्रॉम्प्ट (बाल-अनुकूल, चित्रण शैली)
- reading_level: Ages 5-8
- moral: नैतिक सबक (1 वाक्य)

Appendix C: Domain Setup Instructions

Step 1: Register storybond.in

1. Go to GoDaddy.in or Namecheap.com
2. Search for "storybond.in"
3. Add to cart (₹799-899 for first year)
4. Complete payment
5. Set domain nameservers to Lovable/Vercel (see their docs)

Step 2: Configure DNS

For Lovable:

- Follow Lovable docs to add custom domain
- Add CNAME record: www → your-lovable-project.lovable.app
- Add A record: @ → (Lovable's IP, see docs)
- Wait 24-48 hours for DNS propagation

For Vercel (if using):

- Add custom domain in Vercel dashboard
- Add CNAME: www → cname.vercel-dns.com
- Add A record: @ → 76.76.21.21 (Vercel IP)

Step 3: Verify SSL

- Once DNS propagates, Lovable/Vercel auto-provisions SSL (Let's Encrypt)
- Verify HTTPS works: https://storybond.in
- Force HTTPS redirect in Lovable/Vercel settings

Appendix D: Critical Contacts & Resources

Team Contact Info

Role	Name	Email	Phone	Availability
Product Lead	Binu Chacko	[email]	[phone]	5 hrs/week
AI Engineer	Rajesh K Agarwal	[email]	[phone]	5 hrs/week
Design Lead	Abhigyan Srivatsava	[email]	[phone]	5 hrs/week
Backend	Kumar LR	[email]	[phone]	5 hrs/week
Frontend	Neev Bafna	[email]	[phone]	5 hrs/week
QA/DevOps	Vishawajeet + Zaheer	[email]	[phone]	2.5 hrs/week each

Service Credentials (Store Securely!)

- **Google Cloud Project ID:** storybond-mvp-123456 (for Antigravity auth)
- **Google Cloud API Key:** AIza... (in Antigravity MCP config)
- **OpenAI API Key:** sk-proj-... (in Supabase env vars)
- **Supabase URL:** https://project.supabase.co
- **Supabase Anon Key:** eyJ... (public, safe to use in frontend)
- **Supabase Service Role Key:** eyJ... (SECRET, backend only)
- **Firebase Project:** storybond-mvp (for hosting)
- **Domain Registrar:** GoDaddy login at example@email.com

Emergency Contacts

- **Google Cloud Support:** https://cloud.google.com/support (Antigravity/Stitch issues)
- **Antigravity Community:** https://discuss.ai.google.dev (fastest response)
- **OpenAI Support:** https://help.openai.com (if API issues)
- **Supabase Discord:** https://discord.supabase.com (fast help)
- **AIGF Mentor:** mentor@email (if major blocker)

Appendix E: Daily Check-In Template (WhatsApp)

Format:

- Date: [Date]
- Name: [Your Name]
- Yesterday: [One specific thing completed]
- Today: [One specific thing doing]
- Blocker: [None / Need help with X from @person]
- Hours logged: [X hours]

Example:

- Date: Feb 10
- Name: Rajesh (AI Engineer)
- Yesterday: Wrote GPT-4o system prompt for English ages 2-4
- Today: Building Supabase Edge Function for story generation
- Blocker: None
- Hours logged: 1.5 hours

Appendix G: Antigravity + Stitch Learning Resources

Required Watching (Frontend Lead - Week 0):

1. **"AntiGravity + Stitch Changes Everything"** (17 mins)
   - URL: https://www.youtube.com/watch?v=Uu375Di3qgI
   - Covers: Google Cloud auth, Stitch MCP setup, end-to-end project build
   - Watch: Chapters 3:10-8:18 (setup), 8:18-14:27 (building)
2. **"Google Antigravity Tutorial for Beginners"** (15 mins)
   - URL: https://www.youtube.com/watch?v=1Q5sWISByuw
   - Covers: IDE basics, agent commands, debugging
   - Watch: Full video for IDE navigation
3. **"Google Antigravity + Stitch + Convex"** (20 mins)
   - URL: https://www.youtube.com/watch?v=qFRa4siIL4Y
   - Covers: Complete production workflow, database integration
   - Watch: Timestamp 1:00-10:00 (Stitch UI generation)

Optional Reading:

- Stitch Skills GitHub: https://github.com/google/stitch-skills (prompts and examples)
- Antigravity Docs: https://antigravity.google/docs (official reference)
- Stitch + Firebase Export: https://supergok.com/google-stitch-firebase-export (deployment guide)

Quick Reference Card (Print for Team):

ANTIGRAVITY COMMANDS FOR STORYBOND

/stitch "Design [screen name]..." → Generate UI with Stitch
/debug edge-function [name] → Test Supabase Edge Function
/deploy firebase → Deploy to Firebase Hosting
/preview mobile → Test mobile responsiveness
/connect supabase [url] → Link Supabase project
/export code → Download full codebase
Cmd+K → Open AI chat (any question)
Cmd+R → Live preview refresh

Appendix F: Weekly Meeting Agenda Template

StoryBond Weekly Sync - Week [X]

Date: [Date], 8 PM IST
Duration: 60 minutes

AGENDA:

1. **DEMOS (40 mins - 6-7 mins each)**
   - Product Lead: Share screen and show work
   - AI Engineer: Share screen and show work
   - Design Lead: Share screen and show work
   - Backend: Share screen and show work
   - Frontend: Share screen and show work
   - QA/DevOps: Share screen and show work
2. **BLOCKERS (10 mins)**
   - Who is blocked? On what? Who can help?
3. **NEXT WEEK PLAN (10 mins)**
   - Confirm tasks from Week [X+1] plan
   - Any adjustments needed?
   - Set specific goals for next Sunday

ACTION ITEMS:

- [Person]: [Task] by [date]
- [Person]: [Task] by [date]

RECORDING: [Google Drive link]

FINAL CHECKLIST

Before Starting (Feb 3)

Pre-Week 0: Antigravity + Google Cloud Setup (Complete by Feb 2)

- [ ] **Google Cloud Project Created** (Product Lead creates, shares access with team)
  - Go to console.cloud.google.com → New Project → Name: "storybond-mvp"
  - Enable Vertex AI API, Cloud Run API
  - Create API key (IAM & Admin → API Keys)
- [ ] **Antigravity IDE Installed** (All technical members: Frontend, Backend, AI Engineer)
  - Download from antigravity.google (Mac/Windows/Linux)
  - Authenticate with Google Cloud Project ID
  - Test: Create new project, run "Hello World"
- [ ] **Stitch Skills Installed in Antigravity** (Frontend Lead + AI Engineer)
  - Open Antigravity → MCP Settings → Add Stitch Skills
  - Authenticate Stitch with Google Cloud
  - Test: Generate sample UI with prompt "Design a login form"
  - Verify live preview works

Week 1 Start Checklist:

- All 6 team members have read this complete plan
- All team members committed to 5 hours/week minimum
- Created WhatsApp group for daily check-ins
- Scheduled Sunday 8 PM meetings for 6 weeks
- Set up shared Google Drive folder for docs
- Each member has laptop + internet access
- OpenAI account created (any team member can fund $20)
- **Antigravity** + **Stitch working for Frontend/AI/Backend roles** (pre-requisite complete)

Week 1 Complete

- Supabase project live
- Lovable project created
- All 6 members can access both
- 3 demo scenarios documented
- MVP scope locked (no changes after this)

Week 6 Complete

- Demo rehearsed 3 times
- 3 pre-generated stories ready
- Backup video recorded
- Presentation deck finalized
- Domain storybond.in purchased and configured
- Team confident and excited!

Demo Day (March 28)

- Arrived 30 minutes early
- Tested on presentation equipment
- Backup plans ready (video, screenshots)
- Team knows who presents what
- Demo device charged and ready
- LET'S MAKE EVERY CHILD A HERO!

Document Version: 1.0 - Complete
Created: January 31, 2026
Last Updated: January 31, 2026
Owner: Binu Chacko (Product Lead)
Status: Ready for Team Distribution

This document is designed to be printed as PDF or shared as DOCX. All team members should have access.