# YESTORYD - TECHNICAL DEBT & SESSION WORK PLAN

**Date:** January 16, 2026
**Status Validation:** Based on chat history review

---

## ✅ FEEDBACK ITEMS - ALREADY IMPLEMENTED

### 1. Prorated Refund Calculator

**Feedback:** "No clear policy for partial completion refunds"
**Status:** ✅ **BUILT (Dec 13, 2025)**

**Evidence:**

- SQL termination template created
- `termination_logs` table exists
- Pro-rata calculation: `(Sessions Remaining / 6) × ₹5,999`
- Coach settlement split logic implemented
- Location: SQL template in chat history

**What Works:**

```sql
-- Calculates:
Refund = (Sessions Remaining / 6) × Amount Paid
Coach Settlement = (Sessions Completed × ₹999.83) × Split%
Platform = Remainder
```

**No Action Needed** ✅

---

### 2. Payment Webhook Security

**Feedback:** "Not timing-safe, no idempotency, self-calling HTTP anti-pattern"
**Status:** ✅ **HARDENED (Jan 10, 2026)**

**Evidence:**

- Hardened webhook created with crypto.timingSafeEqual
- Zod payload validation
- Database-backed idempotency (processed_webhooks table)
- Direct function calls (no self-HTTP)
- Request ID tracing

- Location: `/app/api/payment/webhook/route.ts`

**What Works:**

```
typescript

// Timing-safe signature verification
crypto.timingSafeEqual(Buffer.from(expectedSignature), Buffer.from(signature))

// Idempotency check
const { data: existing } = await supabase
  .from('processed_webhooks')
  .select('id')
  .eq('event_id', payload.event)
```

**No Action Needed** ✅

---

**3. Audio Quality Check (Assessment)**

**Feedback:** "Missing audio quality validation before Gemini analysis"
**Status:** ✅ **PARTIALLY BUILT**

**Evidence:**

- Coach assessment has duration-based penalties
- Gemini analyzes audio quality in voice prompt
- Too-short recordings (< 20s) automatically score low
- Location: `/api/coach-assessment/calculate-score/route.ts`

**What Works:**

```
typescript

if (duration < 20) {
  voiceScore = 1;
  voiceAnalysis.notes = `Recording too short (${duration}s)`;
}
```

⚠️ **Missing for Parent Assessment:**

- No pre-flight browser compatibility check
- No fallback if MediaRecorder fails
- No retry prompt for poor quality audio

**Partial Action Needed** ⚠️

---

### 4. Communication Journey Mapped

**Feedback:** "Need complete touchpoint mapping with channels"
**Status:** ✅ **DOCUMENTED (Dec 22-24, 2025)**

**Evidence:**

- 82 communication touchpoints mapped
- Channel allocation defined (WhatsApp, Email, SMS)
- Template names created
- Cost analysis done (₹810/month for 100 children)
- Location: Communication swimlanes document

**What's Mapped:**

- 37 Parent touchpoints
- 17 Coach touchpoints
- 10 Admin alerts
- 8 Lead generation
- 10 Reschedule flow

⚠️ **Not Built:**

- AiSensy templates not created in production
- SendGrid templates not configured
- Automation triggers not implemented

**Implementation Needed** 🔴

---

## 🔴 CRITICAL GAPS - NOT IMPLEMENTED

### 5. Weighted Coach Assignment Algorithm

**Feedback:** "Round-robin treats all coaches equally - need conversion-based weighting"
**Status:** ❌ **NOT BUILT**

**Current Problem:**

```javascript
// Simplistic round-robin
const nextCoach = coaches[currentIndex % coaches.length];
```

**Required Fix:**

```sql
-- Add tracking columns
ALTER TABLE coaches ADD COLUMN conversion_rate DECIMAL(3,2);
ALTER TABLE coaches ADD COLUMN avg_parent_nps DECIMAL(3,2);
ALTER TABLE coaches ADD COLUMN specialty TEXT[];

-- Weighted scoring
coachScore = (conversionRate × 0.40) +
             (parentNPS × 0.30) +
             (availableSlots × 0.20) +
             (specialtyMatch × 0.10)
```

**Impact:** Higher conversion rates, better parent satisfaction

**Priority:** 🟠 P2 (Medium)
**Effort:** 12 hours
**Implementation:** Week 3

---

### 6. Daily Coach Digest (8 AM Email)

**Feedback:** "5-minute pre-session brief too late - send daily digest for mental prep"
**Status:** ❌ NOT BUILT

**Current Problem:**

- Pre-session brief shows 5 min before session
- Coach has no time to mentally prepare
- Rushing between back-to-back sessions

**Required Fix:**

```typescript
```

```
// Cron job at 8:00 AM IST
async function sendDailyCoachDigest() {
  const today = new Date();

  for (const coach of coaches) {
    const sessions = await getCoachSessions(coach.id, today);
    if (sessions.length === 0) continue;

    const digest = generateDigestEmail({
      coach,
      sessions, // Each with: child profile, last session notes, AI focus area
      metrics: coachDailyMetrics(coach.id)
    });

    await sendEmail(coach.email, digest);
  }
}
```

**Impact:** Better session quality, reduced coach stress

**Priority:** 🟠 P2 (Medium)
**Effort:** 4 hours
**Implementation:** Week 3

---

### 7. Breakthrough Human-in-Loop Approval

**Feedback:** "Auto-sending breakthrough celebrations risks false positives"
**Status:** ❌ NOT BUILT

**Current Problem:**

typescript

```typescript
// Auto-sends WhatsApp when Gemini detects breakthrough
if (geminiDetectsBreakthrough) {
  await sendWhatsApp(parent, celebrationMessage);
}
```

**Required Fix:**

typescript

```
// Dashboard alert for coach approval
CREATE TABLE breakthrough_approvals (
  id UUID PRIMARY KEY,
  session_id UUID,
  detected_at TIMESTAMPTZ,
  gemini_analysis TEXT,
  audio_timestamp TEXT,
  coach_action TEXT, -- 'sent', 'edited', 'skipped'
  sent_at TIMESTAMPTZ
);

// Coach dashboard shows:
// "🎉 AI detected breakthrough - Review & Send?"
// [Play Audio Clip] [✅ Send] [Edit] [Skip]
```

**Impact:** Prevents embarrassing false celebrations, builds parent trust

**Priority:** 🟠 P1 (High)
**Effort:** 6 hours
**Implementation:** Week 2

---

### 8. Message Consolidation (82→45 Touchpoints)

**Feedback:** "82 messages cause alert fatigue - consolidate related messages"
**Status:** ❌ **NOT IMPLEMENTED**

**Current Problem:**

```
3:30 PM: Session complete ✅
3:32 PM: Session summary 📝
3:35 PM: Homework assigned 📚
3:40 PM: Next session reminder 📅
```

**Required Fix:**

```
3:35 PM (Single Message):
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━
🎉 Aarav's Session Complete!

📝 Today's Progress:
- Learned 5 new "at" family words
- Reading speed improved to 45 WPM

📚 Homework:
Practice flashcards (10 min daily)

📅 Next Session:
Tuesday, Jan 20 at 5:00 PM
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━
```

**Reduction Strategy:**

```
ELIMINATE:
❌ 1-hour reminders (keep only 24-hour)
❌ Separate homework messages
❌ Weekly progress if no changes
❌ Duplicate emails when WhatsApp sent

CONSOLIDATE:
✅ Post-session: summary + homework + next reminder = 1 message
✅ Progress reports only on parent check-in days
✅ Weekly digest only if important info
```

**Impact:** Better engagement, lower unsubscribe rate

**Priority:** 🟠 P1 (High)
**Effort:** 4 hours
**Implementation:** Week 2

---

### 9. Payment Reconciliation Cron

**Feedback:** "If webhook fails, orphaned payments exist - need daily reconciliation"
**Status:** ❌ **NOT BUILT**

**Current Risk:**

```
Scenario: Parent pays ₹5,999 → Razorpay succeeds → Webhook fails (network issue)
Result: Money received, enrollment NOT created, parent angry
```

**Required Fix:**

```typescript
// Run daily at 11 PM
async function reconcilePayments() {
  // Get payments from Razorpay API (last 7 days)
  const razorpayPayments = await razorpay.payments.all({
    from: sevenDaysAgo,
    to: now,
    status: 'captured'
  });

  // Check which exist in our database
  const orphanedPayments = razorpayPayments.filter(payment => {
    return !existsInDatabase(payment.order_id);
  });

  // Create enrollments for orphaned payments
  for (const payment of orphanedPayments) {
    await createEnrollmentFromOrphanedPayment(payment);
    await notifyAdmin('Recovered orphaned payment', payment);
  }
}
```

**Impact:** Zero revenue loss, customer trust maintained

**Priority:** 🔴 P0 (Critical)
**Effort:** 3 hours
**Implementation:** This Week

---

### 10. Gemini Failure Graceful Degradation

**Feedback:** "If Gemini API is down, assessment crashes - need fallback"
**Status:** ❌ **NOT BUILT**

**Current Problem:**

```typescript
try {
  const analysis = await analyzeWithGemini(audio);
  return analysis;
} catch (error) {
  // Shows error to user? Loses data?
}
```

**Required Fix:**

```typescript
try {
  const analysis = await analyzeWithGemini(audio);
  return { status: 'success', analysis };
} catch (error) {
  // Fallback: Save audio, queue retry
  await supabase.from('pending_assessments').insert({
    child_id,
    audio_url: await uploadToStorage(audio),
    status: 'pending_analysis',
    retry_count: 0
  });

  await queueGeminiRetry(childId, audio);

  return {
    status: 'processing',
    message: 'Your reading is being analyzed! Results in 5 mins via email 📧'
  };
}

// Background retry job
async function retryPendingAssessments() {
  const pending = await getPendingAssessments();
  for (const assessment of pending) {
    try {
      const analysis = await analyzeWithGemini(assessment.audio);
      await sendResultsEmail(assessment.child_id, analysis);
      await markComplete(assessment.id);
    } catch {
      if (assessment.retry_count < 3) {
        await incrementRetryCount(assessment.id);
      } else {
        await notifyAdminFailedAssessment(assessment);
      }
    }
  }
}
```

**Impact:** Zero data loss, better UX during outages

**Priority:** 🔴 P0 (Critical)
**Effort:** 2 hours
**Implementation:** This Week

### 11. Unhappy Path: Audio Quality Retry Flow

**Feedback:** "No retry prompt if assessment audio is poor quality"
**Status:** ❌ **NOT BUILT (Parent Assessment)**

#### Current Flow:

```
Parent records audio → Gemini analyzes → Shows result
(If audio is noisy/quiet/incomplete → Inaccurate result)
```

#### Required Fix:

```
typescript
```

```
// BEFORE Gemini analysis
const audioQuality = await checkAudioQuality(audioFile);

if (audioQuality.score < 0.6) {
  return {
    status: 'retry_needed',
    reason: audioQuality.issue, // 'background_noise', 'too_quiet', 'incomplete'
    message: getRetryMessage(audioQuality.issue)
  }
}

function getRetryMessage(issue) {
  switch(issue) {
    case 'background_noise':
      return "Oops! It was a bit noisy. Find a quieter spot and try again! 🤫";
    case 'too_quiet':
      return "We couldn't hear clearly. Hold the phone closer! 📱";
    case 'incomplete':
      return "The recording cut off. Please read the full passage! 📖";
  }
}

// Track retries
CREATE TABLE assessment_retries (
  id UUID PRIMARY KEY,
  child_id UUID,
  attempt_number INT,
  quality_score DECIMAL(3,2),
  retry_reason TEXT,
  attempted_at TIMESTAMPTZ
);

// UX: Allow 3 attempts, then accept with warning
1st Attempt: Quality = 0.4 → "Try again in quieter spot"
2nd Attempt: Quality = 0.5 → "Still noisy! Find a quiet corner"
3rd Attempt: Quality = 0.55 → Accept but flag in CRM
```

**Impact:** Better data quality, fewer "inaccurate assessment" complaints

**Priority:** 🔴 P0 (Critical)
**Effort:** 4 hours
**Implementation:** This Week

---

### 12. No-Show Cascade Policy

**Feedback:** "No auto-cancel logic if parent no-shows multiple sessions"
**Status:** ❌ **NOT BUILT**

**Current Behavior:** Undefined (manual intervention)

**Required Policy:**

```
1st No-Show: WhatsApp reminder
2nd No-Show: Coach calls parent (logged in CRM)
3rd No-Show: Enrollment flagged "At Risk" in admin
4th No-Show: Auto-pause + admin review
5+ No-Shows: Auto-cancel with prorated refund

Database Tracking:
ALTER TABLE enrollments ADD COLUMN consecutive_no_shows INT DEFAULT 0;
ALTER TABLE enrollments ADD COLUMN total_no_shows INT DEFAULT 0;
ALTER TABLE enrollments ADD COLUMN at_risk BOOLEAN DEFAULT FALSE;

Automation:
// After each no-show
if (enrollment.consecutive_no_shows >= 3) {
  enrollment.at_risk = true;
  await notifyAdmin('Churn risk', enrollment);
}
if (enrollment.total_no_shows >= 5) {
  await autoPauseEnrollment(enrollment.id);
  await triggerAdminReview(enrollment.id);
}
```

**Impact:** Clear expectations, automated churn prevention

**Priority:** 🟡 P2 (Medium)
**Effort:** 3 hours
**Implementation:** Week 3

---

### 13. Session Quality Tracking

**Feedback:** "No feedback loop to know which sessions were effective"
**Status:** ❌ **NOT BUILT**

**Current Gap:** No measurement of session effectiveness

**Required Addition:**

```
sql
```

```
CREATE TABLE session_effectiveness (
  id UUID PRIMARY KEY,
  session_id UUID REFERENCES scheduled_sessions(id),
  parent_rating INT CHECK (parent_rating BETWEEN 1 AND 5),
  child_engagement_score DECIMAL(3,2), -- From Gemini sentiment
  learning_outcome_met BOOLEAN,
  parent_comments TEXT,
  rated_at TIMESTAMPTZ
);

-- Ask parent after every 3rd session
POST /api/sessions/{id}/rate
{
  "rating": 4,
  "comments": "Great session! Aarav is more confident now"
}

-- Display in parent dashboard:
"On a scale of 1-5, how helpful was today's session?"
⭐⭐⭐⭐⭐
[Optional comments]
```

**Impact:** Coach performance insights, continuous improvement

**Priority:** 🟢 P3 (Low)
**Effort:** 6 hours
**Implementation:** Month 2

---

## 🎯 PRIORITIZED IMPLEMENTATION ROADMAP

### This Week (P0 - Critical)

**Total Effort:** 9 hours

| # | Task | Effort | Impact | Owner |
|---|---|---|---|---|
| 9 | Payment Reconciliation Cron | 3h | Zero revenue loss | Backend |
| 10 | Gemini Failure Fallback | 2h | Zero data loss | Backend |
| 11 | Audio Quality Retry Flow | 4h | Better assessment accuracy | Frontend + Backend |

**Goal:** Prevent customer-facing failures before marketing launch

---

### Week 2 (P1 - High Priority)

**Total Effort:** 10 hours

| # | Task | Effort | Impact | Owner |
|---|---|---|---|---|
| 7 | Breakthrough Human-in-Loop | 6h | Build parent trust | Backend + UI |
| 8 | Message Consolidation | 4h | Reduce alert fatigue | Communication |

**Goal:** Polish customer experience

---

### Week 3 (P2 - Medium Priority)

**Total Effort:** 19 hours

| # | Task | Effort | Impact | Owner |
|---|---|---|---|---|
| 5 | Weighted Coach Assignment | 12h | Higher conversion | Backend |
| 6 | Daily Coach Digest | 4h | Better session prep | Backend + Email |
| 12 | No-Show Cascade Policy | 3h | Automated churn handling | Backend |

**Goal:** Operational efficiency improvements

---

### Month 2 (P3 - Nice to Have)

**Total Effort:** 6 hours

| # | Task | Effort | Impact | Owner |
|---|---|---|---|---|
| 13 | Session Quality Tracking | 6h | Performance insights | Backend + UI |

**Goal:** Long-term improvement loop

---

## 📋 CRITICAL IMPLEMENTATION NOTES

**Before Starting Any Work:**

1. **Database Backups:** Ensure daily Supabase backups are configured
2. **Staging Environment:** Test all P0 items in staging first

3. **Error Monitoring:** Verify Sentry is capturing all errors
4. **Rate Limiting:** Confirm Upstash rate limiting is active

**Testing Checklist for Each Item:**

□ Unit tests written
□ Integration tests pass
□ Tested on mobile (Android + iOS)
□ Tested on poor network (4G throttled)
□ Error cases covered
□ Logging added for debugging
□ Documented in technical guide

**Deployment Strategy:**

**P0 Items:** Deploy individually with feature flags **P1-P2 Items:** Bundle in weekly releases **P3 Items:** Include in monthly updates

---

## 🚫 ITEMS EXPLICITLY NOT NEEDED

Based on chat history, these feedback items were misunderstandings or already handled:

1. **E-Learning Phonics First:** ✅ Acknowledged as future work (477-1,178 videos)
2. **Downsell to ₹999 Phonics:** ✅ Strategy agreed but deferred until content ready
3. **Coach Recruitment Deposit:** ❌ Not pursuing (trust-based model preferred)
4. **LLP Conversion:** ✅ Deferred until ₹20L revenue (staying sole proprietorship)

---

## 📊 SUMMARY METRICS

| Priority | Items | Total Effort | Target |
|---|---|---|---|
| 🔴 P0 (Critical) | 3 | 9 hours | This Week |
| 🟠 P1 (High) | 2 | 10 hours | Week 2 |
| 🟡 P2 (Medium) | 3 | 19 hours | Week 3 |
| 🟢 P3 (Low) | 1 | 6 hours | Month 2 |
| ✅ Already Done | 4 | - | Complete |
| **TOTAL** | **13** | **44 hours** | **3 weeks** |

---

## 🎯 NEXT SESSION AGENDA

**Session Focus:** P0 Critical Fixes

1. Review & approve this technical debt prioritization
2. Choose which P0 item to tackle first:
   - Payment Reconciliation Cron (backend-heavy)
   - Gemini Failure Fallback (quick win)
   - Audio Quality Retry Flow (frontend-heavy)
3. Set up staging environment if not exists
4. Begin implementation

**Question for Amit:** Which P0 item should we start with?

---

**Document Version:** 1.0
**Last Updated:** January 16, 2026
**Next Review:** After P0 completion