

Work Order 13 - Inspection Report

Existing Products & Membership Logic

Existing Whop Products (from lib/whop-client.ts)

Subscription Plans:

1. **BARRELS Athlete** - `prod_kNyobCww4tc2p`
 - Currently mapped to: "athlete" tier
 - No pricing defined in code yet
2. **BARRELS Pro** - `prod_04CB6y0IzNJLe`
 - Currently mapped to: "pro" tier
 - No pricing defined in code yet
3. **BARRELS Elite (The Inner Circle)** - `prod_vCV6UQH3K18QZ`
 - Currently mapped to: "elite" tier
 - No pricing defined in code yet
4. **The 90-Day Transformation** - `prod_zH1wnZs0JJKfd`
 - Currently mapped to: "elite" tier
 - One-time purchase

Assessment Products (Placeholders):

1. **Standard Assessment** - `prod_assessment_standard_299` (PLACEHOLDER)
 2. **Pro Assessment** - `prod_assessment_pro_399` (PLACEHOLDER)
-

Existing Membership Logic

1. `lib/tier-access.ts`

Defines tier hierarchy and feature gating:

- **Tiers:** `free` → `athlete` → `pro` → `elite`
- **Feature Access Map:** Maps features to required tiers (e.g., `'52_pitch_assessment': 'pro'`)
- **Functions:**
 - `hasTierAccess(userTier, requiredTier)` : Checks tier hierarchy
 - `checkTierAccess(requiredTier)` : Server-side access control
 - `getTierInfo(tier)` : Display names, colors, icons

2. `lib/assessment-vip-config.ts`

Manages VIP offers and session limits:

- **VIP Logic:**
 - `ASSESSMENT_VIP_DAYS = 60` : VIP offer window
 - Assessment purchases unlock 60-day VIP pricing
- **Session Limits** (per week):

```
typescript
  free: 0,
  athlete: 1,
```

```

    pro: 2,
    elite: 'unlimited'
- Functions:
- canStartNewSession(userTier, sessionsThisWeek) : Enforces weekly caps
- calculateVipExpiryDate() , isVipActive() , getVipDaysRemaining()

```

3. lib/whop-client.ts

Whop API integration:

- **Product Mapping:** getWhopProductTier(productId) maps Whop products to tiers
- **Membership Fetching:**
 - getWhopUserMemberships(whopUserId) : Get single user's memberships
 - getAllWhopMemberships() : Admin sync of all members
- **Assessment Detection:** isAssessmentProduct(productId) (uses placeholder IDs)

4. app/api/videos/upload/route.ts

Enforces session caps:

- Fetches user's membershipTier
- Counts sessions uploaded this week (Monday-Sunday)
- Calls canStartNewSession() to block uploads if cap reached
- **Exempts:** Admins and coaches

5. Database Schema (prisma/schema.prisma)

User model tracks:

- membershipTier : "free" | "athlete" | "pro" | "elite"
 - membershipStatus : "active" | "inactive" | "cancelled" | "expired"
 - membershipExpiresAt : Subscription end date
 - assessmentCompletedAt , assessmentVipExpiresAt , assessmentVipActive : VIP offer tracking
-

✓ What Already Works

1. **Tier Hierarchy:** free < athlete < pro < elite is defined
 2. **Session Limits:** Weekly caps are configured (0, 1, 2, unlimited)
 3. **Session Enforcement:** Upload API blocks users at weekly cap
 4. **Whop Sync:** Admin can sync membership data from Whop
 5. **VIP Offer System:** Assessment → 60-day VIP window logic exists
 6. **Database Schema:** All necessary fields exist
-

⚠ What Needs Work

1. **Pricing Not Defined:** No \$49/\$99/\$199 pricing in code
2. **Assessment Product IDs:** Placeholder IDs need real Whop product IDs
3. **Lessons vs Sessions Terminology:** Code says "sessions", work order says "lessons"
4. **No UI for Usage Display:** Dashboard doesn't show "X/Y lessons used this week"
5. **Assessment 30-Day Window:** Work order mentions 30-day assessment access, but code has 60-day VIP window (different concepts)
6. **No Monthly vs Weekly Clarification:** Work order says "1 lesson/week = 4 per billing month" but code only tracks weekly

🎯 Implementation Strategy

Phase 1: Clarify Business Rules

- **Question:** Are “lessons” and “sessions” the same? (Work order uses “lesson”, code uses “session”)
- **Question:** Assessment product is “one-time purchase with 30-day app access” - is this separate from the 60-day VIP offer?
- **Assumption for now:** 1 lesson = 1 analyzed video upload session

Phase 2: Update Configuration

- Add pricing constants (\$49, \$99, \$199) to config
- Create centralized `MEMBERSHIP_TIERS` config with all details
- Map weekly limits to monthly estimates for UI display

Phase 3: Update Upload Logic

- Ensure upload API correctly enforces limits
- Add clear error messages per tier

Phase 4: Add Dashboard UI

- Show membership tier card with:
- Plan name & price
- Weekly usage: “X / Y lessons used”
- Upgrade CTA when needed

Phase 5: Testing

- Create test users for each tier
- Verify session caps work
- Verify UI displays correctly

📝 Recommendations

1. **Use “Session” Consistently:** Code already uses “session”, recommend sticking with it (or mass-rename to “lesson”)
2. **Clarify Assessment Window:** Is it:
 - Option A: 30-day full app access + 60-day VIP pricing offer?
 - Option B: 30-day access window only?
3. **Add Pricing Config:** Create single source of truth for \$49/\$99/\$199
4. **Update Assessment Product IDs:** Get real Whop product IDs when available