🚀 DEPLOYMENT CHECKLIST - November 20, 2025

Phase 3.7: Grok Tracking + Pricing Fixes

✅ Pre-Deployment Checklist

- [x] **Grok pricing corrected** (10x error fixed)
- [x] **Contribution tracking added** to VerifierAgent
- [x] **Error logging** for failed verifications
- [x] **Post-synthesis tracking** included
- [x] **Code committed to GitHub** (commits: `61c84d1`, `b6773e7`)
- [x] **Code pushed to `main`** branch
- [x] **Documentation created** (GROK_TRACKING_FIX_NOV20.md)

🔄 Deployment Steps

1. Backend Deployment (Render)

- [ ] Go to: https://dashboard.render.com/
- [ ] Select: `vctt-agi-backend` service
- [ ] Click: **"Manual Deploy"** button
- [ ] Select: **"Clear build cache & deploy"**
- [ ] Wait: ~5-10 minutes for deployment
- [ ] Check logs for: `✅ Server successfully started`

2. Environment Variable Check

- [ ] Navigate to **Environment** tab on Render
- [ ] Verify: `XAI_API_KEY` is set
- [ ] If missing:
- [ ] Generate new key at: https://console.x.ai/
- [ ] Add to Render: `XAI_API_KEY=xai-...`
- [ ] Click **"Save"**
- [ ] Redeploy service

3. Frontend (Already Deployed)

- [x] Vercel auto-deployed from GitHub
- [x] Spinner component ready
- [x] WebSocket streaming active
- [ ] Hard refresh after backend deploy: Ctrl+Shift+R

🧪 Testing Protocol

Test 1: Grok Verification

1. [ ] Go to: https://vcttagi-kernal13-peters-projects-3a28ae0e.vercel.app
2. [ ] Ask: **"What is Lord Vishnu in Hinduism?"**
3. [ ] Expected results:
   - [ ] Spinner appears with phase progress

- [ ] Response includes verification badge
- [ ] Committee panel shows: `Grok-4/N(~17%)`
- [ ] Offline count: `0` or very low
- [ ] Sources listed at bottom

Test 2: Committee Stats

1. [ ] Check right panel: **"LLM Committee"**
2. [ ] Should see:
   - [ ] `GPT-5.29/12(~40%)`
   - [ ] `Grok-4/12(~17%)`
   - [ ] `Direct-Claude/12(~50%)`
   - [ ] Offline counts all low (0-2)

Test 3: DevTools Verification

1. [ ] Open Chrome DevTools (F12)
2. [ ] Go to **Console** tab
3. [ ] Look for logs:
   ```
   ✅ Verifier complete - model: grok-4
   🎸 Committee: Grok-grok-4 contribution recorded
   🍄 Mycelium grew by N verified facts
   ```
4. [ ] Go to **Network** tab → **WS** → Check WebSocket messages
5. [ ] Should see: `stream_phase` events during processing

Test 4: API Key Health Check

1. [ ] Run locally:
   ```bash
   curl https://api.x.ai/v1/chat/completions \
     -H "Content-Type: application/json" \
     -H "Authorization: Bearer YOUR_XAI_API_KEY" \
     -d '{"model": "grok-4", "messages": [{"role": "user", "content": "Test"}]}'
   ```
2. [ ] Should return: Valid JSON response
3. [ ] If 401/403: Regenerate API key

📊 Success Criteria

Badge + Committee Sync

- [ ] ✅ Badge shows: "Verified by Grok (N% confidence)"
- [ ] ✅ Committee panel shows: `Grok-4/N(>0%)`
- [ ] ✅ Offline count: `0` or minimal

Cost Tracking

- [ ] Grok costs: ~$0.0002-0.0005 per request
- [ ] Not showing 10x inflated costs
- [ ] Total session cost reasonable (<$0.10 for typical query)

Performance

- [ ] Response time: <30 seconds (with smart culling)
- [ ] Spinner shows all phases smoothly
- [ ] No hanging or timeout errors

Logs

- [ ] Backend logs show: `🎸 Committee: Grok contribution recorded`
- [ ] No errors like: `Grok API error (401)`, `XAI_API_KEY not set`
- [ ] Verification confidence logged correctly

🐛 Troubleshooting Guide

Issue: “Grok still offline in committee”

Diagnosis:

1. [ ] Check Render logs for Grok API errors
2. [ ] Verify `XAI_API_KEY` is set on Render
3. [ ] Test API key with curl command
4. [ ] Check xAI account has credits

Fix:

- [ ] Regenerate API key at console.x.ai
- [ ] Update Render environment variable
- [ ] Redeploy backend
- [ ] Test again

Issue: “Badge works, committee shows 0%”

Diagnosis:

1. [ ] Check database for `LLMContribution` records:

```sql
SELECT * FROM llm_contributions
WHERE model_name LIKE 'grok%'
ORDER BY timestamp DESC LIMIT 10;
```

2. [ ] Verify API endpoint:

```bash
curl https://backend-url/api/committee/session/SESSION_ID
```

Fix:

- [ ] Hard refresh frontend (Ctrl+Shift+R)
- [ ] Clear browser cache
- [ ] Check network tab for API call failures
- [ ] Verify backend logs show contribution recording

Issue: “Spinner not appearing”

Diagnosis:

1. [ ] Check WebSocket connection (DevTools → Network → WS)
2. [ ] Verify phase events are being emitted
3. [ ] Check console for JavaScript errors

Fix:

- [ ] Confirm backend is emitting `stream_phase` events
- [ ] Hard refresh frontend
- [ ] Check frontend logs for connection errors

Issue: “Costs too high”

Verification:

1. [ ] Check `llm.config.ts`:

```typescript
```

```
  'grok-4': {
    inputPer1k: 0.0002,  // Should be 0.0002, not 0.002
    outputPer1k: 0.0005, // Should be 0.0005, not 0.010
  }
```

Fix:
- [ ] Verify latest code is deployed
- [ ] Check commit `61c84d1` is in production
- [ ] Redeploy if pricing not updated

📝 Post-Deployment Tasks

Immediate (Within 1 Hour)

- [ ] Deploy backend to Render
- [ ] Run all 4 test protocols
- [ ] Verify badge + committee sync
- [ ] Check logs for contribution recording
- [ ] Confirm costs are correct (10x fix applied)

Short-term (Within 24 Hours)

- [ ] Monitor Grok offline count (should stay low)
- [ ] Track average verification latency
- [ ] Review total session costs
- [ ] Gather user feedback on spinner UX
- [ ] Check truth mycelium growth rate

Medium-term (This Week)

- [ ] Analyze Grok contribution percentage (should be ~15-20%)
- [ ] Compare Grok vs GPT accuracy on sample queries
- [ ] Optimize verification prompts if needed
- [ ] Consider upgrading to Grok 4.1 (if SuperGrok subscription)
- [ ] Document any recurring issues

🎯 Next Phase Planning

Phase 3.8 (If Everything Works):

- [ ] Implement caching layer for common facts
- [ ] Add agent culling optimization
- [ ] Consider Direct Claude integration
- [ ] Optimize database queries for committee stats
- [ ] Add more detailed cost breakdowns in UI

Phase 4 (Production Hardening):

- [ ] Add rate limiting for LLM calls
- [ ] Implement fallback chains (Grok → GPT → Claude)
- [ ] Set up monitoring alerts for offline models
- [ ] Add cost threshold warnings
- [ ] Implement A/B testing for model selection

📞 Support Resources

- **xAI Console:** https://console.x.ai/
- **Render Dashboard:** https://dashboard.render.com/
- **Vercel Dashboard:** https://vercel.com/dashboard
- **GitHub Repo:** https://github.com/Counterbalance-Economics/vctt-agi-engine
- **Backend URL:** https://vctt-agi-backend.onrender.com
- **Frontend URL:** https://vcttagi-kernal13-peters-projects-3a28ae0e.vercel.app

🎉 Success Indicators

When deployment is successful, you should see:

1. ✅ **Backend Logs:**
   ```
   🥁 Verifier (Grok) starting fact-check...
      ✅ Verifier complete - confidence: 0.98, model: grok-4
      🎸 Committee: Grok-grok-4 contribution recorded
      🍄 Mycelium grew by 3 verified facts
   ```

2. ✅ **Frontend Committee Panel:**
   ```
   LLM Committee
      12 questions
      GPT-5.29/12(42%)
      Grok-4/12(17%)            [0 offline]
      Direct-Claude/12(58%)     [1 offline]
   ```

3. ✅ **User Experience:**
   - Spinner shows phases: 🎬 → 🎸 → 🎺 → 🎻 → 🥁 → ✅
   - Response includes verification badge
   - Sources listed at bottom
   - Total time: 20-30 seconds
   - Cost per query: $0.01-0.05

📊 Status

Current State:
- ✅ Code ready and pushed to GitHub
- ✅ Documentation complete
- ⏳ Awaiting Render deployment
- ⏳ Awaiting testing verification

Ready to deploy! 🚀

Last Updated: November 20, 2025
Latest Commit: `b6773e7` - “Fix: Add Grok contribution tracking to LLM Committee”
Deployed to: Pending