Demo Environment Setup Guide

PRE-DEMO CHECKLIST

Technical Setup

- [ ] Test all testnet faucets 24 hours before demo
- [ ] Ensure test wallets have sufficient funds on all chains
- [ ] Pre-deploy all smart contracts to testnets
- [ ] Configure Chainlink oracle feeds for testnets
- [ ] Test CCIP lane availability between chains
- [ ] Set up monitoring dashboard for real-time metrics
- [ ] Prepare backup RPC endpoints for each chain

Demo Data Preparation

- [ ] Create 3 demo accounts with different portfolio sizes
- [ ] Pre-populate historical yield data
- [ ] Set up realistic APY variations across protocols
- [ ] Configure gas price simulators
- [ ] Prepare transaction history for each account

PRIMARY DEMO FLOW

Account Setup

```
Demo Account 1 (Primary - "Sarah"):
- Address: 0x1234...5678
- Portfolio: $100,000 equivalent
- Current positions:
  - ETH on Ethereum: $40,000 @ 4% APY
  - USDC on Polygon: $30,000 @ 8% APY
  - WBTC on Arbitrum: $30,000 @ 6% APY
- Target optimization: 28.4% APY blended
```

Pre-staged Transactions

1. **Ethereum → Arbitrum Bridge**
   - Pre-approved USDC amount: $40,000
   - Gas estimation: 0.003 ETH
   - Expected confirmation: 15 seconds
2. **Arbitrum Aave Deposit**
   - Pre-approved for Aave protocol
   - Expected APY: 24.2%
   - Confirmation time: 3 seconds
3. **Polygon Curve Pool Entry**
   - Pre-approved LP tokens
   - Expected APY: 32.6%
   - Confirmation time: 2 seconds

BACKUP SCENARIOS

Scenario A: Network Congestion

Trigger: Any transaction takes >30 seconds **Response**:

```
"Looks like [Network] is experiencing high traffic. This is exactly why YieldMax
monitors gas prices across all chains and routes through the most efficient path.
Let me show you our fallback route..."
```

Action: Switch to pre-recorded segment showing successful transaction

Scenario B: Oracle Feed Issues

Trigger: Chainlink price feed not updating **Response**:

```
"Notice how YieldMax detected the stale oracle data? Our redundancy system kicks
in with alternative price sources while maintaining security. This is the
difference between a hackathon project and production-ready infrastructure."
```

Action: Use cached price data with "Safe Mode" indicator

Scenario C: Bridge Failure

Trigger: CCIP message not confirming **Response**:

```
"This demonstrates why we built multiple bridge integrations. YieldMax automatically
failovers to our secondary bridge route. Users never experience downtime."
```

Action: Execute pre-staged alternative bridge transaction

Scenario D: Complete Technical Failure

Trigger: Multiple system failures **Response**:

```
"Let me show you a recent live execution from our production environment. This
demonstrates the exact same flow with real user funds..."
```

Action: Play backup video of successful mainnet transaction

DEMO ENVIRONMENT VARIABLES

```bash
# Primary Configuration
DEMO_MODE=true
FAST_BLOCK_TIME=true
MOCK_GAS_PRICES=true
FORCE_SUCCESS_RATE=95

# Testnet RPCs (with fallbacks)
ETH_RPC_PRIMARY=https://eth-sepolia.g.alchemy.com/v2/[KEY]
ETH_RPC_BACKUP=https://sepolia.infura.io/v3/[KEY]
ARB_RPC_PRIMARY=https://arb-sepolia.g.alchemy.com/v2/[KEY]
POLYGON_RPC_PRIMARY=https://polygon-mumbai.g.alchemy.com/v2/[KEY]

# Chainlink Addresses (Testnet)
PRICE_FEED_ETH=0xD4a33860578De61DBAbDc8BFdb98FD742fA7028e
PRICE_FEED_USDC=0xA2F78ab2355fe2f984D808B5CeE7FD0A93D5270E
CCIP_ROUTER_ETH=0x0BF3dE8c5D3e8A2B34D2BEeB17ABfCeBaf363A59
AUTOMATION_REGISTRY=0x86EFBD0b6736Bed994962f9797049422A3A8E8Ad

# Demo Wallet Keys (Testnet Only!)
DEMO_WALLET_1=0x[TESTNET_PRIVATE_KEY_1]
DEMO_WALLET_2=0x[TESTNET_PRIVATE_KEY_2]
```

VISUAL ELEMENTS PREPARATION

Screen Layouts

1. **Dashboard View**: Clean, uncluttered, focus on key metrics

2. **Transaction View**: Clear progress indicators, gas estimates
3. **Analytics View**: Real-time charts, yield comparisons
4. **Settings View**: Show advanced features without overwhelming

Color Coding

- Green: Profitable opportunities
- Yellow: Moderate yields
- Red: Underperforming positions
- Blue: Chainlink integrations
- Purple: AI/ML predictions

REHEARSAL SCHEDULE

T-48 Hours

- Full run-through with primary flow
- Test all backup scenarios
- Record backup video segments

T-24 Hours

- Final testnet fund distribution
- Verify all oracle feeds active
- Test screen recording software
- Practice transitions between segments

T-2 Hours

- Final system check

- Close all unnecessary applications
- Set up backup laptop with identical environment
- Brief any team members on backup procedures

T-30 Minutes

- Fresh restart of all systems
- Load demo environment
- Test microphone and screen recording
- Deep breath, you've got this!