

YTEmpire Week 0 Execution Plan

Executive Leadership

Role: CEO/Founder

Task 1: Strategic Vision Alignment & Team Kickoff

Description: Conduct comprehensive project kickoff meeting establishing vision, success metrics, and team alignment for the MVP. **Steps:**

1. Prepare vision deck with 90-day targets and success criteria (2 hours)
2. Schedule and conduct all-hands kickoff meeting (2 hours)
3. Document key decisions and action items in shared workspace
4. Establish weekly leadership sync cadence **Duration:** 4 hours **Dependencies:** None **Deliverable:** Vision document, recorded kickoff, meeting cadence established **Priority:** P0

Task 2: Resource Allocation & Budget Approval

Description: Finalize budget allocation across teams and approve critical infrastructure purchases. **Steps:**

1. Review and approve \$200K budget breakdown by department
2. Authorize hardware purchase (Ryzen 9 9950X3D system)
3. Approve API service subscriptions (OpenAI, ElevenLabs, YouTube)
4. Set up financial tracking dashboard **Duration:** 3 hours **Dependencies:** Budget proposals from CTO and VP of AI **Deliverable:** Approved budget sheet, purchase orders initiated **Priority:** P0

Task 3: Beta User Recruitment Strategy

Description: Define and initiate beta user acquisition strategy targeting 10 initial users. **Steps:**

1. Define ideal beta user profile (digital entrepreneurs with \$2-5K budget)
2. Create outreach strategy and messaging
3. Set up beta application form and screening process
4. Initiate first outreach to 5 potential beta users **Duration:** 4 hours **Dependencies:** Product Owner's feature list **Deliverable:** Beta recruitment plan, application form live **Priority:** P1

Role: Product Owner

Task 1: MVP Feature Prioritization Matrix

Description: Create detailed feature priority matrix based on user value and technical feasibility. **Steps:**

1. List all proposed features with user stories

2. Score each feature on value (1-10) and effort (1-10)
3. Create priority matrix visualization
4. Get stakeholder agreement on Week 1-4 features **Duration:** 4 hours **Dependencies:** Technical feasibility input from CTO **Deliverable:** Feature priority matrix with sprint allocation **Priority:** P0

Task 2: Success Metrics Definition

Description: Define measurable success criteria for MVP including user, technical, and business metrics.

Steps:

1. Define primary KPIs (videos/day, cost/video, automation %)
2. Set up tracking methodology for each metric
3. Create dashboard mockup for metrics visualization
4. Document metric calculation formulas **Duration:** 3 hours **Dependencies:** None **Deliverable:** KPI document with tracking plan **Priority:** P0

Task 3: User Journey Mapping

Description: Map complete user journey from signup to first automated video generation. **Steps:**

1. Document step-by-step user flow with decision points
2. Identify friction points and automation opportunities
3. Create wireframe sketches for critical screens
4. Review with UX designer and frontend lead **Duration:** 4 hours **Dependencies:** None **Deliverable:** User journey map with wireframes **Priority:** P1

Technical Leadership

Role: CTO/Technical Director

Task 1: Technical Architecture Documentation

Description: Create comprehensive technical architecture document defining system components, data flow, and integration points. **Steps:**

1. Design high-level system architecture diagram
2. Define microservices boundaries and APIs
3. Document data flow between components
4. Specify technology choices with justifications **Duration:** 4 hours **Dependencies:** Input from VP of AI on ML architecture **Deliverable:** Technical architecture document v1.0 **Priority:** P0

Task 2: Development Environment Standardization

Description: Set up and document standardized development environment for all engineers. **Steps:**

1. Create Docker Compose configuration for local development
2. Set up GitHub repository structure with branch protection
3. Configure VS Code with recommended extensions and settings
4. Document setup process in README **Duration:** 3 hours **Dependencies:** None **Deliverable:** Docker configs, GitHub repos, setup documentation **Priority:** P0

Task 3: CI/CD Pipeline Foundation

Description: Establish basic CI/CD pipeline with GitHub Actions for automated testing and deployment.

Steps:

1. Set up GitHub Actions workflow for automated tests
2. Configure Docker Hub for image registry
3. Create deployment scripts for staging environment
4. Test pipeline with hello-world application **Duration:** 4 hours **Dependencies:** GitHub repository setup **Deliverable:** Working CI/CD pipeline with test deployment **Priority:** P1

Task 4: Security Baseline Implementation

Description: Establish security best practices and initial configurations for the platform. **Steps:**

1. Set up secrets management using environment variables
2. Configure HTTPS with Let's Encrypt
3. Implement basic firewall rules
4. Document security checklist for code reviews **Duration:** 3 hours **Dependencies:** Server access from Platform Ops **Deliverable:** Security configuration, documented best practices **Priority:** P1

Role: VP of AI

Task 1: AI Infrastructure Planning

Description: Design AI/ML infrastructure architecture for model serving and training pipelines. **Steps:**

1. Define model serving architecture (API endpoints, caching)
2. Plan GPU resource allocation strategy
3. Design model versioning and rollback system
4. Document latency and throughput requirements **Duration:** 4 hours **Dependencies:** Hardware specifications from Platform Ops **Deliverable:** AI infrastructure design document **Priority:** P0

Task 2: API Cost Optimization Strategy

Description: Develop comprehensive strategy to achieve <\$3/video cost target through intelligent API usage. **Steps:**

1. Analyze API pricing tiers (OpenAI, ElevenLabs, Google TTS)
2. Design caching strategy for common requests
3. Plan fallback chains (GPT-4 → GPT-3.5 → local models)
4. Create cost tracking framework **Duration:** 3 hours **Dependencies:** None **Deliverable:** Cost optimization strategy document **Priority:** P0

Task 3: Initial Prompt Engineering Framework

Description: Establish prompt templates and testing framework for consistent AI outputs. **Steps:**

1. Create base prompt templates for script generation
2. Design prompt versioning system
3. Set up A/B testing framework for prompt optimization
4. Document prompt engineering best practices **Duration:** 4 hours **Dependencies:** None **Deliverable:** Prompt library, testing framework **Priority:** P1

Backend Team

Role: Backend Team Lead

Task 1: API Architecture Design

Description: Design RESTful API architecture with clear endpoints, authentication, and rate limiting.

Steps:

1. Define API endpoint structure following REST principles
2. Design JWT-based authentication flow
3. Plan rate limiting and quota management
4. Create OpenAPI specification draft **Duration:** 4 hours **Dependencies:** Technical architecture from CTO **Deliverable:** API design document with OpenAPI spec **Priority:** P0

Task 2: Database Schema Design

Description: Design PostgreSQL database schema for users, channels, videos, and analytics. **Steps:**

1. Create ERD with all entities and relationships
2. Define indexes for query optimization
3. Plan data partitioning strategy for scale
4. Write initial migration scripts **Duration:** 4 hours **Dependencies:** Feature requirements from Product Owner **Deliverable:** Database schema, ERD, migration scripts **Priority:** P0

Task 3: Development Environment Setup

Description: Set up local development environment with FastAPI, PostgreSQL, and Redis. **Steps:**

1. Create FastAPI project structure
2. Configure PostgreSQL and Redis connections
3. Set up Alembic for database migrations
4. Create docker-compose.yml for team **Duration:** 3 hours **Dependencies:** Docker environment from CTO **Deliverable:** Working backend development environment **Priority:** P0

Role: API Developer Engineer

Task 1: Authentication Service Scaffolding

Description: Implement basic authentication service with JWT tokens and user registration. **Steps:**

1. Create user registration endpoint
2. Implement JWT token generation
3. Add token refresh mechanism
4. Create basic user profile endpoints **Duration:** 4 hours **Dependencies:** Database schema from Backend Lead **Deliverable:** Working authentication endpoints **Priority:** P1

Task 2: Base API Framework Setup

Description: Set up FastAPI application structure with middleware, error handling, and logging. **Steps:**

1. Configure FastAPI with CORS middleware
2. Implement global error handlers
3. Set up structured logging with correlation IDs
4. Create health check endpoints **Duration:** 3 hours **Dependencies:** API architecture design **Deliverable:** Base FastAPI application **Priority:** P1

Role: Data Pipeline Engineer

Task 1: Message Queue Infrastructure

Description: Set up Redis-based queue system for video processing jobs. **Steps:**

1. Configure Redis for persistent queues
2. Create Celery worker configuration
3. Design job priority system
4. Implement basic job monitoring **Duration:** 4 hours **Dependencies:** Redis setup from Platform Ops **Deliverable:** Working job queue system **Priority:** P1

Task 2: Data Flow Architecture

Description: Design data pipeline architecture for video generation workflow. **Steps:**

1. Map data flow from request to video generation
2. Define queue topics and routing
3. Plan error handling and retry logic
4. Document pipeline monitoring points **Duration:** 3 hours **Dependencies:** System architecture from CTO **Deliverable:** Data pipeline design document **Priority:** P1

Role: Integration Specialist

Task 1: YouTube API Integration Planning

Description: Research and plan YouTube Data API v3 integration with quota management. **Steps:**

1. Study YouTube API quotas and limitations
2. Design 15-account rotation system
3. Plan quota monitoring and alerting
4. Create API client wrapper design **Duration:** 4 hours **Dependencies:** None **Deliverable:** YouTube API integration plan **Priority:** P0

Task 2: N8N Workflow Setup

Description: Install and configure N8N for workflow automation. **Steps:**

1. Deploy N8N using Docker
2. Configure webhook endpoints
3. Create test workflow for video pipeline
4. Document workflow creation process **Duration:** 3 hours **Dependencies:** Docker environment from Platform Ops **Deliverable:** Working N8N instance with test workflow **Priority:** P1

Frontend Team

Role: Frontend Team Lead

Task 1: Frontend Architecture Design

Description: Design React application architecture with state management and routing. **Steps:**

1. Define component hierarchy and structure
2. Choose and configure state management (Zustand)
3. Plan routing structure with React Router

4. Design API integration layer **Duration:** 3 hours **Dependencies:** API design from Backend Lead
Deliverable: Frontend architecture document **Priority:** P0

Task 2: Development Environment Setup

Description: Set up React development environment with TypeScript and Material-UI. **Steps:**

1. Initialize React project with Vite
2. Configure TypeScript and ESLint
3. Install and configure Material-UI
4. Set up hot module replacement **Duration:** 3 hours **Dependencies:** None **Deliverable:** Working frontend development environment **Priority:** P0

Role: React Engineer

Task 1: Component Library Foundation

Description: Create base component library with Material-UI theming. **Steps:**

1. Set up Material-UI theme configuration
2. Create base components (Button, Input, Card)
3. Implement loading and error states
4. Document component usage **Duration:** 4 hours **Dependencies:** Frontend environment setup
Deliverable: Base component library **Priority:** P1

Task 2: Authentication UI Components

Description: Build login, register, and forgot password UI components. **Steps:**

1. Create login form with validation
2. Build registration flow UI
3. Implement password reset interface
4. Add JWT token management **Duration:** 4 hours **Dependencies:** Component library foundation
Deliverable: Authentication UI components **Priority:** P2

Role: Dashboard Specialist

Task 1: Dashboard Layout Design

Description: Design and implement base dashboard layout with navigation. **Steps:**

1. Create responsive dashboard shell
2. Implement sidebar navigation
3. Add header with user menu

4. Set up routing for main sections **Duration:** 4 hours **Dependencies:** Component library from React Engineer **Deliverable:** Dashboard layout component **Priority:** P1

Task 2: Data Visualization Planning

Description: Research and plan implementation of charts using Recharts. **Steps:**

1. Evaluate Recharts capabilities
2. Design chart components architecture
3. Create mock data for testing
4. Build proof-of-concept chart **Duration:** 3 hours **Dependencies:** None **Deliverable:** Visualization plan with POC **Priority:** P2

Role: UI/UX Designer

Task 1: Design System Creation

Description: Create comprehensive design system with colors, typography, and spacing. **Steps:**

1. Define color palette and usage guidelines
2. Set typography scale and hierarchy
3. Create spacing and layout grid system
4. Document in Figma with examples **Duration:** 4 hours **Dependencies:** Brand guidelines from CEO **Deliverable:** Design system in Figma **Priority:** P0

Task 2: Dashboard Wireframes

Description: Create low-fidelity wireframes for main dashboard views. **Steps:**

1. Sketch dashboard overview layout
2. Design channel management interface
3. Create video queue visualization
4. Design analytics displays **Duration:** 4 hours **Dependencies:** User journey from Product Owner **Deliverable:** Dashboard wireframes in Figma **Priority:** P1

Platform Operations Team

Role: Platform Ops Lead

Task 1: Infrastructure Planning & Setup

Description: Plan and initiate setup of local server infrastructure. **Steps:**

1. Verify hardware specifications and order confirmation
2. Plan network topology and security zones

3. Create infrastructure setup checklist
4. Coordinate with ISP for fiber installation **Duration:** 3 hours **Dependencies:** Budget approval from CEO **Deliverable:** Infrastructure plan and setup timeline **Priority:** P0

Task 2: Team Tooling Setup

Description: Set up essential DevOps tools and team access. **Steps:**

1. Create GitHub organization and team permissions
2. Set up Slack workspace with channels
3. Configure password manager for team
4. Set up documentation wiki (Confluence/Notion) **Duration:** 3 hours **Dependencies:** Team member list from CEO **Deliverable:** All team tools operational **Priority:** P0

Role: DevOps Engineer

Task 1: Docker Environment Setup

Description: Create Docker and Docker Compose configurations for all services. **Steps:**

1. Install Docker and Docker Compose on dev servers
2. Create base Dockerfiles for each service
3. Configure Docker networks and volumes
4. Test container orchestration locally **Duration:** 4 hours **Dependencies:** Server access from Platform Ops Lead **Deliverable:** Docker environment ready **Priority:** P0

Task 2: Monitoring Stack Foundation

Description: Set up basic monitoring with Prometheus and Grafana. **Steps:**

1. Deploy Prometheus with Docker
2. Configure Grafana dashboards
3. Set up basic system metrics collection
4. Create alerting rules template **Duration:** 4 hours **Dependencies:** Docker environment **Deliverable:** Basic monitoring operational **Priority:** P1

Role: Security Engineer

Task 1: Security Baseline Configuration

Description: Implement initial security configurations for infrastructure. **Steps:**

1. Configure UFW firewall rules
2. Set up fail2ban for SSH protection

3. Implement SSH key-only authentication
4. Create security checklist document **Duration:** 3 hours **Dependencies:** Server access **Deliverable:** Secured server environment **Priority:** P0

Task 2: Secrets Management Planning

Description: Design secure secrets management strategy. **Steps:**

1. Research secrets management solutions
2. Design environment variable strategy
3. Plan API key rotation procedures
4. Document secrets handling best practices **Duration:** 3 hours **Dependencies:** None **Deliverable:** Secrets management plan **Priority:** P1

Role: QA Engineer

Task 1: Test Framework Selection

Description: Evaluate and select testing frameworks for different layers. **Steps:**

1. Research testing tools (Jest, Pytest, Selenium)
2. Create comparison matrix
3. Set up proof-of-concept tests
4. Document testing strategy **Duration:** 4 hours **Dependencies:** Tech stack decisions from CTO
Deliverable: Testing framework recommendations **Priority:** P1

Task 2: Test Environment Planning

Description: Design test environment architecture and data management. **Steps:**

1. Plan test environment isolation
2. Design test data generation strategy
3. Create test environment setup scripts
4. Document test environment access **Duration:** 3 hours **Dependencies:** Infrastructure plan
Deliverable: Test environment design **Priority:** P2

AI Team

Role: AI/ML Team Lead

Task 1: ML Pipeline Architecture

Description: Design end-to-end ML pipeline for content generation. **Steps:**

1. Map ML model dependencies and data flow

2. Design model serving architecture
3. Plan model versioning and rollback
4. Define performance SLAs for each model **Duration:** 4 hours **Dependencies:** System architecture from VP of AI **Deliverable:** ML pipeline design document **Priority:** P0

Task 2: Model Evaluation Framework

Description: Establish framework for evaluating model performance and quality. **Steps:**

1. Define quality metrics for each model type
2. Create evaluation dataset structure
3. Design A/B testing methodology
4. Set up model performance tracking **Duration:** 3 hours **Dependencies:** None **Deliverable:** Model evaluation framework **Priority:** P1

Role: ML Engineer

Task 1: OpenAI API Integration Setup

Description: Set up OpenAI API integration with rate limiting and error handling. **Steps:**

1. Configure OpenAI Python SDK
2. Implement rate limiting wrapper
3. Add retry logic with exponential backoff
4. Create cost tracking hooks **Duration:** 4 hours **Dependencies:** API keys from VP of AI **Deliverable:** OpenAI integration module **Priority:** P0

Task 2: Local Model Environment

Description: Set up environment for running local ML models as fallbacks. **Steps:**

1. Configure PyTorch with CUDA support
2. Download and test Llama 2 7B model
3. Create model loading utilities
4. Benchmark inference performance **Duration:** 4 hours **Dependencies:** GPU drivers from Platform Ops **Deliverable:** Local model inference setup **Priority:** P1

Role: Data Team Lead

Task 1: Data Schema Design

Description: Design data schema for ML training and analytics. **Steps:**

1. Define feature store schema

2. Design training data structure
3. Plan data versioning strategy
4. Create data quality rules **Duration:** 4 hours **Dependencies:** Database schema from Backend Lead
Deliverable: ML data schema document **Priority:** P0

Task 2: Analytics Pipeline Planning

Description: Plan analytics data pipeline for business metrics. **Steps:**

1. Identify key metrics to track
2. Design ETL pipeline architecture
3. Plan data warehouse structure
4. Define data retention policies **Duration:** 3 hours **Dependencies:** KPIs from Product Owner
Deliverable: Analytics pipeline design **Priority:** P1

Role: Data Engineer

Task 1: Data Collection Infrastructure

Description: Set up infrastructure for collecting training and analytics data. **Steps:**

1. Create data ingestion endpoints
2. Set up data validation pipeline
3. Implement data storage with partitioning
4. Create data backup procedures **Duration:** 4 hours **Dependencies:** Data schema from Data Team Lead
Deliverable: Data collection system **Priority:** P1

Task 2: Feature Engineering Pipeline

Description: Build initial feature engineering pipeline for ML models. **Steps:**

1. Implement feature extraction functions
2. Create feature transformation pipeline
3. Set up feature store connections
4. Document feature definitions **Duration:** 4 hours **Dependencies:** ML pipeline architecture
Deliverable: Feature engineering code **Priority:** P2

Role: Analytics Engineer

Task 1: Metrics Collection Setup

Description: Implement metrics collection for cost and performance tracking. **Steps:**

1. Create metrics collection endpoints

2. Implement cost calculation logic
3. Set up time-series storage
4. Create basic metrics API **Duration:** 4 hours **Dependencies:** Database setup **Deliverable:** Metrics collection system **Priority:** P1

Task 2: Dashboard Data Preparation

Description: Prepare data models for dashboard visualizations. **Steps:**

1. Design aggregation queries
2. Create materialized views for performance
3. Implement caching strategy
4. Document data refresh schedules **Duration:** 3 hours **Dependencies:** Dashboard requirements from Frontend **Deliverable:** Dashboard data models **Priority:** P2

Daily Standup Schedule

Day 1 (Monday)

- **9:00 AM:** All-hands kickoff (CEO)
- **11:00 AM:** Technical architecture review (CTO, VP of AI)
- **2:00 PM:** Infrastructure setup begins (Platform Ops)
- **3:00 PM:** API design session (Backend Team)

Day 2 (Tuesday)

- **9:00 AM:** Team standups begin
- **10:00 AM:** Frontend architecture review
- **2:00 PM:** ML pipeline design review
- **4:00 PM:** Security baseline implementation

Day 3 (Wednesday)

- **9:00 AM:** Team standups
- **10:00 AM:** Integration planning session
- **2:00 PM:** Database schema review
- **4:00 PM:** Cost optimization workshop

Day 4 (Thursday)

- **9:00 AM:** Team standups
- **10:00 AM:** CI/CD pipeline setup

- **2:00 PM:** Testing framework decisions
- **4:00 PM:** API integration reviews

Day 5 (Friday)

- **9:00 AM:** Team standups
- **10:00 AM:** Week 0 retrospective
- **2:00 PM:** Week 1 planning session
- **4:00 PM:** Demo of working components

Success Criteria for Week 0

Must Have (P0)

- ☐ Development environment operational for all teams
- ☐ GitHub repositories created with CI/CD pipeline
- ☐ Database schema designed and reviewed
- ☐ API architecture documented
- ☐ Security baseline implemented
- ☐ Docker environment configured
- ☐ Budget approved and resources ordered

Should Have (P1)

- ☐ Authentication service scaffolding complete
- ☐ Frontend component library started
- ☐ N8N workflow engine deployed
- ☐ Monitoring stack operational
- ☐ ML pipeline architecture defined
- ☐ YouTube API integration planned

Nice to Have (P2)

- ☐ Dashboard wireframes complete
- ☐ Test frameworks selected
- ☐ Feature engineering pipeline started
- ☐ Analytics data models designed

Risk Register

High Priority Risks

1. **Hardware Delivery Delay:** Mitigation - Use cloud resources temporarily
2. **API Quota Limitations:** Mitigation - Implement caching from day 1

3. **Team Onboarding Delays:** Mitigation - Pair programming and documentation

Medium Priority Risks

1. **Technology Integration Issues:** Mitigation - Proof of concept for each integration
2. **Cost Overruns:** Mitigation - Daily cost tracking from day 1
3. **Scope Creep:** Mitigation - Strict adherence to MVP features

Communication Protocols

Slack Channels

- #general - Company-wide announcements
- #dev-backend - Backend team coordination
- #dev-frontend - Frontend team coordination
- #dev-ai - AI/ML team coordination
- #platform-ops - Infrastructure and DevOps
- #leadership - CEO, CTO, VP of AI, Product Owner
- #standup - Daily standup notes
- #blockers - Urgent blocking issues
- #wins - Celebrate victories

Meeting Cadence

- Daily: 9:00 AM standup (15 minutes)
- Monday: Leadership sync (1 hour)
- Wednesday: Technical architecture review (1 hour)
- Friday: Sprint demo and retrospective (2 hours)

Documentation

- GitHub Wiki: Technical documentation
- Confluence/Notion: Process documentation
- Google Drive: Business documents
- Figma: Design files

Tools and Access Checklist

Development Tools

- ☐ GitHub access for all developers
- ☐ Docker Desktop installed

- ☐ VS Code with extensions
- ☐ Postman for API testing
- ☐ pgAdmin for database management

Communication Tools

- ☐ Slack access for all team members
- ☐ Google Workspace accounts
- ☐ Zoom for video calls
- ☐ Figma for design collaboration

Infrastructure Access

- ☐ SSH keys distributed
- ☐ VPN configuration (if needed)
- ☐ AWS/Cloud console access
- ☐ Monitoring dashboard access

Week 1 Handoff Checklist

By end of Week 0, ensure:

- ☐ All P0 tasks completed
- ☐ 80% of P1 tasks completed
- ☐ Development environments verified
- ☐ Team can start coding Monday Week 1
- ☐ All blockers identified and resolved
- ☐ Week 1 sprint planned and assigned
- ☐ Success metrics tracking initiated

Document Version: 1.0

Last Updated: Week 0, Day 1

Next Review: Week 0, Day 5

Owner: CTO/Technical Director