**Detailed Planning**

**Prepared by:** Gia Huy Ly

**I. PROJECT OVERVIEW**

**Project Name:** NexPath  
**Objective:**  
Develop an intelligent software platform for LFAM (large-format additive manufacturing) that integrates AI-assisted toolpath generation, thermal simulation, real-time process monitoring, and hybrid manufacturing support. The system is intended for industrial-scale 3D printing environments.

**Business Model Options:**

* SaaS subscription
* OEM licensing
* Enterprise on-premise deployment
* Custom enterprise solutions

**II. CORE MODULES & ARCHITECTURE**

| **Module** | **Description** | **Technologies** |
| --- | --- | --- |
| **1. UI/UX Layer** | Cross-platform graphical interface for designers, engineers, and operators. | ReactJS, TailwindCSS, ElectronJS (for desktop), Next.js (if web extension is needed) |
| **2. AI Copilot (Toolpath Optimization)** | Automates slicing and G-code generation using AI models trained on geometric and mechanical datasets. | Python, PyTorch or TensorFlow, Scikit-learn, ONNX Runtime |
| **3. Thermal Simulation Engine** | Simulates thermal behavior during deposition to prevent warping, delamination, etc. | C++ or Rust for performance; integrates with a Python frontend; Finite Element Libraries (e.g., deal.II, FEniCS) |
| **4. Toolpath Generator (Traditional)** | Converts 3D models into standard toolpaths (pre-AI layer), includes infill control, layer height, bridging, etc. | Python, CGAL or OpenCASCADE, custom slicer logic |
| **5. Hybrid Manufacturing Module** | Supports additive-subtractive manufacturing coordination, includes CAM integration. | Post-processing via CNC toolpaths; integrates with Fusion 360 API or custom G-code generation |
| **6. Real-Time Monitoring and Feedback** | Captures thermal camera data, deposition rate, material usage, etc. | OpenCV, MQTT or ZeroMQ, C++/Python mix, optional Jetson or edge devices |
| **7. Cloud Backend & Workflow Manager** | Orchestrates tasks, manages print queues, user roles, and resource allocation | Node.js (backend), PostgreSQL, Redis, Kubernetes for scaling |
| **8. Plugin API System** | Allow third-party tools and manufacturers to build custom modules. | REST API, WebAssembly sandbox for plugins, gRPC for high-performance I/O |
| **9. Authentication & Licensing** | License management for SaaS and on-premise options | OAuth 2.0, JWT, Stripe or FastSpring for billing, License server with RSA signing |

**III. DEVELOPMENT STACK OVERVIEW**

**Frontend/UI:**

* React.js
* TailwindCSS
* Electron.js (Desktop)
* Vite/Webpack
* Three.js (3D model visualization)

**Backend/API:**

* Node.js or FastAPI (Python) for APIs
* PostgreSQL (Database)
* Redis (Caching & job queues)
* Docker + Kubernetes (Container orchestration)
* Nginx (Reverse proxy)

**AI/ML Components:**

* PyTorch or TensorFlow (Model training and inference)
* ONNX (Model format for deployment)
* Hugging Face Hub or private ML registry (Model versioning)
* OpenCV (Image processing)

**Simulation & Toolpath Core:**

* C++/Rust (High-performance simulation backend)
* Python bindings (e.g., pybind11)
* FEniCS, deal.II (FEM libraries)
* OpenCASCADE, CGAL (3D geometry processing)

**Monitoring & Device Integration:**

* OpenCV + RTSP (camera stream analysis)
* MQTT or ZeroMQ (for edge device communication)
* Jetson Nano/Orin (on-device AI inference)

**IV. PRODUCT DEVELOPMENT STAGES**

**Phase 1: Research & Prototyping (Month 1–2)**

* Market & competitor analysis
* Proof-of-concept: toolpath generator + AI Copilot prototype
* Initial UI/UX wireframes
* Tech stack finalization

**Phase 2: Core System Development (Month 3–6)**

* Build UI shell and user flow
* Develop G-code generator and AI Copilot
* Integrate thermal simulation (basic version)
* Connect real-time monitoring module
* Setup cloud backend and plugin API system

**Phase 3: Testing & Pilot Deployment (Month 7–9)**

* Closed beta with selected LFAM companies
* Collect feedback for performance, UX, and AI inference accuracy
* Bug fixing, optimization, scalability testing
* Licensing system integration

**Phase 4: Commercial Launch (Month 10–12)**

* Prepare documentation, onboarding tutorial, SDK for plugins
* Launch SaaS platform or distribute on-premise builds
* Begin sales and business development
* Continue adding new machine profiles and hybrid support

**V. SCALABILITY & FUTURE PLANS**

* Expand machine compatibility (FFF, pellet, hybrid, DED, etc.)
* Integrate generative design and topology optimization
* Cloud-based slicing and simulation for remote operation
* AI-assisted defect prediction and correction