# Tech Stack Document

**Document Version:** 1.0  
**Date:** May 16, 2025  
**Project:** VibeCoding

## 1. Introduction

### 1.1 Purpose

This Tech Stack Document outlines the technologies, frameworks, tools, and infrastructure components that will be used to build the VibeCoding platform. It serves as a reference for the development team and stakeholders to understand the technical foundation of the project.

### 1.2 Scope

This document covers the complete technology stack including frontend, backend, database, DevOps, and infrastructure components. It also includes justifications for technology choices and considerations for future scalability.

## 2. Architecture Overview

### 2.1 High-Level Architecture

VibeCoding will follow a microservices architecture to enable scalability, maintainability, and independent deployment of features. The system will consist of:

* Web Application (Frontend)
* Desktop Applications (Cross-platform)
* Mobile Companions (iOS/Android)
* API Gateway
* Microservices for core functionality
* Data Storage Solutions
* AI/ML Services
* Real-time Collaboration Infrastructure
* Authentication and Authorization Services

### 2.2 Architecture Diagram

[Architecture diagram to be included in future versions]

## 3. Frontend Technologies

### 3.1 Web Application

* **Framework**: React 19.0
* **State Management**: Redux Toolkit
* **UI Component Library**: Material-UI v6
* **Build Tools**: Vite 5.0
* **Testing**: Jest, React Testing Library
* **Language**: TypeScript 5.3+

**Justification:** React provides a robust ecosystem for building complex UIs with excellent developer tooling. TypeScript adds strong typing to reduce runtime errors.

### 3.2 Desktop Application

* **Framework**: Electron 28+
* **Core Technologies**: Node.js, Chromium
* **Performance Optimization**: WebAssembly for compute-intensive operations
* **Native Integration**: Native modules for filesystem access
* **Packaging**: Electron Forge

**Justification:** Electron enables cross-platform desktop applications while allowing code reuse between web and desktop platforms.

### 3.3 Mobile Applications

* **Framework**: React Native
* **State Management**: MobX
* **UI Components**: React Native Paper
* **Language**: TypeScript

**Justification:** React Native enables code sharing between iOS and Android while providing near-native performance.

## 4. Backend Technologies

### 4.1 API Layer

* **Framework**: Node.js with NestJS
* **API Style**: RESTful with GraphQL for complex data requirements
* **API Gateway**: Kong
* **Documentation**: OpenAPI/Swagger
* **Language**: TypeScript

**Justification:** NestJS provides a structured approach to building scalable server-side applications with TypeScript.

### 4.2 Microservices

* **Framework**: NestJS for Node.js services
* **Inter-service Communication**: gRPC, Apache Kafka
* **Service Discovery**: Consul
* **Circuit Breaking**: Resilience4j
* **Containerization**: Docker
* **Orchestration**: Kubernetes

**Justification:** Microservices architecture enables independent scaling and deployment of services.

### 4.3 Real-time Collaboration

* **Technology**: WebSockets with Socket.io
* **Consistency**: Operational Transformation (OT)
* **State Synchronization**: CRDT
* **Scaling**: Redis Pub/Sub

**Justification:** These technologies enable real-time collaboration with conflict resolution capabilities.

## 5. Data Storage

### 5.1 Primary Database

* **Database**: PostgreSQL 16+
* **Connection Pooling**: PgBouncer
* **Migration Tool**: Flyway
* **Backup Strategy**: Point-in-time recovery with WAL archiving

**Justification:** PostgreSQL offers robust ACID compliance and excellent JSON support.

### 5.2 Cache Layer

* **Technology**: Redis
* **Use Cases**: Session storage, real-time collaboration state

**Justification:** Redis provides high-performance in-memory data structure storage.

### 5.3 Search Engine

* **Technology**: Elasticsearch
* **Use Cases**: Code search, documentation search

**Justification:** Elasticsearch provides powerful full-text search capabilities.

## 6. AI and Machine Learning

### 6.1 Core ML Infrastructure

* **Framework**: TensorFlow, PyTorch
* **Model Serving**: TensorFlow Serving
* **Language**: Python
* **Infrastructure**: GPU-enabled compute instances

**Justification:** Industry-standard frameworks with extensive community support.

### 6.2 Code Analysis

* **Static Analysis**: Custom analyzers
* **Semantic Analysis**: Graph-based representation
* **Suggestion System**: ML-based recommendations

**Justification:** Enables deep code understanding for intelligent features.

## 7. DevOps & Infrastructure

### 7.1 CI/CD

* **Pipeline**: GitHub Actions
* **Artifact Repository**: Artifactory
* **Quality Gates**: SonarQube
* **Deployment**: Blue-Green deployments

**Justification:** Provides robust CI/CD with quality checks.

### 7.2 Monitoring

* **Metrics**: Prometheus
* **Logging**: ELK Stack
* **Tracing**: Jaeger
* **Alerting**: PagerDuty

**Justification:** Comprehensive observability solution.

### 7.3 Cloud Infrastructure

* **Primary**: AWS
* **Alternative**: Azure (multi-cloud strategy)
* **Regions**: Multi-region deployment

**Justification:** AWS provides comprehensive services while multi-cloud reduces lock-in.

## 8. Security

### 8.1 Authentication

* **Identity Provider**: Auth0
* **Protocol**: OAuth 2.0, OpenID Connect
* **MFA**: TOTP, WebAuthn

**Justification:** Standards-based authentication with flexibility.

### 8.2 Data Protection

* **Encryption**: AES-256, TLS 1.3
* **Key Management**: AWS KMS

**Justification:** Industry-standard encryption protocols.

## 9. Future Considerations

### 9.1 Emerging Technologies

* **Edge Computing**: For reduced latency
* **WebAssembly**: For compute-intensive features
* **Blockchain**: For decentralized version control
* **AR/VR**: For immersive collaboration

### 9.2 Refresh Cycle

* Regular evaluation every 6 months
* Major version upgrades assessed for risk/benefit
* Deprecation strategy for outdated technologies

## 10. Approval

This Tech Stack Document requires approval from:

* Chief Technology Officer
* VP of Engineering
* Security Officer
* Infrastructure Lead