# Setarcos App Technical Stack Summary

## Frontend

* **Framework**: React Native (iOS & Android)
* **UI Library**: React Native Paper (Material Design)
* **Navigation**: React Navigation
* **State Management**: Context API with hooks
* **Testing**: Jest, React Native Testing Library, Detox (E2E)

## Backend

* **Framework**: FastAPI (Python)
* **Database**: PostgreSQL via Supabase
* **Authentication**: Supabase Auth
* **Caching**: Redis
* **Task Queue**: Celery with Redis as broker
* **API Documentation**: OpenAPI/Swagger

## AI Implementation

* **AI Router**: Phased implementation connecting multiple models
* **Models**: Grok, Claude, Google Gemini, ChatGPT
* **Features**: Context-specific tone prompt workflow, cost optimization
* **Lumen AI Companion**: Philosophical guidance agent with cross-feature integration

## DevOps & Infrastructure

* **Hosting**: Render
* **CI/CD**: GitHub Actions
* **Error Tracking**: Sentry
* **Analytics**: PostHog
* **Environments**: Development, Staging, Production

## Core Features

1. **Ask Feature**: AI-driven philosophical dialogues
2. **Quest Feature**: Guided philosophical exploration journeys
3. **Explore Feature**: Interactive concept visualization and mapping
4. **Journal Feature**: Personal philosophical journaling
5. **Forum Feature**: Community discussion and interaction
6. **Profile Feature**: User progression and achievements
7. **Notification Feature**: Contextual user notifications
8. **Lumen AI Companion**: Intelligent guidance system

## Gamification

* **Wisdom XP System**: Cross-feature progression system
* **Achievements**: Philosophical milestones and badges
* **Streaks**: Engagement incentives and rewards

## Security

* **Authentication**: Supabase Auth with secure flows
* **Authorization**: Role-based access control
* **Data Protection**: Privacy controls and encryption

## Mobile App Structure

* **Repository**: setarcos-app
* **Components**: Reusable UI elements
* **Screens**: Feature-specific views
* **Navigation**: Stack, tab, and drawer navigators
* **Services**: API communication layer
* **Hooks**: Custom React hooks for shared logic
* **Context**: Global state management

## API Structure

* **Repository**: setarcos-api
* **Endpoints**: RESTful API (v1)
* **Services**: Business logic layer
* **Models**: Database schema representations
* **Tasks**: Asynchronous processing jobs
* **Middleware**: Request/response processing