BAKAME AI - Complete System Architecture Documentation

🏗 System Overview

BAKAME (Building African Knowledge through Accessible Mobile Education) is a comprehensive AI-powered learning platform that delivers education through voice calls and SMS to feature phones, specifically designed for students without internet access.

📊 High-Level Architecture

Core Components

```
┌─────────────────────────────────────────────────────────────────┐
│                    BAKAME AI ECOSYSTEM                          │
├─────────────────────────────────────────────────────────────────┤
│  Frontend Layer                                                 │
│                                                                 │
│  ┌─────────────────┐  ┌─────────────────┐  ┌─────────────────┐  │
│  │ bakame-frontend │  │  bakame-admin   │  │   Mobile Users  │  │
│  │   (React/Vite)  │  │(React/TypeScript)│  │  (Voice/SMS)    │  │
│  └─────────────────┘  └─────────────────┘  └─────────────────┘  │
├─────────────────────────────────────────────────────────────────┤
│  API Gateway & Communication Layer                              │
│                                                                 │
│  ┌─────────────────┐  ┌─────────────────┐  ┌─────────────────┐  │
│  │   FastAPI       │  │   Twilio API    │  │   WebRTC/HTTP   │  │
│  │   Backend       │  │  Voice/SMS      │  │   Endpoints     │  │
│  └─────────────────┘  └─────────────────┘  └─────────────────┘  │
├─────────────────────────────────────────────────────────────────┤
│  AI Processing Layer                                            │
│                                                                 │
│  ┌─────────────────┐  ┌─────────────────┐  ┌─────────────────┐  │
│  │   OpenAI GPT    │  │  Whisper STT    │  │   Llama LLM     │  │
│  │   (Primary AI)  │  │(Speech-to-Text) │  │ (Alternative)   │  │
│  └─────────────────┘  └─────────────────┘  └─────────────────┘  │
├─────────────────────────────────────────────────────────────────┤
│  Data & Session Management                                      │
│                                                                 │
│  ┌─────────────────┐  ┌─────────────────┐  ┌─────────────────┐  │
│  │   PostgreSQL    │  │     Redis       │  │   File System   │  │
│  │   (Persistent)  │  │   (Sessions)    │  │   (Audio/Logs)  │  │
│  └─────────────────┘  └─────────────────┘  └─────────────────┘  │
└─────────────────────────────────────────────────────────────────┘
```

🎯 Learning Module Architecture

Module System Design

```
┌─────────────────────────────────────────────────────────────────┐
│                    LEARNING MODULES                             │
├─────────────────────────────────────────────────────────────────┤
│                                                                 │
│  ┌─────────────────┐  ┌─────────────────┐  ┌─────────────────┐  │
│  │ English Module  │  │  Math Module    │  │Comprehension    │  │
│  │ • Grammar       │  │ • Mental Math   │  │ • Stories       │  │
│  │ • Pronunciation │  │ • RWF Context   │  │ • Q&A Sessions  │  │
│  │ • Conversation  │  │ • Adaptive Diff │  │ • Analysis      │  │
│  └─────────────────┘  └─────────────────┘  └─────────────────┘  │
│                                                               │
│  ┌─────────────────┐  ┌─────────────────┐                     │
│  │ Debate Module   │  │ General Module  │                     │
│  │ • Critical Think│  │ • Entry Point   │                     │
│  │ • Argumentation │  │ • Module Router │                     │
│  │ • Perspective    │  │ • Help System    │                      │
│  └──────────────────┘  └──────────────────┘                      │
└──────────────────────────────────────────────────────────────────┘
```

Module Processing Flow

```
User Input (Voice/SMS)
         ↓
┌─────────────────┐
│ Webhook Router  │ ← Twilio Integration
└─────────────────┘
         ↓
┌─────────────────┐
│ Speech-to-Text  │ ← OpenAI Whisper (Voice only)
└─────────────────┘
         ↓
┌─────────────────┐
│ Context Manager │ ← Redis Session Storage
└─────────────────┘
         ↓
┌─────────────────┐
│ Module Router   │ ← Determines active module
└─────────────────┘
         ↓
┌─────────────────┐
│ Learning Module │ ← Processes educational content
└─────────────────┘
         ↓
┌─────────────────┐
│ AI Response Gen │ ← OpenAI/Llama + Cultural Context
└─────────────────┘
         ↓
┌─────────────────┐
│ Response Format │ ← TwiML for Voice/SMS
└─────────────────┘
         ↓
┌─────────────────┐
│ User Delivery   │ ← Twilio Voice/SMS
└─────────────────┘
```

🔧 Technical Stack Details

Backend Architecture (FastAPI)

Core Application Structure:

```
bakame-backend/
├── app/
│   ├── main.py                 # FastAPI application entry point
│   ├── config.py              # Environment configuration
│   ├── models/
│   │   ├── database.py        # SQLAlchemy models
│   │   └── auth.py           # Authentication models
│   ├── routers/
│   │   ├── webhooks.py       # Twilio webhook handlers
│   │   ├── admin.py          # Admin dashboard APIs
│   │   ├── auth.py           # Authentication endpoints
│   │   └── content.py        # Content management
│   ├── services/
│   │   ├── twilio_service.py         # Voice/SMS integration
│   │   ├── openai_service.py         # AI text generation
│   │   ├── redis_service.py          # Session management
│   │   ├── logging_service.py        # Analytics & logging
│   │   ├── emotional_intelligence_service.py  # Emotion detection
│   │   ├── gamification_service.py   # Points & achievements
│   │   ├── multimodal_service.py     # Learning style detection
│   │   ├── wellness_service.py       # Mental health support
│   │   ├── economic_empowerment_service.py  # Financial literacy
│   │   ├── community_service.py      # Peer learning
│   │   ├── teacher_service.py        # Educator tools
│   │   ├── predictive_analytics_service.py  # Learning analytics
│   │   ├── adaptive_learning_service.py     # Personalization
│   │   ├── offline_service.py        # Offline capabilities
│   │   ├── llama_service.py          # Alternative LLM
│   │   ├── deepgram_service.py       # Alternative STT
│   │   └── newsapi_service.py        # Current events
│   └── modules/
│       ├── english_module.py         # English learning logic
│       ├── math_module.py           # Mathematics learning
│       ├── comprehension_module.py   # Reading comprehension
│       ├── debate_module.py         # Critical thinking
│       └── general_module.py        # Entry point & routing
├── pyproject.toml              # Poetry dependencies
└── README.md
```

Frontend Architecture

User-Facing Frontend (bakame-frontend): - **Framework:** React 18 + Vite + TypeScript - **UI Library:** Radix UI components - **Styling:** Tailwind CSS - **State Management:** React Query (TanStack) - **Routing:** React Router DOM - **Charts:** Recharts for analytics

Admin Dashboard (bakame-admin): - **Framework:** React 18 + Vite + TypeScript - **UI Library:** Radix UI + shadcn/ui components - **Styling:** Tailwind CSS - **Features:** User management, analytics, curriculum alignment

📊 Data Architecture

Database Schema (PostgreSQL)

```sql
-- Core User Management
users (
    id, phone_number, user_type, name, region,
    school, grade_level, is_active, created_at,
    last_active, total_points, current_level
)

-- Session Tracking
user_sessions (
    id, phone_number, session_id, module_name,
    interaction_type, user_input, ai_response,
    timestamp, session_duration
)

-- Module Analytics
module_usage (
    id, phone_number, module_name, usage_count,
    last_used, total_duration
)

-- Community Features
peer_connections (
    id, user1_phone, user2_phone, connection_type,
    status, created_at, last_interaction
)

learning_groups (
    id, name, description, group_type, region,
    school, grade_level, subject, teacher_phone,
    is_active, created_at, max_members
)

group_memberships (
    id, group_id, user_phone, role, joined_at, is_active
)

peer_learning_sessions (
    id, session_id, group_id, connection_id,
    module_name, topic, participants, started_at,
    ended_at, session_summary
)

-- Authentication
web_users (
    id, email, full_name, hashed_password, role,
    organization, is_active, created_at
)
```

Session Management (Redis)

```json
{
  "user_context:{phone_number}": {
    "current_module": "math",
    "conversation_history": [
      {
        "user": "I want to practice math",
        "ai": "Muraho! Let's practice math with RWF...",
        "timestamp": "2024-01-01T12:00:00Z"
      }
    ],
    "user_state": {
      "math_level": "medium",
      "math_problems_attempted": 15,
      "math_problems_correct": 12,
      "current_math_problem": {
        "question": "150 + 75",
        "answer": 225,
        "operation": "+"
      },
      "requested_module": null
    },
    "user_name": "Jean",
    "phone_number": "+250781234567",
    "session_start": "2024-01-01T11:45:00Z"
  }
}
```

🌐 Communication Flow

Voice Call Processing

```
1. User dials Twilio number
2. Twilio webhook → /webhook/call
3. Welcome message generation
4. Speech collection (Gather)
5. Audio → OpenAI Whisper → Text
6. Text → Module Processing
7. AI Response Generation
8. TwiML Voice Response
9. Audio playback to user
10. Loop for continued interaction
```

SMS Processing

```
1. User sends SMS to Twilio number
2. Twilio webhook → /webhook/sms
3. Text extraction from message body
4. Module Processing
5. AI Response Generation
6. TwiML SMS Response
7. SMS delivery to user
```

🧠 AI Processing Pipeline

Multi-Model AI Architecture

```
┌─────────────────────────────────────────────────────────────────┐
│                     AI PROCESSING LAYER                        │
├─────────────────────────────────────────────────────────────────┤
│  Speech Processing                                              │
│  ┌─────────────────┐  ┌─────────────────┐                     │
│  │ OpenAI Whisper  │  │   Deepgram      │                     │
│  │   (Primary)     │  │ (Alternative)   │                     │
│  └─────────────────┘  └─────────────────┘                     │
├─────────────────────────────────────────────────────────────────┤
│  Text Generation                                               │
│  ┌─────────────────┐  ┌─────────────────┐                     │
│  │   GPT-4o-mini   │  │   Llama LLM     │                     │
│  │   (Primary)     │  │ (Alternative)   │                     │
│  └─────────────────┘  └─────────────────┘                     │
├─────────────────────────────────────────────────────────────────┤
│  Specialized AI Services                                       │
│  ┌─────────────────┐  ┌─────────────────┐  ┌─────────────────┐ │
│  │   Emotional     │  │  Predictive     │  │   Adaptive      │ │
│  │ Intelligence    │  │   Analytics     │  │   Learning      │ │
│  └─────────────────┘  └─────────────────┘  └─────────────────┘ │
└─────────────────────────────────────────────────────────────────┘
```

Cultural Context Integration

Rwanda-Specific AI Prompts: - **English Module:** “You’re a friendly, encouraging English conversation partner who understands Rwandan culture deeply…” - **Math Module:** “You’re an enthusiastic math mentor who makes numbers fun using Rwandan contexts. Use examples with Rwandan francs (RWF)…” - **Comprehension:** “You’re an engaging storyteller who loves Rwandan culture and traditions…” - **Debate:** “You’re a thoughtful discussion partner who understands Rwandan society and values deeply…” - **General:** “You’re BAKAME, a warm and intelligent AI learning companion who understands Rwandan culture deeply…”

🎮 Advanced Features

Gamification System

```
┌─────────────────────────────────────────────────────────────────┐
│                     GAMIFICATION ENGINE                        │
├─────────────────────────────────────────────────────────────────┤
│  Achievement System                                             │
│  • ubuntu_spirit: Community values achievement                 │
│  • hill_climber: Challenge overcomer                          │
│  • knowledge_seeker: Learning streak master                   │
│  • unity_builder: Respectful debate participant               │
│  • math_champion: Mathematics expert                          │
│  • story_master: Comprehension specialist                     │
│  • english_explorer: Language learner                         │
│  • resilience_warrior: Persistence champion                   │
├─────────────────────────────────────────────────────────────────┤
│  Progress Tracking                                             │
│  • Points system across all modules                               │
│  • Level progression (beginner → expert → master)                 │
│  • Module-specific difficulty adaptation                          │
│  • Cultural context rewards                                       │
└───────────────────────────────────────────────────────────────────┘
```

Emotional Intelligence

```
┌───────────────────────────────────────────────────────────────────┐
│                 EMOTIONAL INTELLIGENCE SYSTEM                      │
├───────────────────────────────────────────────────────────────────┤
│  Emotion Detection                                                 │
│  • frustrated, confident, discouraged, motivated                  │
│  • confused, positive                                             │
├───────────────────────────────────────────────────────────────────┤
│  Cultural Response Adaptation                                      │
│  • Kinyarwanda phrases: "Ntugire ubwoba", "Byiza cyane!"     │
│  • Ubuntu philosophy integration                                  │
│  • Rwanda resilience messaging                                    │
│  • Community support emphasis                                     │
├───────────────────────────────────────────────────────────────────┤
│  Adaptive Response Generation                                      │
│  • Emotionally-aware AI responses                               │
│  • Cultural sensitivity                                           │
│  • Motivational messaging                                         │
└───────────────────────────────────────────────────────────────────┘
```

Community & Peer Learning

```
┌───────────────────────────────────────────────────────────────────┐
│                     COMMUNITY FEATURES                             │
├───────────────────────────────────────────────────────────────────┤
│  Regional Learning Groups                                           │
│  • Kigali, Northern, Southern, Eastern, Western regions           │
│  • School-based groups                                            │
│  • Grade-level cohorts                                            │
├───────────────────────────────────────────────────────────────────┤
│  Teacher Integration                                               │
│  • Teacher registration and dashboard                             │
│  • Classroom creation and management                              │
│  • Student progress monitoring                                    │
│  • Analytics and reporting                                        │
├───────────────────────────────────────────────────────────────────┤
│  Peer Connections                                                 │
│  • Study buddy matching                                           │
│  • Mentor-mentee relationships                                  │
│  • Regional peer networks                                         │
│  • Collaborative learning sessions                              │
└───────────────────────────────────────────────────────────────────┘
```

🔒 Security & Authentication

Multi-Layer Security

```
┌───────────────────────────────────────────────────────────────────┐
│                     SECURITY ARCHITECTURE                          │
├───────────────────────────────────────────────────────────────────┤
│  Phone-Based Identity                                              │
│  • No registration required for learners                          │
│  • Phone number as primary identifier                             │
│  • Session-based context management                               │
├───────────────────────────────────────────────────────────────────┤
│  Web Authentication                                                │
│  • JWT-based admin authentication                                 │
│  • Role-based access control (admin, super_admin)                 │
│  • Organization-based permissions                                │
├──────────────────────────────────────────────────────────────────┤
│  Data Protection                                                  │
│  • Encrypted API communications                                  │
│  • Secure database connections                                   │
│  • Privacy-compliant data handling                               │
│  • Session data TTL management                                   │
└──────────────────────────────────────────────────────────────────┘
```

📈 Analytics & Monitoring

Comprehensive Analytics System

```
┌──────────────────────────────────────────────────────────────────┐
│                    ANALYTICS DASHBOARD                            │
├──────────────────────────────────────────────────────────────────┤
│  Usage Statistics                                                 │
│  • Total sessions, unique users                                  │
│  • Module engagement metrics                                     │
│  • Geographic usage patterns                                     │
│  • Learning outcome tracking                                     │
├──────────────────────────────────────────────────────────────────┤
│  Predictive Analytics                                             │
│  • Learning pattern analysis                                     │
│  • Optimal session length detection                              │
│  • Engagement threshold monitoring                               │
│  • Mastery prediction algorithms                                 │
├──────────────────────────────────────────────────────────────────┤
│  Export & Reporting                                               │
│  • CSV data export                                               │
│  • Session logs for quality improvement                          │
│  • Performance metrics for stakeholders                          │
│  • Curriculum alignment reporting                                │
└──────────────────────────────────────────────────────────────────┘
```

🚀 Deployment Architecture

Production Infrastructure

```
┌──────────────────────────────────────────────────────────────────┐
│                    DEPLOYMENT STACK                               │
├──────────────────────────────────────────────────────────────────┤
│  Backend Deployment                                               │
│  • Platform: Fly.io                                              │
│  • URL: https://app-pyzfduqr.fly.dev/                           │
│  • Runtime: Python 3.12 + Poetry                                │
│  • Database: PostgreSQL                                          │
├──────────────────────────────────────────────────────────────────┤
│  Frontend Deployment                                              │
│  • Admin Dashboard: https://project-handling-app-jiwikt4q.devinapps.com/ │
│  • Static hosting with CDN                                       │
│  • React SPA with client-side routing                            │
├──────────────────────────────────────────────────────────────────┤
│  External Services                                                │
│  • Twilio: Voice/SMS infrastructure                              │
│  • OpenAI: Primary AI processing                                 │
│  • Redis: Session management                                     │
│  • NewsAPI: Current events integration                           │
│  • Deepgram: Alternative speech processing                       │
└──────────────────────────────────────────────────────────────────┘
```

🌍 Scalability & Performance

Horizontal Scaling Design

```
┌─────────────────────────────────────────────────────────────────┐
│                        SCALABILITY FEATURES                      │
├─────────────────────────────────────────────────────────────────┤
│  Stateless Architecture                                          │
│  • Session data in Redis (external)                             │
│  • Stateless FastAPI application                                │
│  • Horizontal scaling ready                                     │
├─────────────────────────────────────────────────────────────────┤
│  Performance Optimizations                                       │
│  • Sub-3-second response times                                  │
│  • Async/await throughout                                       │
│  • Connection pooling                                           │
│  • Efficient database queries                                   │
├─────────────────────────────────────────────────────────────────┤
│  Fault Tolerance                                                │
│  • Graceful fallbacks for AI services                          │
│  • Redis memory fallback                                        │
│  • Error handling and logging                                   │
│  • Health check endpoints                                       │
└─────────────────────────────────────────────────────────────────┘
```

📋 Configuration Management

Environment Configuration

```python
# Key Configuration Parameters
class Settings(BaseSettings):
    # Twilio Integration
    twilio_account_sid: str
    twilio_auth_token: str
    twilio_phone_number: str

    # AI Services
    openai_api_key: str
    llama_api_key: str
    use_llama: bool = True
    deepgram_api_key: str
    newsapi_key: str

    # Infrastructure
    redis_url: str = "redis://localhost:6379/0"
    database_url: str

    # Application
    app_env: str = "development"
    debug: bool = True
```

🎯 Key Architectural Decisions

Design Principles

1. **Accessibility First:** No internet or smartphone required
2. **Cultural Integration:** Deep Rwanda context throughout
3. **Scalable Design:** Cloud-native, stateless architecture
4. **Multi-Modal AI:** Voice and text processing capabilities
5. **Educational Focus:** Curriculum-aligned learning modules
6. **Community Building:** Peer learning and teacher integration
7. **Data-Driven:** Comprehensive analytics and adaptation
8. **Fault Tolerant:** Graceful degradation and fallbacks

Technology Choices

- **FastAPI:** High-performance async Python framework
- **React:** Modern, component-based frontend development
- **PostgreSQL:** Robust relational database for complex queries
- **Redis:** High-performance session and cache management
- **Twilio:** Reliable telecommunications infrastructure
- **OpenAI:** State-of-the-art AI language processing
- **Fly.io:** Modern cloud deployment platform

Document Version: 1.0
Last Updated: September 6, 2025
Architecture Status: Production Ready MVP
Next Phase: Community features and teacher integration scaling