EcolithSwap - Complete Platform

EcolithSwap is a comprehensive African-first electric battery swapping and recycling platform designed for urban and rural users in Kenya. The platform enables battery swaps, charging, plastic waste redemption for credit, and environmental impact tracking.

🏗️ Architecture Overview

The EcolithSwap platform consists of three main components:

1. Mobile App (React Native)

- **Location**: `/EcolithSwap/`
- **Platform**: Cross-platform (iOS & Android)
- **Features**: Battery swapping, station finder, plastic waste submission, user dashboard
- **Technology**: React Native, Expo

2. Admin Dashboard (React Web App)

- **Location**: `/ecolithswap-admin/`
- **Platform**: Web-based admin panel
- **Features**: User management, station management, battery tracking, analytics
- **Technology**: React, TypeScript, Vite, Tailwind CSS

3. Backend API (Node.js + MySQL)

- **Location**: `/EcolithSwap-Backend/`
- **Platform**: RESTful API server
- **Features**: Authentication, data management, real-time updates, payment processing
- **Technology**: Node.js, Express, MySQL, JWT, Socket.IO

🚀 Quick Start

Prerequisites

- **Node.js** (v16 or higher)

- **MySQL** (v8.0 or higher)
- **npm** or **yarn**
- **Expo CLI** (for mobile development)
- **Git**

1. Clone the Repository

```
git clone <repository-url>
cd ecolithswap-platform
```

2. Backend Setup

```
# Navigate to backend directory
cd EcolithSwap-Backend

# Install dependencies
npm install

# Set up environment variables
cp .env.example .env
# Edit .env with your MySQL credentials

# Set up database
npm run setup-db

# Seed with sample data (optional)
npm run seed-db

# Start the server
npm start
```

The backend will be running on `http://localhost:3000`

3. Admin Dashboard Setup

```
# Navigate to admin directory
cd ecolithswap-admin

# Install dependencies
npm install

# Set up environment variables
cp .env.example .env
# Edit .env if needed

# Start development server
npm run dev
```

The admin dashboard will be running on `http://localhost:5173`

4. Mobile App Setup

```
# Navigate to mobile app directory
cd EcolithSwap

# Install dependencies
npm install

# Start Expo development server
npx expo start
```

Use the Expo Go app on your phone or an emulator to run the mobile app.

🔐 Default Login Credentials

After running the database seeding script, you can use these test accounts:

Admin Dashboard

- **Admin**: `admin@ecolithswap.com` / `password123`
- **Station Manager**: `manager@ecolithswap.com` / `password123`

Mobile App

- **Customer 1**: `john.doe@email.com` / `password123`
- **Customer 2**: `jane.smith@email.com` / `password123`

📱 Mobile App Features

Core Features

- **Battery Swapping**: Rent and return batteries at charging stations
- **Station Finder**: Locate nearby charging stations with map/list view
- **Plastic Waste Recycling**: Submit plastic waste and earn EcoCredits
- **Payment Integration**: M-Pesa and credit-based payments
- **Impact Tracking**: Monitor environmental impact and savings
- **Offline Support**: Basic functionality works without internet

User Interface

- Clean, intuitive design optimized for low-end Android devices
- Support for multiple languages (English, Swahili)
- Low-bandwidth optimized for intermittent connectivity
- Accessibility features for diverse user base

🖥️ Admin Dashboard Features

Management Capabilities

- **User Management**: View, edit, activate/deactivate user accounts
- **Station Management**: Add, edit, monitor charging stations
- **Battery Management**: Track battery inventory, health, and usage
- **Waste Management**: Verify plastic waste submissions
- **Payment Management**: Monitor transactions and process refunds

Analytics & Reporting

- **Revenue Analytics**: Track income from rentals and services
- **Usage Analytics**: Monitor station utilization and battery performance
- **Environmental Impact**: Track plastic waste processed and CO2 savings
- **User Analytics**: Understand user behavior and engagement

Real-time Features

- Live dashboard updates via WebSockets
- Real-time station status monitoring
- Instant notifications for critical events

🔧 Backend API Features

Core Services

- **Authentication**: JWT-based auth with refresh tokens
- **User Management**: Registration, profile management, role-based access
- **Station Management**: CRUD operations with location-based queries

- **Battery Management**: Inventory tracking with real-time status updates
- **Rental Management**: Complete rental lifecycle management
- **Waste Management**: Plastic waste submission and verification
- **Payment Processing**: M-Pesa integration and credit system
- **Analytics**: Comprehensive reporting and insights

Technical Features

- RESTful API design with consistent error handling
- Database migrations and seeding
- Rate limiting and security middleware
- Real-time updates via Socket.IO
- Comprehensive logging and monitoring
- API documentation and health checks

🗄️ Database Schema

The MySQL database includes the following main tables:

- **users**: User account information
- **user_profiles**: Extended user profile data and credits
- **stations**: Charging station information and locations
- **batteries**: Battery inventory and status tracking
- **battery_rentals**: Rental transactions and history
- **plastic_waste_logs**: Waste submission records
- **payments**: Payment transactions and history

🌍 Environment Configuration

Backend Environment Variables

Key configuration options in `/EcolithSwap-Backend/.env`:

```
# Database
DB_HOST=localhost
DB_PORT=3306
DB_USER=root
DB_PASSWORD=your_password
DB_NAME=ecolithswap

# JWT Authentication
JWT_SECRET=your_secret_key
JWT_EXPIRES_IN=24h

# M-Pesa Integration
MPESA_ENVIRONMENT=sandbox
MPESA_CONSUMER_KEY=your_key
MPESA_CONSUMER_SECRET=your_secret
```

Frontend Environment Variables

Admin dashboard configuration in `/ecolithswap-admin/.env`:

```
VITE_API_BASE_URL=http://localhost:3000/api
VITE_APP_TITLE=EcolithSwap Admin
```

🚦 API Documentation

Authentication Endpoints

- `POST /api/auth/register` - User registration
- `POST /api/auth/login` - User login
- `POST /api/auth/logout` - User logout
- `GET /api/auth/profile` - Get user profile
- `PUT /api/auth/profile` - Update user profile

Station Endpoints

- `GET /api/stations` - List all stations
- `GET /api/stations/:id` - Get station details
- `POST /api/stations` - Create new station (admin)
- `PUT /api/stations/:id` - Update station (admin)

Battery & Rental Endpoints

- `GET /api/batteries` - List batteries
- `POST /api/rentals` - Rent a battery
- `PATCH /api/rentals/:id/return` - Return a battery
- `GET /api/rentals` - Get rental history

Waste Management Endpoints

- `POST /api/waste` - Submit plastic waste
- `GET /api/waste` - Get waste submission history
- `PATCH /api/waste/:id/verify` - Verify waste (admin)

Payment Endpoints

- `POST /api/payments/mpesa` - Process M-Pesa payment
- `POST /api/payments/credits` - Process credit payment
- `GET /api/payments` - Get payment history

🔒 Security Features

- **JWT Authentication**: Secure token-based authentication
- **Role-based Access Control**: Admin, Station Manager, Customer roles
- **Rate Limiting**: Prevent API abuse
- **Input Validation**: Comprehensive data validation
- **SQL Injection Protection**: Parameterized queries
- **CORS Configuration**: Controlled cross-origin requests
- **Security Headers**: Helmet.js for security headers

📊 Monitoring & Analytics

Application Monitoring

- Health check endpoints for system status
- Comprehensive error logging and tracking
- Performance monitoring and optimization
- Database query optimization

Business Analytics

- User engagement and retention metrics
- Revenue tracking and financial reporting
- Environmental impact measurement

- Station utilization and efficiency metrics

🚀 Deployment

Production Deployment Steps

1. **Database Setup**
   - Set up MySQL server
   - Run migrations: `npm run migrate`
   - Configure backup strategy
2. **Backend Deployment**
   - Deploy to cloud provider (AWS, Digital Ocean, etc.)
   - Set production environment variables
   - Configure SSL certificates
   - Set up process manager (PM2)
3. **Admin Dashboard Deployment**
   - Build for production: `npm run build`
   - Deploy to static hosting (Netlify, Vercel, etc.)
   - Configure environment variables
4. **Mobile App Deployment**
   - Build for iOS: `expo build:ios`
   - Build for Android: `expo build:android`
   - Deploy to App Store and Google Play

Docker Deployment (Optional)

Docker configurations are available for containerized deployment.

🧪 Testing

Backend Testing

```
cd EcolithSwap-Backend
npm test
```

Frontend Testing

```
cd ecolithswap-admin
npm test
```

Mobile App Testing

```
cd EcolithSwap
npm test
```

🤝 Contributing

1. Fork the repository
2. Create a feature branch
3. Commit your changes
4. Push to the branch
5. Create a Pull Request

📄 License

This project is licensed under the MIT License - see the LICENSE file for details.

🆘 Support

For support, please contact:
- **Email**: support@ecolithswap.com
- **Documentation**: API Docs
- **Issues**: GitHub Issues

🙏 Acknowledgments

- Ecolith Africa Solutions team
- Open-source community contributors
- Beta testers and early adopters

Built with ❤️ for sustainable technology in Africa