

# ETL Plus - C++ Backend

---

A high-performance, modern C++ backend application for Extract, Transform, Load (ETL) operations with HTTP REST API, authentication, job management, and data transformation capabilities.

## Features

- **HTTP REST API** - Built with Boost.Beast for high-performance HTTP handling
- **Authentication System** - User management, sessions, and role-based access control
- **ETL Job Management** - Schedule, execute, and monitor data processing jobs
- **Data Transformation** - Rule-based transformation engine with multiple data types
- **Database Integration** - PostgreSQL support with connection pooling
- **Multi-threaded Architecture** - Concurrent request and job processing
- **Configuration Management** - JSON-based hierarchical configuration

## API Endpoints

### Health & Monitoring

- `GET /api/health` - Server health check
- `GET /api/monitor/status` - System status
- `GET /api/monitor/metrics` - Performance metrics

### Authentication

- `POST /api/auth/login` - User login
- `POST /api/auth/logout` - User logout
- `GET /api/auth/profile` - User profile

### ETL Jobs

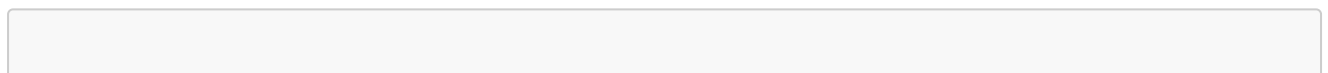
- `GET /api/jobs` - List all jobs
- `POST /api/jobs` - Create new job
- `GET /api/jobs/{id}` - Get job details

## Quick Start

### Prerequisites

- C++20 compiler (GCC 10+ or Clang 12+)
- CMake 3.16+
- Boost 1.70+
- PostgreSQL 12+ (optional)

### Build & Run



```
# Clone and build
git clone <repository-url>
cd CPLUS
mkdir build && cd build
cmake ..
make

# Run
cd bin
cp ../../config/config.json .
./ETLPlusBackend
```

## Test API

```
# Health check
curl http://localhost:8080/api/health

# System status
curl http://localhost:8080/api/monitor/status

# List jobs
curl http://localhost:8080/api/jobs
```

## Project Structure

CPLUS/	
├─ CMakeLists.txt	# Build configuration
├─ config/config.json	# Runtime configuration
├─ include/	# Header files
├─ src/	# Implementation files
├─ build/	# Build artifacts (ignored)
├─ ROADMAP.md	# Development roadmap
├─ ACTION_PLAN.md	# Immediate next steps
└─ DEVELOPMENT.md	# Quick reference

## Development Status

### Completed

- Project structure and build system
- Core HTTP server and API endpoints
- Authentication and session management
- ETL job scheduling and execution
- Data transformation engine
- Configuration management

## In Progress

- HTTP server stability improvements
- PostgreSQL integration
- Enhanced error handling
- Comprehensive test suite

## Next Steps

- JWT authentication
- Data connectors (CSV, JSON)
- Web dashboard frontend
- Performance optimization
- Production deployment

## Documentation

- [ROADMAP.md](#) - Complete development plan and phases
- [ACTION\\_PLAN.md](#) - Immediate tasks and sprint planning
- [DEVELOPMENT.md](#) - Quick reference and commands

## Contributing

1. Fork the repository
2. Create a feature branch
3. Make your changes
4. Add tests
5. Submit a pull request

## License

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

## Technology Stack

- **C++20** - Modern C++ features and performance
- **Boost.Beast** - HTTP server framework
- **PostgreSQL** - Primary database
- **CMake** - Build system
- **JSON** - Configuration and API format

---

**Status:** Active Development

**Version:** 1.0.0-alpha

**Last Updated:** August 8, 2025