

□ SidTech Conduit Documentation

GitHub Repository: <https://github.com/mtaplits/sidtech-conduit>

□ Project Overview

SidTech Conduit is a self-hosted Backend-as-a-Service (BaaS) platform powered by Conduit. It provides ready-made modules for common backend functionality, similar to Firebase or Parse.

Key Features

- **Authentication** - User management, OAuth, JWT
- **Database** - Schema management, CRUD operations
- **Chat** - Real-time messaging
- **Email** - Email sending and templates
- **Forms** - Form handling and validation
- **Push Notifications** - Mobile push notifications
- **Storage** - File storage and management

□ File Storage Locations

Project Files (Your Code)

```
/home/mtaplits/Documents/sidtech-conduit/
├── compose.yml          # Docker Compose configuration
├── env                  # Environment variables
├── loki.cfg.yml         # Logging configuration
├── prometheus.cfg.yml   # Metrics configuration
├── README.md            # Documentation
├── install-docker.sh    # Docker installer
├── install-gh-cli.sh    # GitHub CLI installer
├── setup-conduit.sh     # Conduit setup script
└── .gitignore            # Git ignore rules
```

Docker Persistent Data

```
/var/lib/docker/volumes/  
└── conduit_mongo/_data      # MongoDB database files  
└── conduit_loki/_data       # Loki log storage  
└── conduit_prometheus/_data # Prometheus metrics data
```

Conduit CLI Cache

```
~/.cache/@conduitplatform/cli/  
└── deploy/manifests/v0.16.20/  # Original deployment manifests
```

□ Access Points & Ports

Service	Port	URL	Purpose
Admin Panel	8080	<code>http://localhost:8080</code>	Web-based admin interface
REST API	3000	<code>http://localhost:3000</code>	RESTful API endpoint
GraphQL API	3001	<code>http://localhost:3001</code>	GraphQL API endpoint
Conduit Core	3030-3031	Internal	Core service & gRPC
MongoDB	27017	<code>localhost:27017</code>	Database
Redis	6379	<code>localhost:6379</code>	Cache & sessions
Prometheus	9090	<code>http://localhost:9090</code>	Metrics dashboard
Loki	3100	<code>http://localhost:3100</code>	Log aggregation

⚠ Default Credentials:

Username: `admin`

Password: `admin`

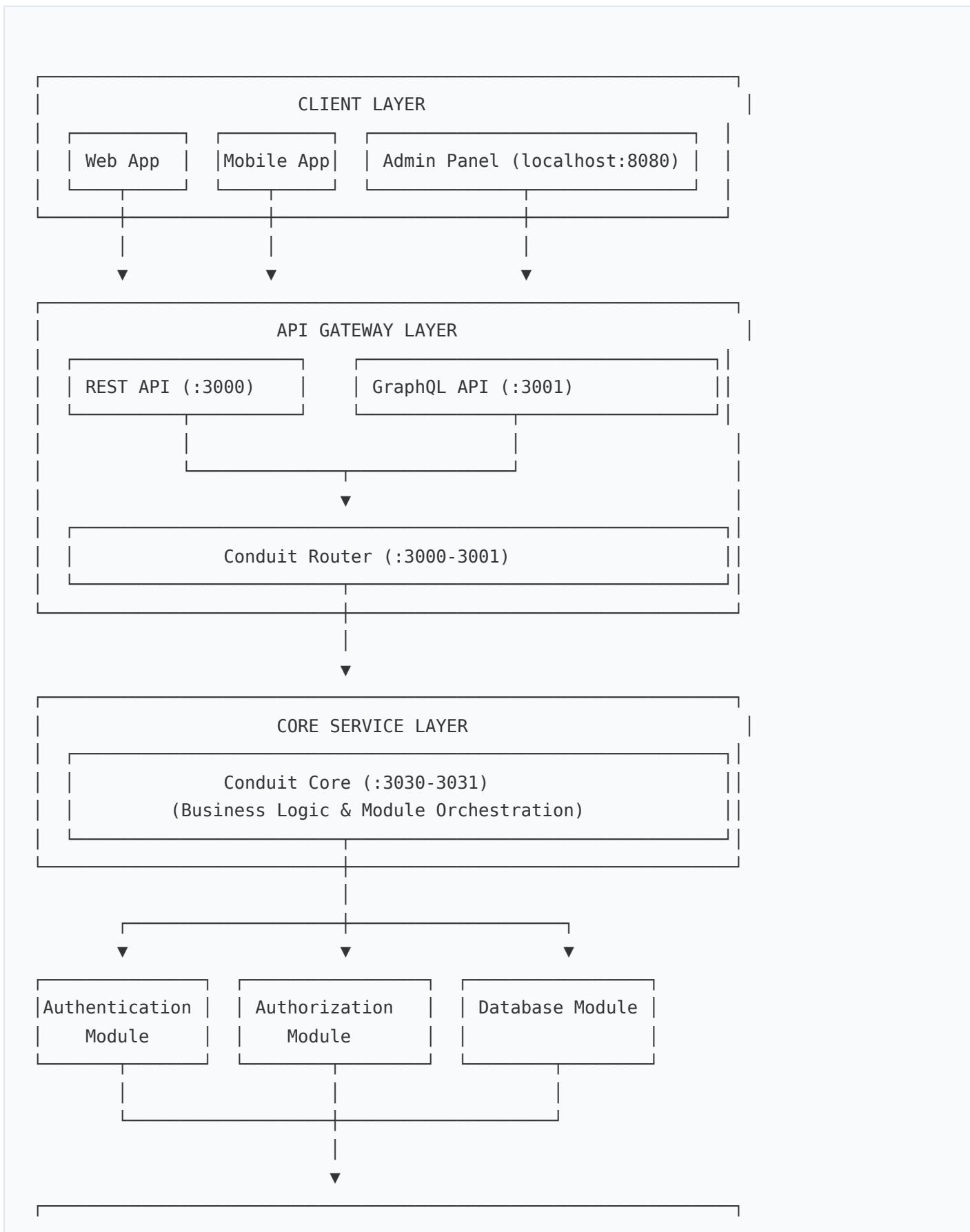
Please change these immediately after first login!

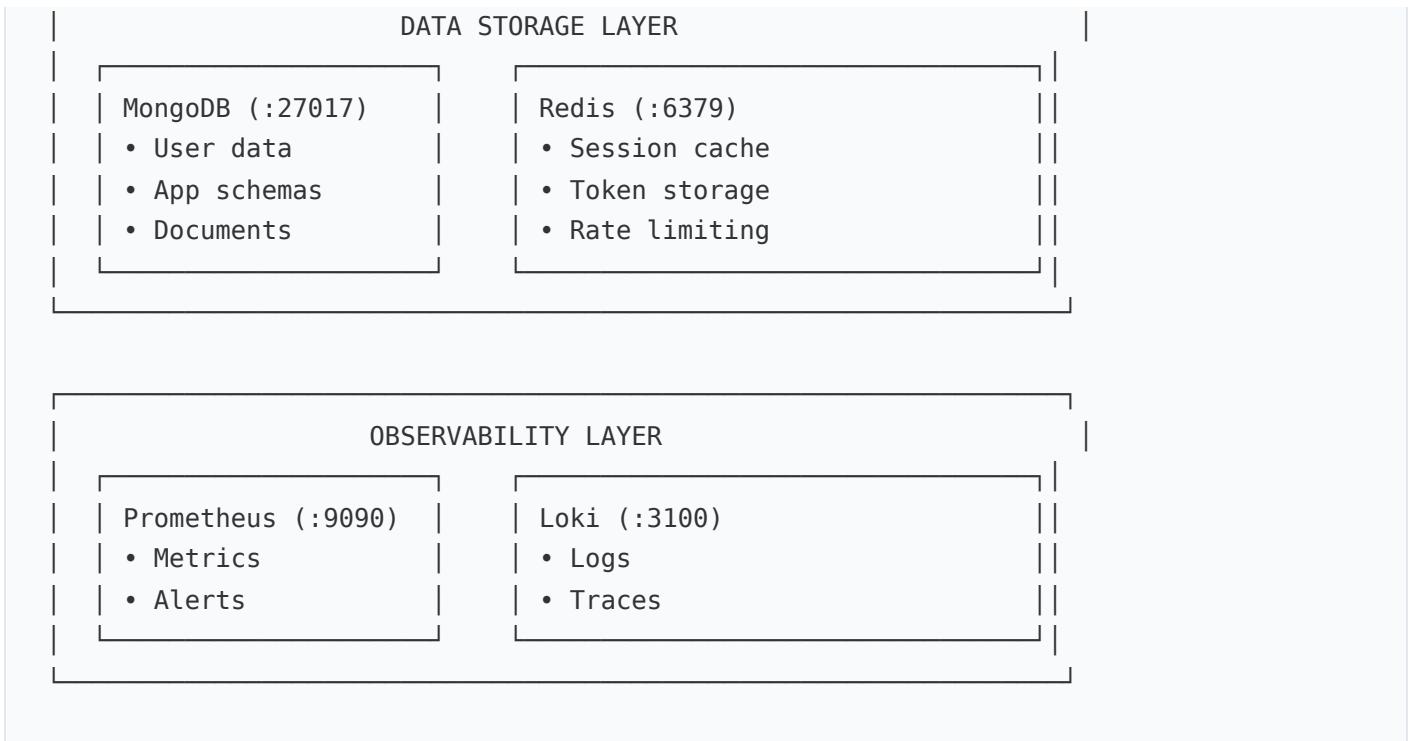
□ Running Containers

Container	Image	Purpose
conduit	<code>conduitplatform/conduit</code>	Core Conduit service
conduit-ui	<code>conduitplatform/conduit-ui</code>	Admin panel web interface
conduit-router	<code>conduitplatform/router</code>	API routing service
conduit-database	<code>conduitplatform/database</code>	Database module

conduit-authentication	conduitplatform/authentication	Authentication module
conduit-authorization	conduitplatform/authorization	Authorization module
conduit-mongo	mongo:4.4.15	MongoDB database
conduit-redis	redis:7.0.2	Redis cache
conduit-prometheus	prom/prometheus	Metrics collection
conduit-loki	grafana/loki	Log aggregation

□ Data Flow Architecture





□ Request Flow Example

1. **Client** sends request to REST (:3000) or GraphQL (:3001)
2. **Router** routes request to Conduit Core
3. **Conduit Core** orchestrates module calls
4. **Authentication** validates JWT (checks Redis for session)
5. **Authorization** verifies permissions
6. **Database Module** performs CRUD on MongoDB
7. **Response** flows back through the stack to client

□ Data Persistence

Data Type	Storage	Persistence
User accounts	MongoDB	Persistent (Docker volume)
Sessions/Tokens	Redis	In-memory (ephemeral)
Application data	MongoDB	Persistent (Docker volume)
Logs	Loki	Persistent (Docker volume)
Metrics	Prometheus	Persistent (Docker volume)

□ Quick Commands

Start Conduit

```
cd /home/mtaplists/Documents/sidtech-conduit  
docker compose up -d
```

Stop Conduit

```
docker compose down
```

View Logs

```
docker compose logs -f
```

Check Status

```
docker compose ps
```

Restart a Service

```
docker compose restart conduit
```

Update Images

```
docker compose pull  
docker compose up -d
```

□ Backup & Restore

Backup MongoDB

```
docker exec conduit-mongo mongodump --out /dump  
docker cp conduit-mongo:/dump ./backup
```

Restore MongoDB

```
docker cp ./backup conduit-mongo:/dump  
docker exec conduit-mongo mongorestore /dump
```

□ Resources

- [Conduit Documentation](#)
- [Conduit GitHub](#)
- [Docker Documentation](#)
- [SidTech Conduit Repository](#)