

CHƯƠNG I: TỔNG QUAN VÀ THIẾT KẾ HỆ THỐNG LGSP

Sơ bộ hệ thống LGSP

- [TRANG 1: Bối cảnh dự án và Tầm nhìn tích hợp](#)
- [TRANG 2: Công nghệ và Các thành phần lõi](#)
- [TRANG 3: Phân tích bài toán liên thông LGSP - NGSP](#)
- [TRANG 4: Chi tiết Cơ sở dữ liệu vận hành \(Core Schema\)](#)

TRANG 1: Bối cảnh dự án và Tầm nhìn tích hợp

1.1. Hiện trạng và Thách thức

Trong kỷ nguyên số, các cơ quan nhà nước tại địa phương thường vận hành các hệ thống phần mềm độc lập (Silos). Điều này dẫn đến việc dữ liệu bị chia cắt, không đồng bộ và gây khó khăn cho người dân khi thực hiện thủ tục hành chính liên thông. Dự án LGSP (Local Government Service Platform) ra đời để giải quyết các nút thắt này.

1.2. Tầm nhìn của trục LGSP

- Trở thành bộ não điều phối:** Kết nối mọi yêu cầu từ Cổng Dịch vụ công tới các CSDL quốc gia.
- Chuẩn hóa thông tin:** Đảm bảo mọi bản ghi (người dùng, doanh nghiệp, văn bản) đều tuân theo cấu chuẩn chung của Bộ Thông tin & Truyền thông.
- Bảo mật và Giám sát:** Mọi giao dịch đi qua trục đều được kiểm soát quyền truy cập và lưu vết (Logging) phục vụ hậu kiểm.

TRANG 2: Công nghệ và Các thành phần lỗi

2.1. Nền tảng WSO2 Micro Integrator (MI)

4.4.0

Hệ thống sử dụng WSO2 MI - một nền tảng mã nguồn mở mạnh mẽ nhất thế giới về ESB. Các ưu điểm chính:

- **Tốc độ xử lý:** Khả năng xử lý hàng ngàn giao dịch/giây với cơ chế non-blocking I/O.
- **Linh hoạt:** Hỗ trợ mọi loại giao thức từ REST, SOAP đến các File System, Kafka, RabbitMQ.
- **Microservices-ready:** Đóng gói gọn nhẹ trong Docker container, dễ dàng triển khai trên cụm K8s.

2.2. Các thành phần phụ thuộc (Middleware)

- **Broker (RabbitMQ):** Đóng vai trò làm "kho đệm" cho các văn bản điện tử quan trọng. Đảm bảo tin nhắn không bị mất khi hệ thống đích quá tải.
- **Streaming (Kafka):** Sử dụng để đẩy các sự kiện log về trung tâm phân tích dữ liệu lớn.
- **Database (MySQL):** Lưu trữ cấu hình runtime và nhật ký vận hành.

TRANG 3: Phân tích bài toán liên thông LGSP - NGSP

3.1. Mô hình kết nối Liên thông Quốc gia (NGSP)

Dự án thực hiện vai trò là Gateway trung gian giữa Địa phương và Trung ương:

1. **Hướng đi lên (Upstream):** Các app địa phương gọi API qua LGSP -> LGSP thực hiện ký số/xác thực -> Gửi lên NDXP/NGSP.
2. **Hướng đi xuống (Downstream):** Dữ liệu từ các Bộ/Ngành gửi về qua NGSP -> LGSP tiếp nhận -> Phân phối về đúng phòng ban/sở ngành tương ứng.

3.2. Đảm bảo tính tin cậy tuyệt đối (Guaranteed Delivery)

Đối với văn bản điện tử (EDXML), hệ thống không chỉ gọi API mà còn thực hiện:

- **Lưu kho (Store):** Lưu payload vào Message Store.
- **Thử lại (Retry):** Nếu gửi thất bại, Message Processor sẽ tự động thử lại theo cơ chế Exponential Backoff (thời gian giãn cách tăng dần).
- **Cảnh báo (Alert):** Nếu sau N lần thử vẫn thất bại, hệ thống sẽ chuyển vào Queue lỗi để Admin xử lý thủ công.

TRANG 4: Chi tiết Cơ sở dữ liệu vận hành (Core Schema)

Hệ thống yêu cầu các bảng dữ liệu sau trong `ESB_DB` để có thể vận hành các tính năng nâng cao:

4.1. Quản lý trạng thái Khóa (sync_lock)

Bảng này cực kỳ quan trọng để đảm bảo tính duy nhất của các tiến trình đồng bộ (Task).

- **Cấu trúc SQL:**

```
CREATE TABLE `sync_lock` (  
  `lock_key` varchar(50) NOT NULL COMMENT 'Mã khóa duy nhất của Task',  
  `is_locked` tinyint(1) DEFAULT '0' COMMENT '1: Đang bận, 0: Sẵn sàng',  
  `locked_at` datetime DEFAULT NULL COMMENT 'Thời điểm bắt đầu chạy',  
  PRIMARY KEY (`lock_key`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

4.2. Quản lý Lập lịch Task động (job_schedule)

Cho phép Admin cấu hình lịch chạy mà không cần sửa code WSO2.

- **Cấu trúc SQL:**

```
CREATE TABLE `job_schedule` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `service_code` varchar(100) NOT NULL COMMENT 'Mã nghiệp vụ (Vd: DM_TTHC)',  
  `cron_expression` varchar(50) NOT NULL COMMENT 'Ví dụ: 0 0 * * * ? (Chạy mỗi giờ)',  
  `sequence_name` varchar(100) NOT NULL COMMENT 'Tên Sequence xử lý chính',  
  `sync_endpoint_path` text COMMENT 'URL API đích',  
  `last_run` datetime DEFAULT NULL COMMENT 'Lần chạy thành công cuối cùng',  
  `status` tinyint(1) DEFAULT '1' COMMENT '1: Kích hoạt, 0: Tái kích hoạt',
```

```
PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

4.3. Quản lý Nhật ký truy cập (api_logs)

- **Cấu trúc SQL:**

```
CREATE TABLE `api_logs` (  
  `id` bigint(20) NOT NULL AUTO_INCREMENT,  
  `api_name` varchar(100) DEFAULT NULL,  
  `request_time` datetime DEFAULT NULL,  
  `log_level` varchar(20) DEFAULT NULL,  
  `message` text DEFAULT NULL,  
  `serviceCode` varchar(100) DEFAULT NULL,  
  `systemCode` varchar(100) DEFAULT NULL,  
  `headers` text DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```