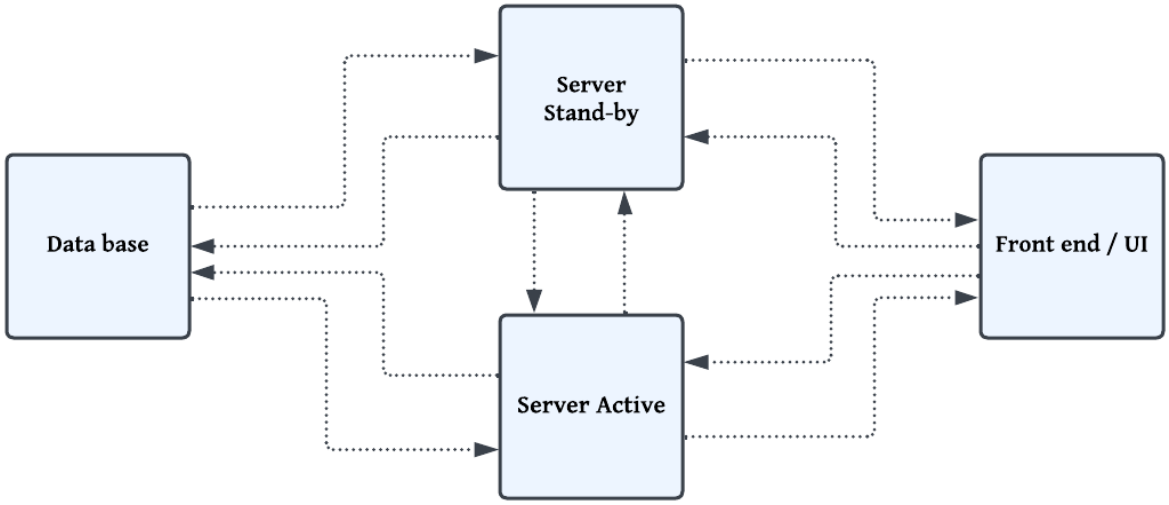
Architecture



1. Front-End Development

A. Responsibilities

1. **Dashboard Design**
   - Real-time monitoring of Baggage Handling System (BHS) status, including flight schedules, baggage flow rates, and error reports.
   - Dynamic visualizations for system operation data.
2. **User Interfaces**
   - Create interfaces for inputting and editing flight schedules.
   - Develop forms for generating and exporting statistical reports.
3. **Operational Data Display**
   - Display flight-wise baggage information dynamically.
   - Show alerts and notifications for system faults or emergencies.
4. **Responsive Design**
   - Ensure the interface is optimized for use on multiple device types (desktop, tablets).
5. **User Roles and Permissions**
   - Create login and role-based access control for operators and administrators.

B. Tools and Technologies

- **Frameworks**: React or Angular
- **Styling**: CSS, Bootstrap, or Tailwind for responsive design.

- **APIs**: Integration with back-end APIs for dynamic data updates.

2. Back-End Development

The back-end will handle the data processing logic, system integration, and communication with external systems.

A. Responsibilities

1. **Core Functionality**
    - Implement logic for flight schedule management, baggage tracking, and statistical analysis.
    - Develop modules for error handling and logging.
2. **System Integration**
    - Interface with external systems (FIS, DCS, CUPPS, Security Systems, BMS, MCS, etc.)
    - Ensure data exchange with these systems using protocols such as REST, SOAP, or specific middleware.
3. **Real-Time Data Processing**
    - Handle real-time data streams for baggage flow monitoring and system health status updates.
4. **API Development**:
    - Create RESTful APIs to serve data to the front end and external systems.
    - Ensure APIs are secure and optimized for high performance.
5. **Redundancy and Failover**
    - Implement server-side redundancy to ensure 24/7 availability..

B. Tools and Technologies

- **Programming Languages**: Python (Flask/Django), Java (Spring Boot).

---

3. Database Development

The database will store all operational, statistical, and system-related data locally.

A. Responsibilities

1. **Table Design**
    - Design tables for flight schedules, baggage data, error logs, system status, and report generation.
    - Ensure normalization to avoid redundancy.
2. **Data Storage**

    - Store real-time operational data, including baggage tracking and flight schedules.
    - Maintain historical data for generating statistical reports.
3. **Backup and Recovery**
    - Implement regular automated backups with redundancy to prevent data loss.
4. **Performance Optimization**
    - Optimize queries for real-time data access and reporting.
5. **Security and Compliance**
    - Implement role-based access control for database operations.

B. Tools and Technologies

- **Database Management System**: PostgreSQL, MySQL, or Microsoft SQL Server.
- **Backup Tools**: pg_dump (PostgreSQL), mysqldump (MySQL), or equivalent tools for automated backups.
- **Monitoring Tools**: pgAdmin, phpMyAdmin, or custom monitoring scripts.

Storage calculation

Data Type	Record Size (KB)	Frequency	Daily Records	Daily Storage (MB)	Yearly Storage (GB)	5-Year Storage (GB)
Baggage Tracking Data	1 KB	100 baggage items/hour	2,400	2.4 MB	0.876 GB	4.38 GB
Flight Schedule Data	5 KB	100 flights/day	100	0.5 MB	0.183 GB	0.92 GB
System Status Updates	0.5 KB	10 updates/minute	14,400	7.2 MB	2.628 GB	13.14 GB
Error Logs	2 KB	50 errors/day	50	0.1 MB	0.036 GB	0.18 GB
External Systems Communication Logs	1 KB	1 message/minute per system	14,400 * 8	115.2 MB	42.048 GB	210.24 GB
Statistical Reports	10 KB	1 report/day	1	0.01 MB	0.004 GB	0.02 GB

Step 2: Total Storage Calculation

Daily Storage:

- Total daily storage = $2.4 + 0.5 + 7.2 + 0.1 + 115.2 + 0.01 = 125.41$ MB/day

Yearly Storage:

- Total yearly storage = $125.41 \times 365 = 45,774.65$ MB/year = **45.77 GB/year**

5-Year Storage:

- Total 5-year storage = $45.77 \times 5 = 228.85$ GB

Server Selection

Category	Details
Processor	Intel Xeon E-2300 series, up to 8 cores and 16 threads, base clock speed 3.0 GHz or higher, 16 MB L3 cache.
Memory	Up to 64 GB DDR4 ECC (Error-Correcting Code) memory, 4 DIMM slots.
Storage	Up to 4 x 3.5" HDDs or 4 x 2.5" SSDs, RAID support with integrated PERC H345 controller, maximum capacity up to 64 TB.
Networking	Dual 1GbE integrated NICs, additional PCIe slots for higher bandwidth NICs.
Expansion	1 x PCIe Gen 4.0 slot for add-ons like network cards or storage controllers.
USB Ports	Front and rear USB 3.0 and USB 2.0 ports for connectivity.