FindMyLeak - Use Case Diagram

FindMyLeak System

Frontend Layer (React)

- Search for Data Breach
- View Breach Results
- Request Data Deletion
- Navigate Pages

Backend Layer (Spring Boot + OSINT Framework)

- Process Scan Request
- Check Cache (< 1 hour)
- Execute OSINT Scan Framework
- Return Cached Results
- Log Search History
- Save Breach Data
- Process Delete Request
- Manage Delete Requests

POST /api/scan
If not cached
If cached
POST /api/delete-request
API Call

MongoDB Database

- Store Breach Records
- Store Search History
- Store Delete Requests

User

Admin

Breach Databases & OSINT Sources

User Flow:
1. Enter email/phone
2. System scans
3. View results
4. Request deletion

Cache Strategy:
• Check MongoDB first
• If found & < 1 hour
• Return cached data
• Saves searching costs

OSINT Integration:
• Searches publicly available resources
• Searches breach DBs
• Returns breach list:
  - Breach names
  - Leaked data types
  - Record counts

MongoDB Collections:
• breach_data
• search_history
• delete_requests

Legend

- Frontend
- Backend
- Database