









<p align="center"> </p> <p align="center"> </p>

SIMBAP - Sistem Informasi Manajemen Berita Acara Pemeriksaan

SIMBAP (Sistem Informasi Manajemen Berita Acara Pemeriksaan) adalah aplikasi berbasis web yang dikembangkan untuk Balai Besar POM di Semarang. Aplikasi ini dirancang untuk mendigitalisasi dan mengotomatisasi proses pembuatan, pengelolaan, dan pelaporan Berita Acara Pemeriksaan (BAP) secara efisien dan terstruktur.

Tentang Aplikasi

SIMBAP memungkinkan petugas untuk:

-  **Membuat Berita Acara Digital** - Form wizard multi-step untuk input data terstruktur
-  **Tanda Tangan Digital** - Integrasi signature pad untuk TTD petugas langsung di aplikasi
-  **Generate PDF Otomatis** - Export BAP ke format PDF dengan layout resmi BPOM
-  **Export Rekapitulasi** - Export data ke Excel/PDF untuk pelaporan berkala
-  **Pencarian & Filter Cepat** - DataTables server-side dengan filtering berdasarkan tahun dan petugas
-  **Multi-role Management** - Akses berbeda untuk Admin dan Petugas
-  **Activity Logging** - Tracking lengkap setiap perubahan data di sistem
-  **Authorization Policy** - Kontrol akses granular per dokumen

Fitur Utama

1. Manajemen Berita Acara Pemeriksaan

- Form input dengan 5 tab: Kelengkapan, Petugas, Objek, Hasil, dan Kepala
- Auto-fill data petugas dari database
- Multi-petugas dalam satu BAP
- Validasi frontend & backend dengan feedback real-time
- Unsaved changes detection untuk mencegah data loss

2. Tanda Tangan Digital

- Draw signature dengan Signature Pad (canvas)
- Upload gambar tanda tangan (PNG/JPG)
- Preview signature sebelum disimpan
- Penyimpanan secure di private storage
- Image optimization otomatis (resize & compress)

3. Export & Cetak Dokumen

- **PDF Individual:** Cetak BAP per dokumen dengan format resmi
- **PDF Rekapitulasi:** Export daftar BAP dalam periode tertentu
- **Excel Export:** Download data ke format .xlsx untuk analisis
- Watermark header & footer BPOM otomatis
- Support konversi teks ke terbilang (tanggal dalam huruf)

4. Role & Permission System

- **Admin:** Full access (CRUD semua data, lihat semua BAP, hapus dokumen, activity log)
- **Petugas:** Restricted access (hanya lihat & edit BAP yang melibatkan dirinya)
- Policy-based authorization dengan Laravel Gates

5. Activity Logging

- Track semua aktivitas: Create, Update, Delete
- Snapshot data sebelum dan sesudah perubahan
- Filter log berdasarkan tanggal
- Informasi lengkap: pelaku, waktu, dan detail perubahan

6. Advanced Search & Filter

- Server-side DataTables untuk performa optimal
- Search by: No. Surat Tugas, Nama Objek, Nama Petugas
- Filter by: Tahun Pemeriksaan, Petugas (untuk admin)
- Sort by: Tanggal Pemeriksaan, Tanggal Buat BAP

Teknologi yang Digunakan

Backend

- **Laravel 12** - PHP Framework modern
- **PHP 8.2+** - Latest PHP version with performance improvements
- **MySQL 8.0** - Relational database

Frontend

- **Tailwind CSS 4** - Utility-first CSS framework
- **Bootstrap 3.4** - Legacy components (DataTables)
- **Vite** - Fast build tool
- **jQuery 3.6** - DOM manipulation & AJAX
- **Signature Pad** - Canvas-based signature drawing

Key Packages

- **barryvdh/laravel-dompdf** (v3.1) - PDF generation
- **intervention/image** (v2) - Image manipulation & optimization
- **maatwebsite/excel** (v3.1) - Excel export/import
- **spatie/laravel-activitylog** (v4.10) - Activity tracking
- **yajra/laravel-datatables** (v12.0) - Server-side DataTables

Instalasi

Prasyarat

Pastikan sistem Anda memiliki:

- **Docker & Docker Compose** (recommended) atau
- PHP 8.2+, Composer, Node.js 20+, MySQL 8.0

Opsi 1: Instalasi dengan Docker (Recommended)

1. Clone Repository

```
git clone https://github.com/daryihsan/beritaAcara.git
cd beritaAcara
```

2. Copy Environment File

```
cp .env.example .env
```

Edit **.env** dan pastikan konfigurasi database sudah sesuai:

```
DB_CONNECTION=mysql
DB_HOST=bap-db      # Nama service di docker-compose
DB_PORT=3306
DB_DATABASE=db_bap
DB_USERNAME=root
DB_PASSWORD=rootpassword
```

3. Build & Run Docker Containers

```
docker-compose up -d --build
```

Services yang akan berjalan:

- **bap-app** (Laravel) → http://localhost:8080
- **bap-db** (MySQL) → localhost:4406
- **bap-phpmyadmin** (phpMyAdmin) → http://localhost:8081

4. Install Dependencies & Setup Database

```
# Masuk ke container
docker exec -it bap-laravel bash
```

```
# Install PHP dependencies
composer install
```

```
# Generate app key
php artisan key:generate
```

```
# Run migrations
php artisan migrate
```

```
# (Optional) Seed default users
php artisan db:seed
```

```
# Install & build frontend assets
npm install
npm run build
```

```
# Exit container
exit
```

5. Set Permissions (Jika Error)

```
docker exec -it bap-laravel bash
chmod -R 775 storage bootstrap/cache
chown -R www-data:www-data storage bootstrap/cache
exit
```

6. Akses Aplikasi

- **Web App**: http://localhost:8080
- **phpMyAdmin**: http://localhost:8081

Opsi 2: Instalasi Manual (Tanpa Docker)

1. Clone Repository

```
git clone https://github.com/daryihsan/beritaAcara.git  
cd beritaAcara
```

2. Install Dependencies

```
# Install PHP dependencies  
composer install
```

```
# Install JavaScript dependencies  
npm install
```

3. Konfigurasi Environment

```
# Copy file environment  
cp .env.example .env
```

```
# Generate application key  
php artisan key:generate
```

Edit file `.env`:

```
DB_CONNECTION=mysql  
DB_HOST=127.0.0.1  
DB_PORT=3306  
DB_DATABASE=db_bap  
DB_USERNAME=root  
DB_PASSWORD=
```

4. Setup Database

```
# Buat database di MySQL  
mysql -u root -p  
CREATE DATABASE db_bap;  
exit;
```

```
# Jalankan migrasi  
php artisan migrate
```

```
# (Optional) Seed default users  
php artisan db:seed
```

5. Build Assets

```
# Development  
npm run dev
```

```
# Production
npm run build
```

6. Set Permissions

```
# Linux/Mac
```

```
chmod -R 775 storage bootstrap/cache
```

```
# Windows (PowerShell as Admin)
```

```
icacls storage /grant Everyone:(OI)(CI)F /T
```

```
icacls bootstrap/cache /grant Everyone:(OI)(CI)F /T
```

7. Jalankan Aplikasi

```
php artisan serve
```

Aplikasi akan berjalan di <http://localhost:8000>

Default Credentials

Setelah menjalankan seeder, Anda bisa login dengan:

Admin:

NIP: (sesuai data seed)

Password: (sesuai data seed)

Petugas:

NIP: (sesuai data seed)

Password: (sesuai data seed)

Note: Untuk keamanan, segera ubah password default setelah login pertama kali.

Struktur Direktori Utama

beritaAcara/

```
|— app/
|   |— DataTransferObjects/ # DTO untuk mapping data
|   |— Exceptions/         # Custom exception handling
|   |— Exports/            # Export Excel classes
|   |— Helpers/            # Helper functions (Date, Terbilang)
|   |— Http/
|   |   |— Controllers/    # Business logic
|   |   |— Middleware/     # Custom middleware (Role)
```

├── Requests/	# Form validation
├── Models/	# Eloquent models
├── Policies/	# Authorization policies
├── Services/	# Service layer (BAP, PDF, Image)
├── database/	
│ ├── migrations/	# Database schema
│ └── seeders/	# Sample data
├── public/	
│ ├── assets/img/	# Images (logo, header, footer PDF)
│ └── build/	# Compiled assets
├── resources/	
│ ├── css/	# Stylesheets
│ ├── js/	# JavaScript modules
│ └── views/	# Blade templates
├── routes/	
│ └── web.php	# Web routes
├── storage/	
│ └── app/private/	# Private file storage (signatures)
├── docker-compose.yml	# Docker orchestration
├── Dockerfile	# Docker image definition
└── vite.config.js	# Vite configuration

Cara Penggunaan

Membuat Berita Acara Baru

1. **Login** ke aplikasi
2. Klik menu "**Berita Acara Baru**" di sidebar
3. Isi form sesuai 5 tab:
 - **Kelengkapan:** No. & Tgl Surat Tugas, Tanggal Pemeriksaan
 - **Petugas:** Tambah petugas pemeriksa + tanda tangan digital
 - **Objek:** Data tempat yang diperiksa
 - **Hasil:** Detail temuan pemeriksaan
 - **Kepala:** Jabatan penandatanganan (Kepala Balai / Plh / Plt)
4. Klik "**Simpan & Cetak BAP**"
5. PDF akan otomatis terbuka di tab baru

Mengedit Berita Acara

- **Petugas:** Hanya bisa edit BAP yang melibatkan dirinya
- **Admin:** Bisa edit semua BAP
- Klik tombol "**Edit**" pada daftar BAP
- Lakukan perubahan → "**Simpan Perubahan BAP**"

Export Rekapitulasi

1. Buka menu **"Daftar Berita Acara"**
2. Pilih **Tahun** dan **Petugas** (untuk admin)
3. Klik tombol **"Excel"** atau **"PDF"**
4. File akan otomatis terdownload

Melihat Activity Log (Admin Only)

1. Buka menu **"Log Aktivitas"**
2. Filter berdasarkan tanggal (opsional)
3. Lihat detail perubahan pada kolom "Detail Perubahan"



Keamanan

- **Authentication:** Session-based login dengan throttling (5 attempts/minute)
- **Authorization:** Policy-based access control per dokumen
- **File Upload:** Image validation (PNG/JPG, max 300KB)
- **Private Storage:** Tanda tangan disimpan di `storage/app/private` (tidak bisa diakses langsung)
- **Secure File Serving:** Route khusus dengan middleware auth untuk serve signature images
- **CSRF Protection:** Token validation pada semua form
- **SQL Injection Prevention:** Eloquent ORM dengan prepared statements
- **XSS Prevention:** Blade templating dengan auto-escaping



Testing

```
# Run all tests  
php artisan test
```

```
# Run specific test  
php artisan test --filter=NamaTest
```



Performa & Optimasi

- **DataTables Server-Side:** Menangani ribuan record tanpa lag
- **Lazy Loading:** Images & signature dimuat on-demand
- **Asset Optimization:** Vite bundling & minification
- **Database Indexing:** Index pada kolom yang sering di-query
- **Image Compression:** Signature otomatis di-resize & compress
- **Caching:** Config & route caching untuk production



Deployment Production

1. Optimize Application


```
# Masuk ke container (jika pakai Docker)
docker exec -it bap-laravel bash
```

```
# Clear & cache config
php artisan config:clear
php artisan config:cache
```

```
# Cache routes
php artisan route:cache
```

```
# Cache views
php artisan view:cache
```

```
# Build production assets
npm run build
```

```
exit
```

2. Update Environment

Edit `.env`:

```
APP_ENV=production
APP_DEBUG=false
```

3. Set Permissions

```
chmod -R 755 storage bootstrap/cache
```

4. Backup Database Regular

```
# Export database
docker exec bap-mysql mysqldump -u root -prootpassword db_bap > backup.sql
```

```
# Import database
docker exec -i bap-mysql mysql -u root -prootpassword db_bap < backup.sql
```



Troubleshooting

Error: "500 Internal Server Error"

```
# Cek log
tail -f storage/logs/laravel.log
```

```
# Clear cache
```

```
php artisan cache:clear  
php artisan config:clear  
php artisan view:clear
```

Error: "Permission Denied" pada storage

```
chmod -R 775 storage bootstrap/cache  
chown -R www-data:www-data storage bootstrap/cache
```

DataTables tidak muncul

- Cek console browser untuk JavaScript errors
- Pastikan jQuery & DataTables loaded (inspect network tab)
- Clear browser cache

PDF tidak generate

- Cek memory limit di `php-custom.ini` (minimal 512M)
- Cek log: `storage/logs/laravel.log`
- Pastikan folder `storage/app/private` writable



Changelog

Lihat [CHANGELOG.md](#) untuk riwayat perubahan lengkap.



Kontribusi

Kontribusi sangat diterima! Untuk berkontribusi:

1. Fork repository ini
2. Buat branch baru (`git checkout -b feature/AmazingFeature`)
3. Commit perubahan (`git commit -m 'Add some AmazingFeature'`)
4. Push ke branch (`git push origin feature/AmazingFeature`)
5. Buat Pull Request

Coding Standards:

- Follow PSR-12 coding style
- Write meaningful commit messages
- Add comments untuk logic yang kompleks
- Test sebelum submit PR



Lisensi

Project ini menggunakan lisensi [MIT License](#).



Developer

Dary Ihsan

- GitHub: [@daryihsan](#)
- Project: <https://github.com/daryihsan/beritaAcara>



Acknowledgments

- [Laravel](#) - The PHP Framework for Web Artisans
- [Tailwind CSS](#) - Utility-first CSS framework
- [DataTables](#) - Advanced table plugin
- [Signature Pad](#) - HTML5 canvas based signature drawing
- [DomPDF](#) - HTML to PDF converter
- [Spatie](#) - Activity logging package

Dikembangkan dengan ❤️ untuk Balai Besar POM
Semarang
 Badan Pengawas Obat dan Makanan Republik
Indonesia </p>



Support

Jika menemukan bug atau butuh bantuan:

1. Buka [Issues](#)
2. Jelaskan masalah secara detail
3. Sertakan screenshot jika perlu
4. Tim akan merespon secepatnya


Sistem Requirements:

- PHP >= 8.2
- MySQL >= 8.0
- Docker >= 20.10 (jika pakai Docker)
- Composer >= 2.6
- Node.js >= 20.x
- Memory >= 512MB

Browser Support:

- Chrome/Edge (Recommended)
- Firefox
- Safari

- Opera

 **Catatan:** Aplikasi ini dioptimalkan untuk desktop. Mobile responsive sudah didukung tapi experience terbaik tetap di desktop/laptop.