

# PERANCANGAN DATABASE PENDAFTARAN IMUNISASI BAYI BERBASIS ONLINE

Made Wira Putra Dananjaya<sup>1)</sup>, Yadhurani Dewi Amritha<sup>2)</sup>,  
Ni Luh Putu Ika Candrawengi<sup>3)</sup> Putu Gita Pujayanti<sup>4)</sup>

Program Studi Teknologi Informasi Fakultas Teknik dan Informatika,  
Universitas Pendidikan Nasional<sup>1)2)3)</sup>

Program Studi Akuntansi, Fakultas Ekonomi,  
Universitas Pendidikan Ganesha<sup>4)</sup>

putradananjaya@undiknas.ac.id<sup>1)</sup>, yadhurani@undiknas.ac.id<sup>2)</sup>  
ikacandrawengi@undiknas.ac.id<sup>3)</sup> gita.pujayanti@student.undiksha.ac.id<sup>4)</sup>

## ABSTRACT

Infant immunization is a crucial effort in the health sector to prevent various potentially dangerous infectious diseases. An accurate and timely immunization data registration and management system is essential to ensure that every baby receives immunization on schedule. However, manual registration systems often face obstacles such as recording errors, service delays, and difficulties in monitoring immunization schedules. This study aims to design and implement an online-based infant immunization registration database system that can be accessed by parents, guardians, and health workers. This system is designed using MySQL for database management and PHP for web application development. In its design, various methods are used such as context diagrams, data flow diagrams, fishbone diagrams, database structures, triggers, and constraints to ensure data integrity and consistency. The results of the study show that this system is able to improve efficiency and accuracy in the infant immunization registration process by reducing manual errors and providing real-time monitoring of the immunization schedule. The implementation of triggers and constraints in the database helps automate several important processes, such as updating immunization status and deleting irrelevant data, as well as maintaining data quality. This system also provides easy access for parents and guardians to register their babies and for health workers to monitor and manage the immunization schedule. With the potential for further development, such as the addition of automatic notification features and integration with other health systems, this system is expected to make a significant contribution to improving the quality of infant immunization services and overall public health.

**Keywords:** Immunization, Database, Trigger, Constraint

## ABSTRAK

Imunisasi bayi merupakan upaya krusial dalam bidang kesehatan untuk mencegah berbagai penyakit menular yang berpotensi berbahaya. Sistem pendaftaran dan pengelolaan data imunisasi yang akurat dan tepat waktu sangat penting untuk memastikan setiap bayi menerima imunisasi sesuai jadwal. Namun, sistem pendaftaran manual sering menghadapi kendala seperti kesalahan pencatatan, keterlambatan layanan, dan kesulitan dalam memantau jadwal imunisasi. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem database pendaftaran imunisasi bayi berbasis online yang dapat diakses oleh orang tua, wali, dan petugas kesehatan. Sistem ini dirancang menggunakan MySQL untuk manajemen basis data dan PHP untuk pengembangan aplikasi web. Dalam perancangannya, digunakan berbagai metode seperti diagram konteks, data flow diagram, fishbone diagram, struktur database, trigger, dan constraint untuk memastikan integritas dan konsistensi data. Hasil penelitian menunjukkan bahwa sistem ini mampu meningkatkan efisiensi dan akurasi dalam proses pendaftaran imunisasi bayi dengan mengurangi kesalahan manual dan

menyediakan pemantauan jadwal imunisasi secara real-time. Implementasi trigger dan constraint pada database membantu otomatisasi beberapa proses penting, seperti memperbarui status imunisasi dan menghapus data yang tidak relevan, serta menjaga kualitas data. Sistem ini juga memberikan kemudahan akses bagi orang tua dan wali untuk mendaftarkan bayi mereka serta bagi petugas kesehatan untuk memantau dan mengelola jadwal imunisasi. Dengan potensi pengembangan lebih lanjut, seperti penambahan fitur notifikasi otomatis dan integrasi dengan sistem kesehatan lainnya, sistem ini diharapkan dapat memberikan kontribusi signifikan dalam meningkatkan kualitas pelayanan imunisasi bayi serta kesehatan masyarakat secara keseluruhan.

**Kata kunci:** Imunisasi, Database, Trigger, Constraint

## **PENDAHULUAN**

Imunisasi bayi merupakan salah satu upaya penting dalam bidang kesehatan untuk mencegah berbagai penyakit menular yang berpotensi berbahaya. Dalam beberapa dekade terakhir, imunisasi telah terbukti efektif dalam mengurangi angka kesakitan dan kematian akibat penyakit-penyakit menular seperti polio, campak, dan hepatitis B. Namun, keberhasilan program imunisasi sangat bergantung pada sistem pendaftaran dan pengelolaan data yang tepat waktu dan akurat. Sistem pendaftaran manual yang masih banyak digunakan di berbagai tempat sering kali menghadapi berbagai kendala, seperti kesalahan dalam pencatatan data, keterlambatan dalam pemberian layanan, dan kesulitan dalam memantau jadwal imunisasi (Hidayatullah, 2016). Kesalahan pencatatan dapat menyebabkan data yang tidak akurat, yang pada gilirannya dapat mengakibatkan bayi terlewat dari jadwal imunisasi yang seharusnya (Ridmadhani et al., 2019). Selain itu, keterlambatan dalam pemberian layanan imunisasi dapat meningkatkan risiko bayi terkena penyakit yang seharusnya dapat dicegah melalui imunisasi.

Dalam era digital saat ini, penggunaan teknologi informasi dalam bidang kesehatan dapat menjadi solusi yang efektif untuk mengatasi masalah-masalah tersebut. Diperlukan sebuah sistem yang dapat meningkatkan efisiensi dan akurasi pengelolaan data imunisasi (Rosidah, 2019). Penelitian ini bertujuan untuk merancang sebuah database pendaftaran imunisasi bayi berbasis online yang dapat diakses dengan mudah oleh orang tua, wali, dan petugas kesehatan. Dengan sistem ini, proses

pendaftaran dapat dilakukan secara cepat dan akurat, mengurangi risiko kesalahan pencatatan. Selain itu, sistem ini juga memungkinkan pemantauan jadwal imunisasi secara real-time, sehingga memudahkan petugas kesehatan dalam memastikan bahwa setiap bayi mendapatkan imunisasi sesuai jadwal (Catur, n.d.). Notifikasi otomatis yang disediakan oleh sistem ini juga akan membantu orang tua dan wali dalam mengingatkan jadwal imunisasi bayi mereka, mengurangi risiko keterlambatan.

Sistem database pendaftaran imunisasi bayi berbasis online yang dirancang dalam penelitian ini diharapkan dapat memfasilitasi proses pendaftaran, mengelola jadwal imunisasi, serta memberikan notifikasi secara otomatis, sehingga seluruh proses dapat berjalan lebih lancar dan terorganisir (Joko, 2023). Dengan demikian, diharapkan penelitian ini dapat memberikan kontribusi signifikan dalam meningkatkan kualitas pelayanan imunisasi bayi serta kesehatan masyarakat secara keseluruhan (Ridwan, 2020). Peningkatan efisiensi dan akurasi dalam pengelolaan data imunisasi tidak hanya akan membantu dalam memastikan setiap bayi mendapatkan perlindungan yang mereka butuhkan, tetapi juga akan mendukung program-program kesehatan masyarakat yang lebih luas dalam upaya pencegahan penyakit menular (Zaki et al., 2023).

## **METODE PENELITIAN**

### **Database**

Database adalah kumpulan data yang terstruktur yang disimpan dan dikelola untuk

memudahkan akses, pengelolaan, dan pembaruan data (ANDREESCU & Mircea, 2012). Dalam penelitian ini, database yang dirancang akan digunakan untuk menyimpan informasi tentang bayi, jadwal imunisasi, dan status imunisasi.

### MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang menggunakan SQL (Structured Query Language) sebagai bahasa untuk mengelola data (Astrahan et al., 1976). MySQL adalah pilihan yang populer untuk pengembangan aplikasi web karena kecepatan, kehandalan, dan kemudahan penggunaannya.

### API

API (Application Programming Interface) adalah sekumpulan definisi dan protokol yang memungkinkan aplikasi perangkat lunak untuk berkomunikasi satu sama lain (Albahiqa, 2022). Dalam konteks sistem pendaftaran imunisasi, API dapat digunakan untuk menghubungkan sistem ini dengan sistem lain, seperti sistem informasi kesehatan yang lebih besar.

### PHP

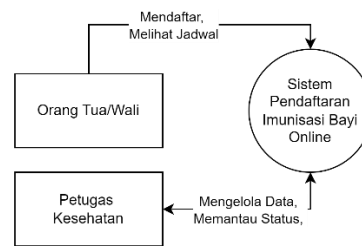
PHP adalah bahasa pemrograman skrip yang terutama digunakan untuk pengembangan aplikasi web (Ceri et al., 2003). PHP memungkinkan integrasi dengan database MySQL untuk membuat aplikasi web yang dinamis dan interaktif.

### UML

UML (Unified Modeling Language) adalah standar untuk memodelkan sistem perangkat lunak (Sekolah Tinggi Teknologi Garut et al., 2021). UML menyediakan berbagai jenis diagram untuk menggambarkan aspek struktural dan perilaku sistem, seperti class diagram, sequence diagram, dan activity diagram.

### Data Flow Diagram (DFD)

DFD Level 0 menggambarkan proses utama dalam sistem dan aliran data antara proses dan entitas eksternal (Akidau et al., 2015). Pada penelitian ini memiliki 2 entitas yang terhubung dalam 1 sistem, yaitu orang tua/wali dan petugas kesehatan. Entitas orang tua/wali memiliki akses untuk mendaftar dan melihat jadwal. Entitas petugas kesehatan dapat akses dua arah untuk mengelola data dan memantau status.



Gambar 1. DFD Level 0

### Struktur Database

Struktur database mencakup tabel-tabel utama, seperti Bayi, OrangTua, JadwalImunisasi, dan StatusImunisasi.

```

CREATE TABLE Bayi (
    id_bayi INT PRIMARY KEY,
    nama VARCHAR(100),
    tanggal_lahir DATE,
    id_orangtua INT,
    tanggal_terakhir_imunisasi DATE,
    FOREIGN KEY
    (id_orangtua) REFERENCES
    OrangTua(id_orangtua)
);

CREATE TABLE OrangTua (
    id_orangtua INT PRIMARY
    KEY,
    nama VARCHAR(100),
    alamat VARCHAR(255),
    no_telepon VARCHAR(15)
);

CREATE TABLE JadwalImunisasi (

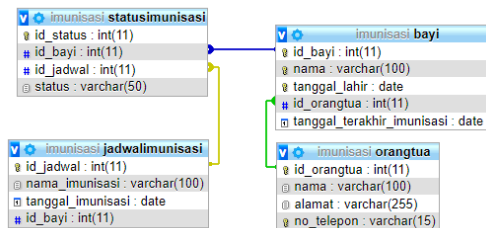
```

```

        id_jadwal INT PRIMARY
KEY,
        nama_imunisasi
VARCHAR(100),
        tanggal_imunisasi DATE,
        id_bayi INT 11,
    );

CREATE TABLE StatusImunisasi (
    id_status INT PRIMARY
KEY,
    id_bayi INT,
    id_jadwal INT,
    status VARCHAR(50),
    FOREIGN KEY (id_bayi)
REFERENCES Bayi(id_bayi),
    FOREIGN KEY (id_jadwal)
REFERENCES
JadwalImunisasi(id_jadwal)
);

```



Gambar 2. Design Database

## HASIL DAN PEMBAHASAN

### Implementasi Trigger dan Constraint

Untuk memastikan integritas dan konsistensi data dalam sistem pendaftaran imunisasi bayi berbasis online, kita dapat menggunakan trigger dan constraint pada database MySQL. Berikut adalah beberapa implementasi trigger dan constraint beserta penjelasannya.

#### Trigger

##### 1. Trigger untuk Memperbarui Status Imunisasi

Setiap kali ada penambahan jadwal imunisasi baru, trigger ini akan secara otomatis menambahkan status "Belum Imunisasi" untuk setiap bayi yang terdaftar dalam jadwal tersebut.

```
DELIMITER $$
```

```

CREATE TRIGGER
update_status_imunisasi
AFTER INSERT ON
JadwalImunisasi
FOR EACH ROW
BEGIN
    INSERT INTO
    StatusImunisasi
    (id_bayi, id_jadwal,
    status)
    VALUES
    (NEW.id_bayi,
    NEW.id_jadwal, 'Belum
    Imunisasi');
END$$
DELIMITER;

```

##### 2. Trigger untuk Memperbarui Tanggal Terakhir Imunisasi

Saat status imunisasi bayi diubah menjadi "Sudah Imunisasi", trigger ini akan memperbarui kolom `tanggal\_terakhir\_imunisasi` pada tabel `Bayi` dengan tanggal saat ini.

```

DELIMITER $$
CREATE TRIGGER
update_tanggal_terakhir_imunis
asi
AFTER UPDATE ON
StatusImunisasi
FOR EACH ROW
BEGIN
    IF NEW.status =
    'Sudah Imunisasi' THEN
        UPDATE Bayi
        SET
        tanggal_terakhir_i
munisasi =
        CURDATE()
        WHERE
        id_bayi =
        NEW.id_bayi;
    END IF;
END$$
DELIMITER;

```

##### 3. Trigger untuk Menghapus Data Status Imunisasi

Saat jadwal imunisasi dihapus, trigger ini akan memastikan semua entri terkait di tabel `StatusImunisasi` juga dihapus, menjaga agar tidak ada data yang tersisa yang tidak relevan.

```

DELIMITER $$
CREATE TRIGGER
delete_status_imunisasi
AFTER DELETE ON
JadwalImunisasi
FOR EACH ROW
BEGIN
    DELETE FROM
    StatusImunisasi
    WHERE id_jadwal =
    OLD.id_jadwal;
END$$
DELIMITER;

```

## Constraint

### 1. Primary Key Constraint

Setiap tabel memiliki kunci utama untuk memastikan bahwa setiap baris data dapat diidentifikasi secara unik

```

ALTER TABLE Bayi
ADD PRIMARY KEY
(id_bayi);

```

```

ALTER TABLE OrangTua
ADD PRIMARY KEY
(id_orangtua);

```

```

ALTER TABLE
JadwalImunisasi
ADD PRIMARY KEY
(id_jadwal);

```

```

ALTER TABLE
StatusImunisasi
ADD PRIMARY KEY
(id_status);

```

### 2. Foreign Key Constraint

Menjaga referensial integritas antara tabel dengan memastikan bahwa nilai di kolom tertentu harus ada di kolom lain (relasi antara tabel). Menjaga integritas referensial antara

tabel. Misalnya, `id\_orangtua` di tabel `Bayi` harus merujuk ke `id\_orangtua` yang ada di tabel `OrangTua`.

```

ALTER TABLE Bayi
ADD CONSTRAINT
fk_orangtua
FOREIGN KEY
(id_orangtua) REFERENCES
OrangTua(id_orangtua);

```

```

ALTER TABLE
StatusImunisasi
ADD CONSTRAINT fk_bayi
FOREIGN KEY (id_bayi)
REFERENCES Bayi(id_bayi);

```

```

ALTER TABLE
StatusImunisasi
ADD CONSTRAINT fk_jadwal
FOREIGN KEY (id_jadwal)
REFERENCES
JadwalImunisasi(id_jadwal);

```

### 3. Unique Constraint

Mencegah duplikasi data pada kolom tertentu. Misalnya, tidak boleh ada dua bayi dengan nama dan tanggal lahir yang sama dalam sistem.

```

ALTER TABLE Bayi
ADD CONSTRAINT unik_bayi
UNIQUE (nama,
tanggal_lahir);

```

```

ALTER TABLE OrangTua
ADD CONSTRAINT
unik_orangtua
UNIQUE (no_telepon);

```

### 4. Not Null Constraint

Memastikan kolom penting tidak boleh dibiarkan kosong. Misalnya, nama bayi dan tanggal lahir harus diisi.

```

ALTER TABLE Bayi
MODIFY COLUMN nama
VARCHAR(100) NOT NULL,

```

```
MODIFY COLUMN
tanggal_lahir DATE NOT NULL;
```

```
ALTER TABLE OrangTua
MODIFY COLUMN nama
VARCHAR(100) NOT NULL,
MODIFY COLUMN no_telepon
VARCHAR(15) NOT NULL;
```

### 5. Check Constraint

Memastikan nilai dalam kolom memenuhi kondisi tertentu. Misalnya, tanggal lahir bayi tidak boleh lebih dari tanggal saat ini, dan status imunisasi harus salah satu dari nilai yang diizinkan.

```
ALTER TABLE Bayi
ADD CONSTRAINT
check_tanggal_lahir
CHECK (tanggal_lahir <=
CURDATE());
```

```
ALTER TABLE
StatusImunisasi
ADD CONSTRAINT
check_status
CHECK (status IN ('Belum
Imunisasi', 'Sudah Imunisasi',
'Tertunda'));
```

Dengan implementasi trigger dan constraint ini, kita dapat memastikan bahwa data dalam sistem pendaftaran imunisasi bayi berbasis online selalu konsisten, akurat, dan memenuhi aturan yang telah ditetapkan.

### Tahapan Pengujian Trigger dan Constraint

Pengujian trigger dan constraint sangat penting untuk memastikan bahwa aturan-aturan yang diterapkan pada database berfungsi dengan benar. Berikut adalah tahapan pengujian yang dapat dilakukan:

#### Pengujian Trigger

1. Pengujian Trigger ``update_status_imunisasi``

Menambahkan entri baru ke dalam tabel ``JadwalImunisasi``. Masukkan data baru ke tabel ``JadwalImunisasi``:

```
INSERT INTO
JadwalImunisasi
(id_jadwal,
nama_imunisasi,
tanggal_imunisasi)
VALUES (1, 'BCG', '2024-
07-15');
```

Periksa tabel ``StatusImunisasi`` untuk memastikan bahwa entri baru telah ditambahkan dengan status "Belum Imunisasi":

```
SELECT * FROM
StatusImunisasi WHERE
id_jadwal = 1;
```

id_status	id_bayi	id_jadwal	status
1	1	1	Belum Imunisasi

Gambar 3. Hasil Pengujian Trigger ``update_status_imunisasi``

2. Pengujian Trigger ``update_tanggal_terakhir_imunisasi``

Mengubah status imunisasi menjadi "Sudah Imunisasi". Update status imunisasi:

```
UPDATE StatusImunisasi
SET status = 'Sudah
Imunisasi'
WHERE id_status = 1;
```

Periksa tabel ``Bayi`` untuk memastikan bahwa ``tanggal_terakhir_imunisasi`` telah diperbarui:

```
SELECT * FROM Bayi WHERE
id_bayi = (SELECT
id_bayi FROM
StatusImunisasi WHERE
id_status = 1);
```

id_bayi	nama	tanggal_lahir	id_orangtua	tanggal_terakhir_imunisasi
1	Agra	2024-07-05	1	2024-07-05

Gambar 4. Hasil Pengujian Trigger ``update_tanggal_terakhir_imuni``

sasi`

3. Pengujian Trigger  
 Menghapus entri dari tabel `JadwalImunisasi`. Hapus data dari tabel `JadwalImunisasi`:

```
DELETE FROM
JadwalImunisasi WHERE
id_jadwal = 1;
```

Periksa tabel `StatusImunisasi` untuk memastikan bahwa entri terkait telah dihapus:

```
SELECT * FROM
StatusImunisasi WHERE
id_jadwal = 1;
```

id_status	id_bayi	id_jadwal	status
-----------	---------	-----------	--------

Gambar 5. Pengujian Pengujian Trigger `delete\_status\_imunisasi`

### Pengujian Constraint

1. Pengujian Primary Key Constraint  
 Menambahkan entri dengan `id` yang sudah ada. Masukkan data duplikat ke tabel `Bayi`:

```
INSERT INTO Bayi
(id_bayi, nama, tanggal_lahir,
id_orangtua)
VALUES (1, 'Bayi Test',
'2023-07-02', 1);
```

```
1 queries executed, 0 success, 1 errors, 0 warnings
Query: INSERT INTO Bayi (id_bayi, nama, tanggal_lahir, id_orangtua) VALUES (1, 'Bayi Test', '2023-07-02', 1)
Error Code: 1062
Duplicate entry '1' for key 'PRIMARY'
Execution Time : 0 sec
Transfer Time : 0 sec
Total Time : 0 sec
```

Gambar 6. Pengujian Primary Key Constraint

2. Pengujian Foreign Key Constraint  
 Menambahkan entri dengan `id\_orangtua` yang tidak ada di tabel `OrangTua`. Masukkan data ke tabel `Bayi` dengan `id\_orangtua` yang tidak valid:

```
INSERT INTO Bayi
(id_bayi, nama, tanggal_lahir,
id_orangtua)
VALUES (2, 'Bayi Test 2',
'2023-08-15', 99);
```

```
1 queries executed, 0 success, 1 errors, 0 warnings
Query: INSERT INTO Bayi (id_bayi, nama, tanggal_lahir, id_orangtua) VALUES (2, 'Bayi Test 2', '2023-08-15', 99)
Error Code: 1452
Column 'id_orangtua' in table 'Bayi' has foreign key constraint fails ('imunisasi`.`Bayi`, CONSTRAINT `bayi_orangt` FOREIGN KEY ('id_orangtua') REFERENCES `orangtua` ('id_orangtua'))
Execution Time : 0 sec
Transfer Time : 0 sec
Total Time : 0.027 sec
```

Gambar 7. Pengujian Foreign Key Constraint

3. Pengujian Unique Constraint  
 Menambahkan entri dengan kombinasi `nama` dan `tanggal\_lahir` yang sudah ada. Masukkan data duplikat ke tabel `Bayi`:

```
INSERT INTO Bayi
(id_bayi, nama, tanggal_lahir,
id_orangtua)
VALUES (3, 'Bayi Test',
'2023-07-02', 1);
```

```
1 queries executed, 0 success, 1 errors, 0 warnings
Query: INSERT INTO Bayi (id_bayi, nama, tanggal_lahir, id_orangtua) VALUES (3, 'Bayi Test', '2023-07-02', 1)
Error Code: 1062
Duplicate entry 'Bayi Test-2023-07-02' for key 'unik_bayi'
Execution Time : 0 sec
Transfer Time : 0 sec
Total Time : 0.026 sec
```

Gambar 8. Pengujian Unique Constraint

4. Pengujian Not Null Constraint  
 Menambahkan entri dengan kolom yang wajib diisi dibiarkan kosong. Masukkan data dengan `nama` kosong:

```
INSERT INTO Bayi
(id_bayi, nama, tanggal_lahir,
id_orangtua)
VALUES (4, NULL, '2023-07-02', 1);
```

```
1 queries executed, 0 success, 1 errors, 0 warnings
Query: INSERT INTO Bayi (id_bayi, nama, tanggal_lahir, id_orangtua) VALUES (4, NULL, '2023-07-02', 1)
Error Code: 1048
Column 'nama' cannot be null
Execution Time : 0 sec
Transfer Time : 0 sec
Total Time : 0 sec
```

Gambar 9. Pengujian Not Null Constraint

5. Pengujian Check Constraint  
 Menambahkan entri yang melanggar aturan check constraint. Masukkan data dengan `tanggal\_lahir` yang melebihi tanggal saat ini:

```

INSERT INTO Bayi
(id_bayi, nama, tanggal_lahir,
id_orangtua)
VALUES (5, 'Bayi Test 3',
'2025-01-01', 1);

```

```

1 queries executed, 0 success, 1 errors, 0 warnings
Query: INSERT INTO Bayi (id_bayi, nama, tanggal_lahir, id_orangtua) VALUES (5, 'Bayi Test 3', '2025-01-01', 1)
Error Code: 1452
Cannot add or update a child row: a foreign key constraint fails ('imunitas'.bayi', CONSTRAINT bayi_inf_1 FOREIGN KEY
('id_orangtua') REFERENCES orangtua ('id_orangtua'))
Execution Time: 0 sec
Transfered Time: 0 sec
Total Time: 0.029 sec

```

Gambar 10. Pengujian Check Constraint

## DAFTAR PUSTAKA

- [1] T. Hidayatullah, "An Explanatory Study of Health Policy Agenda Setting in Indonesian Immunisation Policy for Religious Anti-Vaccination," no. Query date: 2024-07-05 03:23:25, 2016, [Online]. Available: <https://flex.flinders.edu.au/file/8feec3e3-e523-4a29-8c86-4249b2107644/1/ThesisHidayatullah2016.pdf>
- [2] R. Ridmadhani, M. Sanjaya, and ..., "Aplikasi Pengelolaan Stok Vaksin Pada Kantor Kesehatan Pelabuhan Kelas II Bandung," *EProceedings* ..., no. Query date: 2024-07-05 03:23:25, 2019, [Online]. Available: <https://openlibrarypublications.telkomuniversity.ac.id/index.php/appliedscience/article/view/9983>
- [3] O. Rosidah, *Factors associated with influenza vaccination uptake among healthcare workers in tertiary hospitals in Perak/Rosidah Omar*. studentsrepo.um.edu.my, 2019. [Online]. Available: <http://studentsrepo.um.edu.my/11590/>
- [4] B. Catur, "Identification of vaccination experience and understanding of Indonesia Society Public Health Literacy Study," *eprints.poltekkesjogja.ac.id*, no. Query date: 2024-07-05 03:23:25, [Online]. Available: [http://eprints.poltekkesjogja.ac.id/11851/2/IDENTIFICATION%20OF%20VACCINATION%20EXPERIENCE\\_CATUR%20BUDI\\_2021\\_JURNAL%20INTERNASIONAL.pdf](http://eprints.poltekkesjogja.ac.id/11851/2/IDENTIFICATION%20OF%20VACCINATION%20EXPERIENCE_CATUR%20BUDI_2021_JURNAL%20INTERNASIONAL.pdf)
- [5] T. Joko, "Penerapan Hak Akses pada Perancangan Database Akademik untuk Meningkatkan Keamanan Data," *Indones. J. Mach. Learn.* ..., no. Query date: 2024-07-05 03:23:25, 2023, [Online]. Available: <https://eprints.akprind.ac.id/2434/>
- [6] M. Ridwan, *PERANCANGAN SISTEM PENGOLAHAN DATA PASIEN RAWAT INAP DAN RAWAT JALAN YANG ADA DI PUSKESMAS BARUNG-BARUNG BALANTAI BERBASIS* .... repository.upiypk.ac.id, 2020. [Online]. Available: <http://repository.upiypk.ac.id/id/eprint/4865>
- [7] M. Zaki, A. Ramadiani, and ..., "The Impact of the Peduli Lindungi Electronic System on Human Rights and Personal Data Protection," *E3S Web Conf.*, no. Query date: 2024-07-05 03:23:25, 2023, [Online]. Available: [https://www.e3s-conferences.org/articles/e3sconf/abs/2023/25/e3sconf\\_icobar2023\\_04038/e3sconf\\_icobar2023\\_04038.html](https://www.e3s-conferences.org/articles/e3sconf/abs/2023/25/e3sconf_icobar2023_04038/e3sconf_icobar2023_04038.html)
- [8] A. ANDREESCU and M. Mircea, "Perspectives on the role of business rules in database design," *Database Syst. J.*, no. Query date: 2024-07-05 01:23:24, 2012, [Online]. Available: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=fdd91231177f657dcfb8ed10154c5ceabfcb8023>
- [9] M. Astrahan, M. Blasgen, D. Chamberlin, and ..., "System R: Relational approach to database management," ... *Database Syst.* ..., no. Query date: 2024-07-05 01:23:24, 1976, doi: 10.1145/320455.320457.
- [10] M. Albaihaqi, *RANCANG BANGUN RESTFUL API DAN WEBSITE ADMIN UNTUK APLIKASI PRECISION AGRICULTURE*. digilib.unila.ac.id, 2022. [Online]. Available: <http://digilib.unila.ac.id/67693/>
- [11] S. Ceri, P. Fraternali, A. Bongio, M. Brambilla, S. Comai, and ..., *Morgan Kaufmann series in data management systems: Designing data-intensive Web applications*. books.google.com, 2003. [Online]. Available: <https://books.google.com/books?hl=en&lr=&id=yI31Mi738p8C&oi=fnd&pg=PR7&dq=basisdata+trigger+constraint&ots=69psipSFqm&sig=0GICvPhGfleP1NFw97-7iSr8X3M>



- [12] Sekolah Tinggi Teknologi Garut, R. Umar, S. Sarjimin, A. S. Nugroho, A. Dito, and I. Gunawan, "Perancangan Sistem Informasi Keuangan Berbasis Web Multi User Dengan UML," *J. Algoritma*, vol. 17, no. 2, pp. 204–211, Feb. 2021, doi: 10.33364/algoritma/v.17-2.204.
- [13] T. Akidau, R. Bradshaw, C. Chambers, and ..., "The dataflow model: a practical approach to balancing correctness, latency, and cost in massive-scale, unbounded, out-of-order data processing," *Proc. ...*, no. Query date: 2024-07-05 01:23:24, 2015, doi: 10.14778/2824032.2824076.