

# Perancangan Back-end Aplikasi Education Management System Modul Akademik Menggunakan Metode Iterative Incremental Pada SMPN 1 Magetan

1<sup>st</sup> Muhammad Hafeezi Faruqi

*Fakultas Rekayasa Industri*

*Universitas Telkom*

Bandung, Indonesia

faruqihafiz@student.telkomuniversity.a  
c.id

2<sup>nd</sup> Faishal Mufied Al Anshary

*Fakultas Rekayasa Industri*

*Universitas Telkom*

Bandung, Indonesia

faishalmufied@telkomuniversity.ac.id

3<sup>rd</sup> Syfa Nur Lathifah

*Fakultas Rekayasa Industri*

*Universitas Telkom*

Bandung, Indonesia

syfanr@telkomuniversity.ac.id

**Abstrak**— Sistem manajemen pendidikan yang efektif menjadi semakin penting di era digital, terutama dalam mengelola berbagai aspek akademik dan komunikasi antara sekolah, guru, siswa, dan orang tua. Pengelolaan data akademik yang kompleks dan kebutuhan komunikasi yang terus berkembang memerlukan solusi teknologi yang mampu menjawab tantangan tersebut. Sistem yang dapat menyediakan akses yang mudah, transparan, dan terintegrasi untuk semua pemangku kepentingan pendidikan dapat membantu meningkatkan kinerja dan efektivitas administrasi sekolah. Penelitian ini bertujuan merancang back-end aplikasi Education Management System (EMS) modul akademik menggunakan metode Iterative Incremental di SMPN 1 Magetan untuk meningkatkan efisiensi akademik dan komunikasi antara sekolah, guru, siswa, dan orang tua. Penelitian melibatkan wawancara dengan 5 guru, 5 siswa, dan 5 wali murid untuk mengidentifikasi kebutuhan pengguna. Fitur aplikasi berfokus pada kegiatan akademik. Implementasi back-end menggunakan Go dan PostgreSQL, dengan arsitektur Domain-Driven Design untuk fleksibilitas dan skalabilitas. Hasil penelitian menunjukkan aplikasi EMS memenuhi kebutuhan pengguna, meningkatkan transparansi data akademik, serta mempermudah administrasi dan komunikasi di SMPN 1 Magetan. Pengembangan back-end melalui dua fase Iterative Incremental. Pengujian API menunjukkan rata-rata waktu respon 416 milidetik dan tingkat error 0%, menandakan API siap digunakan optimal. Penelitian ini diharapkan memberikan solusi teknologi modern dan efisien untuk pendidikan menengah di Indonesia.

**Kata Kunci**— Education Management System, Iterative Incremental, Go, PostgreSQL, Domain-Driven Design, Modul AkademikPendahuluan

## I. PENDAHULUAN

Pendidikan adalah aspek penting dalam perkembangan sumber daya manusia dan Masyarakat. Proses pendidikan di sekolah tidak hanya menjadi tanggung jawab guru dan siswa, tetapi juga melibatkan peran aktif orang tua. Keterlibatan orang tua dalam pendidikan siswa memiliki dampak yang signifikan terhadap kemajuan dan perkembangan akademis anak-anak mereka [1]. Oleh karena itu, memfasilitasi

keterlibatan orang tua dalam proses pendidikan menjadi suatu hal yang sangat penting. Transformasi pendidikan menjadi digital menjadi suatu keharusan. Karena dapat memastikan pendidikan yang inklusif, inovatif, dan terkini. Pendidikan digital memungkinkan akses yang lebih mudah dan cepat ke berbagai sumber belajar, memfasilitasi penggunaan teknologi dalam proses pembelajaran, dan memberikan kesempatan bagi pengembangan pendidikan yang lebih adaptif dan responsif terhadap kebutuhan siswa [2]. Aplikasi pendidikan memiliki potensi besar untuk meningkatkan interaksi antara sekolah, guru, siswa, dan orang tua. Aplikasi tersebut memungkinkan akses yang mudah dan cepat ke informasi yang diperlukan, memfasilitasi pemantauan kemajuan siswa, serta meningkatkan keterlibatan orang tua dalam pendidikan anak-anak mereka [3].

Permasalahan di SMPN 1 Magetan terkait kesulitan akses data akademik dan absensi serta komunikasi yang kurang efektif antara sekolah dan orang tua dapat diatasi dengan pengembangan aplikasi Education Management System (EMS). Aplikasi berbasis *website* ini menawarkan solusi dengan mempermudah akses *real-time* bagi orang tua untuk memantau perkembangan akademis anak-anak mereka, serta meningkatkan komunikasi antara sekolah dan orang tua. Dengan integrasi teknologi, EMS menyederhanakan proses pendidikan yang masih manual, memperbaiki interaksi, dan mendukung partisipasi orang tua dalam pendidikan anak-anak mereka, sehingga berpotensi meningkatkan kemajuan siswa.

## II. KAJIAN TEORI

### A. Education Management System

Sistem Manajemen Pendidikan (Educational Management System) memegang peran penting dalam meningkatkan efisiensi dan efektivitas lembaga pendidikan. Beberapa penelitian menyoroti implementasi sistem manajemen pendidikan sebagai pendukung dalam administrasi sekolah dan pemantauan kemajuan siswa [4].

### B. Iterative Incremental

Iterative Incremental Development adalah proses pengembangan perangkat lunak yang melibatkan pengulangan siklis untuk mencapai hasil yang diinginkan. Model ini merupakan pendahulu dari Agile dan menekankan analisis serta desain berkualitas untuk jaminan mutu. Pengembangan inkremental menambah kuantitas, ukuran, atau fungsionalitas produk secara bertahap, sementara proses iteratif mengulangi langkah-langkah kecil secara sistematis. Setiap iterasi bertujuan menghasilkan perangkat lunak yang lebih baik, fungsional, teruji, dan dapat digunakan. Mengintegrasikan kedua metode ini membutuhkan strategi yang tepat dengan fokus pada analisis dan desain berkualitas [5].

### C. PostgreSQL

PostgreSQL adalah sistem basis data objek-relasional open source yang kuat dengan lebih dari 30 tahun pengembangan aktif dan reputasi tinggi dalam performa dan kinerja [6]. Aspek objek-relasional PostgreSQL meningkatkan model data relasional dengan dukungan untuk array, pewarisan, dan fungsi. PostgreSQL juga mendukung ekstensibilitas tipe data dan bahasa prosedural, dengan tabel dianggap sebagai kelas, dan baris serta kolom sebagai objek dan atributnya [7].

### D. Go

Go (Go Language) adalah bahasa pemrograman yang dikembangkan oleh Google. Bahasa ini dirancang dengan fokus pada kinerja, keamanan, dan kemudahan penggunaan. Go terkenal dengan kinerjanya yang cepat, dan dapat digunakan untuk pengembangan aplikasi dan perangkat lunak yang bersifat sistem, pemrosesan data, atau pengembangan web [8].

### E. REST API

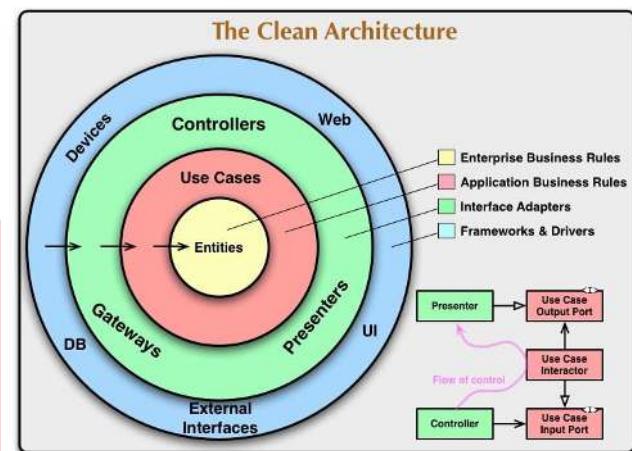
EST API (*representational state transfer application programming interface*) adalah arsitektur yang digunakan untuk merancang layanan yang dikonsumsi di berbagai platform dan lingkungan untuk mendukung interoperabilitas [9].

Metode HTTP yang umum pada REST API mencakup [10]:

1. GET digunakan untuk mengambil data dari server
2. POST digunakan untuk mengirim data baru dari client ke server
3. PUT digunakan untuk memperbarui atau membuat data baru ke server
4. DELETE digunakan untuk menghapus data

### F. Domain-Driven Design

DDD (Domain Driven Design) merupakan sebuah pendekatan untuk pengembangan software kompleks yang memisahkan antara kebutuhan bisnis dan teknis dan implementasi ke model yang terus berkembang [11].



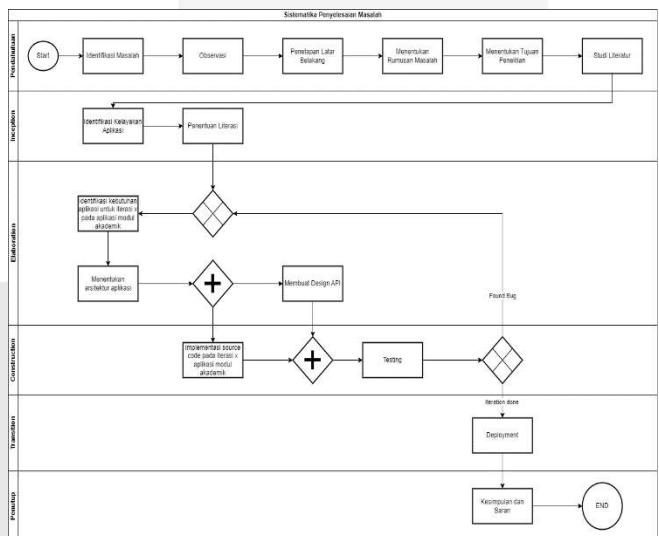
GAMBAR II-1  
Clean Architecture

### G. Load Testing

Pengujian performa *software* bertujuan untuk menguji seberapa kuat performa *software* dalam menerima beban. *Load Test* adalah pengujian yang dilakukan dengan menambahkan beban secara bertahap untuk mengamati perubahan yang terjadi pada sistem. Metode ini dipilih untuk pengujian karena mampu menunjukkan bagaimana sistem berfungsi di bawah beban kerja yang beragam dan bertahap, seperti jumlah pengguna yang mengakses website [12].

## III. METODE

Sistematika penyelesaian masalah adalah sebuah gambaran yang digunakan untuk penyelesaian masalah aplikasi penelitian ini. Sesuai dengan tujuan penelitian dengan menggunakan metode Iterative Incremental.

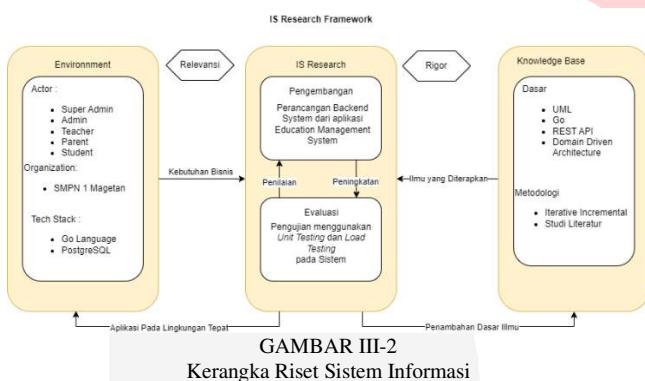


GAMBAR III-1  
Sistematika Penyelesaian Masalah

Dasar pemikiran atau kerangka berpikir merupakan landasan intelektual suatu penelitian yang disusun melalui sintesis informasi dari fakta-fakta, pengamatan, dan studi pustaka. Dengan demikian, kerangka berpikir mencakup teori, prinsip, atau konsep-konsep yang akan menjadi fondasi dalam penelitian. Dalam struktur kerangka pemikiran ini,

variabel-variabel penelitian diuraikan secara komprehensif dan terkait secara signifikan dengan permasalahan yang sedang diselidiki, memberikan dasar yang kokoh untuk mengatasi tantangan penelitian tersebut. Seiring berjalanannya waktu, pembaruan terhadap kerangka berpikir ini dapat mencerminkan perkembangan pengetahuan dan perubahan konteks yang relevan [13]. Kerangka model menggunakan Kerangka Riset Sistem Informasi. Model ini dimulai dengan melakukan analisis kebutuhan untuk menghasilkan aplikasi Education Management System. Tujuannya adalah agar solusi yang diberikan dapat sesuai dengan masalah yang ada dalam lingkungan penelitian. Struktur konseptual melibatkan komponen pendukung seperti aktor, organisasi, dan teknologi, dan hasil akhirnya adalah aplikasi EMS yang menggunakan teknologi backend berbasis Bahasa Go dan Database PostgreSQL.

Pengguna masih menghadapi beberapa masalah dalam sistem akademik saat ini. Pertama, pencetakan hasil ujian yang konvensional mengurangi efisiensi dan menyulitkan pengelolaan data. Kedua, penghitungan dengan tools terbatas meningkatkan risiko kesalahan dan menyulitkan akses. Ketiga, program literasi belum optimal karena kurangnya fasilitas dan dukungan. Keempat, hasil ulangan dan tugas sering kurang transparan dan mudah hilang, membatasi akses wali murid. Terakhir, sistem yang belum maksimal dan terfragmentasi menyulitkan navigasi dan integrasi, menghambat pemanfaatan optimal. Ini menunjukkan area



#### IV. HASIL DAN PEMBAHASAN

Pada penelitian ini terdapat tahap analisis terhadap permasalahan, kebutuhan pengguna dan sistem

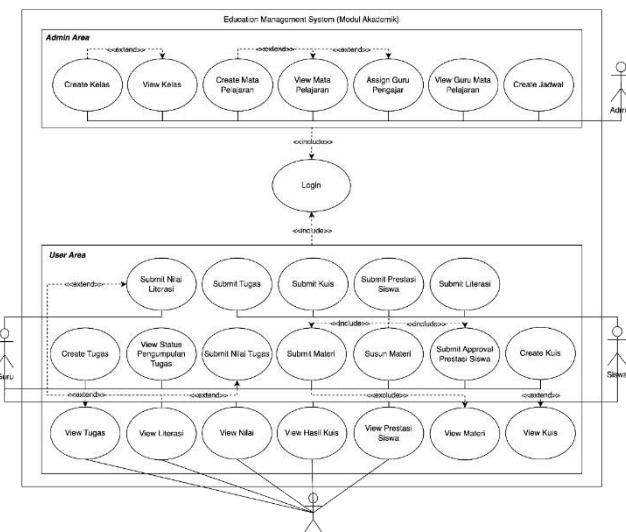
TABEL IV.1  
Analisis Permasalahan

Permasalahan Pengguna	Usability Issue
Masih konvensional dalam pencetakan hasil nilai ujian	Kurangnya efisiensi dalam proses cetak dan distribusi, serta kesulitan mengelola data ujian
Penghitungan terbatas dengan penggunaan tools tertentu dan potensi kesalahan	Kesulitan aksesibilitas dan penggunaan tools. Risiko kesalahan masih sangat tinggi karena bersifat manual
Program literasi belum berjalan maksimal dan kurang regulasi	Kurangnya fasilitas dan dukungan dalam literasi. Masih kurangnya pengelolaan program literasi
Hasil ulangan harian dan tugas kurang transparan kepada wali murid, dan terkadang mudah hilang	Kurangnya kemudahan akses dan penyimpanan hasil ulangan harian dan tugas. Akses yang terbatas kepada wali murid
Penggunaan sistem eksisting yang belum maksimal dan terfragmentasi	Kesulitan dalam navigasi dan integrasi sistem eksisting

yang perlu diperbaiki untuk meningkatkan efisiensi, transparansi, dan aksesibilitas.

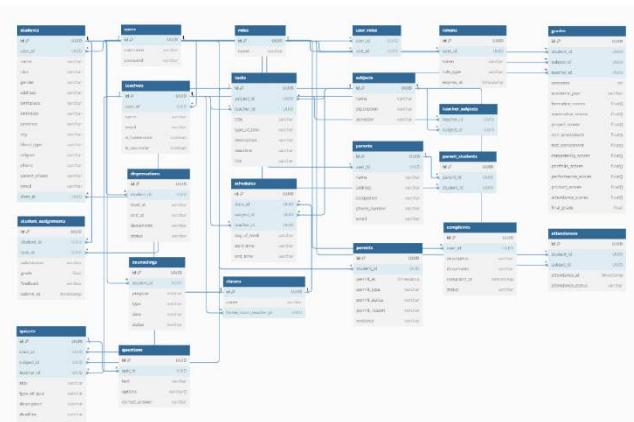
TABEL IV.2  
Analisis Kebutuhan Pengguna

Role	Feature
User (all role)	Fitur Login untuk bisa mengakses aplikasi Fitur Logout untuk keluar dari aplikasi Fitur Ubah Password untuk mengubah kata sandi
Admin	Manajemen Mata Pelajaran: Admin harus dapat membuat, melihat, dan mengelola mata pelajaran Manajemen Guru: Admin harus dapat menambahkan, melihat, dan mengelola profili guru Manajemen Kelas: Admin harus dapat membuat, melihat, dan mengelola kelas Manajemen Wali Kelas: Admin harus dapat menetapkan guru sebagai wali kelas
Guru	Mengelola Nilai: Guru harus dapat mengelola nilai siswa Membuat Tugas: Guru harus dapat membuat dan mengelola tugas untuk siswa Melihat Tugas Siswa: Guru harus dapat melihat tugas yang telah diserahkan oleh siswa Membuat Quiz: Guru harus dapat membuat dan mengelola quiz untuk siswa Melihat Jadwal: Guru harus dapat melihat jadwal mengajar mereka
Siswa	Melihat Nilai: Siswa harus dapat melihat nilai mereka Melihat Jadwal: Siswa harus dapat melihat jadwal pelajaran mereka Melihat Tugas: Siswa harus dapat melihat tugas yang diberikan kepada mereka Input Tugas: Siswa harus dapat mengunggah atau menyerahkan tugas mereka Mengerjakan Quiz: Siswa harus dapat mengerjakan quiz yang diberikan
Orang Tua	Melihat Nilai: Orang tua harus dapat melihat nilai anak mereka



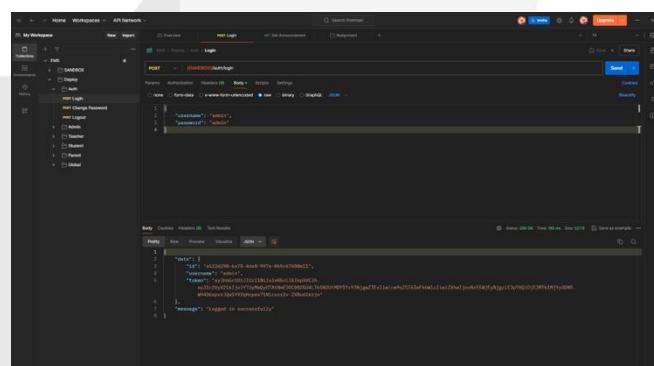
## GAMBAR IV-1 Use Case Diagram

Dan berikut adalah *design entity relationship diagram* pada sistem



## GAMBAR IV-2

Implementasi sistem pada iterasi pertama sebagai berikut



GAMBAR IV-3  
API Login

```

POST - https://api.kelas.vit.ac.id/api/tasks
{
    "name": "Tugas Kelas VIT",
    "subject": "Matematika Kelas VIT",
    "description": "Rumus perhitungan tentang keluarga matematika",
    "url": "https://drive.google.com/file/d/1QzXyfJLjDwvWVgkIwCzqHdOOGmPnZt/test"
}
  
```

The screenshot shows the Postman interface with the API endpoint `/api/tasks`. The request method is `POST`. The body contains JSON data defining a task with a name, subject, description, and a URL.

GAMBAR IV-4  
API Create Task

```

GET - https://api.kelas.vit.ac.id/api/subjects?subject=VIT
  
```

The screenshot shows the Postman interface with the API endpoint `/api/subjects`. The request method is `GET`. The URL includes a query parameter `subject` set to `VIT`.

GAMBAR IV-8  
API Get All Subject with Query Params

```

POST - https://api.kelas.vit.ac.id/api/quizzes
{
    "name": "Quiz Matematika Kelas VIT",
    "type": "Matematika",
    "description": "Quiz Matematika Kelas VIT",
    "url": "https://drive.google.com/file/d/1QzXyfJLjDwvWVgkIwCzqHdOOGmPnZt/test"
}
  
```

The screenshot shows the Postman interface with the API endpoint `/api/quizzes`. The request method is `POST`. The body contains JSON data defining a quiz with a name, type, description, and a URL.

GAMBAR IV-5  
API Create Quiz

```

POST - https://api.kelas.vit.ac.id/api/quizzes/1/questions
{
    "text": "Soal Matematika Kelas VIT",
    "type": "Text",
    "options": [
        "A. 1+1=2",
        "B. 2+2=4",
        "C. 3+3=6",
        "D. 4+4=8"
    ],
    "correct_answer": "A"
}
  
```

The screenshot shows the Postman interface with the API endpoint `/api/quizzes/{quizId}/questions`. The request method is `POST`. The body contains JSON data defining a question with text, type, options, and a correct answer.

GAMBAR IV-9  
API Add Quiz Question

```

POST - https://api.kelas.vit.ac.id/api/tasks
{
    "name": "Tugas Kelas VIT",
    "subject": "Matematika Kelas VIT",
    "description": "Rumus perhitungan tentang keluarga matematika",
    "url": "https://drive.google.com/file/d/1QzXyfJLjDwvWVgkIwCzqHdOOGmPnZt/test"
}
  
```

The screenshot shows the Postman interface with the API endpoint `/api/tasks`. The request method is `POST`. The body contains JSON data defining a task with a name, subject, description, and a URL.

GAMBAR IV-6  
API Submit Task

```

PUT - https://api.kelas.vit.ac.id/api/quizzes/1/questions/1
{
    "text": "Soal Matematika Kelas VIT",
    "type": "Text",
    "options": [
        "A. 1+1=2",
        "B. 2+2=4",
        "C. 3+3=6",
        "D. 4+4=8"
    ],
    "correct_answer": "B"
}
  
```

The screenshot shows the Postman interface with the API endpoint `/api/quizzes/{quizId}/questions/{questionId}`. The request method is `PUT`. The body contains JSON data updating a question with text, type, options, and a correct answer.

GAMBAR IV-10  
Update Quiz Question

```

POST - https://api.kelas.vit.ac.id/api/quizzes
{
    "name": "Quiz Matematika Kelas VIT",
    "type": "Matematika",
    "description": "Quiz Matematika Kelas VIT",
    "url": "https://drive.google.com/file/d/1QzXyfJLjDwvWVgkIwCzqHdOOGmPnZt/test"
}
  
```

The screenshot shows the Postman interface with the API endpoint `/api/quizzes`. The request method is `POST`. The body contains JSON data defining a quiz with a name, type, description, and a URL.

GAMBAR IV-7  
API Submit Quiz

```

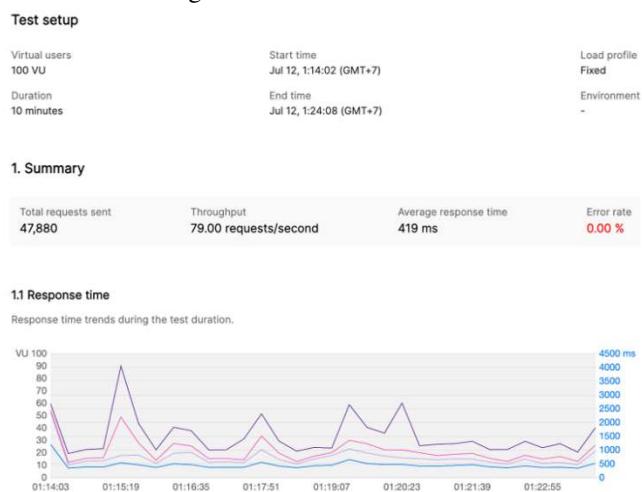
PUT - https://api.kelas.vit.ac.id/api/tasks/1/submissions
{
    "status": "Accepted"
}
  
```

The screenshot shows the Postman interface with the API endpoint `/api/tasks/{taskId}/submissions`. The request method is `PUT`. The body contains JSON data updating a task submission status to `Accepted`.

GAMBAR IV-11  
Update Task Submission

Implementasi sistem pada iterasi kedua

## Hasil Load Testing



GAMBAR IV-12  
Hasil Load Testing

## V. KESIMPULAN

Penelitian berjudul "Perancangan Back-End Aplikasi Education Management System Modul Akademik Menggunakan Metode Iterative Incremental Pada SMPN 1 Magetan" menghasilkan beberapa kesimpulan. Pertama, dirancang berbagai fitur EMS yang mencakup autentikasi pengguna, manajemen mata pelajaran, guru, kelas, dan wali kelas untuk admin; pengelolaan nilai siswa, tugas, quiz, dan jadwal mengajar untuk guru; serta akses nilai, jadwal pelajaran, dan tugas untuk siswa dan orang tua. Kedua, pengembangan aplikasi backend menggunakan metode iterative incremental dengan dua fase: fase pertama menghasilkan fitur manajemen jadwal, mata pelajaran, tugas, quiz, literasi, dan prestasi, sementara fase kedua menghasilkan fitur filtering, penambahan dan pengeditan soal quiz, tugas, dan konten mata pelajaran, dengan arsitektur domain driven design. Ketiga, pengujian API menggunakan unit test dan load test dengan simulasi 100 user menunjukkan rata-rata waktu respons 416 milidetik dan error rate 0%, menandakan API siap digunakan secara optimal.

## REFERENSI

- [1] L. O'Toole, J. Kiely, D. McGillicuddy, E. Z. O'Brien, and C. O'Keefe, "Parental Involvement, Engagement and Partnership in their Children's Education during the Primary School Years," *National Parents Council*, vol. 2, no. April, p. 84, 2019, [Online]. Available: <http://hdl.handle.net/10197/9823>
- [2] A. M. McCarthy, D. Maor, A. McConney, and C. Cavanaugh, "Digital transformation in education: Critical components for leaders of system change," *Social Sciences and Humanities Open*, vol. 8, no. 1, 2023, doi: 10.1016/j.ssho.2023.100479.
- [3] P. J. Muñoz-Merino *et al.*, "Flipping the classroom to improve learning with MOOCs technology," *Computer Applications in Engineering Education*, vol. 25, no. 1, 2017, doi: 10.1002/cae.21774.
- [4] C. Sunaengsih, M. Anggarani, M. Amalia, S. Nurfatmala, and S. D. Naelin, "Principal Leadership in the Implementation of Effective School Management," *Mimbar Sekolah Dasar*, vol. 6, no. 1, p. 79, 2019, doi: 10.17509/mimbar-sd.v6i1.15200.
- [5] I. M. Ibrahim, "Iterative and Incremental Development Analysis Study of Vocational Career Information Systems," *International Journal of Software Engineering & Applications*, vol. 11, no. 5, 2020, doi: 10.5121/ijsea.2020.11502.
- [6] The PostgreSQL Global Development Group, "What Is PostgreSQL?," *Postgresql.Org*, 2021.
- [7] J. Worsley and D. Joshua, *Practical PostgreSQL*. O'Reilly, 2002.
- [8] J. Meyerson, "The go programming language," *IEEE Softw.*, vol. 31, no. 5, 2014, doi: 10.1109/MS.2014.127.
- [9] L. Li, W. Chou, W. Zhou, and M. Luo, "Design Patterns and Extensibility of REST API for Networking Applications," *IEEE Transactions on Network and Service Management*, vol. 13, no. 1, pp. 154–167, 2016, doi: 10.1109/TNSM.2016.2516946.
- [10] S. Kumari, "REST based API," *International Journal of Trend in Scientific Research and Development*, vol. Volume-1, no. Issue-4, pp. 571–575, 2017, doi: 10.31142/ijtsrd2200.
- [11] N. Kratzke, "Domain-driven Design," in *Cloud-native Computing*, 2021. doi: 10.3139/9783446472846.014.
- [12] W. Tejaya, S. Rahman, and A. Munir, "Pengujian Website Invitees Menggunakan Metode Load Testing Dengan Apache Jmeter," *KHARISMA Tech*, vol. 18, no. 1, pp. 99–112, 2023, doi: 10.55645/kharismatech.v18i1.305.
- [13] A. Z. Syahputri, F. Della Fallenia, and R. Syafitri, "Kerangka berfikir penelitian kuantitatif," *Tarbiyah: Jurnal Ilmu Pendidikan dan Pengajaran*, vol. 2, no. 1, pp. 160–166, 2023.