

Implementasi *Service Oriented Architecture* (SOA) pada Sistem Pencatatan Keuangan untuk UMKM

(Implementation of *Service Oriented Architecture* (SOA) in the Financial Recording System for UMKM)

Royana Afwani, Muhammad Iksanul*, Moh. Ali Albar

Prodi Teknik Informatika, Universitas Mataram

Jl. Majapahit No 62, Mataram, Lombok NTB, INDONESIA

Email : royana@unram.ac.id, iksanul15@gmail.com, mohalialbar@unram.ac.id

*Penulis Korespondensi

Abstract One of the indications of a UMKM's success is financial statements. UMKM can use SAK EMKM to prepare financial reports. Many UMKM, on the other hand, have yet to incorporate SAK EMKM. UMKM can use information technology to improve its financial records. With the quantity and scope of UMKM, as well as the demand for information technology, flexible systems must be established on a variety of platforms in order to be sustainable. As a result, SOA must be used in the financial recording system for UMKM. As a result of these issues, a SOA-based Financial Recording System was developed. SOA allows the system to be more adaptable and reusable. The system built consists of a service provider and a cloud consumer. Owners of UMKM can use the Android application to keep financial records and generate reports from service providers. According to the results of the Black Box approach, every service offered at the service provider, as well as the cloud consumer's function, has been performing as predicted. And, according to the proprietor of UMKM, the system has been performing admirably.

Key words: Service Oriented Architecture (SOA), financial reports, UMKM.

I. PENDAHULUAN

Usaha Mikro Kecil Menengah (UMKM) merupakan istilah umum yang merujuk kepada usaha ekonomi produktif yang dimiliki badan usaha maupun perorangan yang memenuhi kriteria sebagaimana yang tercantum pada UU Nomor 20 Tahun 2008 tentang Usaha Mikro, Kecil, dan Menengah. Dalam menjalankan usahanya, UMKM perlu melakukan pencatatan keuangan untuk menghasilkan laporan keuangan.

Dalam membuat laporan keuangan, UMKM dapat mengacu pada Standar Akuntansi Keuangan Entitas Mikro, Kecil, dan Menengah (SAK EMKM). SAK EMKM dikeluarkan oleh Ikatan Akuntan Indonesia (IAI) yang berlaku efektif tanggal 1 Januari 2018, dimana pada SAK EMKM minimal terdapat Laporan posisi keuangan dan Laporan laba rugi selama periode Informasi juga termasuk catatan atas laporan keuangan yang berisi rincian pos-pos tertentu yang relevan. Walaupun terdapat standar

yang diberlakukan, masih cukup banyak UMKM yang belum menerapkannya. Berdasarkan pemetaan yang dilakukan di Kota Tangerang Selatan pada tahun 2018, sekitar 80,4% UMKM di kota tersebut belum melakukan penerapan SAK EMKM pada laporan keuangannya [1].

Dalam era teknologi informasi ini, hampir setiap bidang memanfaatkan teknologi informasi dalam membantu penyelesaian suatu permasalahan pada organisasi atau perusahaannya termasuk pada pencatatan keuangan UMKM. Namun, dengan banyaknya dan tersebarluasnya UMKM, diperlukan sistem yang fleksibel untuk dikembangkan secara berkelanjutan pada berbagai platform. Dengan demikian diperlukan penerapan SOA pada sistem pencatatan keuangan untuk UMKM sehingga dapat digunakan kembali oleh banyak UMKM.

Software as a Service (SaaS) merupakan model distribusi aplikasi dimana penyedia (*cloud provider*) menampung aplikasi berupa *service-service* dan membuatnya tersedia bagi pelanggan (*cloud consumer*) melalui internet. SaaS menerapkan arsitektur *Service Oriented Architecture* (SOA) yang merupakan sebuah arsitektur teknologi informasi yang melakukan modularisasi sistem informasi menjadi *services* dan bertujuan untuk penggunaan kembali logika-logika yang terdapat pada sistem [2]. Penggunaan kembali logika-logika tersebut dapat dilakukan pada berbagai *platform* sehingga menjadikan sistem bersifat reusable. Salah satu *platform* yang dapat menjadi *cloud consumer* adalah perangkat bergerak android, selain itu juga dapat digunakan untuk *web base*, desktop, dan lain-lain.

Dari permasalahan tersebut maka dibuatlah Sistem Pencatatan Keuangan yang mengimplementasikan arsitektur SOA. Dengan mengimplementasikan arsitektur SOA menjadikan sistem tersebut bersifat fleksibel dan reusable, sehingga diharapkan aplikasi dapat dikembangkan di waktu mendatang di berbagai platform dan melalui fitur pada aplikasi diharapkan pemilik UMKM dapat melakukan pencatatan keuangan serta mendapatkan laporan keuangan yang sesuai seperti standart SAK EMKM melalui perangkat bergeraknya.

II. TINJAUAN PUSTAKA DAN DASAR TEORI

Pada bab II ini dibahas tentang penelitian-penelitian terdahulu yang materinya terkait serta teori-teori yang sesuai sebagai referensi penulis saat melakukan penelitian.

A. Tinjauan Pustaka

Pembuatan Sistem pencatatan keuangan dan penerapan *Service Oriented Architecture* (SOA) pernah dilakukan sebelumnya untuk beberapa tahun terakhir. Penelitian sebelumnya akan dijadikan sebagai rujukan ketika pelaksanaan pembuatan sistem ini.

Salah satu sistem pencatatan keuangan yang dibangun memiliki fungsi untuk siklus penyusunan laporan keuangan yang disesuaikan dengan kondisi UKM yakni UKM Kampung Kue Rangkut Surabaya agar dapat membantu dan mempermudah dalam penyusunan laporan keuangan. Sistem ini berbasis web yang dibangun menggunakan bahasa pemrograman PHP dan SQL sebagai basis data [3].

Terdapat pula sistem informasi laporan keuangan (SILK) yang bertujuan untuk mencatat transaksi operasional UMKM serta menghasilkan laporan keuangan. Sistem ini berbasis desktop dan dibangun dengan menggunakan bahasa pemrograman VB.Net. Sistem ini terdiri dari lima aktor yakni pemilik, admin, bagian pembelian, bagian produksi, dan bagian penjualan [4].

Hasil rancangan berupa arsitektur sistem informasi yang menggunakan SOA dan diterapkan dengan mengimplementasikan teknologi REST serta menghasilkan empat buah layanan berupa layanan aset, layanan keuangan layanan kepegawaian, dan layanan persediaan. Penelitian yang dilakukan pada Pusat Penelitian Perkembangan Iptek Lembaga Ilmu Pengetahuan Indonesia ini menghasilkan sebuah prototipe sistem terintegrasi sebagai dashboard untuk layanan-layanan yang dibangun berbasiskan *web*. Pada Implementasi sistem terintegrasi antara aplikasi sisi *server* dan aplikasi sisi *client*, digunakan teknologi *web service* REST yang dibuat dengan menggunakan *framework* Laravel dan basis data MySQL [5].

Implementasi SOA untuk integrasi fungsi akademik dan keuangan berupa sebuah *web service* pada SMK Bina Lestari PUI Kota Tasikmalaya. Analisis penerapan konsep SOA pada kasus dilakukan dengan cara melakukan proses SOA *Lifecycle*. SOA *Lifecycle* memungkinkan penempatan kemampuan *service* melalui tiga tahap, yaitu: *requirements and analysis*, *design and development*, dan *IT operations* [2].

Dengan demikian penulis bermaksud merancang dan membangun sistem pencatatan keuangan yang sesuai SAK EMKM dengan penerapan SOA yang memiliki tujuan menghasilkan sistem pencatatan keuangan yang dapat digunakan oleh banyak UMKM karena bersifat dapat digunakan kembali (*reusable*) dengan keluaran laporan keuangan telah sesuai dengan SAK EMKM.

B. Dasar Teori

Pada bagian ini dijelaskan teori-teori dasar yang digunakan dalam penelitian ini.

B.1. Standar Akuntansi Keuangan Entitas Mikro Kecil dan Menengah (SAK EMKM)

Pada tahun 2009, Dewan Standar Akuntansi Keuangan Ikatan Akuntan Indonesia (DSAK IAI) menerbitkan Standar Akuntansi Keuangan Entitas Tanpa Akuntabilitas Publik (SAK ETAP) untuk diterapkan pada entitas kecil dan menengah. Akan tetapi, melihat kebutuhan standar akuntansi yang lebih sederhana pada entitas mikro kecil dan menengah, DSAK IAI menerbitkan Standar Akuntansi Keuangan Entitas Mikro Kecil dan Menengah (SAK EMKM).

Untuk tujuan kemudahan, laporan keuangan untuk UMKM yang diatur dalam SAK EMKM minimal terdiri dari:

1. Laporan posisi keuangan pada akhir periode Informasi
2. Laporan laba rugi selama periode Informasi
3. Catatan atas laporan keuangan yang berisi tambahan dan rincian pos-pos tertentu yang relevan [6].

B.2. Software as a Service (SaaS)

Software as a Service (SaaS) adalah salah satu model penyampaian layanan di mana perangkat lunak sebagai layanan akan mengubah cara orang membangun, menjual, membeli, dan menggunakan perangkat lunak. Dalam model ini, Perangkat Lunak disediakan sebagai layanan di mana *cloud consumer* dapat mengakses perangkat lunak dari *browser* tanpa khawatir tentang penyebaran atau instalasi dan pemeliharaan. SaaS *cloud computing* memberikan aplikasi yang diinginkan pengguna melalui internet ke ribuan pelanggan menggunakan multitenant architecture. Di sisi lain *cloud user* dapat menjalankan aplikasi atau perangkat lunak tanpa biaya di muka atau investasi dalam basis data, *server* dan lisensi perangkat lunak. Model layanan ini menggambarkan fungsi satu ke banyak sebagai aplikasi tunggal berjalan sebagai layanan di sisi server dengan banyak pengguna dapat menjalankannya dari *browser* mereka secara bersamaan karena mereka terhubung sebagai layanan [7].

B.3. Service Oriented Architecture (SOA)

Service Oriented Architecture (SOA) yaitu sebuah arsitektur yang melakukan modularisasi sebuah sistem informasi menjadi *services*. Konsep berorientasi layanan ini melakukan pendekatan dengan membagi masalah besar menjadi sekumpulan *service* kecil yang bertujuan untuk menyelesaikan permasalahan tertentu [8]. Terdapat *service interface layer* pada SOA yang dibagi menjadi tiga bagian yakni *orchestration service layer*, *business service layer*, dan *application service layer* [9].

1. Orchestration Service Layer

Orchestration service layer adalah *layer* yang memberikan abstraksi paling tinggi dari logika dan aturan bisnis suatu organisasi atau perusahaan serta bagaimana *service* harus berjalan. *Layer* ini digunakan untuk mengatur bagaimana *service-service* yang ada dapat dieksekusi dengan urutan yang sesuai.

Orchestration service layer mempunyai *process services*, yaitu *service* yang dapat mengatur *business service* dan *application service* untuk berjalan dengan aturan dan urutan yang sesuai dengan logika dan aturan bisnis perusahaan yang telah terdefinisi di dalam *process service* tersebut [9].

2. *Business Service Layer*

Business Service Layer bertanggung jawab untuk menyediakan *service* yang berkaitan dengan representasi logika dari bisnis. Tanggung jawab dari *business service* adalah mengimplementasikan logika dan model bisnis perusahaan dalam sebuah *service* [9].

3. *Application Service Layer*

Application Service Layer menjadi dasar untuk membuat fungsionalitas-fungsionalitas aplikasi secara teknis. *Service* yang ada di dalam layer ini biasa disebut sebagai *application services*, di mana tugasnya adalah menyediakan fungsi-fungsi untuk pemrosesan data pada suatu lingkungan aplikasi baik aplikasi modern maupun aplikasi *legacy* (aplikasi yang sudah dipakai sejak lama). Fungsi yang disediakan bersifat *reusable* sehingga selalu dapat digunakan kembali [9].

B.4. *Web Services*

Layanan web dapat di program dalam berbagai bahasa pemrograman yang ada. *Web services* adalah sebuah sistem terdistribusi yang memiliki komponen untuk dapat di-deploy dan diakses menggunakan protokol HTTP (*Hyper Text Transport Protocol*) maupun HTTPS (*HTTP Secure*). Pada *web services* minimal terdapat sebuah *web server* (jaringan penyedia layanan) dan sebuah klien. Klien meminta layanan atau *services* yang ditawarkan oleh *web server* bisa melalui desktop/PC dengan platform aplikasi web ataupun aplikasi internet lainnya maupun melalui aplikasi *mobile* [10].

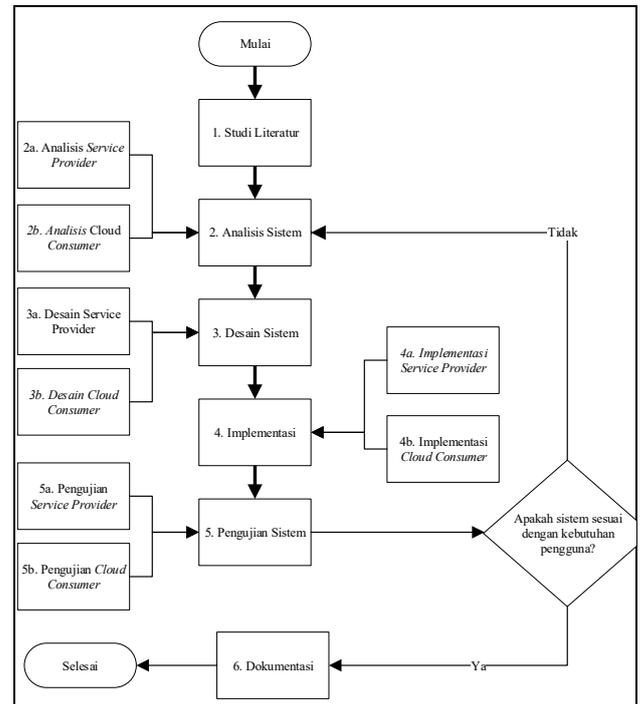
B.5. *Unified Modeling Language (UML)*

Unified Modeling Language (UML) merupakan bahasa visual untuk pemodelan dan komunikasi antar pihak atau team pengembang aplikasi mengenai sebuah sistem dengan menggunakan diagram-diagram dan teks-teks pendukung yang dapat difahami secara seragam oleh semua. UML hanya berfungsi untuk melakukan pemodelan terhadap suatu sistem. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek [11].

III. METODE PENELITIAN

Terdapat beberapa tahapan yang dilakukan dalam penelitian ini yaitu sesuai dengan tahapan yang digunakan pada metode pengembangan sistem *Waterfall* yang terdiri dari lima tahap. Pada penelitian ini hanya empat yang diterapkan yaitu *Requirement*, *Desain*, *Implementation*, dan *Verification*. Tahap-tahap pengembangan sistem tersebut diilustrasikan pada diagram alir atau *flowchart*

berserta tulisan singkat tahapannya yang dapat dilihat pada Gambar 1.



Gambar 1. Diagram alir penelitian

A. *Studi Literatur*

Tahap pertama dalam Implementasi SOA pada Sistem Pencatatan Keuangan untuk UMKM adalah studi literatur. Studi literatur dilakukan dengan mempelajari dan memahami literatur yang berkaitan dengan penelitian. Studi literatur yang telah didapat dari jurnal, buku, dan penelitian-penelitian sebelumnya.

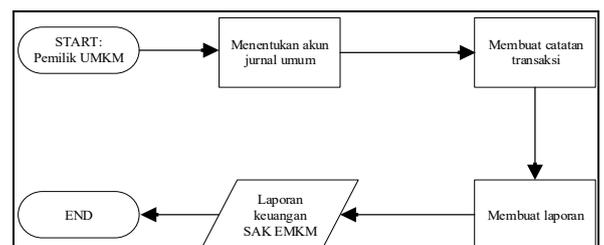
B. *Analisis Sistem*

Dalam analisis sistem pencatatan keuangan untuk UMKM ini dilakukan dua tahapan analisis. Analisis yang pertama yakni analisis terhadap *service provider*, analisis yang kedua yakni analisis terhadap *cloud consumer* yang akan digunakan oleh pemilik UMKM.

B.1. *Analisis Service Provider*

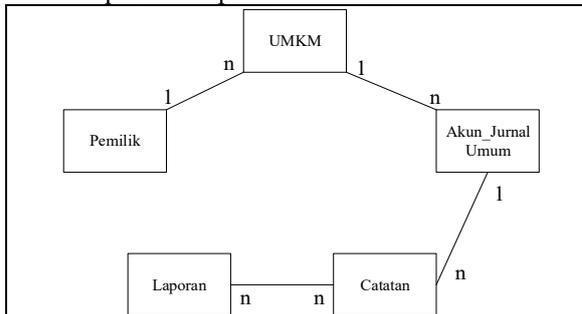
Langkah – langkah penentuan kandidat *service* pada sistem pencatatan keuangan UMKM sebagai berikut:

1. Penentuan alur kerja dan penggambaran ke dalam diagram alir seperti yang digambarkan pada Gambar 2.



Gambar 2. Alur Kerja Service Provider

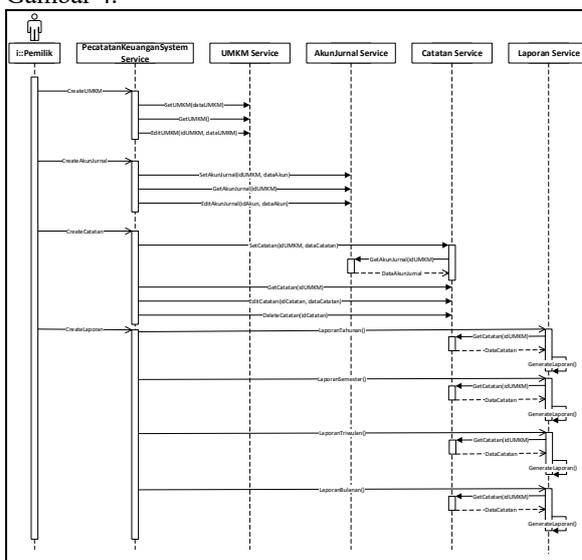
2. Penggambaran *entity model and relations* dengan dilakukannya identifikasi entitas dan relasi di dalam sistem dapat dilihat pada Gambar 3.



Gambar 3. Entity Model and Relations

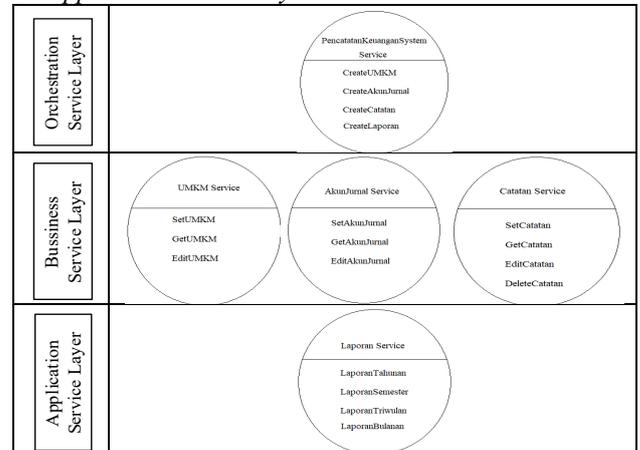
Berikut penjelasan dari objek – objek pada domain model di atas:

- Pemilik : objek pemilik merepresentasikan pemilik dari UMKM yang merupakan pengguna yang nantinya menggunakan sistem ini.
 - UMKM : objek UMKM merepresentasikan UMKM dengan informasi atau data – data UMKM yang dibutuhkan dan diolah nantinya pada sistem.
 - Akun_JurnalUmum : objek Akun_JurnalUmum merepresentasikan akun jurnal umum yang di debit dan di kredit pada saat melakukan pencatatan. Akun jurnal umum dapat berupa bagian akun dari akun – akun yang mencakup pada laporan keuangan.
 - Catatan : objek catatan merepresentasikan setiap catatan dari suatu transaksi yang dilakukan UMKM. Pada objek ini terjadi proses debit maupun kredit pada akun jurnal umum sehingga dapat menghasilkan laporan keuangan.
 - Laporan : objek laporan merepresentasikan laporan keuangan yang dihasilkan dari proses akuntansi yang terjadi.
3. Pembuatan *business use case model (Service Operations Candidate Diagram)* dapat dilihat pada Gambar 4.



Gambar 4. Service Operations Candidate Diagram

4. Penentuan klasifikasi *services* yang dilakukan berdasarkan *business use case model* yang telah dibuat sebelumnya. Penentuan kandidat *service* dilakukan dalam setiap *layer* yang ada pada SOA, yakni *orchestration service layer, business service layer, dan application service layer*.



Gambar 5. Service Operations Candidate Diagram

B.2. Analisis Cloud Consumer

Aplikasi cloud consumer diperlukan sebagai aplikasi yang berinteraksi secara langsung dengan pemilik UMKM yang tersebar luas di berbagai lokasi. Oleh karena itu diperlukan aplikasi bergerak yang dapat mengakses service – service yang telah disediakan oleh service provider sehingga memudahkan pemilik UMKM dalam melakukan pencatatan keuangannya.

C. Desain Sistem

Tahap desain merupakan tahapan perancangan atau penggambaran tentang sistem pencatatan keuangan untuk UMKM. Terdapat dua tahapan yakni tahap desain *service provider* yang fokus pada desain *web service* serta tahap desain *cloud consumer* yang fokus pada desain aplikasi yang berinteraksi secara langsung oleh pemilik UMKM yakni aplikasi berbasis *Android*.

C.1. Desain Service Provider

Berdasarkan hasil analisis kandidat service yang telah dilakukan pada tahap analisis, dapat dipetakan operasi service dalam tahap desain. Seluruh operasi service dapat dilihat pada Tabel I.

TABEL I. SERVICE DAN OPERASI PADA TAHAP ANALISIS

Nama Service	Operasi Service	Service operation pada tahap analisis
UMKM Service	SetUMKM()	Set Data UMKM
	GetUMKM()	Get Data UMKM
	EditUMKM()	Edit Data UMKM
AkunJurnal Service	SetAkunJurnal()	Set Data Akun Jurnal
	GetAkunJurnal ()	Get Data Akun Jurnal
	EditAkunJurnal ()	Edit Data Akun Jurnal
Catatan Service	SetCatatan()	Set Data Catatan
	GetCatatan()	Get Data Catatan
	EditCatatan()	Edit Data Catatan
	DeleteCatatan	Delete Data Catatan

D. Implementasi

Tahapan implementasi adalah menerapkan rancangan sistem ke dalam bentuk bahasa pemrograman. Pada pembuatan *service provider* menggunakan *Laravel* sebagai kerangka kerja dan dalam pembuatan *cloud customer* yakni aplikasi berbasis *Android* dibangun menggunakan bahasa pemrograman *Kotlin*. Sedangkan pada basis data menggunakan *MySQL* dalam implementasinya.

E. Pengujian

Tahap pengujian yaitu tahap dilakukannya uji coba terhadap sistem yang sudah selesai dibuat dari tahapan yang sebelum (implementasi sistem). Pada tahap pengujian dilakukan uji coba untuk mengetahui sistem telah berjalan sesuai dengan rancangan atau tidak. Pengujian terdiri dari pengujian *service provider* dengan menggunakan aplikasi *Postman* yang berfokus pada *web service* yang telah diimplementasikan pada tahap sebelumnya serta pengujian *cloud consumer* yang berfokus pada aplikasi berbasis *Android* sebagai *cloud consumer* yang telah dibangun.

IV. HASIL DAN PEMBAHASAN

Pada bab ini merupakan pembahasan tentang hasil dari penelitian yang dilakukan yang dibagi menjadi dua tahapan yakni implementasi sistem dan pengujian sistem.

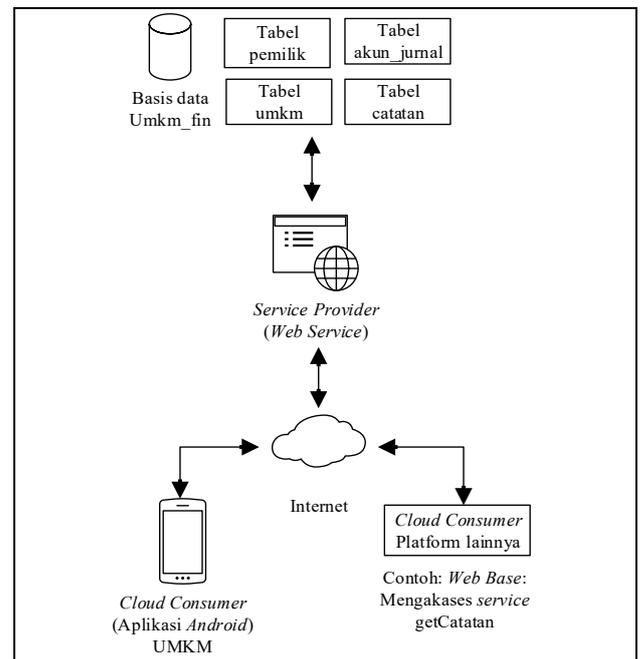
A. Implementasi Sistem

Pengimplementasian sistem dilakukan dengan mengimplementasikan basis data, *web service* sebagai *service provider*, dan aplikasi *android* sebagai *cloud consumer*. Pelaku usaha pada lokasi yang berbeda dapat menggunakan aplikasi *android* secara bersamaan dikarenakan sistem yang bersifat reusable dan fleksibel. Selain dapat diakses melalui aplikasi *android*, *service provider* juga dapat diakses melalui *cloud consumer* dengan platform lain seperti *web*, *desktop*, dan lain – lain. Tiap aplikasi *cloud consumer* dapat mengakses *service* – *service* yang disediakan oleh *service provider* melalui jaringan internet. Selain itu, *cloud consumer* juga dapat mengakses sebuah *service* pada fungsi yang berbeda. Hal tersebut dikarenakan sifat *service provider* yang dapat digunakan kembali sehingga aplikasi *cloud provider* dapat lebih fleksibel dalam pengembangannya. *Service Provider* akan menyimpan data yang diberikan oleh *cloud consumer* ke dalam basis data. *Service Provider* juga akan mengambil data dari basis data untuk diberikan kepada *cloud consumer* sesuai kebutuhannya. Arsitektur pengimplementasian sistem secara lengkap dapat dilihat pada Gambar 8.

A.1. Implementasi Service Provider

Service Provider diimplementasikan dengan menggunakan *web service* di mana setiap *service* yang diperlukan dapat diakses melalui *web service*. *Web service* dibangun menggunakan kerangka kerja *Laravel* dengan bahasa pemrograman *PHP*. *Service Provider* terdiri dari lima buah *service* yakni *service* pemilik, *service* UMKM, *service* akun jurnal, *service* catatan, dan *service* laporan.

Rincian tiap – tiap *service* dapat dilihat pada *route* dari pengimplementasian *service provider* pada Gambar 9.



Gambar 8. Arsitektur Pengimplementasian SOA pada Sistem Pencatatan Keuangan untuk UMKM

```
Route::get('pemilik/profile', 'PemilikController@profile')->middleware('auth:api');
Route::put('pemilik/update', 'PemilikController@update')->middleware('auth:api');

Route::post('auth/register', 'AuthController@register');
Route::post('auth/login', 'AuthController@login');

Route::post('umkm/create', 'UmkmController@postUmkm')->middleware('auth:api'); //Listing 4.2
Route::get('umkm', 'UmkmController@umkm')->middleware('auth:api');
Route::put('umkm/update', 'UmkmController@updateUmkm')->middleware('auth:api');

Route::post('akun/create', 'AkunController@postAkun')->middleware('auth:api'); //Listing 4.3
Route::get('akun/{id_umkm}', 'AkunController@akun')->middleware('auth:api');
Route::put('akun/update', 'AkunController@updateAkun')->middleware('auth:api');

Route::post('catatan/create', 'CatatanController@postCatatan')->middleware('auth:api'); //Listing 4.4
Route::get('catatan/{id_umkm}', 'CatatanController@catatan')->middleware('auth:api');
Route::put('catatan/update', 'CatatanController@updateCatatan')->middleware('auth:api');
Route::delete('catatan/delete/{id_catatan}', 'CatatanController@deleteCatatan')->middleware('auth:api');

Route::get('laporan/tahun/{id_umkm}', 'LaporanController@laporanPerTahun')->middleware('auth:api');
Route::get('laporan/semester/{id_umkm}', 'LaporanController@laporanPerSemester')->middleware('auth:api');
Route::get('laporan/triwulan/{id_umkm}', 'LaporanController@laporanPerTriwulan')->middleware('auth:api');
Route::get('laporan/bulan/{id_umkm}', 'LaporanController@laporanPerBulan')->middleware('auth:api');
```

Gambar 9. Route Service pada Service Provider

A.2. Implementasi Cloud Consumer

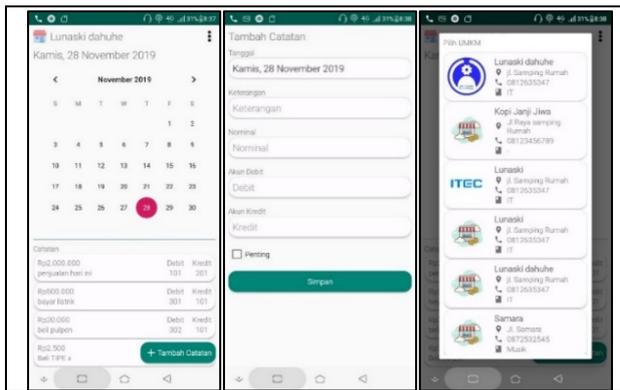
Berikut implementasi tampilan dari *cloud consumer* yakni aplikasi *android*:

1. Tampilan Halaman Catatan

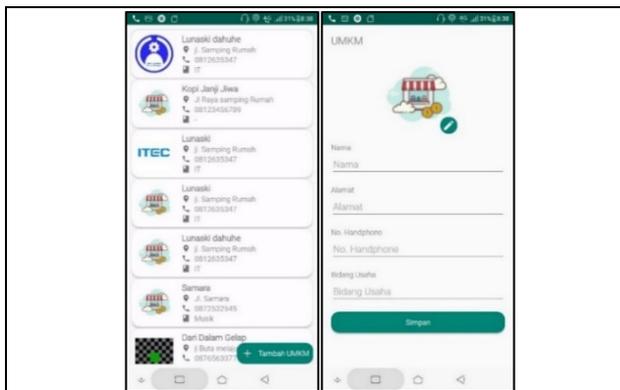
Halaman pencatatan keuangan digunakan untuk melihat informasi terkait catatan keuangan UMKM. Pada halaman ini pemilik UMKM dapat melakukan pengelolaan terhadap catatan keuangan UMKM mulai melihat, menambah, mengubah, serta menghapus catatan. Implementasi tampilan halaman catatan keuangan dapat dilihat pada Gambar 10.

2. Tampilan Halaman UMKM

Halaman UMKM digunakan untuk melihat informasi terkait UMKM. Pada halaman ini terdapat tombol yang digunakan untuk mengubah informasi tentang UMKM serta tombol ubah status UMKM untuk mengubah status UMKM. Implementasi tampilan halaman UMKM dapat dilihat pada Gambar 11.



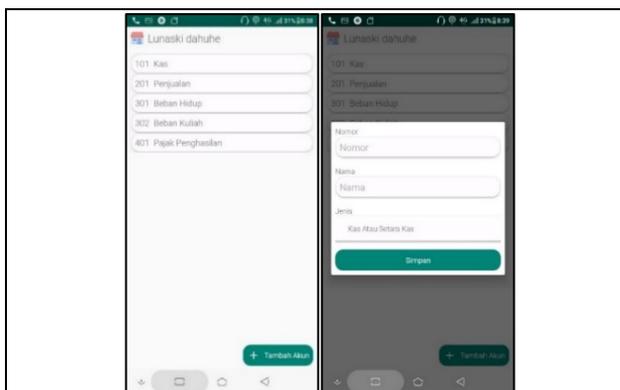
Gambar 10. Halaman UMKM



Gambar 11. Halaman UMKM

3. Tampilan Halaman Akun Jurnal

Halaman akun jurnal umum digunakan untuk melihat informasi terkait akun jurnal umum. Pada halaman ini pemilik UMKM dapat melakukan pengelolaan terhadap akun jurnal umum mulai melihat, menambah maupun mengubah akun. Pada saat menambah atau mengubah akun, maka akan muncul dialog untuk mengisi atau mengubah data akun. Implementasi tampilan halaman akun jurnal umum dapat dilihat pada Gambar 12.

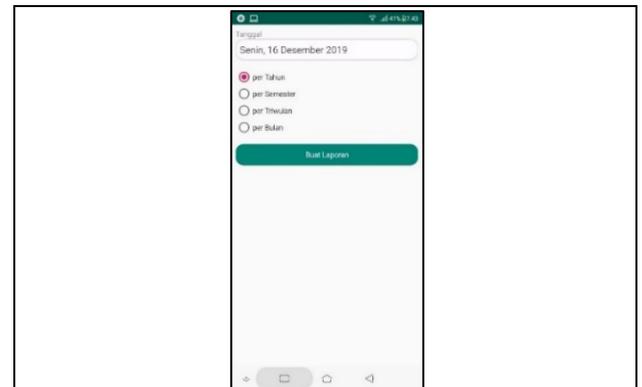


Gambar 12. Halaman Akun Jurnal

4. Tampilan Halaman Membuat Laporan

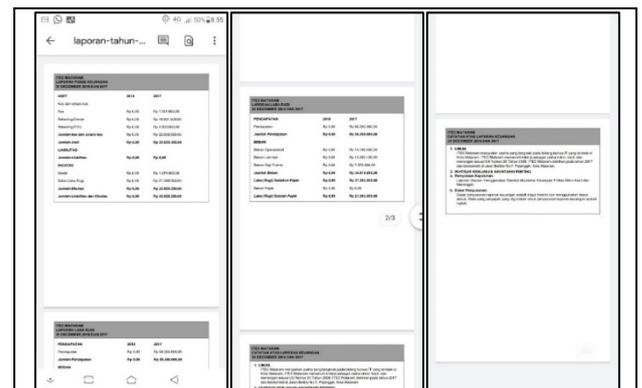
Halaman membuat laporan keuangan digunakan untuk menentukan tanggal laporan dan periode laporan. Pada halaman membuat laporan terdapat *date picker* untuk mengambil tanggal, *checkbox* untuk memilih periode dan

sebuah tombol untuk membuat laporan. Implementasi tampilan dari halaman laporan dapat dilihat pada Gambar 13.



Gambar 13. Halaman Membuat Laporan

Ketika membuat laporan keuangan, sistem akan mengunduh berkas laporan keuangan yang telah dibuat. Laporan yang dibuat terdiri dari tiga buah laporan yakni laporan posisi keuangan, laporan laba rugi, dan catatan atas laporan keuangan. Hasil dari pembuatan laporan keuangan dapat dilihat pada Gambar 14.



Gambar 14. Implementasi Laporan Keuangan

B. Pengujian Sistem

Sistem yang telah menempuh tahap implementasi, akan memasuki tahap pengujian sistem untuk mengetahui sejauh mana keberhasilan sistem dan kualitasnya. Pengujian sistem yang dilakukan antara lain pengujian *web service* yang merupakan *service provider* dan pengujian aplikasi *android* yang merupakan *cloud consumer*. Tiap – tiap pengujian dilakukan dengan menggunakan metode *Black Box*.

B.1. Pengujian Service Provider

Pengujian *Service Provider* dilakukan dengan menggunakan aplikasi *Postman* untuk melakukan sebuah *request* ke *web service*. Pengujian dilakukan dengan menguji setiap *service* yang ada satu – persatu. Berikut merupakan jbaran *service* yang akan diuji:

1. Kelola Data UMKM

TABEL III. PENGUJIAN BLACK BOX PADA SERVICE KELOLA DATA UMKM

Service	Skenario	Hasil yang diharapkan	Hasil	Kesimpulan
Membuat UMKM	Mengirimkan data UMKM yang akan dibuat	Response berupa data UMKM yang telah dibuat	✓	Valid
Mengubah UMKM	Mengirimkan data baru dari UMKM yang akan diubah	Response berupa data baru UMKM yang telah diubah	✓	Valid
Mengambil data UMKM	Mengakses service tanpa mengirim data apa pun	Response berupa data UMKM yang dimiliki	✓	Valid

Pada Tabel III dapat dilihat bahwa pengujian service kelola data UMKM dilakukan dengan tiga skenario yakni membuat UMKM, mengubah UMKM, dan mengambil UMKM. Dari tiga skenario tersebut hasil yang diperoleh yakni sesuai pada setiap skenario sehingga dapat disimpulkan bahwa skenario tersebut valid.

2. Kelola Data Akun Jurnal

TABEL IV. PENGUJIAN BLACK BOX PADA SERVICE KELOLA DATA AKUN JURNAL

Service	Skenario Pengujian	Hasil yang diharapkan	Hasil	Kesimpulan
Membuat akun	Mengirimkan data akun yang akan dibuat	Response berupa data akun yang telah dibuat	✓	Valid
Mengubah akun	Mengirimkan data baru dari akun yang akan diubah	Response berupa data baru akun yang telah diubah	✓	Valid
Mengambil data akun	Mengakses service tanpa mengirim data apa pun	Response berupa data akun yang dimiliki	✓	Valid

Pada Table IV dapat dilihat bahwa pengujian service kelola data akun jurnal dilakukan dengan tiga skenario yakni membuat akun, mengubah akun, dan mengambil data akun. Dari tiga skenario tersebut hasil yang diperoleh yakni sesuai pada setiap skenario sehingga dapat disimpulkan bahwa skenario tersebut valid.

3. Kelola Data Catatan

Pada Table V dapat dilihat bahwa pengujian service kelola data catatan dilakukan dengan empat skenario yakni membuat catatan, mengubah catatan, menghapus catatan, dan mengambil data catatan. Dari empat skenario tersebut hasil yang diperoleh yakni sesuai pada setiap skenario sehingga dapat disimpulkan bahwa skenario tersebut valid.

TABEL V. PENGUJIAN BLACK BOX PADA SERVICE KELOLA DATA CATATAN

Service	Skenario Pengujian	Hasil yang diharapkan	Hasil	Kesimpulan
Membuat catatan	Mengirimkan data catatan yang akan dibuat	Response berupa data catatan yang telah dibuat	✓	Valid
Mengubah catatan	Mengirimkan data baru dari catatan yang akan diubah	Response berupa data baru catatan yang telah diubah	✓	Valid
Menghapus catatan	Mengirimkan data catatan yang akan dihapus	Response sukses menghapus catatan	✓	Valid
Mengambil data catatan	Mengakses service tanpa mengirim data apa pun	Response berupa data catatan yang dimiliki	✓	Valid

4. Laporan

TABEL VI. PENGUJIAN BLACK BOX PADA SERVICE KELOLA DATA CATATAN

Skenario Pengujian	Hasil yang diharapkan	Hasil	Kesimpulan
Membuat laporan per tahun	Mendapatkan berkas laporan ke-uangan untuk tahun yang diminta	✓	Valid
Membuat laporan per semester	Mendapatkan berkas laporan ke-uangan untuk semester yang diminta	✓	Valid
Membuat laporan per triwulan	Mendapatkan berkas laporan ke-uangan untuk tri-wulan yang diminta	✓	Valid
Membuat laporan per bulan	Mendapatkan berkas laporan ke-uangan untuk bu-lan yang diminta	✓	Valid

Pada Table VI dapat dilihat bahwa pengujian service laporan dilakukan dengan empat buah skenario yakni membuat laporan per tahun, per semester, per triwulan, dan per bulan. Dari skenario tersebut diharapkan diperoleh berkas laporan keuangan sesuai dengan periode yang diminta. Hasil yang diperoleh dari skenario tersebut yakni sesuai sehingga dapat disimpulkan bahwa skenario tersebut valid.

B.2. Pengujian Cloud Consumer

Pengujian *Cloud Consumer* dilakukan pada aplikasi *android* yang telah dibuat sebelumnya. Pengujian dilakukan oleh penulis serta pemilik UMKM dengan menguji setiap fungsi dari aplikasi *android* yang ada satu – persatu.. Berikut merupakan jbaran fungsi – fungsi yang diuji:

1. Kelola Data UMKM

Pada Table VII dapat dilihat bahwa pengujian fungsi kelola data UMKM dilakukan dengan tiga skenario yakni membuat UMKM, mengubah UMKM, dan mengambil

UMKM. Dari tiga skenario tersebut hasil yang diperoleh yakni sesuai pada setiap skenario sehingga dapat disimpulkan bahwa skenario tersebut valid.

TABEL VII. PENGUJIAN BLACK BOX PADA FUNGSI KELOLA DATA UMKM

Service	Skenario	Hasil yang diharapkan	Hasil	Kesimpulan
Membuat UMKM	Mengisi form data UMKM yang akan dibuat	Data UMKM berhasil disimpan dan kembali ke halaman UMKM	✓	Valid
Mengubah UMKM	Mengubah data dari UMKM yang diinginkan dan menekan tombol simpan	Data UKMM berhasil diubah dan kembali ke halaman detail UMKM	✓	Valid
Mengambil data UMKM	Membuka halaman UMKM	Aplikasi menampilkan UMKM yang dimiliki pengguna	✓	Valid

2. Kelola Data Akun Jurnal

TABEL VIII. PENGUJIAN BLACK BOX PADA FUNGSI KELOLA DATA AKUN JURNAL

Service	Skenario Pengujian	Hasil yang diharapkan	Hasil	Kesimpulan
Membuat akun	Mengisi form data akun yang akan dibuat	Data akun berhasil disimpan dan kembali ke halaman akun	✓	Valid
Mengubah akun	Mengubah data dari akun yang diinginkan dan menekan tombol simpan	Data akun berhasil diubah dan kembali ke halaman akun	✓	Valid
Mengambil data akun	Membuka halaman akun dan memilih UMKM	Aplikasi menampilkan akun yang dimiliki UMKM tersebut	✓	Valid

Pada Table VIII dapat dilihat bahwa pengujian fungsi kelola data akun jurnal dilakukan dengan tiga skenario yakni membuat akun, mengubah akun, dan mengambil data akun. Dari tiga skenario tersebut hasil yang diperoleh yakni sesuai pada setiap skenario sehingga dapat disimpulkan bahwa skenario tersebut valid.

3. Kelola Data Catatan

Pada Table IX dapat dilihat bahwa pengujian fungsi kelola data catatan dilakukan dengan empat skenario yakni membuat catatan, mengubah catatan, menghapus catatan, dan mengambil data catatan. Dari empat skenario tersebut hasil yang diperoleh yakni sesuai pada setiap skenario sehingga dapat disimpulkan bahwa skenario tersebut valid.

TABEL IX. PENGUJIAN BLACK BOX PADA FUNGSI KELOLA DATA CATATAN

Service	Skenario Pengujian	Hasil yang diharapkan	Hasil	Kesimpulan
Membuat catatan	Mengisi form data catatan yang akan dibuat	Data catatan berhasil disimpan dan kembali ke halaman catatan	✓	Valid
Mengubah catatan	Mengubah data dari catatan yang diinginkan dan menekan tombol simpan	Data catatan berhasil diubah dan kembali ke halaman catatan	✓	Valid
Menghapus catatan	Menekan tombol hapus	Data berhasil dihapus dan kembali ke halaman catatan	✓	Valid
Mengambil data catatan	Membuka halaman catatan dan memilih UMKM serta tanggal	Aplikasi menampilkan catatan yang dimiliki UMKM tersebut pada tanggal yang dipilih	✓	Valid

4. Laporan

TABEL X. PENGUJIAN BLACK BOX PADA FUNGSI KELOLA DATA CATATAN

Skenario Pengujian	Hasil yang diharapkan	Hasil	Kesimpulan
Membuat laporan per tahun	Mengunduh berkas laporan keuangan untuk tahun yang diminta	✓	Valid
Membuat laporan per semester	Mengunduh berkas laporan keuangan untuk semester yang diminta	✓	Valid
Membuat laporan per triwulan	Mengunduh berkas laporan keuangan untuk triwulan yang diminta	✓	Valid
Membuat laporan per bulan	Mengunduh berkas laporan keuangan untuk bulan yang diminta	✓	Valid

Pada Table X dapat dilihat bahwa pengujian fungsi laporan dilakukan dengan empat buah skenario yakni membuat laporan per tahun, per semester, per triwulan, dan per bulan. Dari skenario tersebut diharapkan diperoleh berkas laporan keuangan sesuai dengan periode yang diminta. Hasil yang diperoleh dari skenario tersebut yakni sesuai sehingga dapat disimpulkan bahwa skenario tersebut valid.

B.3. Penilaian Pemilik UMKM

Setelah melakukan pengujian seperti yang dijelaskan diatas pada poin B.1 dan B.2, pemilik UMKM diberikan pertanyaan sebagai penilaian terhadap sistem yang telah dibangun. Terdapat empat buah pertanyaan dan tiap pertanyaan dapat dijawab dengan sangat buruk, buruk, cukup, baik, dan sangat baik. Dari hasil penilaian pemilik UMKM yang melakukan pengujian sistem, sistem yang dibangun sudah baik dalam melakukan fungsi – fungsinya. Hasil dari penilaian pemilik UMKM dapat dilihat pada Tabel XI.

TABEL XI. HASIL PENILAIAN PEMILIK UMKM

No.	Pertanyaan	UMKM 1	UMKM 2
1.	Bagaimana tampilan aplikasi?	Baik	Baik
2.	Bagaimana pengalaman pengguna pada aplikasi?	Cukup	Baik
3.	Bagaimana kinerja aplikasi dalam melakukan pencatatan keuangan?	Sangat Baik	Baik
4.	Bagaimana hasil laporan keuangan dari keluaran aplikasi?	Baik	Baik

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan penelitian dan pembangunan sistem yang telah dilakukan, dapat ditarik beberapa kesimpulan dalam antara lain:

1. Sistem pencatatan keuangan untuk UMKM telah mengimplementasikan *Service Oriented Architecture* (SOA) di mana telah dibangun sebuah *web service* yang merupakan *service provider* yang dapat diakses oleh *cloud consumer* dari berbagai platform karena *service provider* bersifat *reusable*.
2. Pengembangan sistem dapat dilakukan dengan lebih dinamis karena *service provider* yang dibangun bersifat fleksibel.
3. *Service provider* dari sistem pencatatan keuangan untuk UMKM menghasilkan keluaran berupa catatan keuangan yang terdiri dari laporan posisi keuangan, laporan laba rugi, dan catatan atas laporan keuangan sehingga dapat memenuhi SAK EMKM.
4. Pemilik UMKM dapat melakukan pencatatan keuangan hingga membuat laporan keuangan yang telah disediakan oleh *service provider* melalui aplikasi android sebagai *cloud consumer* dari sistem pencatatan keuangan untuk UMKM.
5. Berdasarkan hasil pengujian dengan metode *Black Box*, menunjukkan bahwa *service – service* pada *service provider (web service)* serta fungsi – fungsi pada *cloud consumer (aplikasi android)* dari sistem pencatatan keuangan untuk UMKM telah berjalan sesuai yang diharapkan.
6. Berdasarkan umpan balik yang diberikan oleh pemilik UMKM setelah melakukan pengujian aplikasi *android*, aplikasi yang dibangun telah berjalan dengan baik.

B. Saran

Berikut merupakan beberapa saran untuk dapat membuat sistem ini menjadi lebih baik kedepannya, yaitu:

1. Analisis sistem diperdalam sehingga menghasilkan sistem yang lebih optimal dan mudah digunakan.
2. Oleh karena sistem berkaitan dengan masalah keuangan, diharapkan pengembangan selanjutnya lebih memperhatikan keamanan sistem.

3. Mengembangkan sistem untuk membantu pemilik UMKM untuk mengelola usahanya.
4. Mengoptimalkan UI/UX sistem sehingga sistem lebih ramah terhadap pengguna.
5. Melakukan pengolahan terhadap data dari UMKM sehingga data yang diperoleh dapat digunakan untuk menghasilkan solusi pada suatu permasalahan dengan memanfaatkan teknologi *mechine learning* dan *big data*.

DAFTAR PUSTAKA

- [1] Y. M. Putra, "Pemetaan Penerapan Standar Akuntansi Keuangan Emkm Pada Umkm Di Kota Tangerang Selatan," *Jurnal Profita*, vol. 11, no. 2, p. 201, 2018.
- [2] M. Kharisman, A. Dipalokareswara, and H. Mubarak, "Implementasi Service Oriented Architecture untuk Pengintegrasian Fungsi Akademik dan Keuangan," *Jurnal Penelitian Sitrotika*, vol. 9, no. 2, pp. 221–229, 2014.
- [3] N. A. Latifah Rifani, "Aplikasi sistem informasi akuntansi pada usaha kecil menengah kampung kue rungkut surabaya," *Seminar Nasional Sistem Informasi Indonesia*, pp. 427–436, 2016.
- [4] R. K. Hapsari, A. W. Azinar, and Sugiyanto, "Rancang Bangun Sistem Informasi Laporan Keuangan Untuk Usaha Mikro, Kecil Dan Menengah," *Jurnal Ilmiah NERO*, vol. 3, no. 2, pp. 85–92, 2017.
- [5] W. Warkim and D. I. Sensuse, "Model Integrasi Sistem dengan Pendekatan Metode Service Oriented Architecture dan Model View Controller pada Pusat Penelitian Perkembangan Iptek Lembaga Ilmu Pengetahuan Indonesia," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 3, no. 1, pp. 1–12, 2017.
- [6] Tatik, "Implementasi SAK EMKM (Standar Akuntansi Keuangan Entitas Mikro Kecil dan Menengah) pada Laporan Keuangan," *Jurnal Relasi*, vol. XIV, no. 02, pp. 1–14, 2018.
- [7] K. V. K. M. Kumar, "Software as a Service for Efficient Cloud Computing," *International Journal of Research in Engineering and Technology.*, vol. 7, p. 10, 2014.
- [8] T. Erl, *Service-Oriented Architecture: Concepts, Technology, and Design*. 2005.
- [9] J. B. Wijanarko and R. Afwani, "Integrasi Sistem Manajemen Tata Usaha Dengan Metode Service Oriented Architecture (SOA) Pada SMP Negeri 10 Mataram," *J-COSINE*, vol. 3, no. 2, pp. 162–171, 2019.
- [10] R. Roslidar and R. Dawood, "Rancang Bangun Layanan Web (Web Service) Untuk Aplikasi Rekam Medis Praktik Pribadi Dokter," *Jurnal Online Teknik Elektro*, vol. 2, no. 1, 2017.
- [11] R. A. Sukamto and M. Shalahudin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika, 2016.