

**MEMBANGUN *APPLICATION PROGRAMMING INTERFACE*
SISTEM INFORMASI VILLA TERMURAH CIATER
(Studi Kasus Simulasi Kemitraan)**

Jajang Nurjaman¹, Chalifah Chazar²

¹Teknik Informatika, STMIK IM. jajangnur2021@gmail.com

²Teknik Informatika, STMIK IM. chalifa.chazar@gmail.com

ABSTRACT

Various design styles or architectural styles of information exchange are created to make the exchange of information very easy, but how is the exchange of information easy to do? The information system villatermurahciater.com is one source of information that provides information on villa rentals. Service Oriented Architecture (SOA) is a style of software architecture that relies on communication between services, by defining how two or more parts of a service can interact with each other, where one part can run another part. Application Programming Interface (API) is a virtual interface technology that provides data for potential villa partners in developing their information system. Using the REST API design style for web service creation, it is hoped that this partnership simulation can be implemented well with other information systems.

Keywords: Service Oriented Architecture (SOA), Application Programming Interface (API), REST API, Web service.

ABSTRAK

Berbagai gaya perancangan atau architectural style pertukaran informasi diciptakan untuk membuat pertukaran informasi menjadi sangat mudah, tapi bagaimana cara pertukaran informasi itu mudah dilakukan? Sistem informasi villatermurahciater.com adalah salah satu sumber informasi yang menyediakan informasi-informasi penyewaan villa. Dengan arsitektur *Service Oriented Architecture* (SOA) adalah suatu gaya arsitektur *software* yang bertumpu pada komunikasi antar *service*, dengan mendefinisikan bagaimana dua bagian atau lebih *service* dapat saling berinteraksi, dimana suatu bagian dapat menjalankan bagian lainnya. *Application Programming Interface* (API) merupakan suatu teknologi antarmuka virtual yang menyediakan data-data untuk calon mitra vila dalam pengembangan sistem informasinya. Menggunakan gaya perancangan *REST API* untuk pembuatan *web servicenya* diharapkan simulasi kemitraan ini dapat diimplementasikan dengan sistem informasi lain secara baik.

Kata kunci: Service Oriented Architecture (SOA), Application Programming Interface (API), REST API, Web service.

1. PENDAHULUAN

Kemudahan dalam mendapatkan dan bertukar informasi adalah salah satu bagian dari perkembangan dan kemajuan teknologi yang dibangun sampai dengan saat ini, *Architectural style* untuk pertukaran informasi dan data dibangun untuk mempermudah pertukaran sebuah data ke dalam aplikasi dan sistem informasi lainnya untuk kepentingan perkembangan sebuah teknologi, bisnis, instansi, organisasi dan lain sebagainya. Akan tetapi bagaimana cara pertukaran informasi itu mudah dilakukan? IPCI atau Ikatan Pramuwisata Ciater adalah kelompok masyarakat yang mengelola dalam jasa pemasaran vila-vila yang disewakan kepada pengunjung di tempat wisata sari ater, IPCI sekaligus pemilik sistem informasi villatermurahciater.com akan mengembangkan usaha dan sistem informasinya dengan bermitra bersama perusahaan jasa lainnya.

API (*Application Programming Interface*) terdiri dari dokumentasi yang terdiri dari antarmuka, fungsi, kelas, struktur, dan sebagainya untuk membangun perangkat lunak. Dengan API ini, kemudian membuat programmer untuk "membongkar" perangkat lunak untuk kemudian dikembangkan atau diintegrasikan dengan perangkat lunak lain (Sontana, Rahmatulloh dan Rachman, 2019).

“*SOA is basically a group of services, where the communication involves either two or more services regarding some activity or it can involve simple data passing*” (Alanazi dkk., 2019). *SOA (Service Oriented Architecture)* merupakan arsitektur pendekatan dalam sistem komunikasi antara 2 aplikasi atau lebih, dimana data yang telah disediakan oleh layanan *web service* dapat digunakan oleh *rest client* atau suatu sistem dan aplikasi lainnya. Menggunakan gaya perancangan *RESTful* merupakan gaya pengembangan *web service* yang memanfaatkan *HTTP (Hyper Text Transfer Protocol)*, dalam *service request data* atau layanan permintaan datanya menggunakan kata kerja atau *method* yaitu *GET* untuk *request client server* akan secara otomatis direspon oleh *web server* dan *method* untuk *web service* yaitu *GET, POST, PUT* dan *DELETE* dalam pengelolaan data-datanya.

Oleh karena itu dari paparan diatas untuk studi kasus simulasi kemitraan dalam hal ini mengintegrasikan sistem informasi villatermurahciater.com diperlukan pembuatan *API*. Dengan membuat *web service* atau *web API* menggunakan gaya perancangan *Representational State Transfer (REST)* dan *Representational State Transfer Application Programming Interface (RESTful API)* diharapkan dapat menyediakan data-data diatas untuk calon mitra (*rest client*) dan pengelolaan data untuk villatermurahciater.com (*rest server*).

2. METODOLOGI

2.1 Metode Pengumpulan Data

Kebutuhan dalam penyusunan skripsi ini mempunyai cara dalam mengumpulkan data-data untuk kebutuhan analisa, kebutuhan dalam penyediaan data vila, tentunya hal tersebut akan disediakan untuk keperluan *service data* yang menangani *service request data* dari *rest client*. Pada tahap pengumpulan data dilakukan dengan menggunakan metode observasi, studi pustaka dan studi literatur.

2.1.1 Observasi

Hasil dari metode ini adalah kumpulan data-data yang akan disediakan untuk calon mitra bisnis IPCI yaitu tipe atau nama vila, harga masing-masing vila per-hari, jumlah kapasitas orang yang bisa ditampung untuk masing-masing vila, fasilitas untuk masing-masing vila, gambar masing-masing vila dan kontak *person* yang akan melayani pesanan vila, data-data tersebut akan menjadi data utama. Data-data yang sudah diperoleh akan dijadikan layanan data yang disediakan oleh *web service villatermurahciater.com* yang nantinya layanan data tersebut bisa diakses atau “dikonsumsi” oleh *rest client* atau calon mitra.

2.1.1.1 Analisa Data Vila

Data vila adalah data utama dalam *service provider data* yang nantinya *service costumer* akan mengirim *service request* dan kemudian *service request* tersebut harus direspon oleh *service provider*.

2.1.1.2 Analisa Pemesanan Vila

Pada pola pemesanan vila calon penyewa vila menghubungi langsung kepada nomor kontak personal pelayanan pemesanan vila untuk langsung melakukan pemesanan vila kemudian calon penyewa vila dan pic jasa pengantar vila membuat janji untuk bertemu dan melakukan pembayaran sewa vila ditempat.

2.1.1.3 Analisa Pemasaran Vila

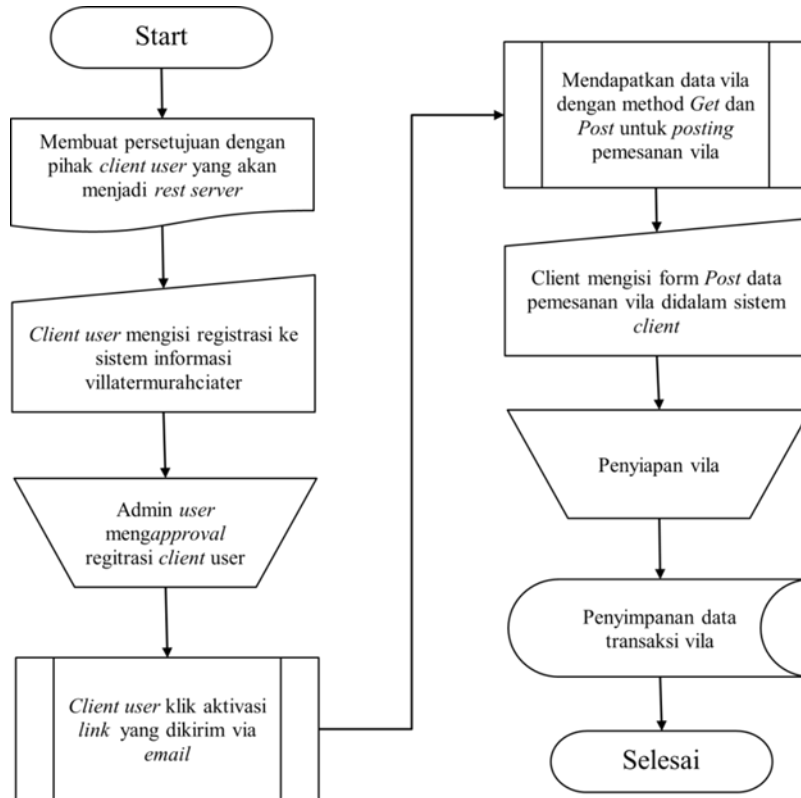
Untuk pemasaran vila, selain pemanfaatan web site untuk pemasaran dan penyediaan informasi vila-vilanya, pemilik informasi *villatermurahciater.com* juga memanfaatkan sosial media seperti facebook, instagram dan menawarkan secara langsung kepada calon penyewa vila.

2.1.1.4 Analisa Pengembangan Sistem Informasi

Dari metode observasi yang telah dilakukan, maka akan ada poin-poin pengembangan sistem informasi *villatermurahciater.com* dalam penyediaan layanan data vila dan beberapa aturan dasar untuk menjadi rest server *villatermurahciater.com*, dimana data vila tersebut dapat direusable atau digunakan kembali kedalam sistem calon mitra untuk keperluan informasi dan tujuan bisnis di dalam sistem informasi calon mitra secara fleksibel, sebagai gambaran simulasi layanan data vila itu berjalan, penulis mencoba menggambarkan proses awal membuat persetujuan, proses registrasi menjadi rest

server dan penggunaan data vila yang telah didapatkan, akan tetapi gambaran aturan bermitra ini bukan menjadi sebuah aturan baku dalam pembuatan aturan bisnis dari ke 2 belah pihak.

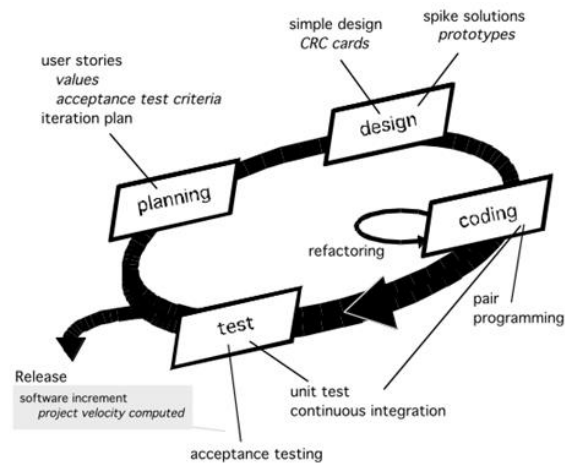
Poin-poin pengembangan sistem informasi tersebut akan disajikan pada Gambar 1.1.



Gambar: 1. 1 *Flowchart* pengembangan sistem informasi villatermurahciater.com

2.2 Metode Pengembangan Sistem

Metode pengembangan sistem villatermurahciater.com adalah metode *Extreme Programming* yang akan berkonsentrasi pada 4 poin penting yaitu planing (perencanaan), design (perancangan), coding (pengkodean) dan testing (pengujian) kemudian tahapan release.



Gambar 1. Kerangka Kerja *Extreme Programming* (Sontana dkk, 2019:3)

1. Planing (Perencanaan)

Penjadwalan pengembangan sistem informasi, dimana dibuatkan time table mulai dan berakhirnya pengembangan sistem informasi atau suatu proyek.

Analisis data merupakan kebutuhan yang berdasarkan pada fungsionalitas seperti user name dan password untuk login yang nantinya memiliki hak akses yang dapat mengelola fitur-fitur yang ada di dalam web service yaitu GET, POST, PUT dan DELETE.

Spesifikasi minimum perangkat keras yang dibutuhkan untuk menjalankan aplikasi berbasis web, yaitu PC yang terhubung dengan jaringan internet. Selibuhnya aplikasi ini tidak membutuhkan antarmuka perangkat keras yang spesifik.

2. Design (Perancangan)

Pada tahap ini memberikan identifikasi aktor, skenario, flowchart diagram, use case diagram, class diagram dan CRC (class-responsibility-collaborator).

3. Coding (Pengkodean)

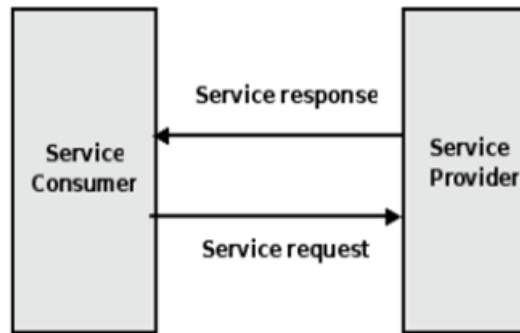
Pada tahap ini pengerjaan perancangan awal dilakukan, dengan tidak merubah code yang sudah ada melainkan mengembangkan serangkaian pengujian unit yang telah dibuat yang nantinya akan dimasukkan ke dalam tahap rilis.

4. Test (Pengujian)

Pada tahap ini pengujian unit dari hasil pengkodean apakah sudah sesuai dengan output perancangan yang telah dibuat dan dari fungsionalitas apakah sudah sesuai dengan fungsinya. Hal ini dilakukan untuk meminimalisir adanya kesalahan (error).

API (*Application Programming Interface*) terdiri dari dokumentasi yang terdiri dari antarmuka, fungsi, kelas, struktur, dan sebagainya untuk membangun perangkat lunak. Dengan API ini, kemudian membuat programmer untuk "membongkar" perangkat lunak untuk kemudian dikembangkan atau diintegrasikan dengan perangkat lunak lain (Sontana dkk, 2019).

“SOA is basically a group of services, where the communication involves either two or more services regarding some activity or it can involve simple data passing” (Alanazi dkk., 2019). SOA (Service Oriented Architecture) merupakan arsitektur pendekatan dalam sistem komunikasi antara 2 aplikasi atau lebih, dimana data yang telah disediakan oleh layanan *web service* dapat digunakan oleh *rest client* atau suatu sistem dan aplikasi lainnya.



Gambar 2. Dasar komunikasi dalam *Service Oriented Architecture* (Alanazi dkk., 2019)

Menggunakan gaya perancangan *RESTful* merupakan gaya pengembangan *web service* yang memanfaatkan HTTP (*Hyper Text Transfer Protocol*), dalam *service request* data atau layanan permintaan datanya menggunakan kata kerja atau method yaitu *GET* untuk request client server akan secara otomatis direspon oleh *web server* dan method untuk *web service* yaitu *GET*, *POST*, *PUT* dan *DELETE* dalam pengelolaan data-datanya.

3. HASIL DAN PEMBAHASAN

3.1 Penjadwalan

Penjadwalan pengembangan sistem informasi *villatermurahciater.com* disajikan pada tabel 3.1 dimulai dari minggu pertama pada bulan September sampai dengan bulan Desember.

Tabel: 3. 1 Penjadwalan pengembangan sistem informasi *villatermurahciater.com*

Bulan	Sepetember				Oktober				November				Desember			
Minggu Ke	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
Perencanaan																
Perancangan																
Pengkodean																
Pengujian																

3.1.1 Perencanaan

Analisis kebutuhan fungsionalitas data untuk pengembangan sistem informasi villatermurahciater.com. disajikan pada tabel 3.2

Tabel: 3. 2 Analisis kebutuhan data

No	Nama Data	Atribut
1	Data Pengelola <i>web API</i>	<i>email</i> dan <i>password</i>
2	Data vila	Tipe atau nama vila, harga sewa vila, fasilitas dan gambar vila.

Tabel 3.2 merupakan analisa kebutuhan data yang berdasarkan observasi pada tahap pengumpulan data yang nantinya akan disajikan oleh sistem informasi villatermurahciater.com yang menjadi portal informasi dalam pengembangan sistem informasinya dan data tersebut bisa *reuseable* atau digunakan kembali oleh aplikasi lain atau calon mitra villatermurahciater.com.

3.1.2 Perancangan

3.1.2.1 Perancangan *Use Case Diagram*

Gambaran grafis dari beberapa atau semua *actor*, *use case*, dan interaksi diantaranya yang memperkenalkan suatu sistem. *Use case* diagram tidak menjelaskan secara detail tentang penggunaan *use case*, tetapi hanya memberi gambaran singkat hubungan antara *use case*, *actor*, dan sistem, Berikut adalah tabel dan beberapa gambar *use case* diagram sistem.

Tabel: 3. 3 Identifikasi aktor

No	Aktor	Deskripsi
1	<i>Client User</i>	Orang yang mendaftar sebagai <i>REST client</i> . Sebelum <i>client user</i> tersebut meminta dan mendapatkan data informasi villatermurahciater.com. dengan menggunakan <i>metode HTTP</i> yaitu <i>Get</i> .
2	<i>Mitra User</i>	Orang yang dapat melihat informasi tentang vila, dapat mendaftar sebagai mitra villatermurahciater.com, dapat <i>upload</i> sejumlah data-data vila dan merubah atau <i>edit</i> data-data vila sesuai dengan data vila yang telah <i>upload</i>

3	<i>Admin User</i>	Orang yang dapat melihat informasi vila, informasi <i>client user</i> , mitra <i>user</i> dan dapat melihat data vila sesuai yang diupload oleh mitra <i>user</i> , serta dapat mengelola data vila dengan menggunakan metode <i>HTTP</i> yaitu <i>Get, Put, Post</i> dan <i>Delete</i> .
---	-------------------	---

3.1.2.2 Identifikasi Use Case

Tabel: 3. 4 Identifikasi use case mitra user

No	Nama Use Case	Deskripsi
1	Melihat halaman utama <i>web site</i>	<i>Use Case</i> menggambarkan kegiatan mitra <i>user</i> dan <i>client user</i> dapat melihat halaman utama <i>villatermurahciater.com</i>
2	<i>Sign up</i> Sebagai mitra	<i>Use Case</i> menggambarkan kegiatan memasukan <i>email</i> dan <i>password</i> oleh mitra <i>user</i> untuk mendaftar sebagai <i>member</i> dari sistem informasi <i>villatermurahciater.com</i> , kemudian mendapatkan link aktivasi dikirim ke <i>email user</i>
3	<i>Login</i>	<i>Use Case</i> menggambarkan kegiatan memasukan <i>email</i> dan <i>password</i> oleh mitra <i>user</i> untuk masuk kedalam sistem informasi <i>villatermurah.com</i>
4	<i>Upload</i> data vila	<i>Use Case</i> menggambarkan kegiatan <i>user</i> mengunggah data vila ke <i>villatermurahciater.com</i>
5	<i>Edit</i> data vila	<i>Use Case</i> menggambarkan kegiatan <i>user</i> mengubah data vila yang sudah diunggah sebelumnya
6	<i>View</i> Data vila	<i>Use Case</i> menggambarkan kegiatan <i>user</i> melihat data vila dengan memilih sub menu <i>edit</i>
7	<i>Logout</i>	<i>Use Case</i> menggambarkan kegiatan <i>user</i> keluar dari sistem informasi <i>villatermurahciater.com</i>

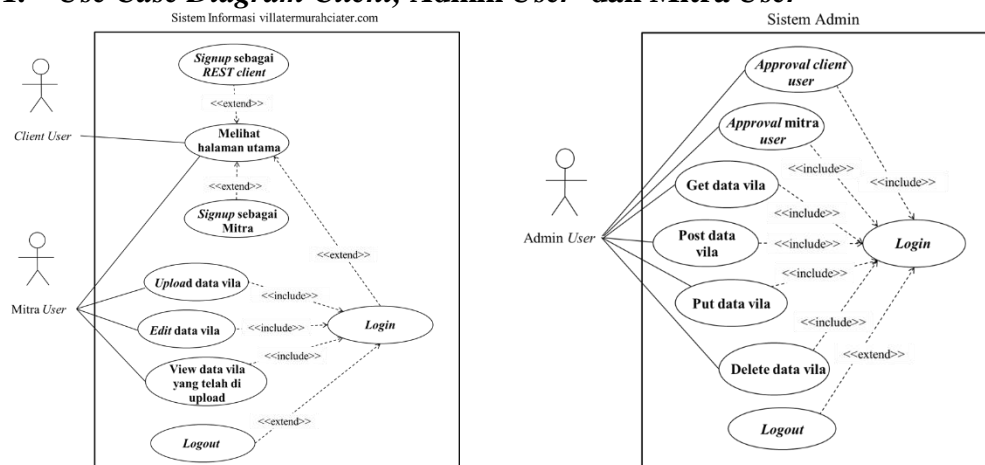
Tabel: 3. 5 Identifikasi use case client user

No	Nama Use Case	Deskripsi
1	<i>Sign up</i> sebagai <i>Rest client</i>	<i>Use Case</i> menggambarkan kegiatan <i>user</i> mendaftar sebagai <i>Rest client</i> .

Tabel: 3. 6 Identifikasi *use case admin user*

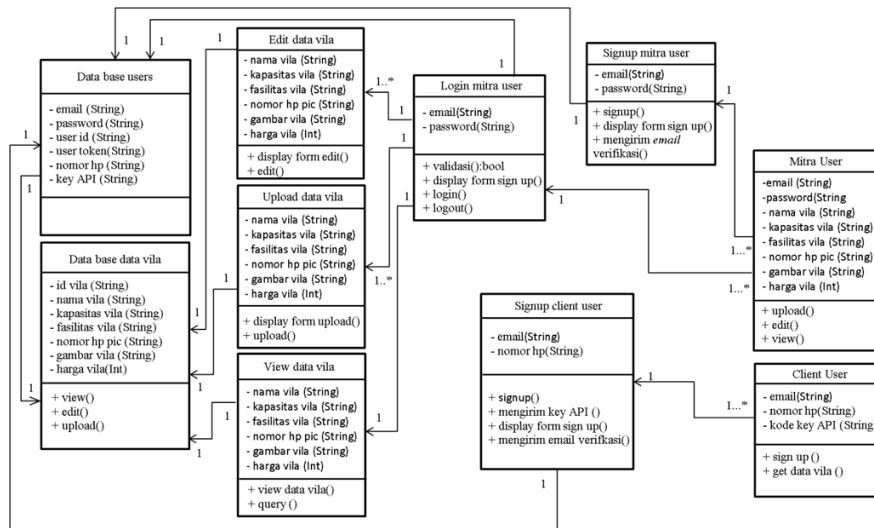
No	Nama Use Case	Deskripsi
1	<i>Login</i>	<i>Use Case</i> menggambarkan kegiatan memasukan <i>email</i> dan <i>password</i> oleh Admin untuk masuk kedalam sistem informasi admin villatermurahciater.com
2	<i>Approval</i> registrasi mitra <i>user</i>	<i>Use Case</i> menggambarkan kegiatan admin memproses <i>approval</i> registrasi mitra <i>user</i> dengan cara masuk ke dalam sistem informasi admin villatermurahciater.com.
3	<i>Approval</i> registrasi <i>client user</i>	<i>Use Case</i> menggambarkan kegiatan admin memproses <i>approval client user</i> dengan cara masuk ke dalam sistem informasi admin villatermurahciater.com.
4	<i>Get data vila</i>	<i>Use Case</i> menggambarkan kegiatan admin meminta data vila ke <i>web server</i> villatermurahciater.com
5	<i>Put data vila</i>	<i>Use Case</i> menggambarkan kegiatan admin mengubah data vila ke <i>web server</i> villatermurahciater.com
6	<i>Post data vila</i>	<i>Use Case</i> menggambarkan kegiatan admin mengunggah data vila ke <i>web server</i>
7	<i>Delete data vila</i>	<i>Use Case</i> menggambarkan kegiatan admin menghapus data vila ke <i>webs server</i>
8	<i>Logout</i>	<i>Use Case</i> menggambarkan kegiatan admin keluar dari sistem

1. Use Case Diagram Client, Admin User dan Mitra User



Gambar 3. 1 Use case diagram client, admin user dan mitra user

2. Class Diagram



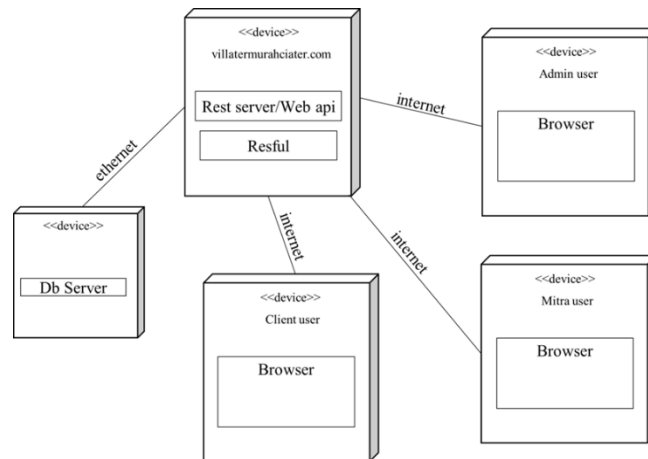
Gambar 3. 2 Class diagram sistem informasi villatermurahciater.com

3. Class Responsibility Collaboration (CRC)

Tabel: 3. 7 Class Responsibility Card (CRC)

Menu Utama	
Responsibilitie	Collabolor
Melihat data vila	Database vila
upload data vila	Class upload, Database vila
edit data vila	Class edit, Database vila
sign up	Class sign up, Database vila
login	Class login, Database vila
logout	Class logout

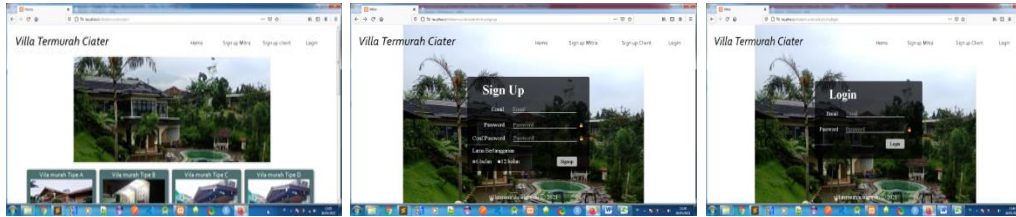
4. Analisa Deployment Diagram



Gambar 3. 3 Deployment diagram sistem informasi villatermurahciater.com

3.2 Implementasi

3.2.1 Antar muka *website villatermurahciater.com*



4. PENUTUP

4.1 Kesimpulan

Dari hasil pembahasan dan penelitian yang telah dilakukan, maka didapatkan beberapa kesimpulan sebagai berikut:

Adanya gaya arsitektur SOA (Service Oriented Architecture) serta gaya perancangan REST (Representational State Transfer) dan RESTful sangatlah membantu untuk menentukan konsep penyediaan layanan informasi serta data villatermurahciater.com bagi calon mitra yang akan berkerja sama dalam usaha dibidang layanan jasa. Pertukaran informasi dan data menjadi sangat mudah, efisien, fleksibel, bisa digunakan oleh cross platform. Sehingga dapat dikembangkan kedalam sistem informasi lain yang lebih kompleks dan besar.

4.2 Saran

Melihat dari kesimpulan diatas penulis ingin memberikan saran dalam penelitian ini yang bertujuan untuk pengembangan penelitian selanjutnya dalam pengamanan pertukaran data dimana key yang telah dibuat masih bisa dipakai oleh banyak client user tanpa memberikan autentikasi yang khusus untuk client user.

DAFTAR PUSTAKA

- Alanazi, s. T. *Et al.* (2019) 'evaluation approaches of service oriented architecture (soa) - a survey', *2nd international conference on computer applications and information security, iccais 2019*, pp. 1–6. Doi: 10.1109/cais.2019.8769543.
- Hendini, a. (2016) 'permodelan uml sistem informasi moniotring penjualan dan stok barang (studi kasus: distro zhezha pontianak)', *crop science*, 23(2), pp. 201–205. Doi: 10.2135/cropsci1983.0011183x002300020002x.
- Putri, r. A. (2020) 'diktat sistem basis data', p. 65.
- Ridlo, i. A. (2017) 'pedoman pembuatan flowchart', *academia.edu*, p. 14. Available_at: https://www.academia.edu/34767055/pedoman_pembuatan_flowchart.
- Sontana, i., rahmatulloh, a. And rachman, a. N. (2019) 'application programming interface google picker sebagai penyimpanan data sistem informasi arsip berbasis cloud', *jurnal nasional teknologi dan sistem Informasi*, 5(1), pp. 25–32. doi: 10.25077/teknosi.v5i1.2019.25-32.