

RANCANG BANGUN APLIKASI *BROKER INSURANCE* SYSTEM MENGGUNAKAN METODE *RAPID* *APPLICATION DEVELOPMENT* PADA PT. CIPTA INTEGRA DUTA

Arfin Mustofa^{1*}, Rahmawati¹

¹Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspiptek No. 46,
Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: ^{1*}arfather26@gmail.com, ²dosen02394@unpam.ac.id

(* : coressponding author)

Abstrak– PT. Cipta Integra Duta yang dikenal sebagai ebagai integra merupakan perusahaan yang bergerak dibidang pialang asuransi. Dalam pelaksanaan operasional perusahaan integra sudah tersedia sebuah aplikasi IBOSS namun masih menggunakan teknologi lama yaitu dengan codeigniter versi 3 disatukan dengan senchajs versi 3. Hal itu mengakibatkan perusahaan mempunyai banyak kendala dalam operasional maupun penambahan fitur aplikasi. Dalam aplikasi tersebut tidak terdapat *log activity* tiap departemen, Serta dalam penginputan *memberships* nya masih menggunakan *form* yang diinputkan satu persatu sehingga menjadi kurang efisien apabila terdapat data dalam jumlah yang banyak. Untuk penyimpanan file - file perusahaan masih menggunakan dokumen cetak. Berdasarkan pada permasalahan tersebut, maka dilakukan penelitian untuk membuat sebuah aplikasi *broker insurance system*. Metodologi penelitian yang digunakan ialah metode wawancara, observasi, dan studi pustaka sedangkan metodologi pengembangan sistem menggunakan metode *Rapid Application Development* (RAD) serta *Unified Modelling Language* (UML) sebagai penggambaran sistem. Untuk pembuatan sistemnya menggunakan bahasa pemrograman php dengan *framework* laravel sebagai Open API, reactjs sebagai *visualisasi* data dan Amazon S3 sebagai penyimpanan *storage data*. Hasil Penelitian ini adalah aplikasi ini dapat mempermudah dalam melacak aktivitas dari data *client*, mencari file yang dibutuhkan melalui aplikasi, proses memasukkan data *membership client* hanya cukup dengan menggunakan file *excel* serta pemrosesan data tersimpan secara terpusat dan terintegrasi ke dalam *database*.

Kata Kunci: Rancang Bangun, PT. Cipta Integra Duta, *Broker Insurance*, *Rapid Application Development*, *Unified Modeling Language*, *Laravel*, Open API, ReactJs.

Abstract– PT. Cipta Integra Duta, known as *Integra* is engaged in the insurance brokerage business. In the operational implementation of the integrated company, an IBOSS application is available but it still uses the old technology, namely codeigniter version 3 combined with Senchajs version 3. This causes the company to have many obstacles in following the digitalization era. In this application there is no activity log for each department, and in inputting memberships, it still uses forms that are entered one by one so that it becomes less efficient if there is a large amount of data. For the storage of company files, they still use printed documents. Based on these problems, a research was conducted to create an broker insurance system application. The research methodology used is the method of interview, observation, and literature study, while the system development methodology uses the Rapid Application Development (RAD) method and the Unified Modeling Language (UML) as a system description. For the manufacture of the system using the php programming language with the laravel framework as an Open API, reactjs as data visualization and Amazon S3 as data storage. The results of this study are that this application can make it easier to track activities from client data, find the files needed through the application, the process of entering client membership data is only enough by using an excel file and processing data stored centrally and integrated into the database.

Keywords: Design, PT. Cipta Integra Duta, *Broker Insurance*, *Rapid Application Development*, *Unified Modeling Language*, *Laravel*, Open API, ReactJs.

1. PENDAHULUAN

Dalam era transformasi digital yang sangat erat dengan kemajuan teknologi dan perkembangan untuk menuju sistem yang lebih cerdas daripada sebelumnya memaksa setiap perusahaan di Indonesia harus memikirkan dan melakukan revolusi baru untuk dapat memenuhi tuntutan dari perkembangan zaman.

Dalam setiap aktivitas yang dilakukan oleh masyarakat, khususnya aktivitas yang berkaitan dengan finansial, resiko merupakan suatu hal yang tidak dapat dihindari. Salah satu hal yang dapat meminimalisir resiko tersebut adalah dengan jasa penyedia asuransi.

Perkembangan pialang asuransi atau *broker insurance* menunjukkan peningkatan yang cukup signifikan dari tahun ke tahun. Data menurut investor.id tahun 2022 jumlah pialang asuransi atau *broker* di Indonesia sudah mencapai 196 pialang asuransi. yang terdiri atas 155 Perusahaan Pialang Asuransi dan 41 Perusahaan Pialang Reasuransi.

PT. Cipta Integra Duta yang dikenal sebagai INTEGRA. adalah perusahaan pialang asuransi yang berfokus pada *employee benefit* dan mengedepankan kesejahteraan bagi klien. INTEGRA dimulai sebagai sebuah *departemen* dari perusahaan *broker insurance* terkemuka di Indonesia (IBS GROUP), dan kemudian resmi memisahkan diri dan secara legalitas menjadi satu perusahaan INTEGRA pada tahun 1997.

Dalam pelaksanaan operasional perusahaan integra sudah tersedia sebuah aplikasi IBOSS namun masih menggunakan teknologi lama yaitu dengan codeigniter versi 3 digabung dengan senchajs versi 3, sehingga banyak kendala dalam operasional maupun penambahan fitur aplikasi, serta akan menjadi sulit mencari sumber daya manusia (pegawai) yang mengetahui teknologi tersebut.

Untuk melakukan penyelesaian masalah yang terjadi direkomendasikan untuk melakukan pengembangan sistem *broker insurance* yang dapat mengakomodir segala proses bisnis yang terjadi pada INTEGRA sesuai dengan standar prosedur operasi yang berlaku. Maka dilakukan *revamp system* aplikasi terhadap sistem yang sekarang berjalan. Oleh karena itu penulis akan merancang aplikasi *broker insurance system* menggunakan laravel sebagai *Open API*, *reactjs* sebagai *visualisasi data* dan *Amazon S3* sebagai penyimpanan *storage data*.

2. METODOLOGI PENELITIAN

Metode pengembangan yang digunakan dalam penelitian ini adalah *Rapid Application Development* adalah metode pengembangan perangkat lunak incremental yang menekan pada jangka waktu pengembangan yang pendek. Model RAD ini merupakan adaptasi “berkecepatan tinggi” dari model waterfall, dimana kecepatan pengembangan dicapai melalui pendekatan berbasis komponen. Jika kebutuhan dan cakupan sistem dapat dimengerti pada awal pengembangan, model RAD ini memungkinkan pengembang untuk membuat sistem yang fungsional dalam jangka waktu yang pendek. Metode RAD mempunyai 4 tahapan utama.

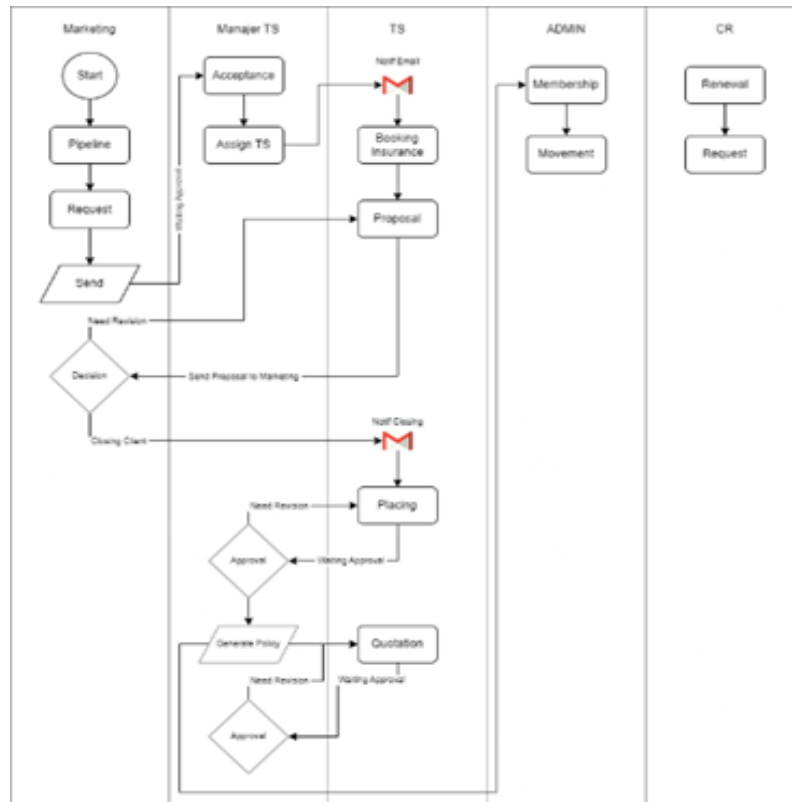
Berikut adalah penjelasan masing-masing fase dalam penelitian ini.

1. Fase Perencanaan
Syarat-Syarat Pada tahap ini dilakukan pengidentifikasi tujuan aplikasi atau sistem serta untuk mengidentifikasi syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut. Pada tahap ini penulis melakukan analisis kebutuhan.
2. Fase Perancangan
Pada tahap ini adalah melakukan proses desain dan melakukan perbaikan - perbaikan apabila masih terdapat ketidaksesuaian desain antara *user* dan *analyst*. Untuk tahap ini maka keaktifan user yang terlibat sangat menentukan untuk mencapai tujuan, karena *user* bisa langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain.
3. Fase Konstruksi
Setelah desain dari sistem yang akan akan dibuat sudah disetujui baik itu oleh masing masing departemen, maka pada tahap ini penulis mengembangkan desain menjadi suatu program. Hal terpenting adalah keterlibatan *user* sangat diperlukan supaya sistem yang dikembangkan dapat memberi kepuasan kepada *user*.
4. Fase Pelaksanaan
Pada tahap ini dilakukan pengujian sistem, dengan melakukan pengujian mandiri yang akan dilakukan penulis dan pengujian yang akan dilakukan oleh *user*.

3. ANALISA DAN PEMBAHASAN

Membangun serta mengubah sistem terdahulu menjadi lebih baik adalah harapan yang merupakan hasil analisa dari mengidentifikasi masalah, hambatan serta kebutuhan pengguna sehingga dapat dijadikan sebagai informasi bahan perbaikan.

Penerapan sistem ini untuk mempermudah dalam melakukan operasional diperusahaan. Detail kebutuhan sistem perlu digambarkan dalam sebuah bagan atau diagram alir berupa Flowchart, untuk mempermudah pemahaman terhadap urutan proses Aplikasi *Broker Insurance*.



Gambar 1. Flowchart Sistem

Peneliti terlebih dahulu melakukan identifikasi aktor yang berguna dalam menentukan entitas apa saja yang bisa berinteraksi secara langsung dengan *system*

1. *Marketing*

Aktor yang mempunyai hak akses untuk mengelola data *pipeline*, mengelola data *request*, melakukan *submit request* untuk meminta *proposal* kepada *technical service*, melakukan *closing* apabila proposal telah sesuai, dan *mendownload quotation*.

2. *Manager TS*

Aktor yang mempunyai hak akses untuk dapat melihat data *pipeline* serta *request*, memilih *pic/ts* yang akan bertanggung jawab, menerima atau menolak *request*, menyetujui *placing* dan *quotation*.

3. *Technical Service*

Aktor yang mempunyai hak akses untuk mendaftarkan/*booking* asuransi bagi *client*, membuat dan mengupload proposal, mengirim *proposal* ke *marketing*, membuat *placing* serta *mengedit quotation*.

4. *Client Relationship*

Aktor yang mempunyai hak akses untuk *renewal* data *client*.

5. *Admin*

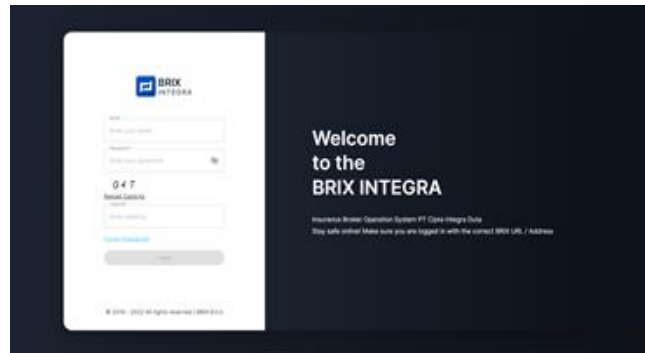
Aktor yang mempunyai hak akses untuk mencatat data karyawan pada perusahaan *client*.

4. IMPLEMENTASI

4.1 Implementasi

1. Halaman Login

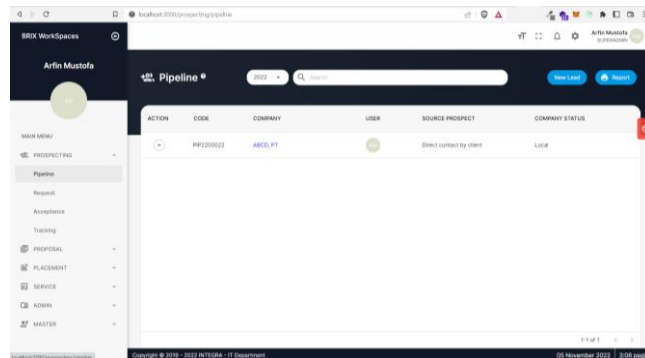
Halaman ini untuk memasukkan *email* dan *password* agar halaman sistem dapat diakses. Pada Halaman ini juga *level* akses akan ditentukan oleh sistem sehingga halaman yang ditampilkan sudah sesuai dengan hak akses.



Gambar 2. Halaman Login

2. Halaman Pipeline

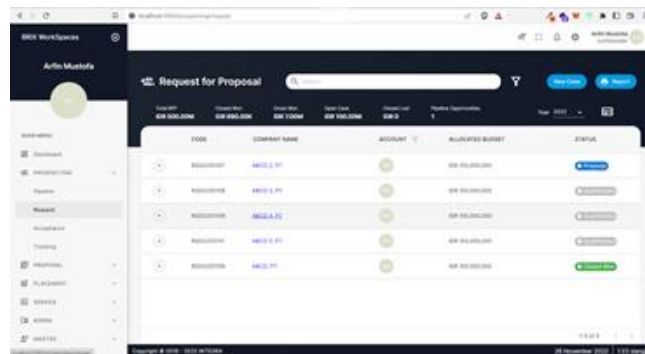
Untuk memasukan data *prospect client* dengan informasi *client* secara umum.



Gambar 3. Halaman Pipeline

3. Halaman Request

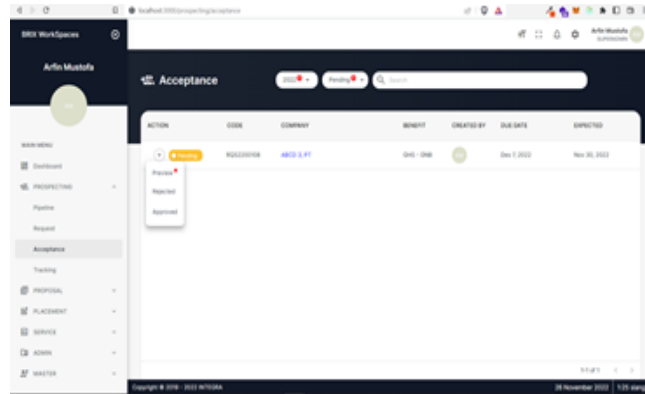
Halaman ini untuk melengkapi data *pipeline* secara spesifik terhadap kebutuhan *client* yang akan mengambil asuransi.



Gambar 4. Halaman Request

4. Halaman *Acceptance*

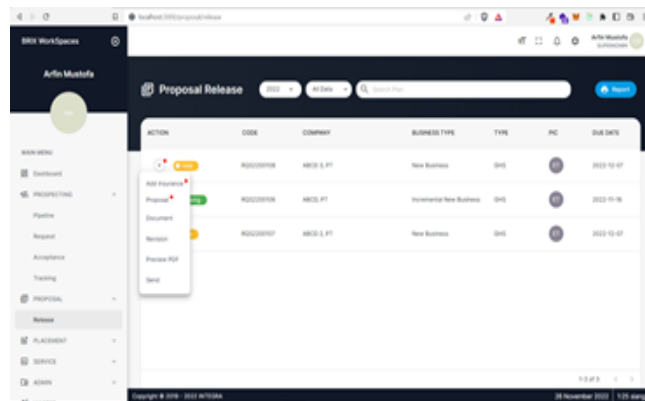
Halaman ini untuk persetujuan dari actor manager *technical service* dari data *pipeline & request* yang kemudian akan dipilhkan *pic technical service*.



Gambar 5. Halaman *Acceptance*

5. Halaman *Release*

Halaman ini untuk memilih asuransi mana yang cocok dengan *form* kebutuhan *client* & mengupload *proposal* yang akan diteruskan kepada actor *marketing*.



Gambar 6. Halaman *Release*

6. Halaman *Tracking*

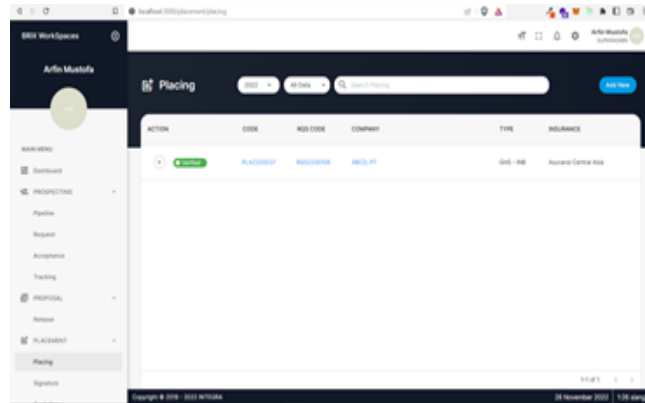
Halaman ini untuk pemantauan data *client* yang sudah masuk dalam aplikasi brix



Gambar 7. Halaman *Tracking*

7. Halaman *Placing*

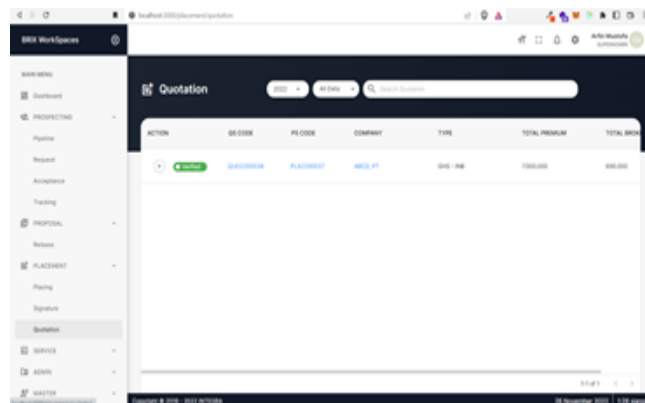
Halaman ini untuk pembuatan surat perjanjian *client* yang telah *closing* antara pihak integra dengan pihak asuransi



Gambar 8. Halaman *Placing*

8. Halaman *Quotation*

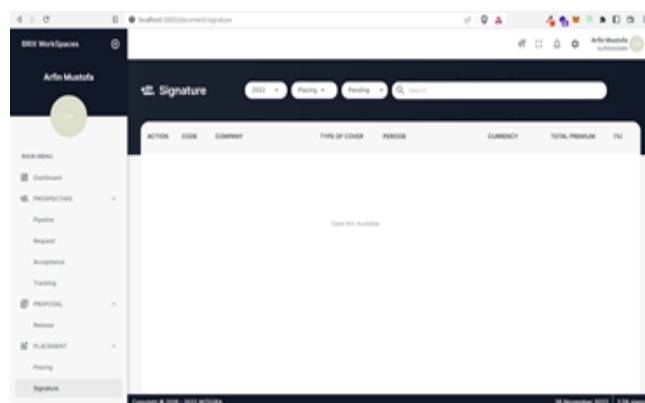
Halaman ini untuk pembuatan surat perjanjian antara pihak integra dengan pihak *client*



Gambar 9. Halaman *Quotation*

9. Halaman *Signature*

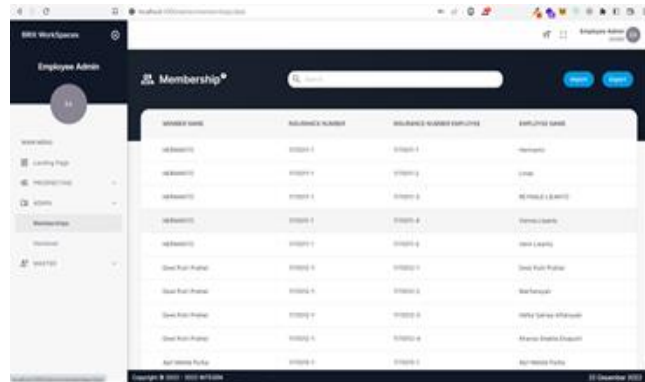
Halaman ini untuk persetujuan dari *manager technical service* terhadap *placing/quotation* yang telah dibuat



Gambar 10. Halaman *Signature*

10. Halaman *Memberships*

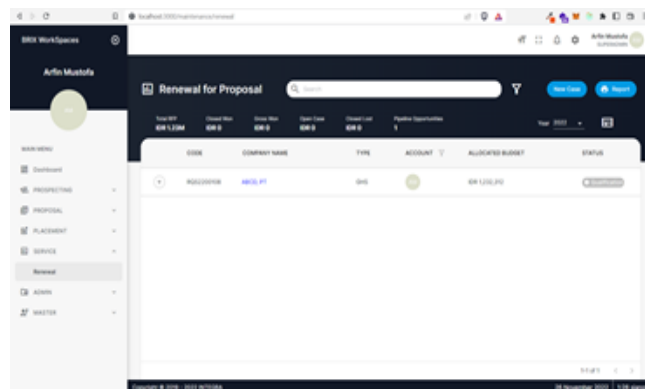
Halaman ini digunakan untuk menginput data pegawai di perusahaan pemegang asuransi / *client*.



Gambar 11. Halaman *Memberships*

11. Halaman *Renewal*

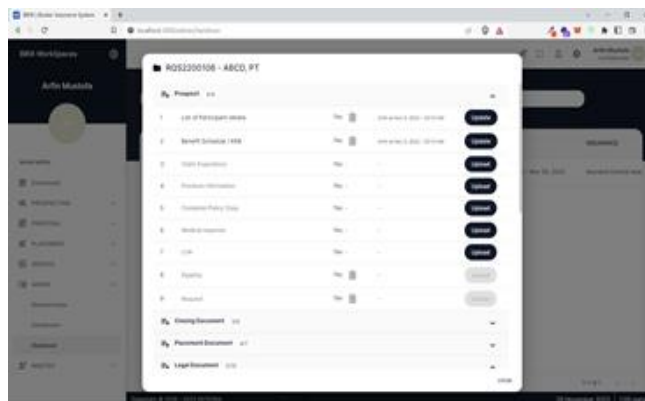
Halaman ini untuk perpanjangan masa kontrak policy dari client yang telah memiliki asuransi.



Gambar 12. Halaman *Renewal*

12. Halaman *Handover*

Halaman ini untuk menampung semua *file client* mulai dari *prospect* hingga periode *policy* berakhir.



Gambar 13. Halaman *Handover*

5. KESIMPULAN

5.1 Kesimpulan

Setelah melewati semua tahapan penelitian mulai dari Perencanaan, Perancangan, Kontruksi, Pelaksanaan hingga pengujian yang telah diuraikan pada *Broker Insurance System* Dalam Operasional perusahaan di PT Cipta Integra Duta dapat disimpulkan sebagai berikut:

1. Aplikasi ini dapat menyimpan *file* dan berbagi *file* di *Cloud Storage*.
2. Sistem ini berjalan dengan baik serta mudah dalam pengoperasiannya. Sebab perancangan sistem dibuat sesederhana mungkin serta dipisahkan sesuai kebutuhan tiap *departement*.
3. Proses untuk menambah data *member* menjadi lebih mudah karena *admin* hanya perlu mengupload data *excel* ke dalam sistem.
4. *Log Request Client* dapat di *tracking* oleh masing - masing user *department*.

5.2 Saran

Saran-saran yang bisa disampaikan untuk pengembangan aplikasi ini adalah sebagai berikut:

1. Sistem ini diharapkan dapat dikembangkan lebih baik lagi, seperti halnya penambahan fitur-fitur baru yang lebih mudah dipahami oleh pengguna.
2. Perlunya mengikuti perkembangan ilmu teknologi guna menyempurnakan sistem yang ada, sehingga kinerja sistem yang berjalan pun akan semakin baik.
3. Perlunya *tracking activity client* yang telah bergabung dalam perusahaan

REFERENCES

- Abdul Kadir. 2008. *Tuntunan Praktis Belajar Database Menggunakan MySQL*, C.V. Andi Offset. Yogyakarta.
- Adam, Muhammad. (2015). *Manajemen Pemasaran Jasa*, Bandung: Alfabeta.
- Andria, Kusriani, & Amborowati, A. (2016). Evaluasi Kualitas Web Portal STT Dharma Iswara Madiun Menggunakan Metode McCall. *Jurnal Ekonomi Dan Teknik Informatika*. Vol 5, Hal. 33-43.
- Arianto, O. D., & Susetyo, Y. A. (2022). Penerapan restful web service Dengan Framework Laravel Untuk Pembangunan Sistem Informasi manajemen sumber Daya Manusia. *JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 7(2), 522-532.
- CORPORATION, N.A.T.I.O.N.A.L. L.E.A.R.N.I.N.G. (2019). *Insurance broker*. PASSBOOKS.
- Firman A., Wowor H.F., Najoan X., 2016. Sistem Informasi Perpustakaan Online Berbasis Web. *E-Jurnal Teknik Elektro dan Komputer*. Vol.5 No.2., Hal. 29-36
- Herdiyatomoko, H. F. (2022). Desain Sistem Backend berbasis rest api menggunakan framework laravel 7. *SKANIKA*, 5(2), 136-144.
- Minnick, C. (2022). *Beginning react JS foundations building user interfaces with reactjs*. <https://doi.org/10.1002/9781119685630>
- Nugroho, A., & Mustofa, T. K. (2012). Implementasi Komputasi Awan Menggunakan teknologi google app engine (GAE) Dan Amazon Web Services (AWS). *Jurnal Teknik Informatika*, 1(1).
- Pressman, Roger, S. (2012). *Rekayasa Perangkat Lunak. Pendekatan Praktisi*. Edisi 7. Yogyakarta: Andi
- Sugiarti, Yuni, 2013, *Analisis dan Perancangan UML (Unified Modeling Language) Generated VB.6*, Yogyakarta, Graha Ilmu.
- Sukanto, R. A., dan Shalahudin, M. 2014, *Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek*. Bandung: Informatika Bandung.
- Using Amazon EC2/S3 as a Data Warehouse Solution. (2010). *Pro Python System Administration*, 349-372.