

Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi
 Jl. Ahmad Yani, K.M. 33,5 - Kampus STMIK Banjarbaru
 Loktabat – Banjarbaru (Tlp. 0511 4782881), e-mail: puslit.stmikbjb@gmail.com
 e-ISSN: 2685-0893
 p-ISSN: 2089-3787

Kajian Pengembangan *Enterprise Architecture* Pada Industri *Software House*

Saiful Azhari Muhammad^{1*}, Richardus Eko Indrajit², Erick Dazki³

Magister Teknologi Informasi, Universitas Pradita

Jl. Gading Serpong Boulevard No.1 Curug Sangereng, Tangerang, Indonesia

*e-mail *Corresponding Author*: saiful.azhari@student.pradita.ac.id

Abstract

The presence of a local software house company is a solution choice for companies on a small, medium and large scale. The software house company has been able to provide solutions to fulfill information systems and produce good quality software products. However, local software house companies face operational problems with the company's internal information technology in completing projects obtained from their customers. To solve the company's internal information technology problems, the company applies the enterprise architecture method to streamline the company's operations. So far, information technology operations in software house companies have experienced ineffective operations with various application systems that have been implemented. Changes in information technology operations by rearranging using the Enterprise Architecture method, and applying it to the company's software house. This study aims to develop Enterprise Architecture in a software house company, where all are designed using Information Technology Planning based on the TOGAF-based Enterprise Architecture Framework. With the application of Enterprise Architecture, information technology operations can be changed to be more organized and more efficient in supporting company operations.

Keyword: *Software House; Enterprise Architecture; TOGAF; Information System*

Abstrak

Hadirnya perusahaan *software house* lokal menjadi pilihan solusi bagi perusahaan dalam skala kecil, skala medium dan skala besar. Perusahaan *software house* telah mampu memberikan solusi pemenuhan sistem informasi dan menghasilkan produk *software* yang berkualitas baik. Namun perusahaan *software house* lokal menghadapi permasalahan operasional informasi teknologi internal perusahaan dalam menyelesaikan proyek yang didapat dari pelanggannya. Untuk menyelesaikan permasalahan informasi teknologi internal perusahaan, perusahaan menerapkan metode *enterprise* arsitektur dalam mengefektifkan operasionalnya. Selama ini operasional teknologi informasi di perusahaan *software house* mengalami operasional yang tidak efektif dengan berbagai sistem aplikasi yang telah diterapkan. Perubahan operasional teknologi informasi dengan menata ulang menggunakan metode *Enterprise Architecture*, dan menerapkannya ke dalam perusahaan *software house*. Penelitian ini bertujuan untuk mengembangkan *Enterprise Architecture* pada perusahaan *software house*, dimana semua dirancang dengan menggunakan *Information Technology Planning* dengan *Framework Enterprise Architecture* berbasis TOGAF. Dengan penerapan *Enterprise Architecture* dapat mengubah operasional informasi teknologi menjadi lebih tertata dan lebih efisien dalam mendukung operasional perusahaan.

Kata kunci: *Software House; Enterprise Architecture; TOGAF; Sistem Informasi*

1. Pendahuluan

Transformasi digital memerlukan komitmen seluruh elemen yang berada dalam organisasi perusahaan karena hal ini mempengaruhi proses bisnis, sumber daya manusia untuk dapat memanfaatkan teknologi dalam semua proses bisnis sehingga mendapatkan model baru untuk meningkatkan pendapatan. Proses transformasi digital ini melibatkan lintas departemen dalam struktur organisasi dan tentunya komitmen yang tinggi dari pimpinan perusahaan untuk secara berkelanjutan melaksanakan proses ini.

Industri *Software House* banyak berperan dalam pemasokan aplikasi kepada industri-industri lain yang sedang melakukan transformasi digital. Dalam penggunaan teknologi digital suatu industri tidak akan lepas dari aplikasi. Karena aplikasi adalah salah satu komponen yang digunakan untuk menjalani transformasi digital.

Permasalahan dalam perusahaan perangkat lunak saat ini adalah pengelolaan dalam menyelesaikan proyek-proyek dari pelanggan. Salah satu permasalahannya karena banyak sistem aplikasi yang digunakan tidak terfokus mendukung operasional informasi teknologi internal perusahaan. Sistem aplikasi yang tidak mendukung akan dilakukan penghapusan dan memulai merancang sistem yang mendukung kegiatan operasional. Fokus dalam penelitian ini hanya menyelesaikan dengan merancang kembali design dari informasi teknologi.

Hal ini harus mencakup semua aspek dalam perusahaan sehingga terbentuk aliran yang jelas mulai dari bisnis, informasi, data dan teknologi yang sesuai dengan strategi transformasi digital dan perkembangan teknologi. Sehingga dibutuhkan suatu *enterprise architecture* yang akan mencakup semua aspek tersebut. Permintaan konsumen akan aplikasi baru dan menarik pada *platform* digital telah mendorong dunia *software house* agar meningkatkan kecepatan pengembangan dan pengalaman pengguna yang lebih berkualitas.

Tujuan dari penelitian ini, menyelesaikan rancangan informasi teknologi dengan menggunakan metode *Enterprise Architecture* berbasis TOGAF. Penggunaan metode TOGAF menjadikan perusahaan *Software House* mempunyai *Information Technology Planning*. Pengembangan sistem internal nantinya mengikuti *Planning* yang sudah direncanakan. Penerapan dari *Enterprise Architecture* nantinya akan menerapkan sistem yang dapat mendukung operasional informasi teknologi menjadi lebih tertata dan menjadi lebih efektif, karena hanya sistem aplikasi yang mendukung operasional saja yang digunakan.

2. Tinjauan Pustaka

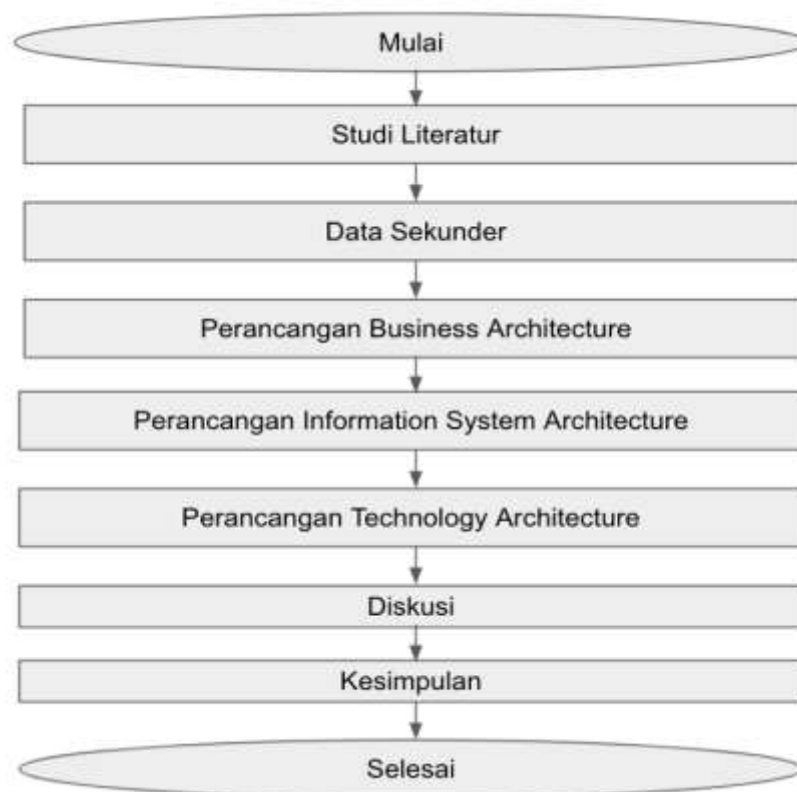
Beberapa penelitian yang membahas tentang studi *Enterprise Architecture* dapat dijadikan tinjauan pustaka untuk memperkuat bahwa penelitian ini terdapat kajian atau pembahasan mengenai topik penelitian *Enterprise Architecture*. Berikut beberapa penelitian dengan topik *Enterprise Architecture*.

Tabel 1. Topik Penelitian Bidang *Enterprise Architecture*

| No | Topik Penelitian | Kelebihan | Kekurangan |
|----|---|--|--|
| 1 | Designing Optimal Enterprise Architecture for Digital Industry: State and Prospects [1] | Pembahasan mengenai <i>Enterprise Architecture</i> pada industri digital, tahapan bisnis arsitektur, aplikasi arsitektur, informasi arsitektur, teknologi arsitektur. | Pembahasan mengenai tahapan migrasi, tahap opportunity dan solusi belum dibahas. |
| 2 | Adaptasi Enterprise Arsitektur pada perusahaan pengembang perangkat lunak dan manfaat yang dihasilkan [2] | Pembahasan mengenai <i>Enterprise Architecture</i> pada industri pengembangan perangkat lunak dengan tahapan bisnis arsitektur, aplikasi arsitektur, informasi arsitektur, teknologi arsitektur, tahapan Opportunities and Solutions | Pembahasan mengenai tahapan strategi tahapan Migration Planning belum dibahas lebih rinci. |
| 3 | Towards the Definition of Enterprise Architecture Debts [3] | Pembahasan mengenai <i>Enterprise Architecture</i> pada industri pengembangan perangkat lunak dengan tahapan bisnis arsitektur, aplikasi arsitektur, informasi arsitektur, teknologi arsitektur dan <i>opportunity solution</i> | Pembahasan mengenai tahapan strategi penerapan belum dibahas lebih rinci. |

Dari hasil penelitian sebelumnya yang terdapat pada Table 1, kebanyakan penelitian hanya berfokus pada bahasan Arsitektur Bisnis, Arsitektur Aplikasi, Arsitektur Informasi dan Aplikasi Teknologi. Penelitian mengenai “Kajian Pengembangan *Enterprise Architecture* Pada Industri *Software House*” membahas dengan menambahkan strategi penerapan atau strategi implementasi. *Research Gap* di penelitian-penelitian sebelumnya tidak mengintegrasikan tahapan strategi implementasi. Pembahasan dengan tahapan strategi implementasi belum dibahas. Keterbatasan pada penelitian ini tidak membahas lebih lengkap pada semua tahapan, tetapi melengkapi dari beberapa penelitian yang ada pada Tabel 1. Kontribusi penelitian ini melengkapi dari penelitian-penelitian sebelumnya dengan menambahkan tahapan strategi dengan menggunakan *Framework TOGAF*.

3. Metodologi



Gambar 1. Usulan metode penelitian

Enterprise Architecture (EA) mendefinisikan keadaan masa depan saat ini dan yang diinginkan dari proses organisasi, kemampuan, sistem aplikasi, data, dan infrastruktur TI dan menyediakan peta jalan untuk mencapai suatu target [4]. Dokumen EA merekam adaptasi aktual dalam elemen organisasi yang berbeda, kapan dan mengapa hal itu terjadi. Ini menjadikan EA sebagai alat yang berguna untuk analisis fleksibilitas [5]. *Enterprise Architecture* menyediakan sarana sistematis untuk identifikasi awal dan respons cepat terhadap tren dan peristiwa penting baik di dalam maupun di luar perusahaan [6].

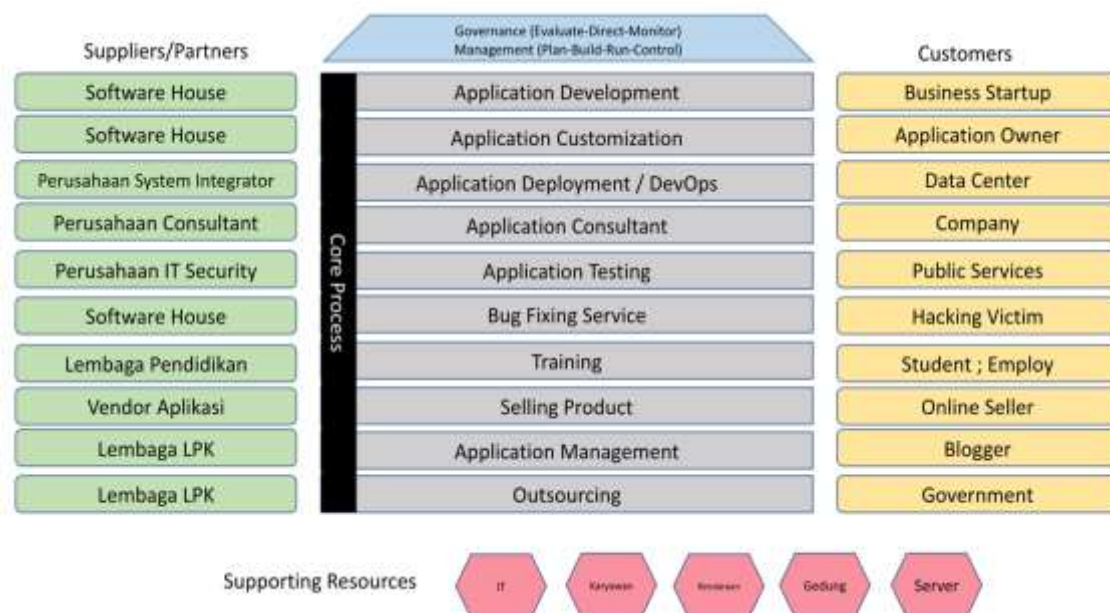
Benefit dengan adanya *Enterprise Architecture* pada suatu industry adalah : Mengidentifikasi ketergantungan sumber daya, Mengidentifikasi sinergi sumber daya, Meningkatkan pengembangan solusi, Meningkatkan kepuasan pelanggan, Meningkatkan kepuasan karyawan, Meningkatkan kualitas informasi, Meningkatkan keselarasan organisasi, Meningkatkan komunikasi organisasi, Meningkatkan keselarasan sumber daya dan Meningkatkan efisiensi [7].

Enterprise Architecture menggunakan tampilan dan lapisan untuk menggambarkan elemen arsitektur untuk mengelola kompleksitas (misalnya proses, data, dan teknologi).

Masing-masing pandangan menggambarkan perspektif berbeda yang bermakna bagi pemangku kepentingan tertentu. Layering menguraikan sistem menjadi kelompok komponen terkait yang prosesnya menyediakan layanan ke lapisan berikutnya [8]. *ArchiMate* adalah standar pemodelan untuk menggambarkan arsitektur perusahaan. Bahasa ini diatur dalam enam lapisan, yaitu Strategi, Bisnis, Aplikasi, Teknologi, Fisik, dan Implementasi & Migrasi [9]. *ArchiMate* diterbitkan oleh *The Open Group* (TOG) pada tahun 2009 sebagai solusi untuk pemodelan Enterprise Architetur. Dirancang khusus untuk menghasilkan model yang mewakili keseluruhan perusahaan [10]. Dengan *ArchiMate* seorang arsitek dapat memodelkan antara lain, produk dan layanan organisasi, bagaimana produk dan layanan ini direalisasikan/disampaikan oleh proses bisnis, dan bagaimana pada gilirannya, proses ini didukung oleh sistem informasi dan infrastruktur TI yang mendasarinya [11]. Dalam pemodelan *enterprise architecture* banyak penulis menyarankan penerapan *ArchiMate* untuk memodelkan arsitektur perusahaan mengikuti *The Open Group Architecture Framework* (TOGAF) [12].

4. Hasil

Model bisnis secara umum yang menggambarkan komponen *core process*, *supplier*, pelanggan, dan *supporting resource* perusahaan pada industri *software house* yang diperoleh dari hasil wawancara ahli dapat disajikan pada gambar berikut:



Gambar 2. Core Process, Suppliers, Customers dan Supporting Resources

Menurut para ahli yang kami wawancarai, ada 3 *core proses* yang merupakan penyumbang pendapatan terbesar dalam industri *software house*, yaitu: *Application Development*, *Application Customization*, dan *Application Testing* [13].

4.1. Arsitektur Bisnis

Pada bagian ini akan dijelaskan aspek-aspek yang mempengaruhi jalannya bisnis pada industry *software house*. Aspek-aspek tersebut dapat digambarkan dalam bentuk *Business Model Canvas* sebagaimana gambar 3:

1) Key Partnerships

- a. *Software House*. Sesama perusahaan yang bergerak di bidang *software house* bisa saling membantu dan berbagi *resource* demi tercapainya tujuan masing-masing.
- b. *Sistem integrator*. Dalam pengimplementasian aplikasi yang berskala besar dan terhubung ke banyak sistem akan membutuhkan arsitektur jaringan dan server yang kompleks [14], hal ini akan lebih mudah jika berkolaborasi dengan *Sistem Integrator*.

Sehingga *Software House* dapat fokus dalam mengembangkan aplikasi dan *Sistem Integrator* akan mengurus *intrastructure*.

- c. Perusahaan Konsultan. Pada keadaan tertentu *software house* juga membutuhkan konsultan untuk mendapatkan penyelesaian masalah dengan lebih baik.
- d. Perusahaan *IT Security*. Untuk memastikan aplikasi yang telah dikembangkan aman dari bahaya *hacker*, *Software House* membutuhkan *IT security* untuk melakukan pengujian keamanan aplikasi. Tujuan utama kolaborasi *software house* dengan perusahaan *IT security* adalah untuk menghilangkan atau meminimalkan risiko terjadinya ancaman informasi dengan memanfaatkan serangkaian kegiatan perencanaan, organisasi, teknis, dan pengendalian [15].
- e. Lembaga Pendidikan dan Lembaga LPK. Perekrutan programmer akan lebih mudah jika telah bekerja sama dengan lembaga pendidika, sehingga dapat melihat potensi yang masih dapat dikembangkan lagi kedepannya.

| | | | | |
|---|---|---|---|---|
| Key Partnerships 1. Software House 2. System Integrator 3. Perusahaan Konsultan 4. Perusahaan IT Security 5. Lembaga Pendidikan 6. Vendor Aplikasi 7. Lembaga LPK | Key Activities 1. Application Development 2. Application Consultant 3. Bug Fixing Service 4. Training 5. Application management 6. Application Customization 7. Application Testing Key Resources 1. Karyawan 2. Office 3. Servers | Value Propositions 1. Teknologi Terkini 2. Standard International 3. 24/7 Support 4. SDA berkompeten | Customer Relationships 1. Forum 2. Sosial Media Channels 1. Telephone 2. Email 3. Instant Messaging 4. Drive Sharing | Customer Segments 1. Staff IT 2. Company 3. Pemerintahan 4. BUMN |
| Cost Structure 1. Listrik 2. Internet 3. Colocation 4. Karyawan 5. Logistik | | Revenue Streams 1. Business startup 2. Application Owner 3. Company 4. Government | | |

Gambar 3. Business Model Canvas industri Software House

2) Value Proposition

- a. Teknologi terkini. Penggunaan teknologi terkini dalam pembangunan aplikasi membuat pelanggan merasa meliki aplikasi dengan teknologi terbaru sehingga meningkatkan kepuasan pelanggan.
- b. Standard International. Aplikasi dibuat dengan standard yang baik. Kualitas suatu sistem dinilai dari kinerja, efisiensi, interoperabilitas, ketersediaan, keamanan, kegunaan, akurasi, perawatan, dan kesesuaian [16]. Sehingga membuat pelanggan merasa nyaman dalam menggunakan aplikasi dan tidak terlalu khawatir dengan data yang ada didalam aplikasi.
- c. 24/7 Support. Dengan ketersediaan support yang baik diharapkan segala keluhan dan masalah yang terjadi pada aplikasi dapat dengan cepat diatasi. Sehingga pengguna tetep merasa nyaman dalam penggunaan aplikasi.
- d. SDA berkompeten. Kualitas SDA sangat mempengaruhi kualitas produk yang dihasilkan, dengan adanya SDA yang berkualitas maka produk yang dihasilkanpun akan lebih baik. Sehingga pelanggan dapat memperoleh aplikasi yang diinginkan sesuai dengan expektasi mereka.

3) Key Activities

- a. *Application Development*. Service pembuatan aplikasi sesuai dengan spesifikasi yang telah diberikan oleh pelanggan.
- b. *Application Consultant*. Service consultant dalam perancangan aplikasi untuk perusahaan-perusahaan yang telah memiliki divisi *Application Development* tetapi belum

memiliki banyak pengalaman dalam suatu jenis aplikasi yang ingin mereka kembangkan. Sehingga membutuhkan sudut pandang pihak ketiga yang lebih berpengalaman.

- c. *Bug Fixing Service*. Service memperbaiki aplikasi milik pelanggan yang mengalami kegagalan sistem atau ditemukan adanya celah yang membahayakan data didalamnya.
 - d. *Training*. Service memberikan pelatihan pembuatan aplikasi.
 - e. *Application Management/Maintenance*. Service pengelolaan aplikasi milik perusahaan ataupun badan pemerintahan.
 - f. *Application Customization*. Service kustomisasi aplikasi yang telah dimiliki oleh pelanggan atau bisa juga kustomisasi aplikasi *opensource* untuk memenuhi kebutuhan pelanggan yang terkadang sedikit berbeda dengan kebutuhan pada umumnya. Kustomisasi memainkan peran penting dalam penyediaan aplikasi untuk pelanggan yang berbeda karena sangat mungkin logika bisnis, antarmuka, dan data akan berbeda untuk setiap pelanggan [17].
 - g. *Application Testing*. Service pengecekan aplikasi yang mencakup *load testing*, *stress testing*, *performance testing* dan juga *security testing*. Service ini merupakan salah satu penyumbang pendapatan terbanyak karena pengujian perangkat lunak, dikenal sebagai proses yang memakan waktu dan mahal [18].
- 4) *Customer Relationships*
 - a. Forum dapat menjadi tempat berdiskusi secara lebih teratur dan terstruktur baik sesama developer maupun dengan para pelanggan.
 - b. Melalui sosial media penyampaian informasi kepada para pelanggan akan lebih menarik dan menyenangkan.
 - 5) *Customer Segment*
 - a. Staf IT, *service training* akan lebih banyak dibutuhkan oleh staf IT karena perkembangan teknologi membuat mereka harus bisa mengikuti agar perusahaan mereka juga dapat mengikuti *trend* teknologi.
 - b. *Company*. Tidak semua perusahaan memiliki divisi *application development*, sehingga mereka sangat membutuhkan *software house* untuk memenuhi kebutuhan aplikasi.
 - 6) *Revenue Stream*
 - a. *Business Startup*. Kebutuhan awal seperti *website company profile* dan aplikasi absensi banyak dibutuhkan oleh *startup* aplikasi, dimana *software house* dapat membantu mengembangkan aplikasi-aplikasi tersebut sesuai dengan kebutuhan mereka.
 - b. Perusahaan yang telah memiliki aplikasi juga masih membutuhkan *software house* ketika mereka ingin memperbaiki aplikasi tersebut ataupun menambahkan fitur tertentu.
 - c. Lembaga Pemerintahan.

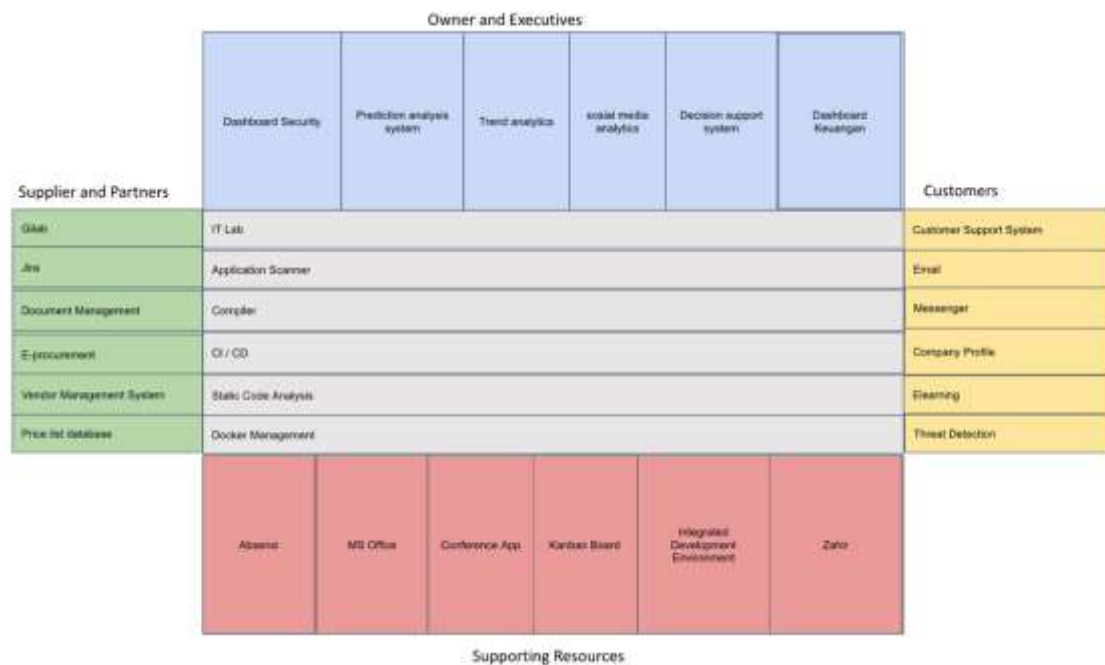
4.2. Arsitektur Aplikasi

Aplikasi-aplikasi yang dibutuhkan pada komponen-komponen arsitektur mulai dari *supplier*, *core process*, *support*, *owner and executive*, hingga *customer*. Arsitektur aplikasi digambarkan melalui visualisasi pada gambar 4.

Aplikasi dan sistem yang dibutuhkan untuk menunjang bisnis utama atau *core process* di industri *Software House* diantaranya adalah:

- a. IT Lab, dibutuhkan untuk melakukan *research* teknologi-teknologi baru sebelum mulai diimplementasikan ke produk. Sehingga teknologi baru yang siap diimplementasikan ke produk adalah teknologi yang sudah stable dan telah teruji.
- b. *Application Scanner*, dibutuhkan untuk mengidentifikasi celah keamanan pada aplikasi yang telah dibuat. Dengan begitu dapat diketahui lebih dini jika terdapat celah keamanan dan dapat segera dilakukan perbaikan.
- c. *Compiler*, dibutuhkan untuk merubah *code* program menjadi *binary* aplikasi yang dapat dijalankan pada suatu sistem operasi tertentu.
- d. *Continuous Integration and Continuous Deployment*, digunakan untuk mengontrol *source code* aplikasi dan sebagai otomatisasi proses *deploy* aplikasi baik di dalam *environment development* maupun *environment production*.

- e. *Static Code Analysis* adalah sistem yang dapat mengidentifikasi kesalahan pengkodean aplikasi sehingga menimbulkan celah keamanan. Dengan adanya sistem ini diharapkan celah keamanan yang ditimbulkan karena adanya kesalahan pengkodean dapat diminimalisir.
- f. *Docker Management* berfungsi mengatur docker-docker yang digunakan di lingkungan perusahaan serta dapat memonitoring kinerja setiap docker yang ada.



Gambar 4. Arsitektur Aplikasi pada industri *Software House*

Aplikasi dan sistem yang dibutuhkan untuk membantu kerjasama dan komunikasi dengan supplier antara lain adalah:

- a. *Gitlab*, platform ini yang sangat berguna untuk saling berbagi *source code* aplikasi antara team *development* internal dan team *development* pihak ketiga.
- b. *Jira*, merupakan platform team management yang sangat berguna untuk memonitor pekerjaan yang sedang dikerjakan bersama dengan partner.
- c. *Document Management*, digunakan untuk berbagi document produk dengan lebih flexibel dan terstruktur.
- d. *E-procurement*, untuk memudahkan partner melakukan penawaran dan *bidding* proyek.
- e. *Vendor Management System*, untuk manage data-data vendor yang telah bekerjasama.
- f. *Price list database*, untuk mendata seluruh produk baik dari pihak ketiga maupun produk sendiri agar lebih mudah dalam pengelolaan.

Aplikasi dan sistem yang dibutuhkan oleh *executive* dan *owner* adalah:

- a. *Dashboard Security*, dashboard yang menunjukkan status keamanan jaringan perusahaan, dimana *dashboard* ini menampilkan serangan secara *realtime*, sehingga dapat segera diketahui jika ada pihak-pihak yang tidak bertanggung jawab berusaha masuk kedalam sistem perusahaan.
- b. *Prediction Analysis Sistem*, merupakan *machine learning* yang membantu *executive* memprediksi trend yang akan terjadi berdasarkan history dan kejadian pada saat ini. Sehingga dapat membantu untuk membuat plan bisnis jangka panjang.
- c. *Trend Analytics*, merupakan sistem *machine learning* yang mempelajari *trend* yang sedang terjadi pada saat ini. Sistem ini berguna untuk membantu memetakan inovasi yang cocok untuk segera dikembangkan.
- d. *Social Media Analytics*, merupakan sistem *big data* yang mengolah data dari berbagai sosial media untuk melihat sentimen dan kebutuhan market terhadap suatu produk. Sehingga *executive* dapat menentukan produk yang tepat untuk dikembangkan.

- e. *Decision Support System*, sistem *artificial intelegent* yang dapat membantu *executive* mengambil keputusan dengan dasar *history* dan kasus-kasus yang pernah ada. Sehingga *executive* dapat mengambil keputusan dengan lebih tepat dan akurat.
 - f. *Dashboard* Keuangan untuk memonitor keuangan perusahaan.
Aplikasi dan sistem yang dibutuhkan untuk menjalin komunikasi dan bisnis dengan para *customers* diantaranya adalah sebagai berikut :
 - a. *Customer Support System*, berfungsi untuk berkomunikasi dengan customer baru yang ingin menanyaan produk atau *service* ke perusahaan.
 - b. *Email*, untuk berkomunikasi dengan cara yang lebih formal, biasanya digunakan untuk mengirim berkas.
 - c. *Messenger*, untuk berkomunikasi dengan *customer* setia yang sudah lama berlangganan, sehingga response dapat lebih cepat.
 - d. *Company Profile*, sangat penting untuk mengenalkan perusahaan kita kemasyarakat dan calon pelanggan baru.
 - e. *Elearning*, digunakan untuk terus saling terhubung dengan *customer* yang pernah mengambil training, dimana *customer* tetap terus dapat mengakses materi dari *elearning*.
 - f. *Threat Detection*, sebagai alat pendeteksi *customer/oknum* nakal yang berusaha menyerang sistem perusahaan.
- Aplikasi yang dibutuhkan oleh *supporting resources* dalam menjalankan tugas-tugasnya:
- a. Absensi, aplikasi untuk memonitor kehadiran karyawan.
 - b. *MS Office*, aplikasi untuk membantu karyawan menyusun dokumen.
 - c. *Conference Application*, aplikasi untuk *conference* baik sesama karyawan maupun dengan *customer*.
 - d. *Kanban Board*, aplikasi berbagi tugas dan *task management*.
 - e. *Integrated Development Environment*, aplikasi yang membantu *programmer* dalam melakukan pengkodean aplikasi.
 - f. *Zahir*, aplikasi *accounting* untuk membantu mencatat keuangan perusahaan.

4.3. Arsitektur Informasi

Arsitektur informasi serta hubungan masing-masing data yang tersimpan dalam bentuk database pada aplikasi-apliaksi yang telah dibahas sebelumnya ditunjukkan pada gambar 5:

a. Owner dan Executive

Database yang dibutuhkan untuk *owner* dan *executive* ada 3, yaitu *database* media sosial, *database* media *online* dan *database trend* digital dimana setelah data-data tersebut diolah akan ditambihkan diempat aplikasi yang berbeda yaitu dashboard *prediction analysis*, *social media analytics*, *trend analitycs* dan *decision support system*. Adapun dashboard *security* mengambil data dari database *threat* yang ada pada segmen *customers*.

b. Supplier

Segmen *supplier* membutuhkan 2 database, yaitu: database vendor yang digunakan untuk aplikasi *vendor management*, *price list database* dan juga *e-procurement* kemudian *database document* digunakan oleh aplikasi *Gitlab*, *Jira* dan *document mangement*.

c. Core Process

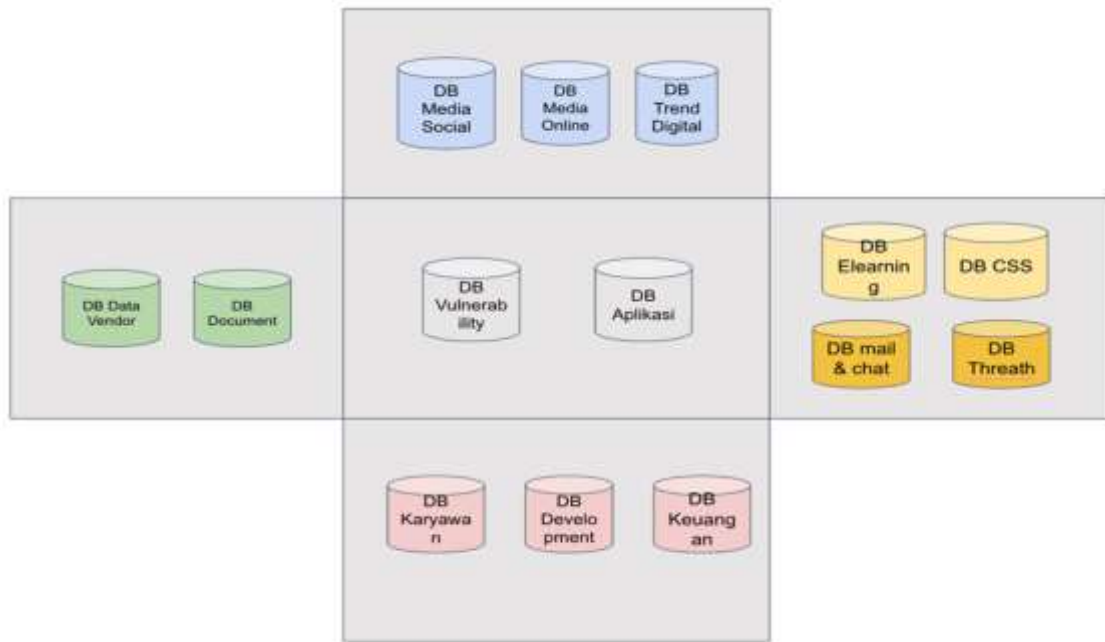
Segmen *core process* memiliki 2 database, yaitu : database *vulnerability* yang digunakan oleh aplikasi *scanner* dan aplikasi *static code analysis*. Database aplikasi digunakan oleh *system continuous integration and continuous deployment*.

d. Customer

Pada segmen ini terdapat 4 database, yaitu : database e-learning untuk menyimpan data aplikasi e-learning, database main dan *chat* untuk menyimpan data email dan *messenger*, database CSS untuk menyimpan data *costumer service system* dan database *threat* untuk menyimpan log serangan.

e. Supporting Resources

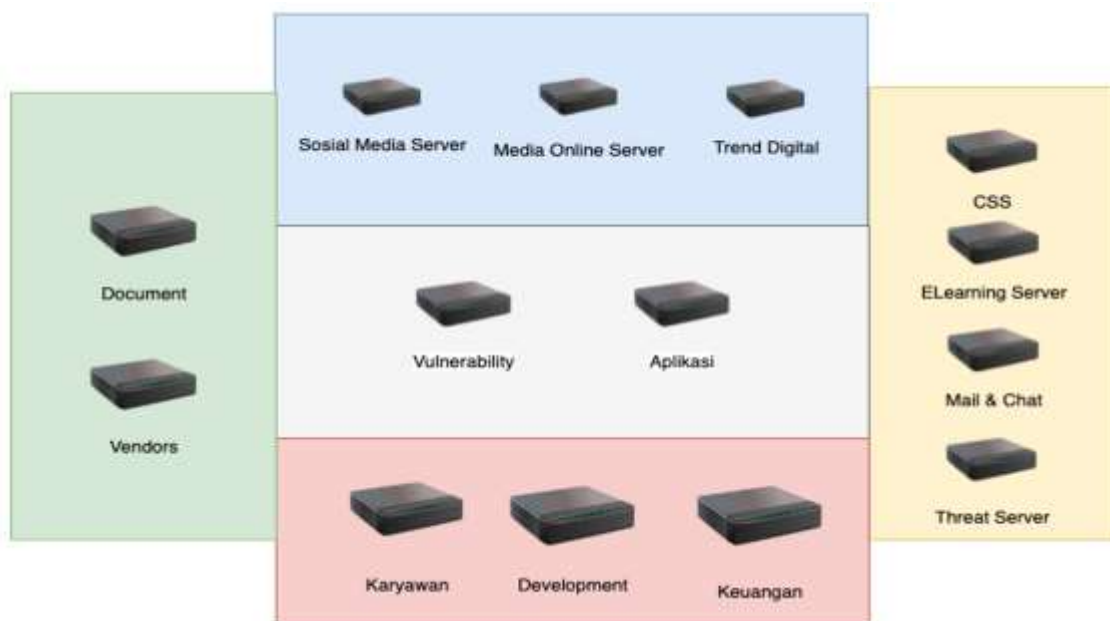
Segmen ini memiliki 3 database, yaitu database karyawan untuk menyimpan data dari aplikasi absensi, database *development* untuk menyimpan data *kanban board* dan database keuangan untuk mensuplay data ke aplikasi *zahir*.



Gambar 5. Arsitektur Informasi Dan Data Industri *Software House*

4.4. Arsitektur Teknologi

Arsitektur teknologi dirancang berdasarkan Arsitektur Data dan Informasi serta disesuaikan dengan kebutuhan Arsitektur Aplikasi dan Arsitektur Bisnis. Arsitektur Teknologi yang dirancang untuk kebutuhan industri *Software House* ditunjukkan pada Gambar 6.

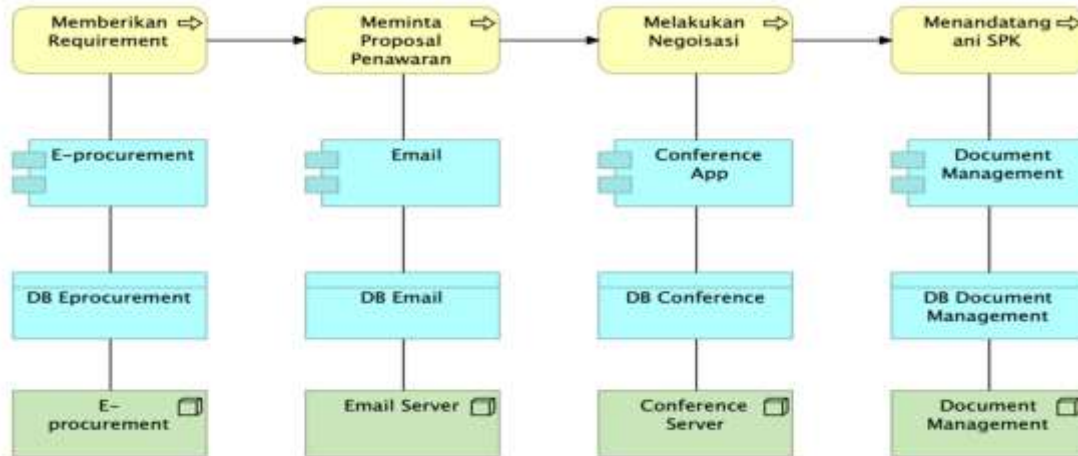


Gambar 6. Arsitektur Teknologi *Software House*

Secara keseluruhan terdapat 30 aplikasi dan sistem yang mendukung berjalannya industri *software house*. Dari sejumlah aplikasi tersebut tidak semuanya menggunakan database terpisah, ada beberapa aplikasi yang sudah bundling dengan database. Sehingga hanya ada 14 database yang dikelola. Pada arsitektur teknologi terdapat 14 server, karena ada beberapa aplikasi yang tidak banyak membutuhkan *resource* digabungkan menjadi satu agar lebih hemat.

Pada penelitian ini pemodelan arsitektur bisnis difokuskan pada 3 *core process* yang paling banyak menghasilkan profit.

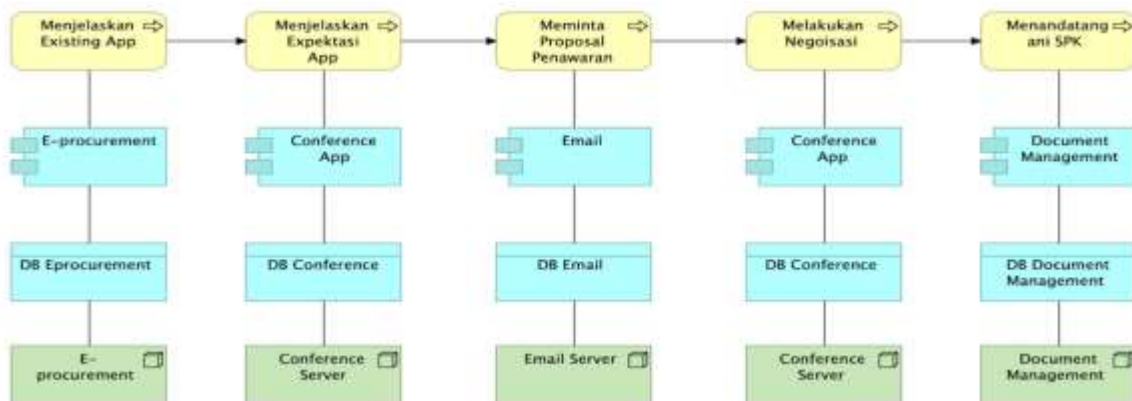
1) *Application Development*, skema arsitektur bisnis *Application Development*



Gambar 7. Arsitektur bisnis *Application Development*

Ketika *customer* akan memesan aplikasi, hal pertama yang perlu dilakukan adalah memberikan *requirement* kepada perusahaan *software house* melalui aplikasi *E-procurement*. Setelah *requirement* diterima, tahap selanjutnya adalah memintal proposal penawaran dari pihak *software house* melalui email ataupun *E-procurement*. Tahap berikutnya setelah proposal didapatkan adalah melakukan negoisasi harga untuk mendapat harga yang lebih murah dari yang ditawarkan oleh pihak *software house*, tahap ini bisa dilakukan dengan cara tatap muka maupun melalui *conference application*. Jika negoisasi telah mencapai kesepakatan, maka dilakukan penandatanganan surat perjanjian kerjasama oleh kedua belah pihak. Semua proses disimpan dan diproses diserver masing-masing aplikasi pendukung.

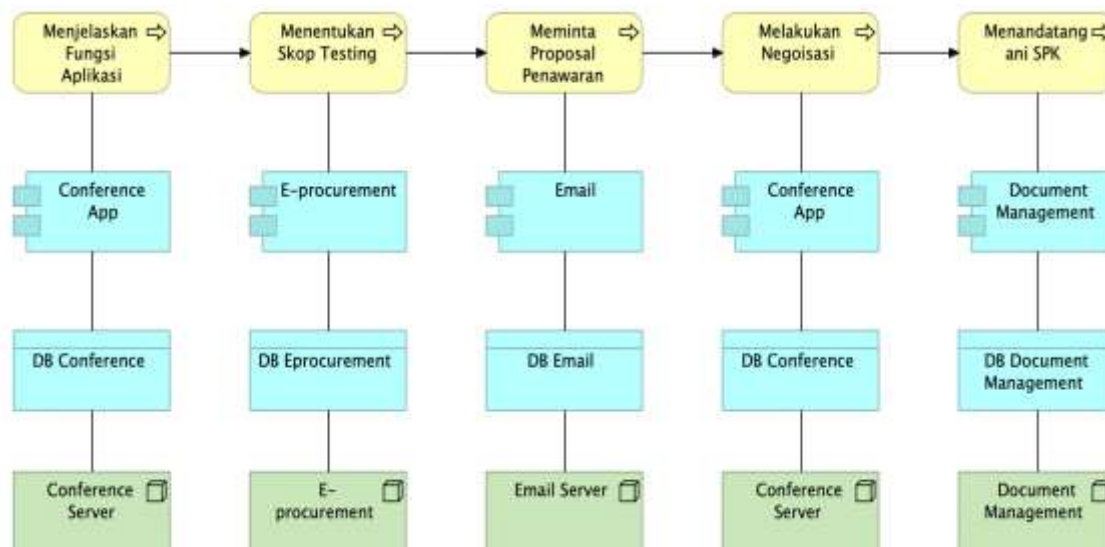
2) *Application Customization*, skema arsitektur bisnis *Application Customization*:



Gambar 8. Arsitektur bisnis *Application Customization*

Hal yang dilakukan oleh *customer* ketika membutuhkan *service Application Customization* adalah yang pertama menjelaskan keadaan aplikasi saat ini melalui *E-procurement*. Kemudian yang kedua menjelaskan kebutuhan dan expektasi yang diinginkan melalui *conference application*. Tahap ketiga meminta proposal penawaran dari *software house* melalui *email*. Setelah penawaran didapatkan, tahap keempat adalah melakukan negoisasi harga untuk mendapatkan harga yang paling cocok. Jika negoisasi telah mendapatkan kesepakatan harga, tahap selanjutnya adalah penandatanganan surat pernyataan kerja sama oleh kedua belah pihak.

3) *Application Testing* Berikut adalah arsitektur bisnis *application testing*



Gambar 9. Arsitektur bisnis *application testing*

Untuk mengajukan *service application testing*, yang harus dilakukan oleh *customer* yang pertama adalah menjelaskan fungsi aplikasi yang akan dilakukan testing, selanjutnya menentukan skop *testing* yang akan dikerjakan oleh pihak *software house*, kemudian meminta proposal penawaran dari pihak *software house*, tahap berikutnya melakukan negoisasi untuk mendapatkan harga yang disepakati dan jika telah sepakat maka dilakukan penandatanganan surat pernyataan kerja sama oleh kedua belah pihak. Semua proses tersebut tercatat pada database aplikasi yang mendukung proses terkait dan diproses di server masing-masing aplikasi.

4.5. Pembahasan

Dari seluruh proses yang telah dilakukan mulai dari perancangan bisnis arsitektur, aplikasi arsitektur, informasi arsitektur, teknologi arsitektur hingga penerapannya dapat menyelesaikan permasalahan sistem menjadi lebih tertata dan lebih efektif dan efisien. Pentingnya perancangan *enterprise architecture* dapat mengubah *information technology planning* sesuai dengan kebutuhan sistem yang efisien. Penerapan *Enterprise Architecture* dengan metode TOGAF, tidak sekedar melakukan pengembangan sistem, tetapi juga merencanakan sistem yang dikembangkan sesuai dengan *IT Planning* perusahaan *Software House*, sebagaimana dalam [19].

5. Kesimpulan

Industri *software house* paling tidak memiliki 10 *core* bisnis yang dapat menghasilkan keuntungan. Pada penelitian kali ini kami membahas 3 *core* bisnis yang paling banyak menghasilkan pendapatan menurut para ahli yang telah kami wawancarai, ketiga *core* bisnis tersebut adalah *application development*, *application customization*, dan *application testing*. Perancangan Arsitektur *Enterprise* pada 3 *core* bisnis tersebut telah digambarkan menggunakan *archimate core framework*. Dengan mengintegrasikan semua elemen dalam organisasi tersebut diharapkan dapat mencapai target perusahaan dengan lebih cepat dan lebih efisien serta memudahkan dalam membuat perencanaan implementasi teknologi informasi untuk perusahaan di *software house*, sehingga terjadinya resiko kegagalan dalam proses implementasi informasi dan teknologi dapat diperkecil.

Daftar Referensi

- [1] V. V. Martynov, D. N. Shavaleeva, and A. I. Salimova, "Designing Optimal Enterprise Architecture for Digital Industry: State and Prospects," *Proc. - 2018 Glob. Smart Ind. Conf. GloSIC 2018*, pp. 1–7, 2018, doi: 10.1109/GloSIC.2018.8570159.
- [2] K. Jamróz, D. Pitulej, and J. Werewka, "Adapting enterprise architecture at a software

- development company and the resultant benefits," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8627 LNCS, pp. 170–185, 2014, doi: 10.1007/978-3-319-09970-5_16.
- [3] S. Hacks, H. Hofert, J. Salentin, Y. C. Yeong, and H. Lichter, "Towards the definition of enterprise architecture debts," *Proc. - IEEE Int. Enterp. Distrib. Object Comput. Work. EDOCW*, vol. 2019-October, pp. 9–16, 2019, doi: 10.1109/EDOCW.2019.00016.
- [4] G. Shanks, M. Gloet, I. Asadi Someh, K. Frampton, and T. Tamm, "Achieving benefits with enterprise architecture," *J. Strateg. Inf. Syst.*, vol. 27, no. 2, pp. 139-156, 2018, doi: 10.1016/j.jsis.2018.03.001.
- [5] Y. Gong, J. Yang, and X. Shi, "Towards a comprehensive understanding of digital transformation in government: Analysis of flexibility and enterprise architecture," *Gov. Inf. Q.*, vol. 37, no. 3, p. 101487, 2020, doi: 10.1016/j.giq.2020.101487.
- [6] J. J. Korhonen and M. Halen, "Enterprise architecture for digital transformation," in *Proceedings - 2017 IEEE 19th Conference on Business Informatics, CBI 2017*, pp. 349-358, 2017, vol. 1, doi: 10.1109/CBI.2017.45.
- [7] E. Niemi and S. Pekkola, "The Benefits of Enterprise Architecture in Organizational Transformation," *Bus. Inf. Syst. Eng.*, vol. 62, no. 6, pp. 585-597, 2020, doi: 10.1007/s12599-019-00605-3.
- [8] Z. Pourzolfaghar, V. Bastidas, and M. Helfert, "Standardisation of enterprise architecture development for smart cities," *J. Knowl. Econ.*, vol. 11, no. 4, pp. 1336-1357, 2020, doi: 10.1007/s13132-019-00601-8.
- [9] T. P. Sales, B. Roelens, G. Poels, G. Guizzard, N. Guarino, and J. Mylopoulos, "A Pattern Language for Value Modeling in ArchiMate," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11483, pp. 230-245, 2019, doi: 10.1007/978-3-030-21290-2_15.
- [10] A. Ellerm and M. E. Morales-Trujillo, "Modelling Security Aspects with ArchiMate: A Systematic Mapping Study," *IEEE*, pp. 577-584, 2020, doi: 10.1109/SEAA51224.2020.00094.
- [11] S. de Kinderen, K. Gaaloul, and H. A. Proper, "Bridging value modelling to ArchiMate via transaction modelling," *Softw. Syst. Model.*, vol. 13, no. 3, pp. 1043-1057, 2014, doi: 10.1007/s10270-012-0299-z.
- [12] A. Sadovykh, A. Bagnato, A. J. Berre, and S. Walderhaug, "ArchiMate as a Specification Language for Big Data Applications - DataBio Example," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer, vol. 12055, pp. 191-199, 2020 doi: 10.1007/978-3-030-39306-9_14.
- [13] M. Varl, J. Duhovnik, and J. Tavčar, "Customized product development supported by integrated information," *J. Ind. Inf. Integr.*, no. 25, p. 100248, 2021, doi: 10.1016/j.jii.2021.100248.
- [14] A. E. Coronado Mondragon and C. E. Coronado Mondragon, "Managing complex, modular products: how technological uncertainty affects the role of systems integrators in the automotive supply chain," *Int. J. Prod. Res.*, vol. 56, no. 20, pp. 6628-6643, 2018, doi: 10.1080/00207543.2018.1424362.
- [15] E. K. Szczepaniuk, H. Szczepaniuk, T. Rokicki, and B. Klepacki, "Information security assessment in public administration," *Comput. Secur.*, vol. 90, pp. 101709, 2020, doi: 10.1016/j.cose.2019.101709.
- [16] S. R. Mirsalari and M. Ranjbarfard, "A model for evaluation of enterprise architecture quality," *Eval. Program Plann.*, vol. 83, p. 101853, 2020, doi: 10.1016/j.evalprogplan.2020.101853.
- [17] A. Q. Ali, A. B. M. Sultan, A. A. A. Ghani, and H. Zulzalil, "A Systematic Mapping Study on the Customization Solutions of Software as a Service Applications," *IEEE*, vol. 7, pp. 88196-88217, 2019, doi: 10.1109/ACCESS.2019.2925499.
- [18] R. Gao, Y. Wang, Y. Feng, Z. Chen, and W. Eric Wong, "Successes, challenges, and rethinking – an industrial investigation on crowdsourced mobile application testing," *Empir. Softw. Eng.*, vol. 24, no. 2, pp. 537-561, 2019, doi: 10.1007/s10664-018-9618-5.
- [19] B. Hanafi and R.D.H. Purba, "Perancangan Enterprise Architecture Dengan Modified Togaf Adm Pada PT Ilmu Komputercom Braindevs Sistema," *Journal of Information System, Informatics and Computing*, vol. 5, no. 2, pp. 222-231, 2021.